

2110101 : การทำโปรแกรมคอมพิวเตอร์

ระบบคอมพิวเตอร์เบื้องต้น

ภาควิชาวิศวกรรมคอมพิวเตอร์
จุฬาลงกรณ์มหาวิทยาลัย

วัตถุประสงค์ของ 2110101

- รู้จักคอมพิวเตอร์
- ประยุกต์คอมพิวเตอร์
- แก้ปัญหาทางวิศวกรรมด้วยคอมพิวเตอร์
- **ทำโปรแกรมคอมพิวเตอร์**

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

2

คอมพิวเตอร์

- เครื่องประมวลผล จัดเก็บ และสื่อสารข้อมูล
 - ประมวลผลข้อมูลได้รวดเร็ว
 - จัดเก็บข้อมูลได้มากมาย
 - สื่อสารกันได้อย่างไร้พรมแดน
- การใช้งาน
 - บันทึกลง
 - เกมส์ ดูนั่ง ฟังเพลง ...
 - จัดเก็บและค้นคืน
 - WWW, ระบบทะเบียนราษฎร์ ระบบห้องสมุด ...
 - ธุรกิจ
 - email ระบบบุคลากร, วิเคราะห์ข้อมูล, วางแผนทรัพยากร ...
 - ...



คุณลักษณะ

- ซื่อสัตย์ ทำตามคำสั่ง
- ไม่บ่น ทำแต่เรื่องซ้ำๆ ซากๆ
- จำได้มาก จำได้แม่น
- ทำงานรวดเร็ว
- ไม่ฉลาด สอนอย่างไรก็ทำอย่างนั้น

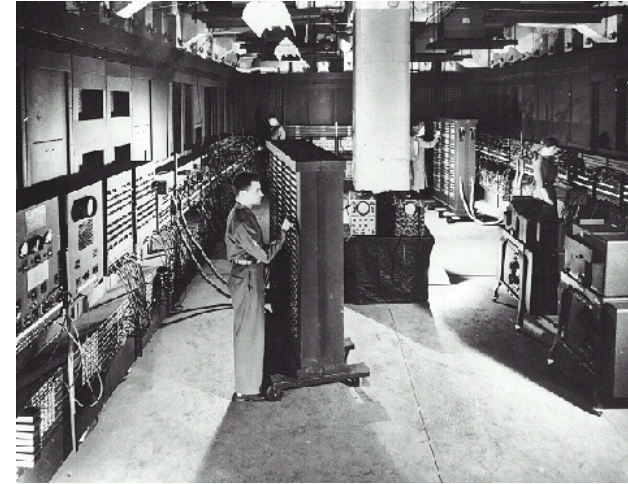


เราเพิ่งมีคอมพิวเตอร์ใช้กันแค่ 70 กว่าปี

- เล็กลง (ร้อยล้านทรานซิสเตอร์ปลายนิ้วก้อย)
- เร็วขึ้น (พันล้านคำสั่งในวินาที)
- จุข้อมูลมากขึ้น (แสนล้านตัวอักษรในฝ่ามือ)
- กินไฟน้อยลง (ใช้ไฟทั้งอำเภอลดเหลือเท่าทีวี)
- ราคาถูกลง (ไบต์ละบาทเป็นแสนไบต์บาท)

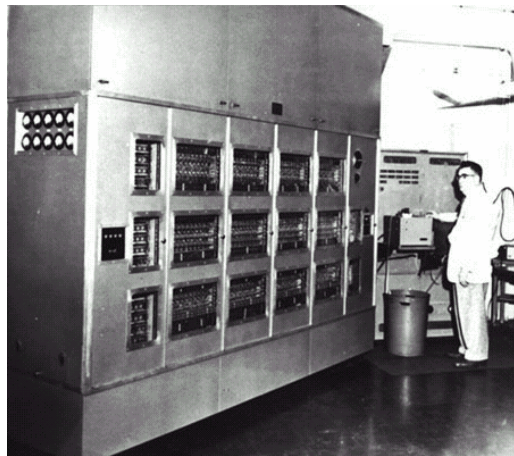
Moore's Law : เทคโนโลยีทางคอมพิวเตอร์มี
ประสิทธิภาพมากขึ้นสองเท่าทุกๆ 18 เดือน

ENIAC (1946)



1946 : ENIAC เดิมใช้ห้องขนาด 30x50 ตารางฟุต
1996 : ENIAC chip ขนาด 7.44 x 5.29 ตาราง มม.

ILLIAC I (1952)



ใช้ 2,800 หลอด กว้างxยาวxสูง = $2 \times 10 \times 8.5$ ฟุต³
หนัก 5 ตัน มี memory 5.12K และ magnetic drum
64K ให้บริการเป็นเวลา 10 ปี

Apple Computers



1976 : Apple I



1977 : Apple II



1980 : Apple III



1983 : Apple IIe



1983 : Lisa



1984 : Mac 128k



1986 : Mac plus



1987 : Mac II



1989 : Mac portable

...



2002 : iMac



2007 : iMac

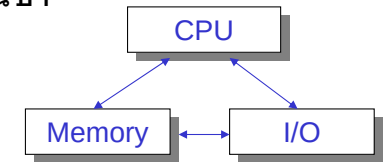
องค์ประกอบของคอมพิวเตอร์

- hardware เพื่อประมวลผล จัดเก็บ และสื่อสารข้อมูล
- software เพื่อสั่งให้ hardware ทำงานตามขั้นตอนที่กำหนดไว้



Hardware

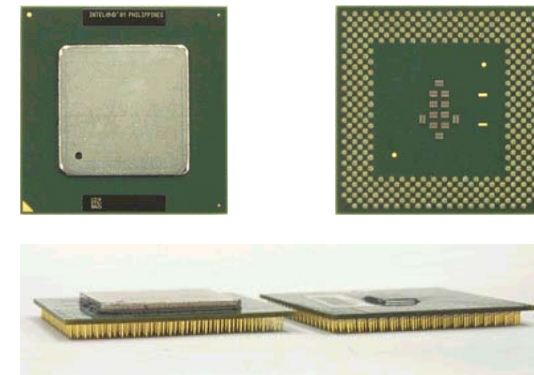
- Central Processing Unit (CPU)
 - เพื่อประมวลผลทางคณิตศาสตร์, ตรรกะ, การทดสอบจริงเท็จ, การทำงานซ้ำ
- Memory
 - เพื่อจัดเก็บและค้นคืนข้อมูล
- I/O
 - เพื่อรับส่งสื่อสารข้อมูล (จอภาพ แป้นพิมพ์ เมาส์ โมเด็ม เครื่องพิมพ์ กล้อง ...)



Central Processing Unit

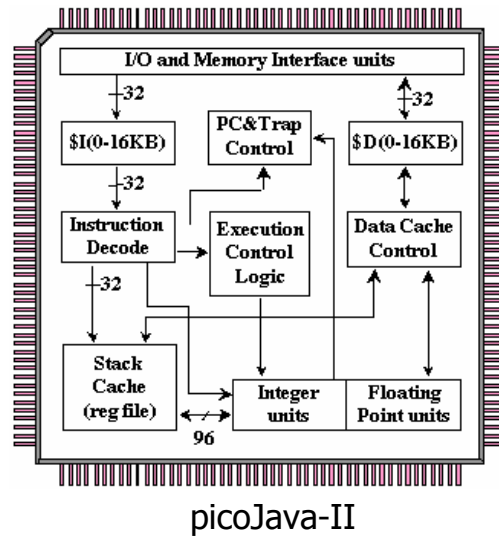
- ทำงานตามชุดคำสั่งที่กำหนดให้
- ทำงานได้รวดเร็ว (พันล้านคำสั่งในวินาที)
- ควบคุมการทำงานของส่วนอื่นๆ
- ชุดคำสั่ง
 - เป็นรหัสฐานสอง
 - เก็บในหน่วยความจำ
 - เปลี่ยนแปลงได้
- ยี่ห้อ
 - Intel Pentium, Intel Celeron, Intel StrongARM, AMD Xeon, SunSparc, IBM PowerPC, ...

CPU

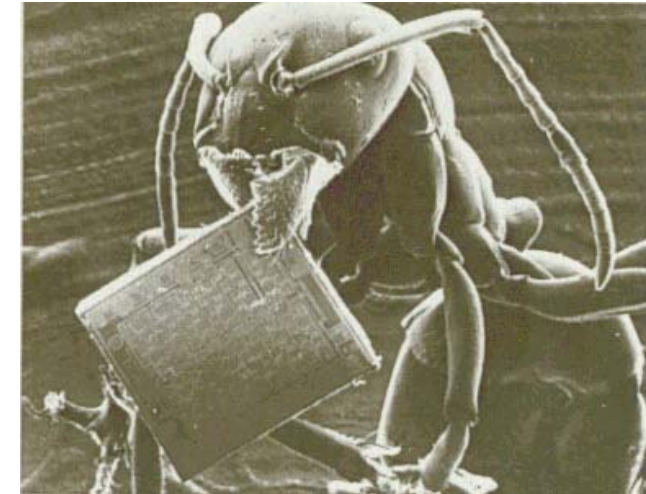


microprocessor

CPU



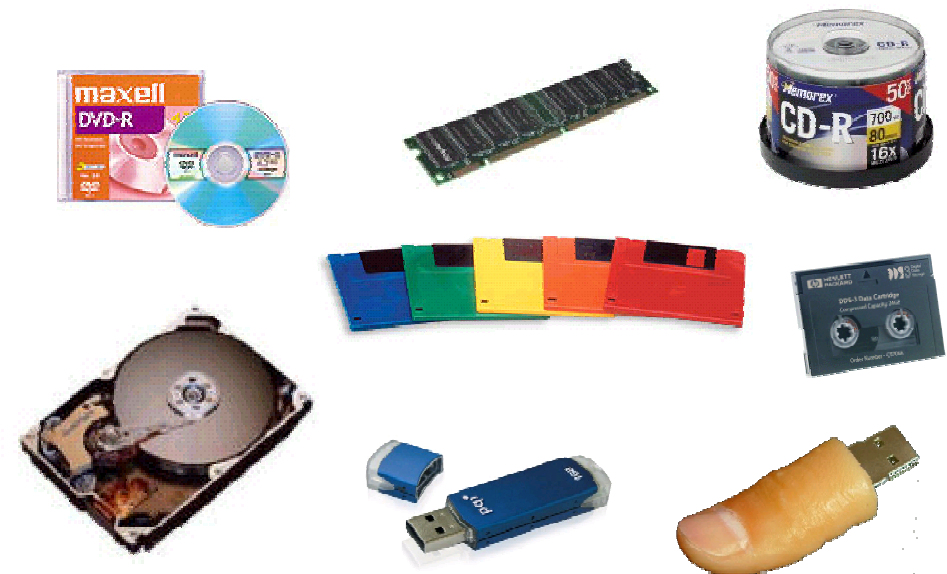
มดกับชิป



Memory

- เก็บชุดคำสั่งการทำงานของ CPU
- เก็บข้อมูลสำหรับการประมวลผล
- แบ่งเป็น
 - main memory
 - ที่เก็บชั่วคราว
 - เร็วแต่ราคาแพง (10^{-8} วินาที แล่นไบต์บาท)
 - RAM (random access memory)
 - secondary storage
 - ที่เก็บถาวร
 - ช้ากว่าแต่ราคาถูก (HD : 0.01 วินาที สิบล้านไบต์บาท)
 - hard disk, CD-ROM, DVD-ROM, flash drive, ...

Memory



Input/Output Devices

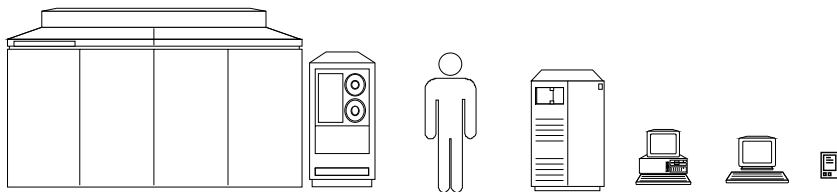
- อุปกรณ์รับข้อมูล
 - แป้นพิมพ์ เมาส์ แป้นกดเกมส์ กล้อง สแกนเนอร์ โมเด็ม ไมโครโฟน เครื่องอ่านบาร์โค้ด เครื่องอ่านบัตร
- อุปกรณ์ส่งหรือแสดงข้อมูล
 - จอภาพ เครื่องพิมพ์ โมเด็ม ลำโพง



หน่วยต่างๆ ที่น่าสนใจ

- bit - หน่วยเล็กสุดแทนสองสภาวะ (binary digit)
- byte - 8 bits
- word - จำนวน bytes ที่เครื่องประมวลผลได้สะดวก
- CPU : 900MHz, 2.5GHz
- RAM : 512MB, 2GB
- Harddisk : 200GB, 7200rpm
- Printer : 600dpi, 10ppm
- Sound : 44.1KHz 16-bit stereo
- Modem : 56Kbps

ประเภท : ขนาดและน้ำหนัก



- Mainframe
- Mini-computer
- Personal Computer
- Network Computer
- Personal Digital Assistant



Reliability

Type	Platform	Outages/year
Mainframe	IBM S/390 (Sysplex)	10 min.
	Tandem Nonstop	1.7 Hrs.
Mini	Digital VAX	18.9 Hrs.
	Unix-based	23.8 Hrs.
PC	Windows-NT based	224.5 Hrs.

ตัวอย่าง : คอมพิวเตอร์

- CPU : Intel® Core 2 Duo Processor 2.6 GHz
- Memory

- 2GB DDR2-667
- HD 320 GB SATA, 7200RPM
- Diskette drive : 1.44MB
- DVD-RW 16X SuperMulti Function



25,000 บาท

- I/O

- NVIDIA® GeForce® 7100 graphics
- Monitor : LCD 19" 1280 x 1024
- Sound card : Audio Codec 7.1 Surround Sound
- Gigabit Ethernet LAN on Board
- 10 USB 2.0 ports (4 Front 6 Rear)
- Internal 56K fax-modem

Software

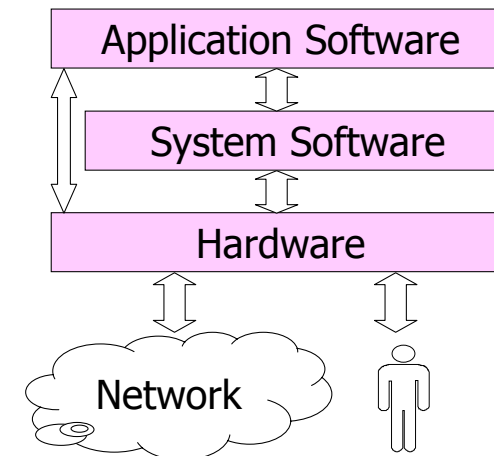
- ชุดคำสั่งต่างๆ ซึ่งสั่งให้ hardware ทำงานตามขั้นตอนที่กำหนดไว้
- เราเก็บ software มากมายไว้ใน secondary storage (harddisk, CD-ROM, DVD-ROM)
- ระบบนำ (หรือ load) software ที่ต้องการลงใน main memory เพื่อใช้งาน



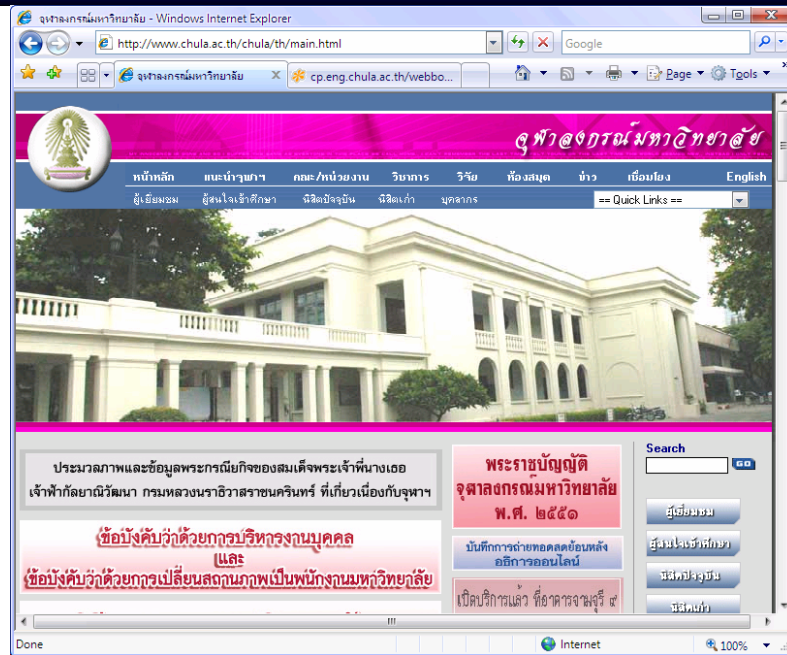
Software

- System software
 - Operating system (Windows XP, Linux, Mac OS, ...)
 - System tools (antivirus, zip/unzip, backup/restore, ...)
 - Development tools (compiler, debugger, ...)
 - ...
- Application software
 - Office (email, word processor, spreadsheet, ...)
 - Network applications (browser, chat, ...)
 - Entertainment (movies, musics, games, ...)
 - ...

Hardware & Software



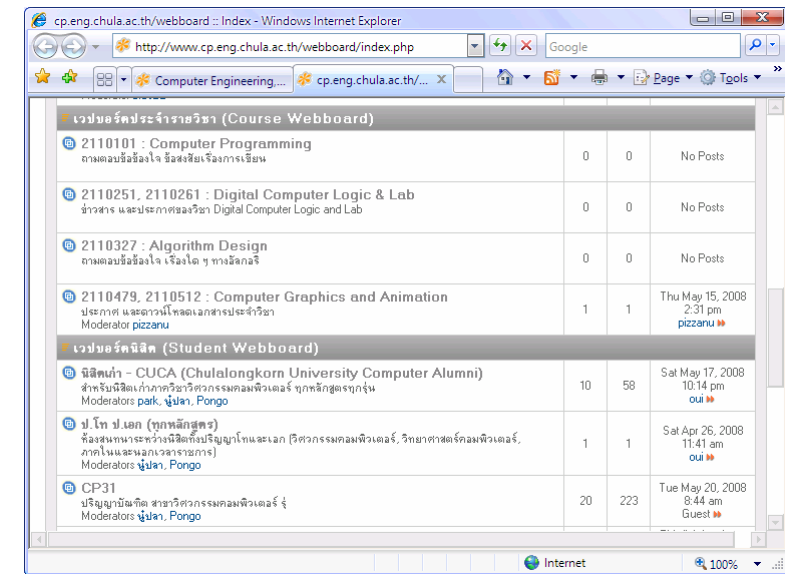
Web Browser



2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

25

Web Board

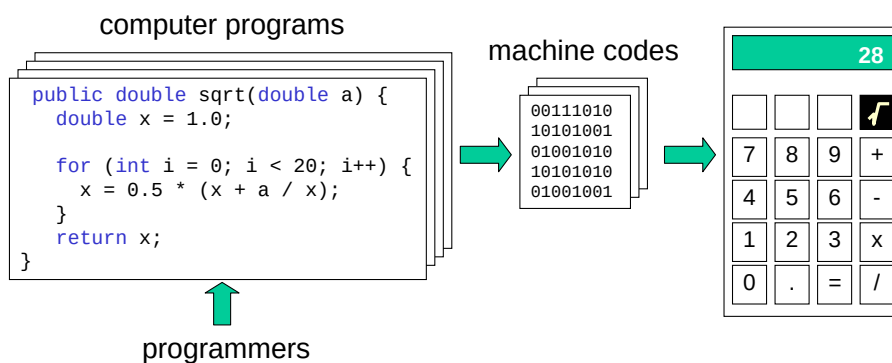


2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

26

Software

- เราพัฒนา software ได้ด้วยการเขียนโปรแกรมคอมพิวเตอร์ (computer program)
 - โปรแกรมคือชุดคำสั่งซึ่งเขียนด้วยภาษาคอมพิวเตอร์ เพื่อสั่งงานคอมพิวเตอร์ให้บริการตามข้อกำหนด



2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

27

ขั้นตอนการพัฒนา software

- กำหนดความต้องการ (ปัญหา)
- วิเคราะห์ตัวปัญหา
- ออกแบบขั้นตอนวิธีแก้ไขปัญหา
- ทำโปรแกรมตามขั้นตอนวิธีที่ออกแบบ
- ทดสอบความถูกต้อง
- ปรับปรุงแก้ไขโปรแกรม

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

28

ตัวอย่าง

- ความต้องการ
 - ค่าเฉลี่ยของชุดข้อมูล
- วิเคราะห์
 - แหล่งข้อมูล
 - ผู้ใช้ป้อนจำนวนข้อมูล
 - ผู้ใช้ป้อนข้อมูลต่างๆ ผ่านทางแป้นพิมพ์
 - ผลลัพธ์
 - แสดงค่าเฉลี่ยออกสู่จอภาพ

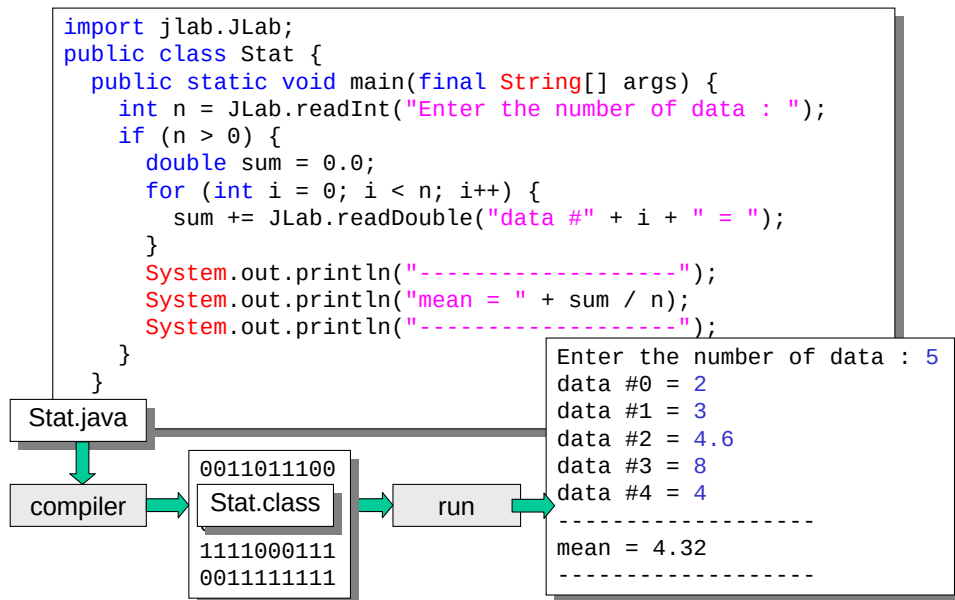
ตัวอย่าง

- ออกแบบ
 - รับจำนวนข้อมูลจากผู้ใช้ n
 - รับข้อมูลต่างๆ จากผู้ใช้ $x_1, x_2, \dots, x_n$
 - ผลลัพธ์คำนวณได้จากสูตร

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

- แสดงผลลัพธ์ออกสู่จอภาพ

ตัวอย่าง



2110101 : การทำโปรแกรมคอมพิวเตอร์

แนวคิดการทำโปรแกรม

ภาควิชาวิศวกรรมคอมพิวเตอร์
จุฬาลงกรณ์มหาวิทยาลัย

Program, Class, Method, Statement

```
public class Nat {  
    public class DWindow {  
        public class Tree {  
            public static void main(String[] args) {  
                DWindow w = new DWindow(250, 250);  
                Nat nat = new Nat(Nat.HAPPY);  
                w.add(nat, 165, 245 - nat.getHeight()/2);  
                drawTree(w, 125, 245, 100, 3.14159/2, 10);  
            }  
            public static void drawTree(DWindow w, int x, int y,  
                int len, double r, int lvl) {  
                if (lvl == 0) return;  
                int x1 = (int)(x + len * Math.cos(r));  
                int y1 = (int)(y - len * Math.sin(r));  
                w.drawLine(x, y, x1, y1);  
                drawTree(w, x1, y1, len*6/10, r+0.5, lvl-1);  
                drawTree(w, x1, y1, len*6/10, r-0.5, lvl-1);  
            }  
        }  
    }  
}
```

หนึ่งโปรแกรมมีหลายคลาส หนึ่งคลาสมีหลายเมทอด หนึ่งเมทอดมีหลายคำสั่ง

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

2

หนึ่งโปรแกรมมีหลายคลาส

```
public class Nat {  
    public class DWindow {  
        public class Tree {  
            public static void main(String[] args) {  
                DWindow w = new DWindow(250, 250);  
                Nat nat = new Nat(Nat.HAPPY);  
                w.add(nat, 165, 245 - nat.getHeight()/2);  
                drawTree(w, 125, 245, 100, 3.14159/2, 10);  
            }  
            public static void drawTree(DWindow w, int x, int y,  
                int len, double r, int lvl) {  
                if (lvl == 0) return;  
                int x1 = (int)(x + len * Math.cos(r));  
                int y1 = (int)(y - len * Math.sin(r));  
                w.drawLine(x, y, x1, y1);  
                drawTree(w, x1, y1, len*6/10, r+0.5, lvl-1);  
                drawTree(w, x1, y1, len*6/10, r-0.5, lvl-1);  
            }  
        }  
    }  
}
```

ชื่อคลาส

หนึ่งคลาสมีหลายเมทอด

```
public class Nat {  
    public class DWindow {  
        public class Tree {  
            public static void main(String[] args) {  
                DWindow w = new DWindow(250, 250);  
                Nat nat = new Nat(Nat.HAPPY);  
                w.add(nat, 165, 245 - nat.getHeight()/2);  
                drawTree(w, 125, 245, 100, 3.14159/2, 10);  
            }  
            public static void drawTree(DWindow w, int x, int y,  
                int len, double r, int lvl) {  
                if (lvl == 0) return;  
                int x1 = (int)(x + len * Math.cos(r));  
                int y1 = (int)(y - len * Math.sin(r));  
                w.drawLine(x, y, x1, y1);  
                drawTree(w, x1, y1, len*6/10, r+0.5, lvl-1);  
                drawTree(w, x1, y1, len*6/10, r-0.5, lvl-1);  
            }  
        }  
    }  
}
```

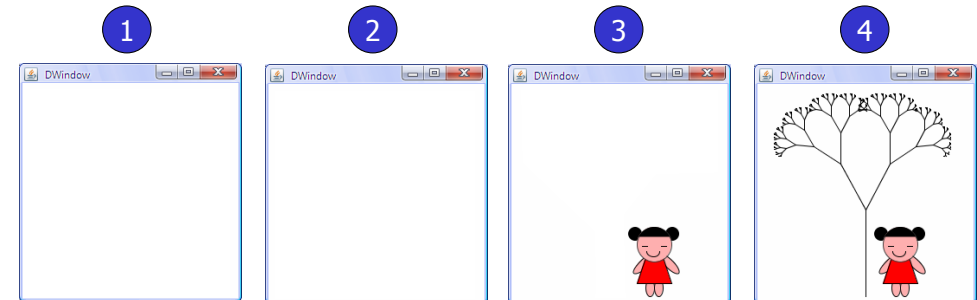
ชื่อเมทอด

หนึ่งเมทอดมีหลายคำสั่ง

```
public class Nat {
public class DWindow {
public class Tree {
    public static void main(String[] args) {
        DWindow w = new DWindow(250, 250);
        Nat nat = new Nat(Nat.HAPPY);
        w.add(nat, 165, 245 - nat.getHeight()/2);
        drawTree(w, 125, 245, 100, 3.14159/2, 10);
    }
    public static void drawTree(DWindow w, int x, int y,
                                int len, double r, int lvl) {
        if (lvl == 0) return;
        int x1 = (int)(x + len * Math.cos(r));
        int y1 = (int)(y - len * Math.sin(r));
        w.drawLine(x, y, x1, y1);
        drawTree(w, x1, y1, len*6/10, r+0.5, lvl-1);
        drawTree(w, x1, y1, len*6/10, r-0.5, lvl-1);
    }
}
```

เมทอดทำทีละคำสั่งจากบนลงล่าง

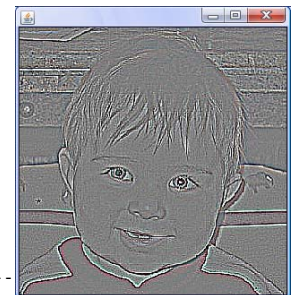
- 1 DWindow w = new DWindow(250, 250);
- 2 Nat nat = new Nat(Nat.HAPPY);
- 3 w.add(nat, 165, 245 - nat.getHeight()/2);
- 4 drawTree(w, 125, 245, 100, 3.14159/2, 10);



คำสั่งมีหลายแบบ

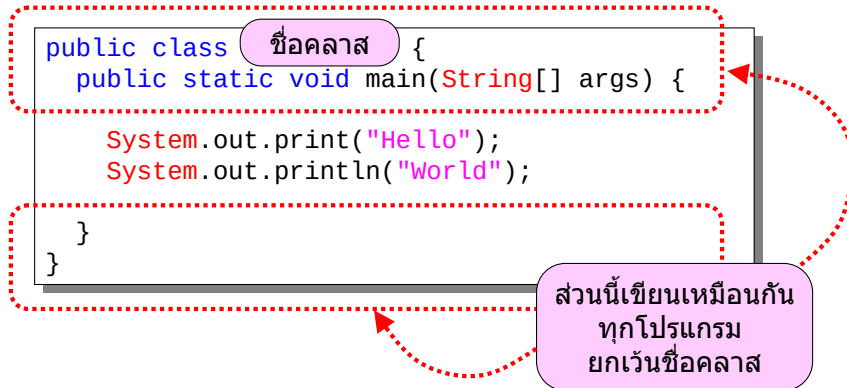
- เตรียมที่เก็บ `double n, c = 0;`
- คำนวณ `n = len * Math.sin(r);`
- ดัดสินใจ `if (n == 0) return;`
- ทำซ้ำ `while(n > 0) {`
`n = n/2;`
`c++;`
`}`
- แสดงผล `System.out.println(c);`
- และอื่น ๆ ...

```
import java.awt.image.DPImage;
public class Filter {
    static int[][] SHARPEN = new int[][]{{ 0, -1, 0 },{-1, 5, -1},{ 0, -1, 0 }};
    static int[][] BLUR = new int[][]{{ -2, -2, -2 },{-2, 12, -2},{ -2, -2, -2 }};
    static int[][] EMOSS = new int[][]{{ -2, -1, 0 },{-1, -1, -1},{ 0, -1, -1 }};
    public static void main(String[] args) {
        int w = 300, h = 300;
        DPImage in = new DPImage(w, h, "baby.jpg");
        DPImage out = new DPImage(w, h);
        in.setVisible(true);
        int[] px = filter(in.getPixelMap(), EMOSS);
        out.setPixelMap(px); out.setVisible(true);
        out.setLocation(in.getX() + in.getWidth(), out.getY());
    }
    public static int[] filter(int[][] b, int[][] f) {
        int w = b.length, h = b[0].length;
        int[] out = new int[w*h];
        int fw = f.length, fh = f[0].length;
        int fsum = 0;
        for (int i = 0; i < fw; i++)
            for (int j = 0; j < fh; j++)
                fsum += f[i][j];
        int sw = fw / 2, sh = fh / 2;
        for (int x = sw; x < w - sw; x++)
            for (int y = sh; y < h - sh; y++) {
                int sumr = 0, sumg = 0, sumb = 0;
                for (int i = -sw; i <= sw; i++)
                    for (int j = -sh; j <= sh; j++) {
                        sumr += getR(b[x+i][y+j]) * f[i+sw][j+sh];
                        sumg += getG(b[x+i][y+j]) * f[i+sw][j+sh];
                        sumb += getB(b[x+i][y+j]) * f[i+sw][j+sh];
                    }
                if (fsum != 0) {
                    sumr /= fsum; sumg /= fsum; sumb /= fsum;
                } else {
                    sumr += 128; sumg += 128; sumb += 128;
                }
                out[x][y] = mixRGB(sumr, sumg, sumb);
            }
        return out;
    }
    public static int getR(int color) {
        return (color >> 16) & 0xFF;
    }
    ...
}
```

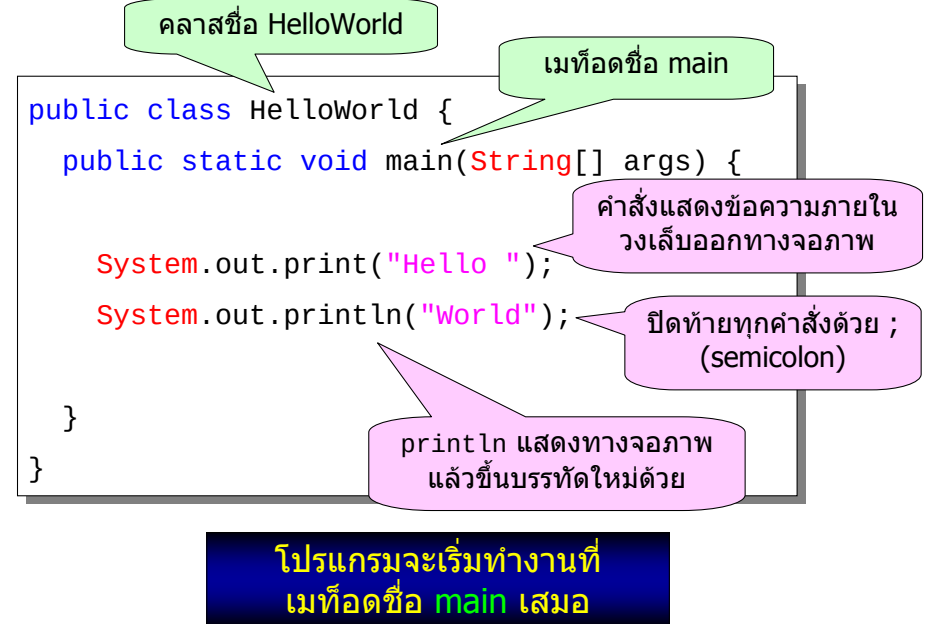


เพื่อความง่าย

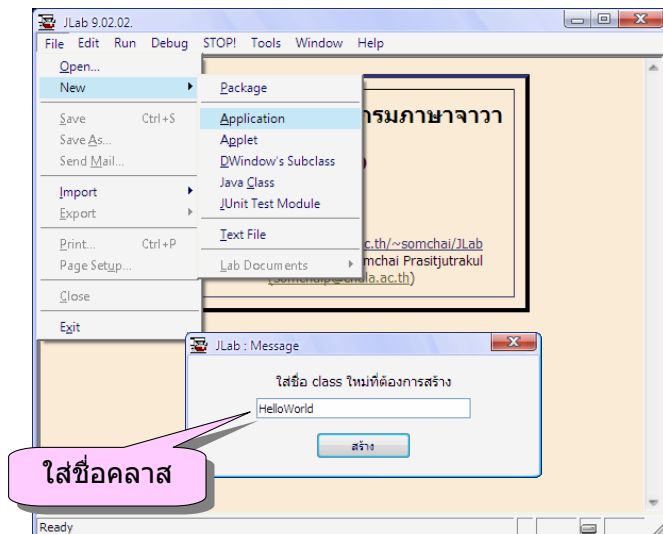
- โครงสร้างของภาคการศึกษา
 - หนึ่งโปรแกรม มีหนึ่งคลาส
 - หนึ่งคลาส มีหนึ่งเมทอด
 - หนึ่งเมทอด มีหลายคำสั่ง



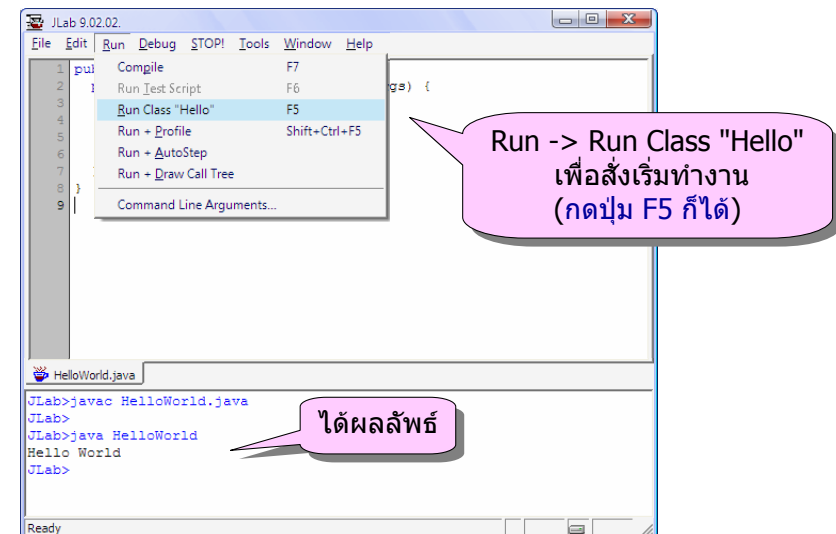
เริ่มด้วยโปรแกรมยอดนิยม : Hello World



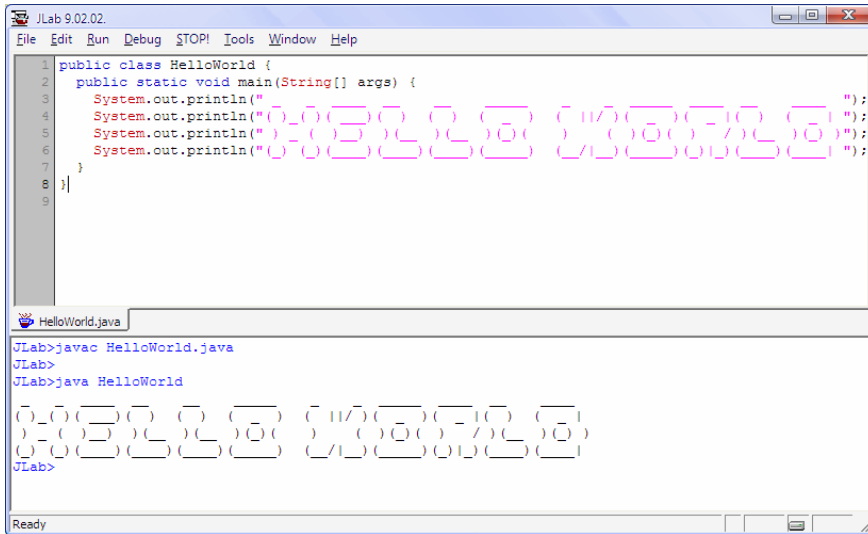
สร้างคลาส



เขียนและสั่งทำงาน



Hello World แบบพยายามให้สวย



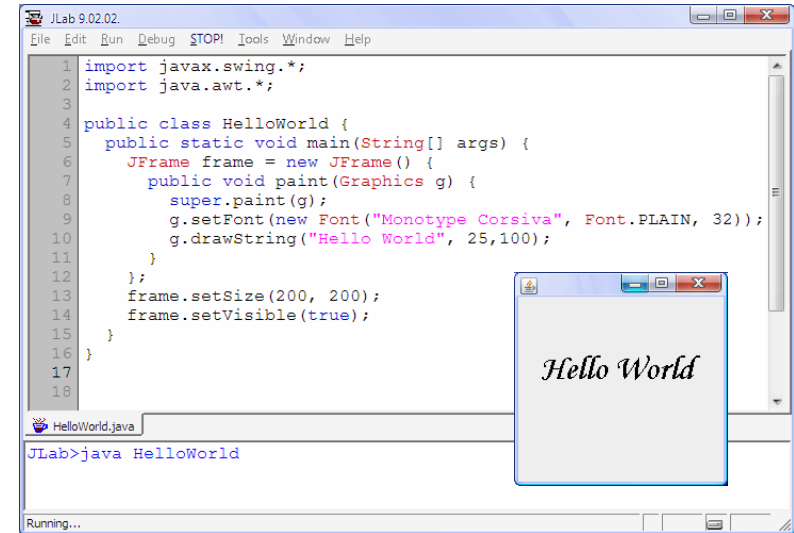
```
1 public class HelloWorld {
2     public static void main(String[] args) {
3         System.out.println(" ");
4         System.out.println("  _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ ");
5         System.out.println(" _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ ");
6         System.out.println(" _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ ");
7     }
8 }
9
```

JLab>javac HelloWorld.java
JLab>
JLab>java HelloWorld

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
 _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
 _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Ready

Hello World แบบวินโดว์



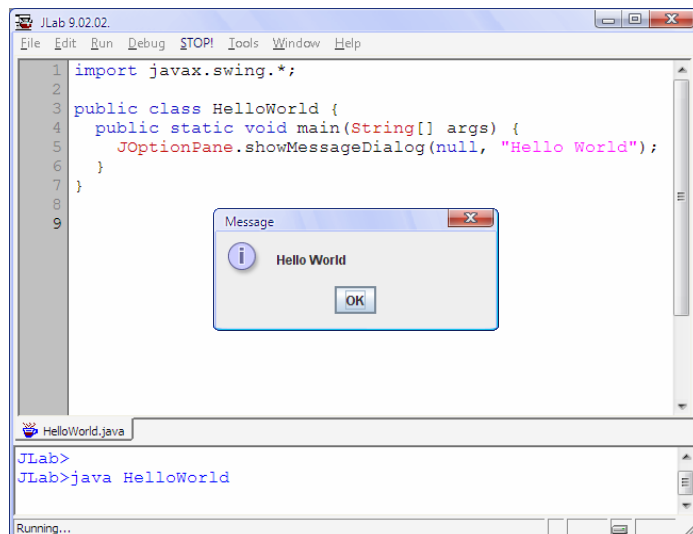
```
1 import javax.swing.*;
2 import java.awt.*;
3
4 public class HelloWorld {
5     public static void main(String[] args) {
6         JFrame frame = new JFrame() {
7             public void paint(Graphics g) {
8                 super.paint(g);
9                 g.setFont(new Font("Monotype Corsiva", Font.PLAIN, 32));
10                g.drawString("Hello World", 25,100);
11            }
12        };
13        frame.setSize(200, 200);
14        frame.setVisible(true);
15    }
16 }
17
18
```

JLab>java HelloWorld

Running...

Hello World

Hello World แบบ dialog box



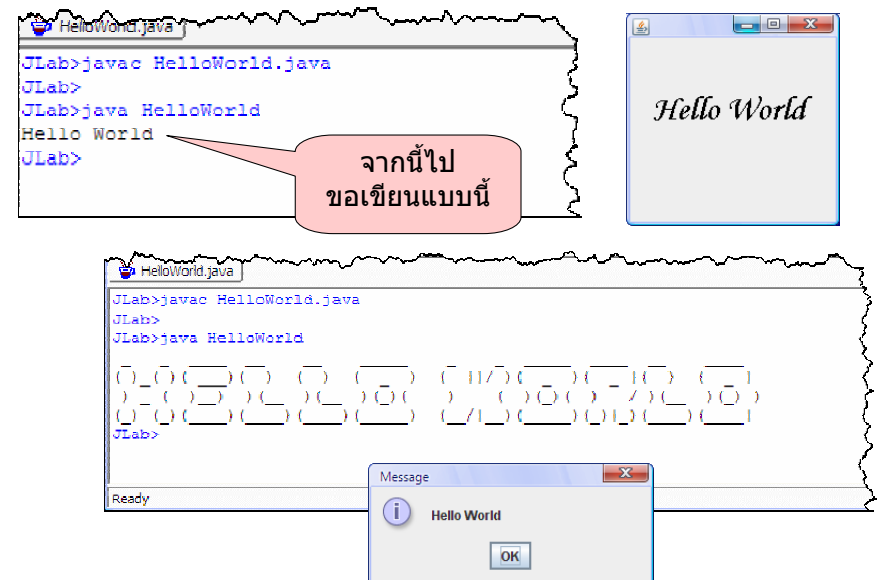
```
1 import javax.swing.*;
2
3 public class HelloWorld {
4     public static void main(String[] args) {
5         JOptionPane.showMessageDialog(null, "Hello World");
6     }
7 }
8
9
```

JLab>
JLab>java HelloWorld

Running...

Message
Hello World
OK

เพื่อความง่าย : ขอใช้แบบ System.out



```
1 import javax.swing.*;
2
3 public class HelloWorld {
4     public static void main(String[] args) {
5         JOptionPane.showMessageDialog(null, "Hello World");
6     }
7 }
8
9
```

JLab>javac HelloWorld.java
JLab>
JLab>java HelloWorld

Running...

Message
Hello World
OK

จากนี้ไป
ขอเขียนแบบนี้

อีกหลาย ๆ ตัวอย่าง

```
public class Example1 {
    public static void main(String[] args) {
        System.out.println("อดดา หิ อดดโน นาโถ");
        System.out.println("คนเป็นที่พึ่งแห่งตน");
    }
}
```

JLab>java Example1
อดดา หิ อดดโน นาโถ
คนเป็นที่พึ่งแห่งตน

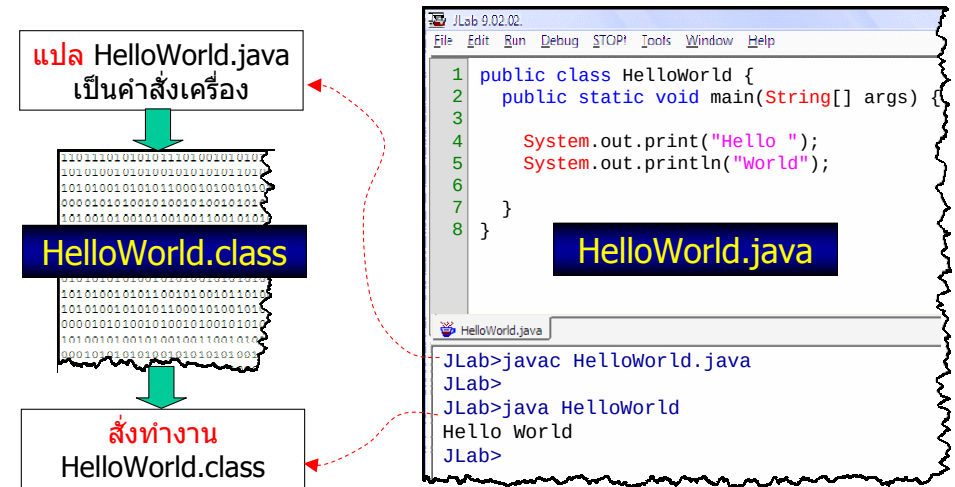
```
public class Example2 {
    public static void main(String[] args) {
        System.out.println(1+2+3+4+5+6+7+8+9);
    }
}
```

JLab>java Example2
45

```
public class Example3 {
    public static void main(String[] args) {
        System.out.println(new java.util.Date());
    }
}
```

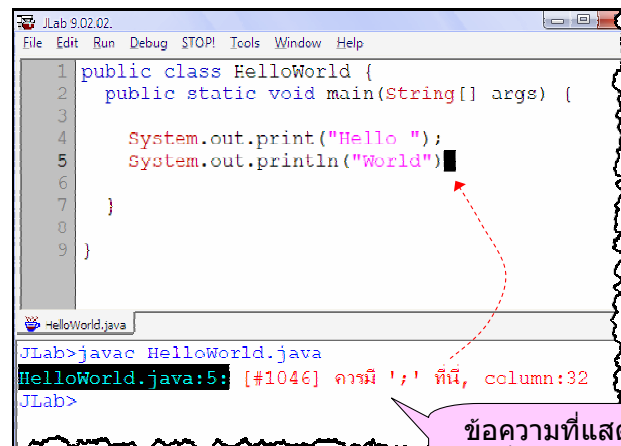
JLab>java Example3
Tue Feb 26 12:18:29 ICT 2008

ตัวแปลและตัวสั่งทำงาน (compiler & launcher)



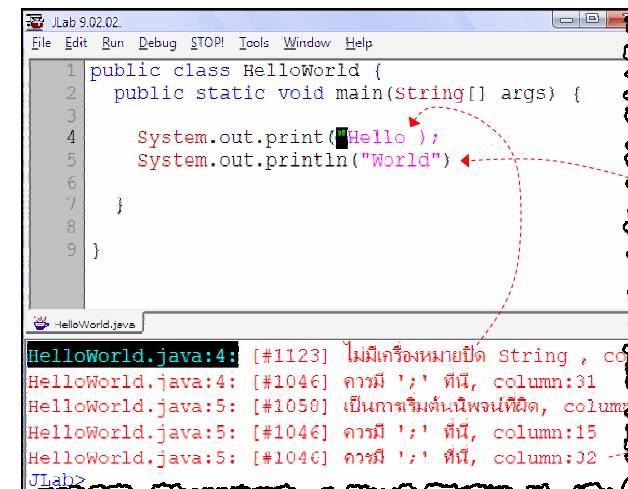
ใน JLab กด F5 เพื่อแปลและสั่งทำงาน

ถ้าเขียนผิดกฎ ก็แปลไม่ผ่าน



ข้อความที่แสดงอาจ
สื่อความหมาย

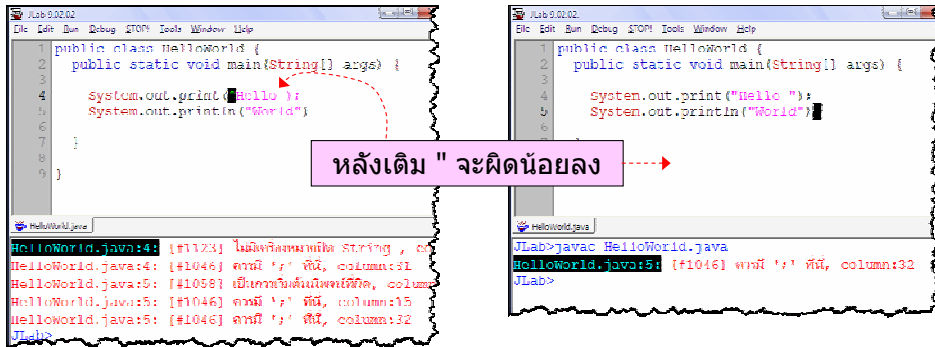
ถ้าเขียนผิดกฎ ก็แปลไม่ผ่าน



อาจง ถ้าผิดหลายที่

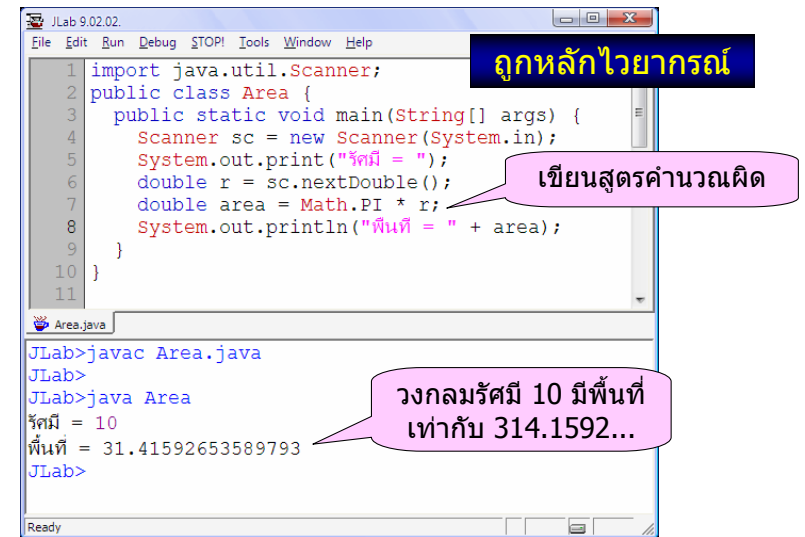
โปรแกรมต้องถูกหลักไวยากรณ์

- ภาษาโปรแกรมคอมพิวเตอร์ ไม่กำกวม
- ตัวแปลภาษาจู้จี้มาก ผิดนิดผิดหน่อยไม่ได้
- ต้องฝึกทักษะการตีความ error message
- โดยทั่วไป แก้ที่ผิดพลาดอันแรก ๆ แล้วลองสิ่งใหม่



โปรแกรมต้องทำงานถูกต้องตามข้อกำหนด

จงเขียนโปรแกรมแสดงพื้นที่ของวงกลมจากรัศมี
ที่ผู้ใช้กำหนดให้ทางแป้นพิมพ์



โปรแกรมต้องสวย อ่านเข้าใจง่าย

```
public class HelloWorld{public static void
main(String[] args){System.out.print("Hello ");
System.out.println
("World");
} }
```

สองโปรแกรมนี้อีกถูกหลักไวยากรณ์และทำงานเหมือนกัน
แต่โปรแกรมไหนอ่านง่ายกว่ากัน ?

```
public class HelloWorld {

    public static void main(String[] args) {
        System.out.print("Hello ");
        System.out.println("World");
    }

}
```

ใช้ช่องว่าง การย่อหน้า และการขึ้นบรรทัดใหม่
เพื่อให้โปรแกรมอ่านง่าย และอ่านง่ายขึ้น

สวยเกินไปอาจผิดกฎ

```
*****
* public class Hello {
*
*     PUBLIC STATIC VOID main(String[] args) {
*
*         System.out.print("Hello ");
*
*         System.out.println("2110101");
*
*     }
*
* }
*****
```

Comment คือข้อความที่เขียนให้คนอ่าน

```
import java.util.Scanner;
/*
 * โปรแกรมคำนวณพื้นที่จากรัศมีที่กำหนดให้ทางแป้นพิมพ์
 */
public class Area {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("รัศมี = ");
        double r = sc.nextDouble(); // อ่านรัศมีจากแป้นพิมพ์
        double area = Math.PI * r * r; // คำนวณพื้นที่
        System.out.println("พื้นที่ = " + area);
    }
}
```

comment แบบ
หลายบรรทัด

จาก // จนจบ
บรรทัดคือ
comment

- ข้อความระหว่างเครื่องหมาย /* กับ */ คือ comment
- ข้อความหลังเครื่องหมาย // จนจบบรรทัดคือ comment
- เขียน comment ให้คน (โปรแกรมเมอร์) อ่าน
- ตัวแปลภาษาไม่สนใจ comment

การใช้สีเพื่อให้อ่านง่ายขึ้น

- สีเขียว : comment
- สีน้ำเงิน : คำสงวนของภาษา
- สีแดง : คำที่พบในคลังคำสั่งมาตรฐาน
- สีแสด : อักขระที่อยู่ภายในเครื่องหมาย " ... "
- สีม่วง : อักขระที่อยู่ภายในเครื่องหมาย ' ... '
- สีดำ : อื่นๆ

ซอฟต์แวร์ที่ใช้เขียน
โปรแกรมจะให้สีอัตโนมัติ

```
// 2110101 : Computer Programming
public class HelloWorld {
    public static void main(final String[] args) {
        System.out.print("Hello ");
        System.out.print("World");
        System.out.print('A');
    }
}
```

สีช่วยสังเกตความผิดพลาด

```
public Class HelloWorld {
    public static void main(String[] args) {
        System.out.print("Hello");
        System.out.print("World");
        System.out.print('A');
    }
}
```

นำเป็นสีฟ้า - ต้องเป็นขีดตัวเล็ก

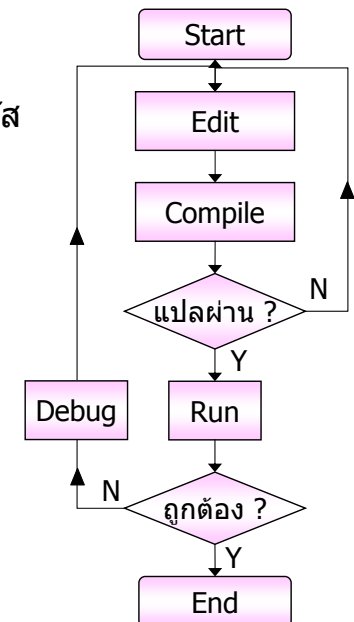
วงเล็บควรเป็นสีดำ
- ลืมใส่ " ปิด

นำเป็นสีฟ้า
- ต้องเป็น
ตัวแอลเล็ก

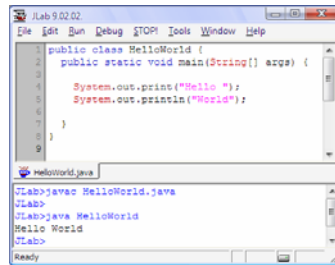
System นำเป็นสีแดง
- สะกดผิด

วงจรการพัฒนาโปรแกรม

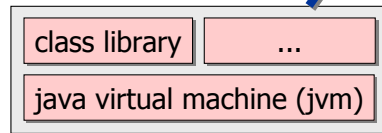
- Edit : เขียนโปรแกรมต้นฉบับ (source program, source code)
- Compile : แปล source code เป็นรหัสเครื่อง (byte code)
- Run : สิ่งทำงานรหัสเครื่องที่ได้
- Debug : หาตำแหน่งที่เราเขียนแล้วทำงานผิด (จะสาธิตให้ชมในอนาคต)
- ใช้ซอฟต์แวร์ช่วยพัฒนาโปรแกรมใน "Edit-Compile-Run-Debug"
- สำหรับวิชานี้เราจะใช้ JLab เป็นซอฟต์แวร์ในการช่วยพัฒนาโปรแกรม



JLab & Java

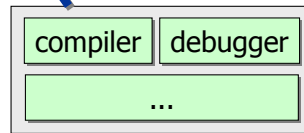


JLab
ช่วยเราพัฒนาโปรแกรม
ภาษาจาวาได้สะดวกขึ้น
(edit-compile-run-debug)



Java Runtime Environment (JRE)

ให้ซอฟต์แวร์ที่เขียนด้วยภาษาจาวาทำงานได้



Java Development Kit (JDK)

ให้เราพัฒนาโปรแกรมภาษาจาวาได้



การติดตั้ง JLab และ Java

- ในแผ่น CD-ROM มีตัวติดตั้ง JLab และ Java
- ใส่แผ่น ระบบจะเริ่มติดตั้งโดยอัตโนมัติ
- ในกรณีที่เครื่องไม่มีจาวา จะติดตั้งจาวาเพิ่มให้ด้วย
(การติดตั้งจาวาใช้เวลาหลายนาที รอน้อย)



JLab

<http://java.sun.com>
<http://www.cp.eng.chula.ac.th/~somchai/JLab>

สรุป

- หนึ่งโปรแกรมมีหลายคลาส
- หนึ่งคลาสมีหลายเมทอด
- หนึ่งเมทอดมีหลายคำสั่ง
- โปรแกรมจาวาเริ่มทำงานที่เมทอดชื่อ main
- การทำงานจะทำทีละคำสั่งตามลำดับจากบนลงล่าง
- โปรแกรมต้องถูกหลักไวยากรณ์
- โปรแกรมต้องทำงานถูกต้องตามข้อกำหนด
- โปรแกรมต้องอ่านเข้าใจได้ง่าย ๆ
- วงจรการพัฒนาโปรแกรม
edit – compile – run – debug

2110101 : การทำโปรแกรมคอมพิวเตอร์

จำนวน ตัวแปร และตัวปฏิบัติการพื้นฐาน

ภาควิชาวิศวกรรมคอมพิวเตอร์
จุฬาลงกรณ์มหาวิทยาลัย

หัวข้อ

- สิ่งที่ต้องรู้ก่อนทำโปรแกรม
- ประเภทข้อมูล
 - จำนวนเต็ม
 - จำนวนจริง
- ตัวแปร
 - การประกาศ
 - กฎการตั้งชื่อ
- ตัวปฏิบัติการ + - * / %
- นิพจน์

System.out.println

```
public class Print {  
    public static void main(String[] args) {  
        System.out.print("x = ");  
        System.out.println(2.5);  
        System.out.println("x = " + 2.5);  
        System.out.print("a = " + 1);  
        System.out.println(", b = " + 2);  
        System.out.println("a = " + 1 + ", b = " + 2);  
    }  
}
```

```
JLab>java Print  
x = 2.5  
x = 2.5  
a = 1, b = 2  
a = 1, b = 2
```

"x = "

เรียกว่าสตริง (String) แทนข้อความ

"x " + "= "

เครื่องหมาย + กับสตริง คือ การต่อสตริง

บวก ลบ คูณ หาร

```
public class Operator1 {  
    public static void main(String[] args) {  
        System.out.println(4 + 2);  
        System.out.println(4 - 2);  
        System.out.println(4 * 2);  
        System.out.println(4 / 2);  
    }  
}
```

```
JLab>java Operator1  
6  
2  
8  
2
```

```
public class Operator2 {  
    public static void main(String[] args) {  
        System.out.println("4 + 2 = " + (4 + 2) );  
        System.out.println("4 - 2 = " + (4 - 2) );  
        System.out.println("4 * 2 = " + (4 * 2) );  
        System.out.println("4 / 2 = " + (4 / 2) );  
    }  
}
```

```
JLab>java Operator2  
4 + 2 = 6  
4 - 2 = 2  
4 * 2 = 8  
4 / 2 = 2
```

ต้องใส่วงเล็บ

บวก ลบ คูณ หาร

```
public class Operator3 {
    public static void main(String[] args) {
        System.out.println("1+2+...+5 = " + (1+2+3+4+5) );
        System.out.println("1+2+...+5 = " + ((5*6)/2) );
    }
}
```

```
JLab>java Operator3
1+2+...+5 = 15
1+2+...+5 = 15
```

```
public class Operator4 {
    public static void main(String[] args) {
        System.out.println("วงกลมที่มีรัศมี 4 หน่วย มีพื้นที่ " +
            (3.14159*4*4));
    }
}
```

```
JLab>java Operator4
วงกลมที่มีรัศมี 4 หน่วย มีพื้นที่ 50.26544
```

ค่าส่งยาวเกินไป ขึ้นบรรทัดใหม่ได้

จำนวนเต็ม – จำนวนจริง

```
public class IntegerAndRealNumber {
    public static void main(String[] args) {
        System.out.println("5/2 = " + (5/2) );
        System.out.println("5/2.0 = " + (5/2.0) );
        System.out.println("5.0/2 = " + (5.0/2) );
        System.out.println("5.0/2.0 = " + (5.0/2.0) );
        System.out.println("1.0/3.0 = " + (1.0/3.0) );
    }
}
```

```
JLab>java IntegerAndRealNumber
5/2 = 2
5/2.0 = 2.5
5.0/2 = 2.5
5.0/2.0 = 2.5
1.0/3.0 = 0.3333333333333333
```

- นำจำนวนเต็มมาคำนวณ จะตัดเศษทิ้ง
- ถ้ามีจำนวนที่มีจุดทศนิยมในการคำนวณ จะไม่ตัดเศษ

ตัวแปร (Variables) – ที่เก็บข้อมูลใน RAM

```
public class Variable1 {
    public static void main(String[] args) {
        int a; // ประกาศตัวแปรจำนวนเต็มชื่อ a
        double y; // ประกาศตัวแปรจำนวนจริงชื่อ y
        String s; // ประกาศตัวแปรสตริงชื่อ s
        a = -12; y = 19.5; s = "a = ";
        System.out.println(s + a);
        System.out.println("y = " + y);
    }
}
```

```
a -12
y 19.5
s "a = "
```

```
JLab>java Variable1
a = -12
y = 19.5
```

- ต้องประกาศตัวแปรก่อนที่จะนำมาใช้
- ตัวแปรทุกตัวต้องมีประเภทข้อมูลและชื่อกำกับ
- ตัวแปรมีหลายประเภท
 - int จำนวนเต็ม
 - double จำนวนจริง
 - String สตริง (ข้อความ)
 - อื่น ๆ ...



double y;

การประกาศตัวแปร

- `int i;` ประกาศตัวแปร int ชื่อ i
- `int i, j, k, a;` ประกาศหลายตัวแปรก็ได้ (คั่นด้วย comma)
- `int i = 0;` ประกาศพร้อมกับให้ค่าเริ่มต้นก็ได้
- `int i = 0, j = 8;` ประกาศและให้ค่าเริ่มต้นหลายตัวก็ได้
- `int i, j, k;`
`int k;` ประกาศซ้ำไม่ได้
- `int i, double x;` ประกาศคนละประเภท ต้องแยกคำสั่ง
- `int i;`
`Double x;`
`string s;` `int` – ตัวเล็กหมด, `double` – ตัวเล็กหมด
`String` – S ตัวใหญ่
สังเกตสี `int` และ `double` สีฟ้า `String` สีแดง
- `k = 27;`
`int k;` ต้องประกาศตัวแปร ก่อนใช้งาน

กฎการตั้งชื่อตัวแปร (และชื่อคลาส)

- ชื่อประกอบด้วยตัวอักษร ตัวเลข ตัว \$ หรือ _ ก็ได้
- ชื่อห้ามขึ้นต้นด้วยตัวเลข
- ชื่อยาวๆ ได้ไม่เป็นไร
- ตัวอักษรตัวใหญ่ไม่เหมือนตัวเล็ก
- ต้องไม่ซ้ำกับคำสงวนของภาษาจาวา
- ต้องไม่ซ้ำกับชื่ออื่นๆ ของโปรแกรมที่ใช้มาก่อน

ตัวอย่างถูก

```
int17 power Public radius area day_of_week
```

ตัวอย่างผิด

```
7zean String public ohOH! ed-edd-n-eddy
```

คำสงวนในภาษาจาวา

<u>abstract</u>	<u>assert</u>	<u>boolean</u>	<u>break</u>	<u>byte</u>
<u>case</u>	<u>catch</u>	<u>char</u>	<u>class</u>	<u>const</u>
<u>continue</u>	<u>default</u>	<u>do</u>	<u>double</u>	<u>else</u>
<u>extends</u>	<u>false</u>	<u>final</u>	<u>finally</u>	<u>float</u>
<u>for</u>	<u>goto</u>	<u>if</u>	<u>implements</u>	<u>import</u>
<u>instanceof</u>	<u>int</u>	<u>interface</u>	<u>long</u>	<u>native</u>
<u>new</u>	<u>null</u>	<u>package</u>	<u>private</u>	<u>protected</u>
<u>public</u>	<u>return</u>	<u>short</u>	<u>static</u>	<u>strictfp</u>
<u>super</u>	<u>switch</u>	<u>synchronized</u>	<u>this</u>	<u>throw</u>
<u>throws</u>	<u>transient</u>	<u>true</u>	<u>try</u>	<u>void</u>
<u>volatile</u>	<u>while</u>			

ห้ามนำคำสงวนมาตั้งเป็นชื่อตัวแปร
(เราจะศึกษาเฉพาะที่ขีดเส้นใต้)

ธรรมเนียมนิยมในการตั้งชื่อ

- มักขึ้นต้นชื่อตัวแปรด้วยตัวอักษรตัวเล็ก (ชื่อคลาสมักขึ้นต้นด้วยตัวใหญ่)
- หลีกเลี่ยงการใช้ _ และ \$
- สื่อความหมาย
- ถ้าชื่อประกอบด้วยคำหลายคำ เขียนแบบ "หลังอูฐ"

```
numStudents dayOfWeek imageBufferWithAlphaChannel
```

```
tvChannel customerID initialVelocity
```

imageBufferWithAlphaChannel

การให้ค่ากับตัวแปร และการคำนวณ

```
public class Circle {
    public static void main(String[] args) {
        String s1 = "เส้นรอบวง = ";
        String s2 = "พื้นที่ = ";
        double radius = 5;
        double circumference, area;
        circumference = 2 * 3.14159 * radius;
        area = 3.14159 * radius * radius;
        System.out.println(s1 + circumference);
        System.out.println(s2 + area);
    }
}
```

```
JLab>java Circle
เส้นรอบวง = 31.4159
พื้นที่ = 78.53975
```

เครื่องหมาย = คือการนำค่าทางขวา ให้กับตัวแปรทางซ้าย

ค่าที่ให้กับตัวแปรใด ต้องเป็นประเภทเดียวกับตัวแปรนั้น
(ยกเว้นว่า แปลงค่าได้แน่ ๆ `double ← int`)

ตัวอย่าง

`x = y + 2;` ให้ x มีค่าเท่ากับ y + 2

`x = x + 1;` เพิ่มค่าของตัวแปร x อีก 1 $x \leftarrow x + 1$

`x = 2 * x;` เพิ่มค่าของตัวแปร x อีกเท่าตัว

`x = 0 - x;` `x = -1 * x;` `x = -x;` เหมือนกันทั้งสามคำสั่ง

`dx = x1 - x2;`
`dy = y1 - y2;`
`len2 = dx*dx + dy*dy;` $len2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$

`double d = 1234;` นำจำนวนเต็มใส่ในตัวแปรจำนวนจริงได้

ตัวอย่างผิดๆ

`5 = x;` `x + 2 = x;` `x = 2a + 8;`

`String s = 5;` `int k = "4";` `int k = 1.5;`

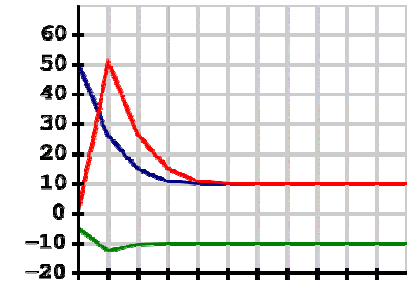
ตัวอย่าง : การหารากที่สอง

- โจทย์ : ต้องการหารากที่สองของ a
- ใช้ Babylonian method

$$x_0 \approx \sqrt{a}$$

$$x_k = \frac{1}{2} \left(x_{k-1} + \frac{a}{x_{k-1}} \right)$$

$$\sqrt{a} = \lim_{n \rightarrow \infty} x_n$$



a = 100.0
 $x_0 = -5.0$
 $x_1 = -12.5$
 $x_2 = -10.25$
 $x_3 = -10.00$

a = 100.0
 $x_0 = 50.0$
 $x_1 = 26.0$
 $x_2 = 14.92$
 $x_3 = 10.81$
 $x_4 = 10.03$
 $x_5 = 10.00$

a = 100.0
 $x_0 = 1.0$
 $x_1 = 50.5$
 $x_2 = 26.24$
 $x_3 = 15.03$
 $x_4 = 10.84$
 $x_5 = 10.03$
 $x_6 = 10.00$

โปรแกรม : การหารากที่สอง

```
public class SquareRoot{
    public static void main(String[] args) {
```

```
double a = 100.0;
double x0 = 1.0;
double x1 = (x0 + (a / x0)) / 2.0;
double x2 = (x1 + (a / x1)) / 2.0;
double x3 = (x2 + (a / x2)) / 2.0;
double x4 = (x3 + (a / x3)) / 2.0;
double x5 = (x4 + (a / x4)) / 2.0;
double x6 = (x5 + (a / x5)) / 2.0;
System.out.println("รากที่สองของ " + a + " = " + x6);
}
```

JLab>java SquareRoot
 รากที่สองของ 100.0 = 10.000000000107445

$$x_k = \frac{1}{2} \left(x_{k-1} + \frac{a}{x_{k-1}} \right)$$

โปรแกรม : การหารากที่สอง (ปรับปรุง)

```
double a = 100;
double x0 = 1;
double x1 = (x0 + (a / x0)) / 2.0;
double x2 = (x1 + (a / x1)) / 2.0;
double x3 = (x2 + (a / x2)) / 2.0;
double x4 = (x3 + (a / x3)) / 2.0;
double x5 = (x4 + (a / x4)) / 2.0;
double x6 = (x5 + (a / x5)) / 2.0;
```

ใช้ตัวแปร 8 ตัว

```
double a = 100;
double x0 = 1, x1;
x1 = (x0 + (a / x0)) / 2.0;
x0 = (x1 + (a / x1)) / 2.0;
x1 = (x0 + (a / x0)) / 2.0;
x0 = (x1 + (a / x1)) / 2.0;
x1 = (x0 + (a / x0)) / 2.0;
x0 = (x1 + (a / x1)) / 2.0;
```

ใช้ตัวแปร 3 ตัว

```
double a = 100;
double x = 1;
x = (x + (a / x)) / 2.0;
x = (x + (a / x)) / 2.0;
x = (x + (a / x)) / 2.0;
x = (x + (a / x)) / 2.0;
x = (x + (a / x)) / 2.0;
x = (x + (a / x)) / 2.0;
```

ใช้ตัวแปร 2 ตัว

x ได้รับค่าใหม่

ค่าของ x ปัจจุบัน

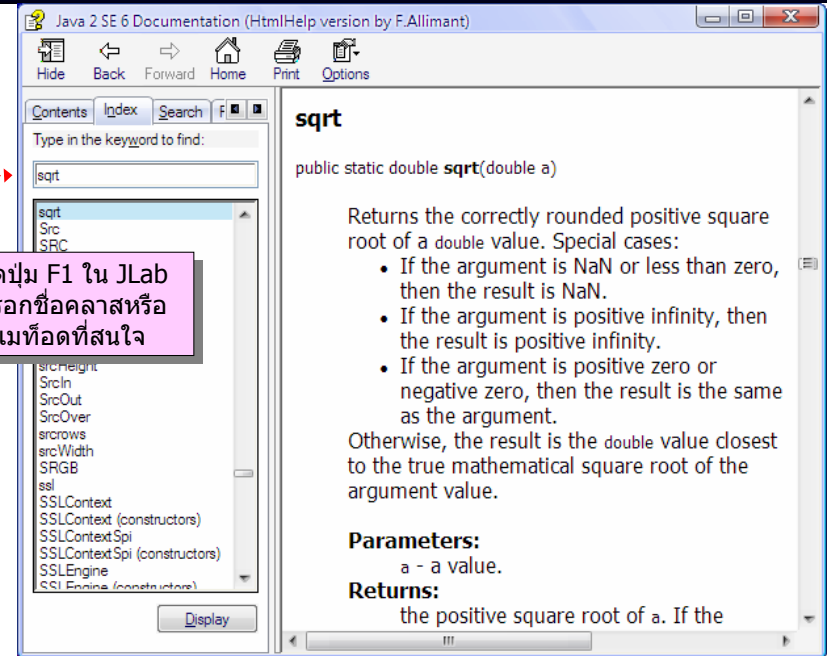
คลาสมาตรฐาน Math

- จาวามีคลาส **Math** ที่มีฟังก์ชันทางคณิตศาสตร์มากมาย เช่น

```
double s, t;
double radian, degree;
radian = Math.toRadians(90); // เปลี่ยนองศาเป็นเรเดียน
degree = Math.toDegrees(radian); // เปลี่ยนเรเดียนเป็นองศา
t = Math.sin(radian); // ฟังก์ชันไซน์ (sine)
s = Math.cos(radian); // ฟังก์ชันโคไซน์ (cosine)
s = Math.sqrt(100.0); // ฟังก์ชันหารากที่สอง
s = Math.abs(s - t); // ฟังก์ชันหาค่าสัมบูรณ์ (absolute)
s = Math.max(s, t); // คืนค่ามากกว่าระหว่าง s กับ t
s = Math.min(s, t); // คืนค่าน้อยกว่าระหว่าง s กับ t
s = Math.random(); // คืนค่าสุ่มระหว่าง 0.0 ถึง 1.0 [0.0, 1.0)
```



อ่านรายละเอียดของคลาสมาตรฐานจาก Help



การรับข้อมูลจากแป้นพิมพ์

```
import java.util.Scanner;

public class Keyboard {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);

        System.out.print("a = ");
        double a = kb.nextDouble();
        System.out.print("b = ");
        double b = kb.nextDouble();

        double max = Math.max(a,b);
        System.out.println("ตัวมาก = " + max);
    }
}
```

ให้โปรแกรมรู้จักคลาส Scanner

สร้างตัวอ่านจากแป้นพิมพ์

แสดงข้อความให้ผู้ใช้ทราบ

รอรับข้อมูลแบบ double

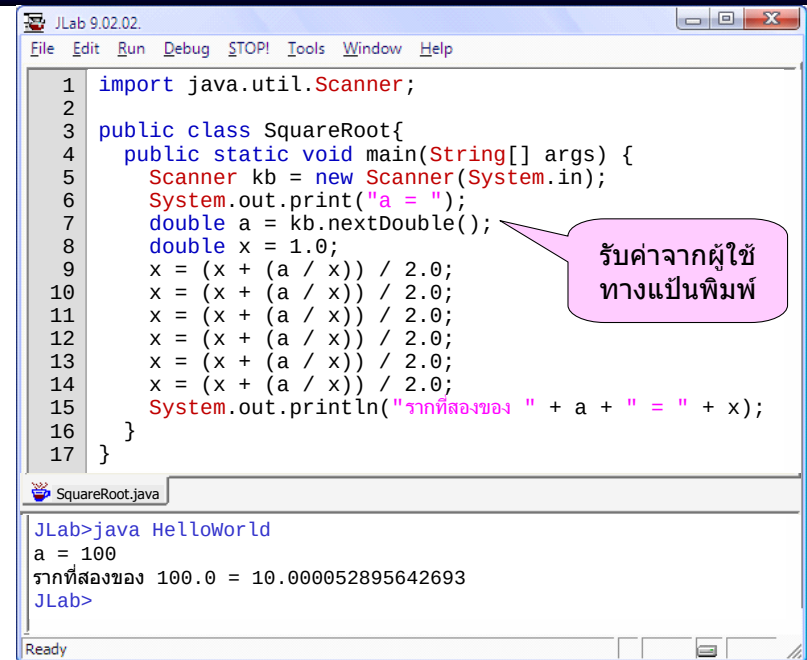
นี่คือตัวเลขที่ผู้ใช้ป้อนทางแป้นพิมพ์

จำ kb.nextDouble() อ่านจำนวนจริง
 kb.nextInt() อ่านจำนวนเต็ม
 kb.nextLine() อ่านสตริง

JLab>java Keyboard

a = 10
 b = 8.2
 ตัวมาก = 10.0

ตัวอย่าง : การหารากที่สอง (รับจากแป้นพิมพ์)



สาธิตการใช้ debugger

```

1 import java.util.Scanner;
2
3 public class SquareRoot{
4     public static void main(String[] args) {
5         Scanner kb = new Scanner(System.in);
6         System.out.print("a = ");
7         double a = kb.nextDouble();
8         double x = 1.0;
9         x = (x + (a / x)) / 2.0;
10        x = (x + (a / x)) / 2.0;
11        x = (x + (a / x)) / 2.0;
12        x = (x + (a / x)) / 2.0;
13        x = (x + (a / x)) / 2.0;
14        x = (x + (a / x)) / 2.0;
15        System.out.println("รากที่สองของ " + a + " = " + x);
16    }
17 }

```

ตัวอย่าง : การหารากของสมการกำลังสอง

• โจทย์ : ต้องการหารากสมการ $ax^2 + bx + c = 0$

• ใช้สูตร $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

• วิเคราะห์

- รับค่า a, b และ c จากผู้ใช้ทางแป้นพิมพ์
- แสดงรากของสมการทั้งสองค่าทางจอภาพ

$r1 = ((-b) + \text{Math.sqrt}((b*b) - (4*a*c))) / (2*a);$

$$r_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

โปรแกรม : การหารากของสมการกำลังสอง

```

import java.util.Scanner;
public class QuadraticRoot{
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("a = ");
        double a = kb.nextDouble();
        System.out.print("b = ");
        double b = kb.nextDouble();
        System.out.print("c = ");
        double c = kb.nextDouble();
        double r1 = ((-b) + Math.sqrt((b*b) - (4*a*c)))/(2*a);
        double r2 = ((-b) - Math.sqrt((b*b) - (4*a*c)))/(2*a);
        System.out.print("รากของสมการ " +
            a + "x^2 + " + b + "x + " + c + " คือ");
        System.out.println(r1 + ", " + r2);
    }
}

```

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

JLab>java QuadraticRoot

a = 1
b = -5
c = 6

รากของสมการ $1.0x^2 + -5.0x + 6.0$ คือ 3.0, 2.0

การใช้ตัวแปร

1. ตัวแปรเพื่อเก็บข้อมูลขาเข้า (input)
2. ตัวแปรเพื่อเก็บผลลัพธ์ (output)
3. ตัวแปรเพื่อเก็บข้อมูลชั่วคราวระหว่างการประมวลผล

```

Scanner kb = new Scanner(System.in);
System.out.print("a = ");
double a = kb.nextDouble();
System.out.print("b = ");
double b = kb.nextDouble();
System.out.print("c = ");
double c = kb.nextDouble();
double t = Math.sqrt((b*b) - (4*a*c));
double r1 = ((-b) + t) / (2*a);
double r2 = ((-b) - t) / (2*a);

```

```

double r1 = ((-b) + Math.sqrt((b*b) - (4*a*c)))/(2*a);
double r2 = ((-b) - Math.sqrt((b*b) - (4*a*c)))/(2*a);

```

ลำดับการคำนวณ

$$\frac{9c}{5} + 32$$

นิพจน์ (expression)

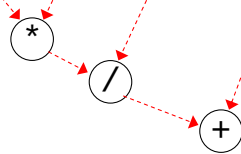
$$((9.0 * c) / 5.0) + 32.0$$

* กับ / ทำจากซ้ายไปขวา

$$(9.0 * c / 5.0) + 32.0$$

* กับ / ทำก่อน + กับ -

$$9.0 * c / 5.0 + 32.0$$



ลำดับการคำนวณ

$$\frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

นิพจน์ (expression)

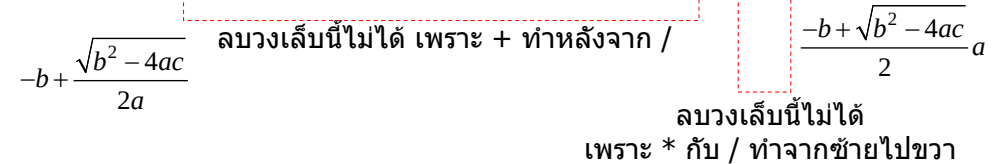
$$((-b) + \text{Math.sqrt}((b*b) - (4*a*c)))/(2*a)$$

ติดลบทำก่อน

$$(-b + \text{Math.sqrt}((b*b) - (4*a*c)))/(2*a)$$

* กับ / ทำก่อน + กับ -

$$(-b + \text{Math.sqrt}(b*b - 4*a*c))/(2*a)$$



สรุปลำดับการคำนวณ

- ทำในวงเล็บทำก่อน : $-7 + 2 * \text{Math.sqrt}(5+4)$
- การเรียกใช้ method : $-7 + 2 * \text{Math.sqrt}(9)$
- ติดลบ : $-7 + 2 * 3$
- คูณหาร * / : $-7 + 2 * 3$
- บวก ลบ + - : $-7 + 6$
- = : -1

ในกรณีมี operators
หลายตัวในระดับเดียวกัน
ให้ทำจากซ้ายไปขวา

$$8 / 4 * 2$$

$$2 * 2$$

$$4$$

โจทย์ : ดัชนีมวลกาย (Body Mass Index)

- เขียนโปรแกรมรับข้อมูลทางแป้นพิมพ์
 - น้ำหนัก (หน่วยเป็นกิโลกรัม)
 - ความสูง (หน่วยเป็นเซนติเมตร)
- แสดงดัชนีมวลกาย ซึ่งคำนวณได้จากสูตร

$$BMI = \frac{\text{weight(kg.)}}{\text{height}^2(\text{m}^2)}$$

- หมายเหตุ : หุ่นดีต้องมี BMI ระหว่าง 18.5 – 25
น้อยไปผอม มากไปอ้วน

โปรแกรม : ดัชนีมวลกาย

```
import java.util.Scanner;

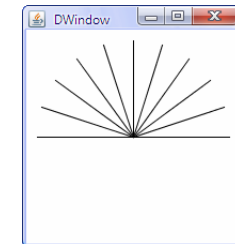
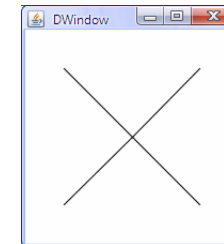
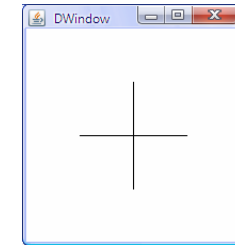
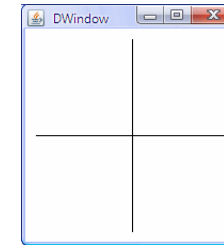
public class BMI {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("น้ำหนัก (kg.) = ");
        double weight = kb.nextDouble();
        System.out.print("ความสูง (m.) = ");
        double height = kb.nextDouble();

        double bmi = 
        System.out.println("ดัชนีมวลกาย = " + bmi);
    }
}
```

JLab>java BMI
 น้ำหนัก (kg.) = 65
 ความสูง (m.) = 173
 ดัชนีมวลกาย = 21.718066089745733

ตัวอย่าง : ลากเส้น

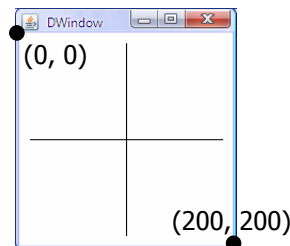
- ใช้คลาสพิเศษของ jlab ชื่อ DWindow ในการวาดรูป



ตัวอย่าง : ลากสองขีด

```
import jlab.graphics.DWindow;

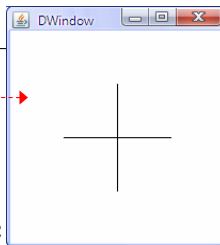
public class LineDrawing1 {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        w.drawLine(100, 10, 100, 190);
        w.drawLine(10, 100, 190, 100);
    }
}
```



```
import jlab.graphics.DWindow;

public class LineDrawing2 {
    public static void main(String[] args) {
        double width = 200, height = 200, gap = 10;
        DWindow w = new DWindow(width, height);
        w.drawLine(width / 2, gap, width / 2, height - gap);
        w.drawLine(gap, height / 2, width - gap, height / 2);
    }
}
```

เปลี่ยน gap = 50



การใช้ตัวแปร ทำให้ปรับแก้โปรแกรมได้ง่าย

ตัวอย่าง : ลากสองขีด – สีขีด

```
import jlab.graphics.DWindow;

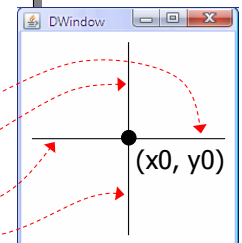
public class LineDrawing2 {
    public static void main(String[] args) {
        double width = 200, height = 200, gap = 10;
        DWindow w = new DWindow(width, height);
        w.drawLine(width / 2, gap, width / 2, height - gap);
        w.drawLine(gap, height / 2, width - gap, height / 2);
    }
}
```

สองขีด

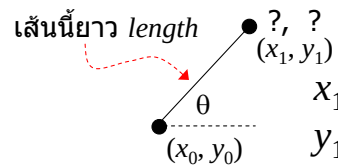
```
import jlab.graphics.DWindow;

public class LineDrawing3 {
    public static void main(String[] args) {
        double width = 200, height = 200, gap = 10;
        DWindow w = new DWindow(width, height);
        double x0 = width / 2, y0 = height / 2;
        w.drawLine(x0, y0, width - gap, y0);
        w.drawLine(x0, y0, x0, gap);
        w.drawLine(x0, y0, gap, y0);
        w.drawLine(x0, y0, x0, height - gap);
    }
}
```

สี่ขีด



ตัวอย่าง : ลากเส้นคำนวณจากจุดและมุม

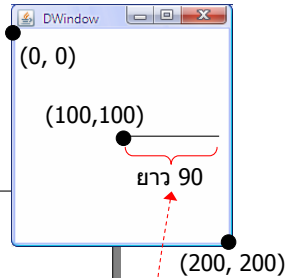


$$x_1 = x_0 + \text{length} \cdot \cos(\theta)$$

$$y_1 = y_0 - \text{length} \cdot \sin(\theta)$$

```
import jlab.graphics.DWindow;
```

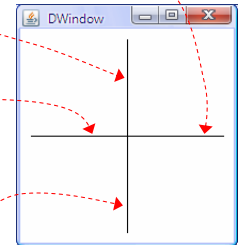
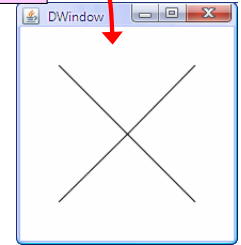
```
public class LineDrawing4 {
    public static void main(String[] args) {
        int width = 200, height = 200, len = 90;
        DWindow w = new DWindow(width, height);
        double x0 = width / 2, y0 = height / 2;
        double x1, y1;
        double angle = 0;
        //-----
        x1 = x0 + len * Math.cos(Math.toRadians(angle));
        y1 = y0 - len * Math.sin(Math.toRadians(angle));
        w.drawLine(x0, y0, x1, y1);
        ...
    }
}
```



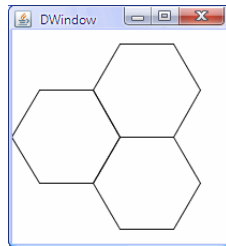
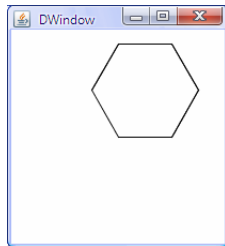
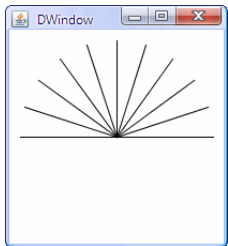
ตัวอย่าง : ลาก 4 เส้น (คำนวณจากจุดและมุม)

ถ้าเปลี่ยนมุมเริ่มต้นเป็น 45 องศา

```
int width = 200, height = 200, len = 90;
DWindow w = new DWindow(width, height);
double x0 = width / 2, y0 = height / 2;
double x1, y1, angle = 0, delta = 90;
//-----
x1 = x0 + len * Math.cos(Math.toRadians(angle));
y1 = y0 - len * Math.sin(Math.toRadians(angle));
w.drawLine(x0, y0, x1, y1);
angle = angle + delta;
x1 = x0 + len * Math.cos(Math.toRadians(angle));
y1 = y0 - len * Math.sin(Math.toRadians(angle));
w.drawLine(x0, y0, x1, y1);
angle = angle + delta;
x1 = x0 + len * Math.cos(Math.toRadians(angle));
y1 = y0 - len * Math.sin(Math.toRadians(angle));
w.drawLine(x0, y0, x1, y1);
angle = angle + delta;
x1 = x0 + len * Math.cos(Math.toRadians(angle));
y1 = y0 - len * Math.sin(Math.toRadians(angle));
w.drawLine(x0, y0, x1, y1);
```



โจทย์ : จงเขียนโปรแกรมเพื่อแสดงรูปข้างล่างนี้



2110101 : การทำโปรแกรมคอมพิวเตอร์

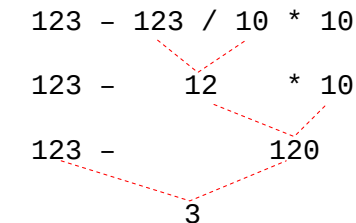
ตัวปฏิบัติการและจำนวน (เพิ่มเติม) while(true)

ภาควิชาวิศวกรรมคอมพิวเตอร์
จุฬาลงกรณ์มหาวิทยาลัย

ตัวอย่าง : หลักหน่วย

```
import java.util.Scanner;

public class UnitDigit {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("จำนวนเต็ม = ");
        int n = kb.nextInt();
        int unitDigit = n - n / 10 * 10;
        System.out.println("หลักหน่วย = " + unitDigit);
    }
}
```

$$u = n - 10 \left\lfloor \frac{n}{10} \right\rfloor$$


ตัวอย่าง : เศษจากการหาร (modulo) %

```
import java.util.Scanner;

public class Remainder {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("a = ");
        int a = kb.nextInt();
        System.out.print("b = ");
        int b = kb.nextInt();
        int q = a / b;
        int r = a - a / b * b;
        System.out.println("a/b = " + q);
        System.out.println("เศษของ a/b = " + r);
    }
}
```

$$r = a - b \left\lfloor \frac{a}{b} \right\rfloor$$

เรียก % ว่า modulo operator
ใช้ได้กับทั้ง int และ double

5 % 2	ได้	1
2008 % 4	ได้	0
5.5 % 2	ได้	1.5
5.5 % 1	ได้	0.5

ตัวอย่าง : แยกเลขโดดออกจากจำนวนเต็ม

- รับจำนวนเต็มจากผู้ใช้ทางแป้นพิมพ์
- แสดงเลขหลักสิบและหลักพันของจำนวนที่รับเข้ามา

สนใจหลักสิบ 41829 4182 2
 ↓/10 ↓%10

```
import java.util.Scanner;

public class Digits {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("จำนวนเต็ม = ");
        int n = kb.nextInt();
        int d10 = (n / 10) % 10;
        int d1000 = (n / 1000) % 10;
        System.out.println("หลักสิบ = " + d10);
        System.out.println("หลักพัน = " + d1000);
    }
}
```

ตัวปฏิบัติการพิเศษ : ++ --

- **a++** เพิ่มจำนวนในตัวแปร a ขึ้นอีก 1
- **++a** " "
- **a--** ลดจำนวนในตัวแปร a ลงอีก 1
- **--a** " "
- อยู่หน้า, อยู่หลัง มีข้อแตกต่างเล็กน้อย

```
int a = 10, b = 5;
a++; // a เพิ่มขึ้น 11
--b; // b ลดเป็น 4
int c = (a++) + (--b);

int d = (++a) + (b--);

(a++)++; // ผิด เขียนไม่ได้
```

ลดค่าก่อนแล้วค่อยนำไปใช้

```
--b;
int c = a + b;
a++;
```

ใช้ก่อนแล้วค่อยเพิ่มค่า

```
++a;
int c = a + b;
b--;
```

จำนวนเต็ม – จำนวนจริง

- int และ double เป็นข้อมูลพื้นฐานที่ฮาร์ดแวร์ประมวลผลได้รวดเร็ว
- int และ double ใช้เนื้อที่จำกัด (จึงเก็บข้อมูลได้จำกัด)
 - int หนึ่งตัวใช้ 4 ไบต์
 - double หนึ่งตัวใช้ 8 ไบต์
- int เก็บจำนวนเต็มได้ทุกค่าในช่วง $\pm$ สองพันล้าน
- double เก็บค่าประมาณของจำนวนจริงในช่วง $\pm 10^{300}$
 - แทนจำนวนจริงแบบ floating point
 - 10000 $\rightarrow 1 \times 10^4$ (ในจาวาเขียน 1E4 แทน 1×10^4)
 - 0.00014 $\rightarrow 1.4 \times 10^{-4}$ (เขียน 1.4E-4 ก็ได้)
 - 10000001 $\rightarrow 1.0000001 \times 10^7$

e ตัวเล็กก็ได้

เลขฐานสิบ - เลขฐานสอง

$10^3 \ 10^2 \ 10^1 \ 10^0 \quad 10^{-1} \ 10^{-2} \ 10^{-3}$

9 8 1 2 . 5 6 5

$2^3 \ 2^2 \ 2^1 \ 2^0 \quad 2^{-1} \ 2^{-2} \ 2^{-3}$

1 0 0 1 . 1 0 1

$$2^3 + 2^0 + 2^{-1} + 2^{-3} = 9.625$$

ฐานสิบ : $(1/3)_{10} \approx 0.33333333...$

แทนบางจำนวนไม่ได้
ได้แค่ค่าประมาณ

ฐานสอง : $(1/10)_{10} \approx 0.000110011001100110011...$

จำนวนเต็มฐานสอง

- ถ้าแทนจำนวนเต็มด้วยเลขฐานสอง 4 บิต สามารถแทนจำนวนได้ในช่วง -2^3 ถึง $2^3 - 1$
- จำนวนแบบ int ในจาวาใช้ 4 ไบต์ (32 บิต) จึงแทนจำนวนเต็มได้ในช่วง -2^{31} ถึง $2^{31} - 1$

-2147483648 ถึง 2147483647

ง่าย ๆ int เก็บค่าได้ในช่วง $\pm$ สองพันล้าน

- ถ้าต้องการเก็บจำนวนมากกว่านี้ ก็ใช้ int ไม่ได้ ต้องใช้แบบอื่น (ไม่ขอกล่าวถึง)

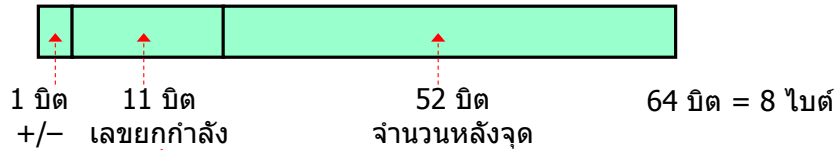
ฐานสอง	ฐานสิบ
0000	= 0
0001	= 1
0010	= 2
0011	= 3
0100	= 4
0101	= 5
0110	= 6
0111	= 7
1000	= -8
1001	= -7
1010	= -6
1011	= -5
1100	= -4
1101	= -3
1110	= -2
1111	= -1

ไม่ขออธิบายรายละเอียด

การแทนจำนวนจริง

- จาวาแทนจำนวนจริงแบบ floating point

9.25 แทนด้วย 1001.101 → 1.001101×2^3



$X.XXX...XXXX \times 2^{xxx}$

$1.11111...11111111 \times 2^{1023}$ ← ค่ามากที่สุด

$1.7976931348623157 \times 10^{308}$

ง่าย ๆ double เก็บค่าได้ในช่วง $\pm 10^{300}$
ความละเอียดหลังจุดได้ประมาณ 16 หลัก

จาวามีข้อมูลอีกประเภท
ชื่อ float ขนาด 4 ไบต์
double หมายถึงมีความ
ละเอียดกว่า float สองเท่า

$0.1 + 0.1 + 0.1 \neq 0.3$

```
public class DoublePrecision {
    public static void main(String[] args) {
        double a = 0.1;
        double b = a + a + a;
        double c = 0.3;
        System.out.println(b);
        System.out.println(c);
        System.out.println(b - c);
    }
}
```

```
JLab>java DoublePrecision
0.30000000000000004
0.3
5.551115123125783E-17
```

แทน 0.1 (ฐานสิบ) ด้วยจำนวนฐานสองแบบ floating point ไม่ได้
แทนได้แต่ค่าประมาณ 0.000110011001100110011...

คำเตือน : อย่าใช้ double เก็บเงิน (อาจปัดงบดุลไม่ได้)

จำนวนเต็มบวกเพิ่มค่าไปเรื่อย ๆ จะติดลบ

```
public class DoublePrecision {
    public static void main(String[] args) {
        int a = 2147483646;
        System.out.println(a);
        a++; System.out.println(a);
        a++; System.out.println(a);
    }
}
```

```
JLab>java DoublePrecision
2147483646
2147483647
-2147483648
```

ค่ามากที่สุดที่ int เก็บได้คือ
2147483647

ถ้าเพิ่มอีก จำนวนบวกจะเป็นจำนวนลบ
เพราะรูปแบบของจำนวนฐานสอง

คำเตือน : อย่าเก็บข้อมูลเกินช่วงที่รับได้

0000	=	0
0001	=	1
0010	=	2
0011	=	3
0100	=	4
0101	=	5
0110	=	6
0111	=	7
1000	=	-8
1001	=	-7
1010	=	-6
1011	=	-5
1100	=	-4
1101	=	-3
1110	=	-2
1111	=	-1

ทำไมต้องใช้ int ?

- การคำนวณบางอย่างต้องการแบบเต็มหน่วย
 - การนับ, ตำแหน่งพิกัดบนจอภาพ, ...
- คำนวณด้วย int เร็วกว่า double
- int ใช้ 4 ไบต์น้อยกว่า double
- คำสั่งบางอย่างใช้กับ int ได้เท่านั้น
- นิสัยของโปรแกรมเมอร์ที่ติดมาจากภาษาอื่น

ถึงแม้ว่า double จะเก็บค่าประมาณของจำนวนจริง
แต่จำนวนเต็มทุกค่าที่ int เก็บได้ double ก็เก็บได้แม่นยำเช่นกัน

การให้ค่าข้อมูลผิดประเภท

```

1 public class TypeCheck {
2     public static void main(String[] args) {
3         int a = 25;
4         double x = a / 2.0;
5         a = a / 2.5;
6         a = x;
7     }
8 }

```

นำค่า double ไปใส่ในตัวแปร int ไม่ได้

```

a = (int)(a / 2.5);
a = (int)(x);

```

แก้ไขเพื่อบังคับให้เปลี่ยนค่าจาก double เป็น int โดยยอมให้ปัดเศษทิ้ง

JLab>javac TypeChecking.java
TypeChecking.java:5: [#1102] [#1155] อาจสูญเสียความแม่นยำของข้อมูลได้
TypeChecking.java:6: [#1102] [#1155] อาจสูญเสียความแม่นยำของข้อมูลได้
JLab>

การเปลี่ยนจาก double เป็น int

ต้องการเปลี่ยนค่าของนิพจน์ที่ได้ค่า double เป็น int

1. ใส่วงเล็บครอบนิพจน์นั้น

2. ใส่ (int) ไว้ข้างหน้า

(int) จะทำก่อน operator ตัวอื่น

```

int a;
double x=1.2, y=2.5;
a = Math.sqrt(x*x + y*y);
a = 20*Math.random();
a = 10 + 10*Math.random();

```

```

int a;
double x=1.2, y=2.5;
a = (int)(Math.sqrt(x*x + y*y));
a = (int)(20*Math.random());
a = 10+(int)(10*Math.random());

```

`a = (int)20 * Math.random();` ผิด (int) แปลงเฉพาะ 20

`a = (int)Math.random() * 20;` ไม่ผิด แต่ได้ 0 เสมอ

Math.random() คืนค่าสุ่มในช่วง [0,1)

การเปลี่ยนจาก int เป็น double

- สามารถนำค่าที่เป็น int เก็บใส่ตัวแปรแบบ double ได้เลย

```

int a = 5, b = 2;
double x = 2 + a;

```

- แต่ถ้าต้องการเปลี่ยนระหว่างการคำนวณ

```

int a = 5, b = 2;
double w = a / b; // ได้ 2
double x = (double)(a/b); // ได้ 2
double y = (double) a / (double) b; //2.5
double z = (double) a / b; //2.5

```

เปลี่ยน a เป็น double

หารกับ double จะแปลง b เป็น double

คำถาม

```
double x = (1/2 + 1/2);
```

x มีค่าเท่าใด ?

```
int y = (int) 1.0/2.0;
```

y มีค่าเท่าใด ?

```
double u = 0.0 + 1/2;
```

u มีค่าเท่าใด ?

```
double v = (0.0 + 1)/2;
```

v มีค่าเท่าใด ?

```
double f = 212.0;
double c = (5/9) * (f - 32);
```

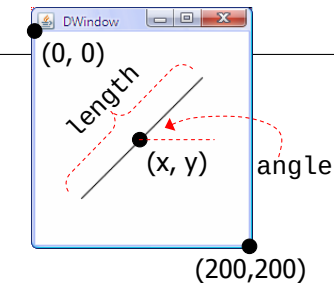
c มีค่าเท่าใด ?

DWindow – while(true)

DWindow : ทบทวน

```
import jlab.graphics.DWindow;

public class LineDrawing5 {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double x = 100, y = 100;
        double length = 80, angle = 45;
        double lenX = length / 2 * Math.cos(Math.toRadians(angle));
        double lenY = length / 2 * Math.sin(Math.toRadians(angle));
        w.drawLine(x - lenX, y + lenY, x + lenX, y - lenY);
    }
}
```



2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

18

while(true) { ... }

JLab>java InfiniteLoop

```
0
1
2 public class InfiniteLoop {
3     public static void main(String[] args) {
4         int i = 0;
5         while (true) {
6             System.out.println(i);
7             i++;
8         }
9     }
10 }
11
12
13
14
15
16
17
18
19
20
```

วนซ้ำไม่สิ้นสุด จนกว่าจะเลิกโปรแกรม

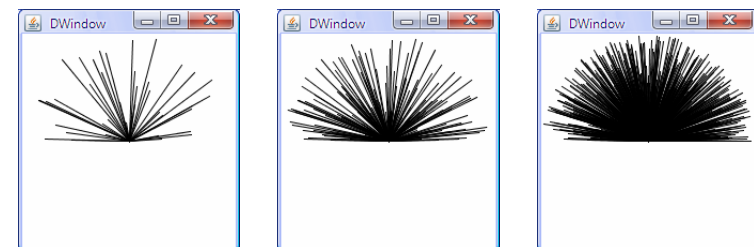
2110101

19

while(true) { ... }

```
import jlab.graphics.DWindow;
public class RandomLines {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double x = 100, y = 100;
        while (true) {
            double length = 100 * Math.random();
            double angle = 180 * Math.random();
            double lenX = length * Math.cos(Math.toRadians(angle));
            double lenY = length * Math.sin(Math.toRadians(angle));
            w.drawLine(x, y, x + lenX, y - lenY);
        }
    }
}
```

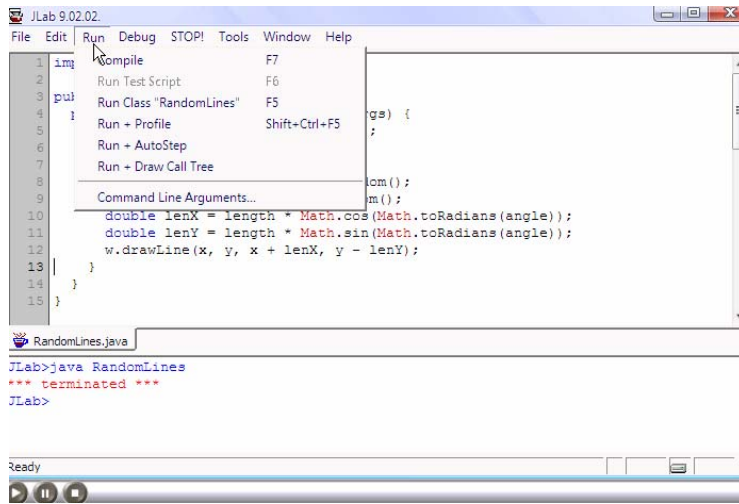
รูปถ่ายหน้าจอขณะทำงานได้
50 รอบ, 200 รอบ, 1000 รอบ



2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

20

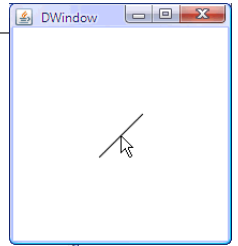
ลองทำที่ละคำสั่งให้ดู



ตัวอย่าง : เส้นเคลื่อนตามเมาส์

```
import jlab.graphics.DWindow;

public class MovingLine {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        while (true) {
            double xm = w.getMouse().getX();
            double ym = w.getMouse().getY();
            w.clearBackground();
            w.drawLine(xm - 20, ym + 20, xm + 20, ym - 20);
        }
    }
}
```

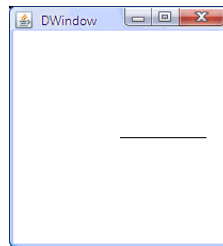


w.clearBackground() → การล้างภาพที่ปรากฏในวินโดว์ w
w.getMouse().getX() → ได้พิกัด x ของหัวลูกศรเมาส์
w.getMouse().getY() → ได้พิกัด y ของหัวลูกศรเมาส์

ตัวอย่าง : เส้นหมุนทวนเข็มนาฬิกา

```
import jlab.graphics.DWindow;

public class RotatingLine {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        int angle = 0;
        int length = 80;
        while (true) {
            w.clearBackground();
            double lenX = length * Math.cos(Math.toRadians(angle));
            double lenY = length * Math.sin(Math.toRadians(angle));
            w.drawLine(100, 100, 100 + lenX, 100 - lenY);
            angle = angle + 5;
            w.sleep(100);
        }
    }
}
```



เพิ่มคำสั่ง **w.sleep(100)** ในวงวน เพื่อให้หยุดทำงานชั่วคราวระหว่างรอบ 100 คือในหยุด 100ms. แล้วก็ทำงานต่อ (เปลี่ยน 100 เป็นจำนวนอื่นได้)

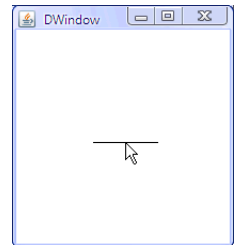
โจทย์ : เส้นหมุนเคลื่อนตามเมาส์

```
import jlab.graphics.DWindow;

public class MovableRotatingLine {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double length = 60;

        while (true) {
            double x = w.getMouse().getX();
            double y = w.getMouse().getY();

            w.clearBackground();
            w.drawLine(x, y, x + length * Math.cos(Math.toRadians(Math.atan2(y - y0, x - x0))), y + length * Math.sin(Math.toRadians(Math.atan2(y - y0, x - x0))));
            w.sleep(100);
        }
    }
}
```



จงแก้ไขโปรแกรมข้างบนนี้ให้ทำงานตามที่แสดง

โจทย์ : งูเลื้อยตามเมาส์

```
import jlab.graphics.DWindow;

public class MovingSnake {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);

        while (true) {
            double xm = w.getMouse().getX();
            double ym = w.getMouse().getY();

            w.clearBackground();
            w.drawLine(
            );
            w.drawLine(
            );
            w.drawLine(
            );
            w.sleep(100);
        }
    }
}
```

จงแก้ไขโปรแกรมข้างบนนี้ให้ทำงานตามที่แสดง

โจทย์ : เกล็ดหิมะ

```
import jlab.graphics.DWindow;
public class FallingSnow {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        w.setBackground(w.BLACK);
        w.setForeground(w.WHITE);

        }
    }
}
```

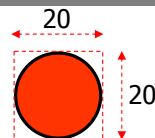
จงแก้ไขโปรแกรมข้างบนนี้ให้ทำงานตามที่แสดง

ตัวอย่าง : ลูกบอลเคลื่อนที่

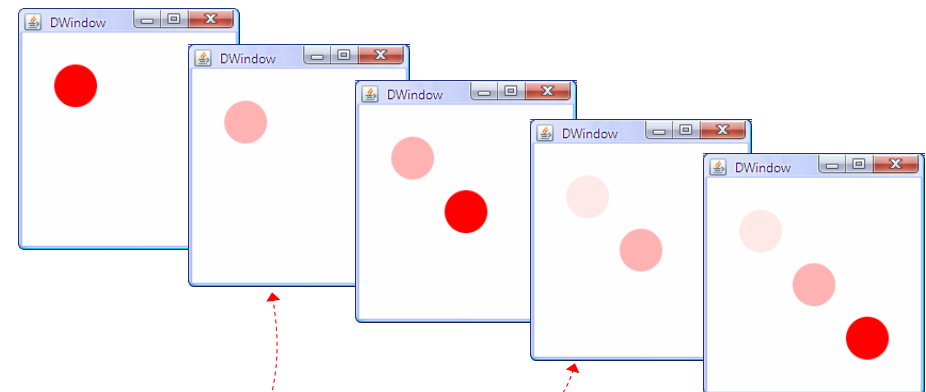
```
import jlab.graphics.DWindow;

public class Ball {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double x = 100, y = 100;
        double dx = 3, dy = 2;
        while (true) {
            x = (x + dx) % 200;
            y = (y + dy) % 200;
            w.clearBackground();
            w.fillEllipse(w.RED, x, y, 20, 20);
            w.sleep(50);
        }
    }
}
```

จุดศูนย์กลาง
ของวงรี



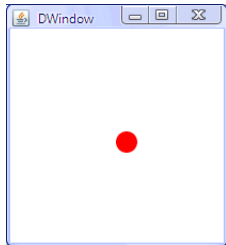
การทำให้จาง : w.fade(level)



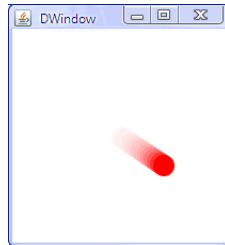
```
DWindow w = new DWindow(200, 200);
w.fillEllipse(w.RED, 50, 50, 40, 40);
w.fade(0.7);
w.fillEllipse(w.RED, 100, 100, 40, 40);
w.fade(0.7);
w.fillEllipse(w.RED, 150, 150, 40, 40);
```

ตัวอย่าง : ลูกบอลมีหาง

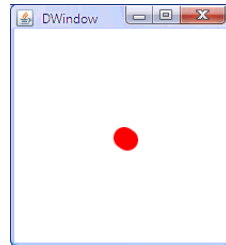
```
DWindow w = new DWindow(200, 200);  
double x = 100, y = 100;  
double dx = 3, dy = 2;  
while (true) {  
    x = (x + dx) % 200;  
    y = (y + dy) % 200;  
    w.fade( ??? );  
    w.fillEllipse(w.RED, x, y, 20, 20);  
    w.sleep(50);  
}
```



w.fade(0)

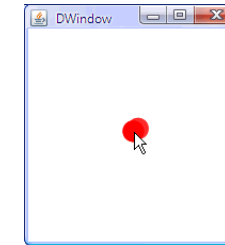


w.fade(0.2)



w.fade(1)

โจทย์ : ลูกบอลมีหางไล่ตามหัวลูกศรเมาส์



จงเขียนโปรแกรมให้ทำงานตามตัวอย่างที่แสดงข้างบนนี้

2110101 : การทำโปรแกรมคอมพิวเตอร์

Selection : if, if-else

ภาควิชาวิศวกรรมคอมพิวเตอร์

จุฬาลงกรณ์มหาวิทยาลัย

ตัวอย่าง : หาคากที่สอง

```
import java.util.Scanner;
public class SquareRoot{
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("a = ");
        double a = kb.nextDouble();
        double x = 1.0;
        x = (x + (a / x)) / 2.0;
        x = (x + (a / x)) / 2.0;
        x = (x + (a / x)) / 2.0;
        x = (x + (a / x)) / 2.0;
        x = (x + (a / x)) / 2.0;
        x = (x + (a / x)) / 2.0;
        System.out.println("รากที่สองของ " + a + " = " + x);
    }
}
```

$$x_k = \frac{1}{2} \left(x_{k-1} + \frac{a}{x_{k-1}} \right)$$

ต้องทำกี่ครั้ง ?

```
JLab>java SquareRoot
a = 100
รากที่สองของ 100.0 = 10.000052895642693
```

```
JLab>java SquareRoot
a = 10000
รากที่สองของ 10000.0 = 177.01806998329744
```

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

2

ตัวอย่าง : หาคากที่สอง (ปรับปรุง)

```
import java.util.Scanner;
public class SquareRoot{
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("a = ");
        double a = kb.nextDouble();
        double x = 1.0;
        while (true) {
            x = (x + (a / x)) / 2.0;
            if ( (x*x - a) < 1e-5) break;
        }
        System.out.println("รากที่สองของ " + a + " = " + x);
    }
}
```

$$x \approx \sqrt{a}$$

$$x_k^2 - a \geq 0$$

ดูรายละเอียดหน้าถัดไป

ถ้า $x^2 - a < 10^{-5}$ ให้หลุดออกจากวงวน

```
JLab>java SquareRoot
a = 1000000
รากที่สองของ 1000000.0 = 1000.00000000000118
```

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

3

พิสูจน์ : $x^2 - a \geq 0$

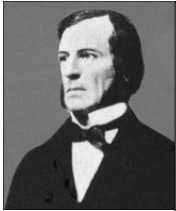
$$\begin{aligned} x_k &= \frac{1}{2} \left(x_{k-1} + \frac{a}{x_{k-1}} \right) & x_k^2 - a &= \frac{x_{k-1}^4 + 2ax_{k-1}^2 + a^2 - 4ax_{k-1}^2}{4x_{k-1}^2} \\ x_k &= \frac{x_{k-1}^2 + a}{2x_{k-1}} & &= \frac{x_{k-1}^4 - 2ax_{k-1}^2 + a^2}{4x_{k-1}^2} \\ x_k^2 &= \left(\frac{x_{k-1}^2 + a}{2x_{k-1}} \right)^2 & &= \left(\frac{x_{k-1}^2 - a}{2x_{k-1}} \right)^2 \\ x_k^2 - a &= \left(\frac{x_{k-1}^2 + a}{2x_{k-1}} \right)^2 - a & &\geq 0 \end{aligned}$$

$$x_k^2 - a \geq 0$$

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

4

while(true) + if ... break



George Boole
(1815-1864)

```
while (true) {  
    if ( Boolean-expression ) break;  
}
```

นิพจน์บูลีน
ให้ค่าจริง
หรือเท็จ

```
int n = 1;  
while (true) {  
    System.out.print(n + " ");  
    n++;  
    if (n > 5) break;  
}
```

แสดง 1 2 3 4 5

```
int k = 1;  
while (true) {  
    //...  
    k++;  
    if (k > n) break;  
}
```

ทำงานในวงวน n รอบ

ตัวอย่าง : หาค่าเฉลี่ยจากแป้นพิมพ์ (แบบที่ 1)

- หาค่าเฉลี่ยของจำนวนจริงที่รับจากแป้นพิมพ์ 4 จำนวน

```
import java.util.Scanner;  
  
public class Average1 {  
    public static void main(String[] args) {  
        Scanner kb = new Scanner(System.in);  
        double sum = 0;  
        int k = 1, n = 4;  
        while (true) {  
            System.out.print("จำนวนที่ " + k + " = ");  
            sum = sum + kb.nextDouble();  
            k++;  
            if (k > n) break;  
        }  
        System.out.println("ค่าเฉลี่ย = " + (sum / n));  
    }  
}
```

JLab>java Average1
จำนวนที่ 1 = 10
จำนวนที่ 2 = 11
จำนวนที่ 3 = 12
จำนวนที่ 4 = 13
ค่าเฉลี่ย = 11.5

ตัวอย่าง : หาค่าเฉลี่ยจากแป้นพิมพ์ (แบบที่ 2)

- หาค่าเฉลี่ยของจำนวนจริงที่รับจากแป้นพิมพ์
- จะถามผู้ใช้อย่างไรก่อนว่าจะป้อนกี่จำนวน

```
import java.util.Scanner;  
  
public class Average2 {  
    public static void main(String[] args) {  
        Scanner kb = new Scanner(System.in);  
        System.out.print("จำนวนข้อมูล = ");  
        int n = kb.nextInt();  
        double sum = 0;  
        int k = 1;  
        while (true) {  
            System.out.print("จำนวนที่ " + k + " = ");  
            sum = sum + kb.nextDouble();  
            k++;  
            if (k > n) break;  
        }  
        System.out.println("ค่าเฉลี่ย = " + (sum / n));  
    }  
}
```

JLab>java Average2
จำนวนข้อมูล = 3
จำนวนที่ 1 = 10
จำนวนที่ 2 = 16
จำนวนที่ 3 = 1
ค่าเฉลี่ย = 9.0

ตัวอย่าง : หาค่าเฉลี่ยจากแป้นพิมพ์ (แบบที่ 3)

- หาค่าเฉลี่ยของจำนวนจริงบวกที่รับจากแป้นพิมพ์
- พบจำนวนลบเมื่อใดให้เลิกรับ

```
import java.util.Scanner;  
  
public class Average3 {  
    public static void main(String[] args) {  
        Scanner kb = new Scanner(System.in);  
        double sum = 0;  
        int k = 1;  
        while (true) {  
            System.out.print("จำนวนที่ " + k + " = ");  
            double v = kb.nextDouble();  
            if (v < 0) break;  
            sum = sum + v;  
            k++;  
        }  
        System.out.println("ค่าเฉลี่ย = " + (sum / (k-1)));  
    }  
}
```

JLab>java AverageN
จำนวนที่ 1 = 12
จำนวนที่ 2 = 10
จำนวนที่ 3 = 5
จำนวนที่ 4 = -1
ค่าเฉลี่ย = 9.0

ตัวเปรียบเทียบ และการเชื่อมนิพจน์บูลีน

- ตัวเปรียบเทียบ ได้แก่
 $<$ (น้อยกว่า) $>$ (มากกว่า)
 $<=$ (น้อยกว่าหรือเท่ากับ) $>=$ (มากกว่าหรือเท่ากับ)
 $==$ (เท่ากับ) $!=$ (ไม่เท่ากับ)
- ตัวเชื่อมนิพจน์ ได้แก่
 $\&\&$ (และ) $||$ (หรือ) $!$ (ไม่)

```
if ( x != 0 ) ...           // ถ้า x ไม่เท่ากับ 0
if ( (n % 2) == 1 ) ...    // ถ้า n เป็นเลขคี่
if ( (0 <= x) && (x <= 10) ) ... // ถ้า 0 ≤ x ≤ 10
if ( !(x < 0 || x > 10) ) ... // ถ้า 0 ≤ x ≤ 10
```

$$\neg(p \vee q) \Leftrightarrow (\neg p) \wedge (\neg q)$$

$$\neg(p \wedge q) \Leftrightarrow (\neg p) \vee (\neg q)$$

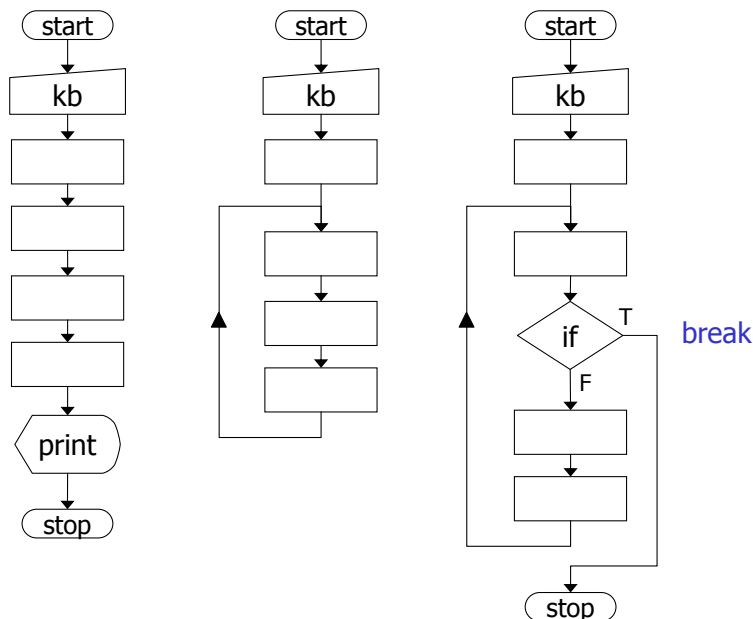
De Morgan's laws

ตารางความจริง : $\&\&$, $||$

A	B	A $\&\&$ B
true	true	true
true	false	false
false	true	false
false	false	false

A	B	A $ $ B
true	true	true
true	false	true
false	true	true
false	false	false

Flowchart – ฟังงาน



โปรแกรม - ฟังงาน

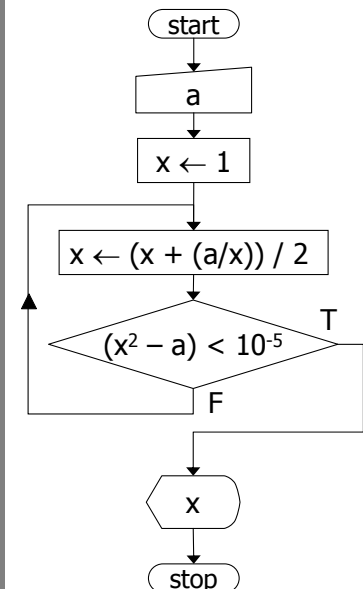
```
import java.util.Scanner;
public class SquareRoot{
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("a = ");
        double a = kb.nextDouble();

        double x = 1.0;

        while (true) {
            x = (x + (a / x)) / 2.0;

            if ((x*x - a) < 1e-5) break;
        }

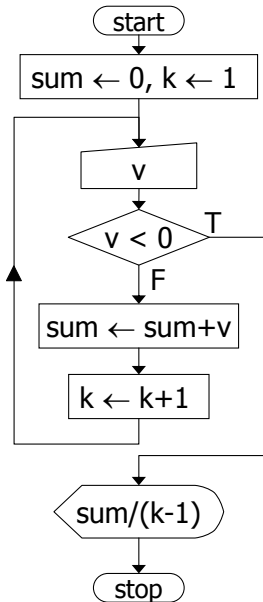
        System.out.println(x);
    }
}
```



โปรแกรม - ผังงาน

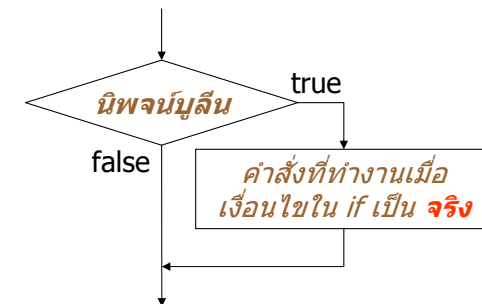
```
import java.util.Scanner;

public class Average3 {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        double sum = 0;
        int k = 1;
        while (true) {
            System.out.print("จำนวนที่ " + k + " = ");
            double v = kb.nextDouble();
            if (v < 0) break;
            sum = sum + v;
            k++;
        }
        System.out.println("ค่าเฉลี่ย = " +
            (sum / (k-1)));
    }
}
```



คำสั่ง if อีกแบบ

```
if ( นิพจน์ที่ได้ค่า จริง หรือ เท็จ ) {
    คำสั่งที่ทำงานเมื่อเงื่อนไขใน if เป็น จริง
}
```

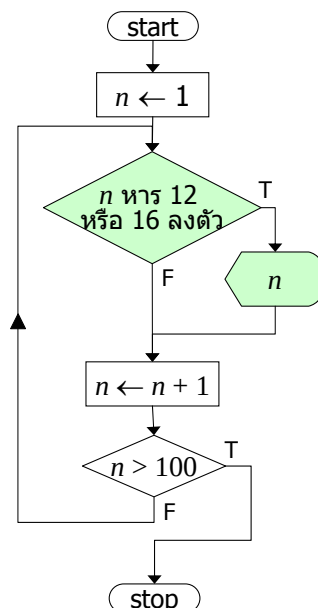


```
if ( (n % 7) == 0 || (n % 11) == 0 ) {
    System.out.print(n + ",");
}
```

จำนวนใดใน 1 ถึง 100 ที่หารด้วย 12 หรือ 16 ลงตัว

```
public class Div1216 {
    public static void main(String[] args){
        int n = 1;
        while (true) {
            if ( (n%12) == 0 || (n%16) == 0 ) {
                System.out.print(n + ",");
            }
            n++;
            if (n > 100) break;
        }
    }
}
```

JLab>java Div1216
12,16,24,32,36,48,60,64,72,80,84,96,



ตัวอย่าง : หาค่าประมาณของ π ที่ดีกว่า 22/7

- $|\pi - 22/7| \approx 0.001264...$
- หาค่า n และ d ที่ทำให้ $|\pi - n/d| < 10^{-6}$

n/d	$(\pi - n/d)$	
22/7 =	-0.0013	32/10 = -0.0584
22/8 =	0.3916	32/11 = 0.2325
23/8 =	0.2666	33/11 = 0.1416
24/8 =	0.1416	34/11 = 0.0507
25/8 =	0.0166	35/11 = -0.0402
26/8 =	-0.1084	35/12 = 0.2249
26/9 =	0.2527	36/12 = 0.1416
27/9 =	0.1416	37/12 = 0.0583
28/9 =	0.0305	...
29/9 =	-0.0806	352/112 = -0.0013
29/10 =	0.2416	352/113 = 0.0265
30/10 =	0.1416	353/113 = 0.0177
31/10 =	0.0416	354/113 = 0.0088
		355/113 = -0.0000003



祖冲之
พบจำนวน $\frac{355}{113}$
เมื่อปี ค.ศ. 480

ตัวอย่าง : หาค่าประมาณของ π ที่ดีกว่า 22/7

```
public class ApproxPI {
    public static void main(String[] args) {
        int n = 22, d = 7;
        double approx = (double)n/d;
        double pi = 3.141592653589793;
        while (true) {
            if (Math.abs(pi - approx) < 1e-6) break;
            if (pi < approx) { d++; }
            if (pi > approx) { n++; }
            approx = (double)n/d;
        }
        System.out.println((n + "/" + d + " = " + approx));
        System.out.println("error = " + (pi - approx));
    }
}
```

n/d	$(\pi - n/d)$
22/7	= -0.0013
22/8	= 0.3916
23/8	= 0.2666
...	...

```
JLab>java ApproxPI
355/113 = 3.1415929203539825
error = -2.667641894049666E-7
```

ตัวอย่าง : โปรแกรมแสดงชื่อเดือน

- รับหมายเลขเดือน (1 – 12) จากผู้ใช้ทางแป้นพิมพ์
- แสดงชื่อเดือนทางจอภาพ

```
JLab>java Month
หมายเลขเดือน = 3
มีนาคม
```

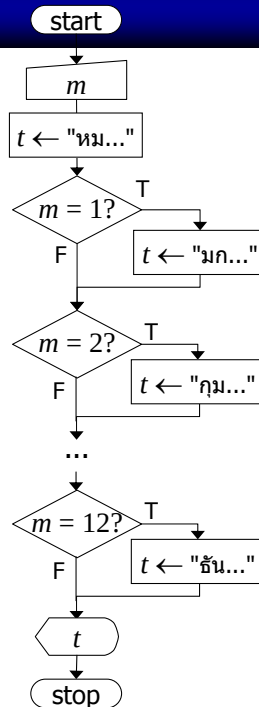
```
JLab>java Month
หมายเลขเดือน = 12
ธันวาคม
```

```
JLab>java Month
หมายเลขเดือน = 15
หมายเลขต้องอยู่ในช่วง 1 - 12
```

```
import java.util.Scanner;
public class Month {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("หมายเลขเดือน = ");
        int month = kb.nextInt();
        if (month == 1) { System.out.println("มกราคม"); }
        if (month == 2) { System.out.println("กุมภาพันธ์"); }
        // เติมที่เหลือเอง ...
        if (month == 12) { System.out.println("ธันวาคม"); }
        if (month < 1 || month > 12) {
            System.out.println("หมายเลขต้องอยู่ในช่วง 1 - 12");
        }
    }
}
```

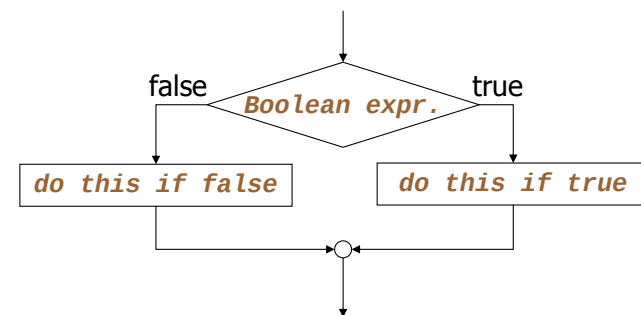
ตัวอย่าง : แสดงชื่อเดือน

```
import java.util.Scanner;
public class Month {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("หมายเลขเดือน = ");
        String t = "หมายเลขต้องอยู่ในช่วง 1 - 12";
        int month = kb.nextInt();
        if (month == 1) { t = "มกราคม"; }
        if (month == 2) { t = "กุมภาพันธ์"; }
        if (month == 3) { t = "มีนาคม"; }
        if (month == 4) { t = "เมษายน"; }
        if (month == 5) { t = "พฤษภาคม"; }
        if (month == 6) { t = "มิถุนายน"; }
        if (month == 7) { t = "กรกฎาคม"; }
        if (month == 8) { t = "สิงหาคม"; }
        if (month == 9) { t = "กันยายน"; }
        if (month == 10) { t = "ตุลาคม"; }
        if (month == 11) { t = "พฤศจิกายน"; }
        if (month == 12) { t = "ธันวาคม"; }
        System.out.println(t);
    }
}
```



คำสั่ง if - else

```
if ( นิพจน์ที่ได้ค่าจริง หรือ เท็จ ) {
    คำสั่งที่ทำงานเมื่อเงื่อนไขใน if เป็นจริง
} else {
    คำสั่งที่ทำงานเมื่อเงื่อนไขใน if เป็นเท็จ
}
```

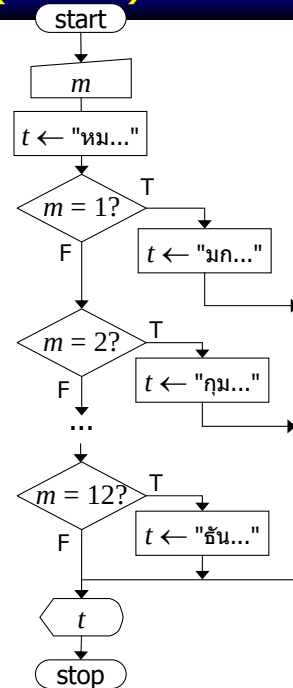


ตัวอย่าง : แสดงชื่อเดือน (if-else)

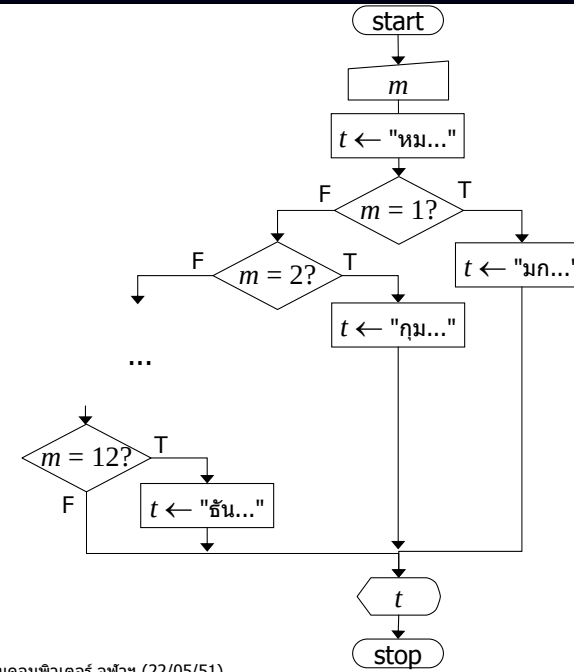
```
import java.util.Scanner;

public class Month {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("หมายเลขเดือน = ");
        String t = "หมายเลขต้องอยู่ในช่วง 1 - 12";
        int month = kb.nextInt();
        if (month == 1) { t = "มกราคม"; }
        else if (month == 2) { t = "กุมภาพันธ์"; }
        else if (month == 3) { t = "มีนาคม"; }
        // เติมที่เหลือเอง ...

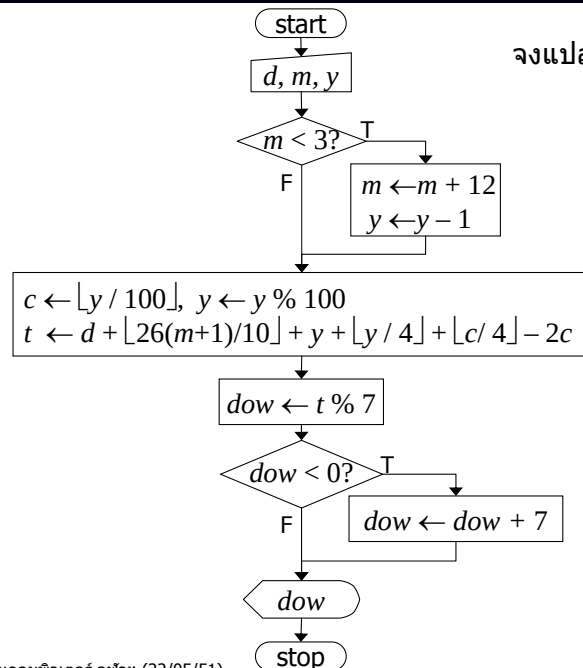
        else if (month == 12) { t = "ธันวาคม"; }
        System.out.println(t);
    }
}
```



ผังงาน : การแสดงชื่อเดือน (if-else)



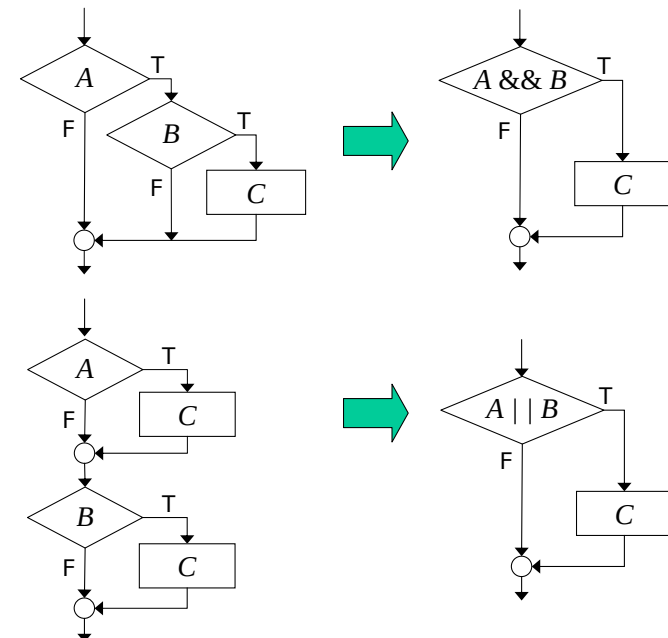
โจทย์ : วันอะไร ? (วิธีของคุณ Zeller)



จงแปลงผังงานนี้เป็นโปรแกรม

y คือ ปี ค.ศ.
dow = 0 คือวันเสาร์
= 1 คือวันอาทิตย์
...
= 6 คือวันศุกร์

การรวมเงื่อนไขการทดสอบ



โจทย์ : ภาษีเงินได้

ขั้นเงินได้สุทธิ	อัตราภาษี
1 ถึง 100,000	5%
100,001 ถึง 500,000	10%
500,001 ถึง 1,000,000	20%
1,000,001 ถึง 4,000,000	30%
4,000,001 บาทขึ้นไป	37%

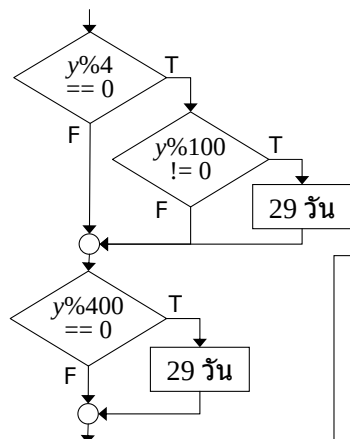
จงเขียนโปรแกรมรับเงินได้สุทธิทางแป้นพิมพ์
เพื่อคำนวณและแสดงภาษีเงินได้ส่วนบุคคลตาม
เกณฑ์ในตารางข้างบนนี้

การทดสอบปีอธิกสุรทิน

- เดือนกุมภาพันธ์ของปี y (ค.ศ.) มี 29 วันเมื่อ
 - เลขปี (ค.ศ.) ต้องหารด้วย 4 ลงตัว แต่ต้องหารด้วย 100 ไม่ลงตัว ยกเว้นกรณีที่หารด้วย 400 ลงตัว เช่น
 - ปี 2002 ก.พ. มี 28 วัน (หารด้วย 4 ไม่ลงตัว)
 - ปี 2004 ก.พ. มี 29 วัน (หารด้วย 4 ลงตัว และหารด้วย 100 ไม่ลงตัว)
 - ปี 2100 ก.พ. มี 28 วัน (หารด้วย 100 ลงตัว)
 - ปี 2000 ก.พ. มี 29 วัน (หารด้วย 400 ลงตัว)
- ให้ y คือเลขปี (ค.ศ.)
 - $(y \% 4) == 0$ แทนการทดสอบว่าหารด้วย 4 ลงตัว ?
 - $(y \% 100) != 0$ แทนการทดสอบว่าหารด้วย 100 ไม่ลงตัว ?
 - $(y \% 400) == 0$ แทนการทดสอบว่าหารด้วย 400 ลงตัว ?

การทดสอบปีอธิกสุรทิน

- เลขปี (ค.ศ.) ต้องหารด้วย 4 ลงตัว แต่ต้องหารด้วย 100 ไม่ลงตัว ยกเว้นกรณีที่หารด้วย 400 ลงตัว



```

if (y%4 == 0) {
    if (y%100 != 0) {
        d = 29;
    }
}
if (y%400 == 0) {
    d = 29;
}

```

```

if ((y%4 == 0)&&(y%100 != 0)) {
    d = 29;
}
if (y%400 == 0) {
    d = 29;
}

```

```

if ( ((y%4 == 0)&&(y%100 != 0)) || (y%400 == 0) ) {
    d = 29;
}

```

ลำดับการปฏิบัติการ ! && ||

- วงเล็บทำก่อน
- NOT : !
- AND : &&
- OR : ||

```

if ( ((y%4 == 0)&&(y%100 != 0)) || (y%400 == 0) ) {
    d = 29;
}

```

เหมือน

```

if ( y%4 == 0 && y%100 != 0 || y%400 == 0 ) {
    d = 29;
}

```

เหมือน

```

if ( y%4 == 0 && !(y%100 == 0) || y%400 == 0 ) {
    d = 29;
}

```

Block : กลุ่มคำสั่ง

- Block คือกลุ่มคำสั่ง ครอบด้วย { }
- ถ้ามีคำสั่งเดียวหลัง if หรือ else, ไม่ต้องใส่ใน block ก็ได้

```
if (Boolean Expression)
    a statement ;
```

```
if (Boolean Expression)
    a statement ;
else
    a statement ;
```

```
if (Boolean Expression){
    statements ;
}
```

```
if (Boolean Expression){
    statements ;
} else {
    statements ;
}
```

```
if (x < 0) {
    x = -x;
}
```

```
if (x < 0)
    x = -x;
```

```
if (x < 0) x = -x;
```

เหมือน

เหมือน

```
if (n > 100) break;
```

```
if (n > 100) { break; }
```

เหมือน

ตัวแปรแบบ boolean

- int - จำนวนเต็มในช่วงประมาณ $\pm$ สองพันล้าน
- double - ค่าประมาณจำนวนจริงในช่วง $\pm 10^{300}$
- String - ข้อความ
- boolean - true กับ false

หาผลรวมของจำนวน 10 จำนวน และตรวจสอบด้วยว่าใน 10 จำนวนนี้มีจำนวนใดเป็นเลขคี่หรือไม่

```
import java.util.Scanner;
public class SumHasOdd {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        double sum = 0;
        boolean hasOdd = false;
        int k = 1, n = 10;
        while (true) {
            System.out.print("กรณำป้อนจำนวน = ");
            double v = kb.nextDouble();
            if (v % 2 == 1) hasOdd = true;
            sum += v;
            if (++k > n) break;
        }
        System.out.println("sum = " + sum + ", hasOdd = " + hasOdd);
    }
}
```

เตรียมตัวแปร boolean ไว้ เก็บว่าเคยพบจำนวนคี่หรือไม่

ตัวอย่าง : ลูกบอล (อีกครั้ง)

```
import jlab.graphics.DWindow;

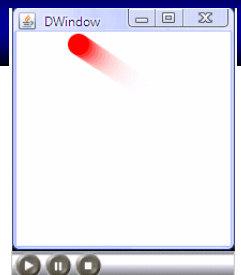
public class Ball {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double x = 100, y = 100;
        double dx = 3, dy = 2;
        while (true) {
            x = (x + dx) % 200;
            y = (y + dy) % 200;
            w.fade(0.2);
            w.fillEllipse(w.RED, x, y, 20, 20);
            w.sleep(50);
        }
    }
}
```

ตัวอย่าง : ลูกบอลเด้งผนัง

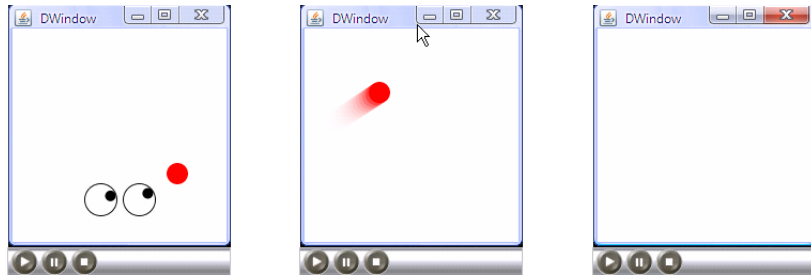
```
import jlab.graphics.DWindow;

public class BouncingBall {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double x = 100, y = 100;
        double dx = 3, dy = 2;
        double r = 10;
        while (true) {
            if ((x-r <= 0) || (x+r >= 200)) {
                dx = -dx;
            }
            if ((y-r <= 0) || (y+r >= 200)) {
                dy = -dy;
            }
            x = x + dx;
            y = y + dy;
            w.fade(0.2);
            w.drawEllipse(x, y, 20, 20);
            w.sleep(50);
        }
    }
}
```

ถ้าตกขอบวินโดว์ให้กลับทิศ



โจทย์ : ลูกบอลแดง ๆ



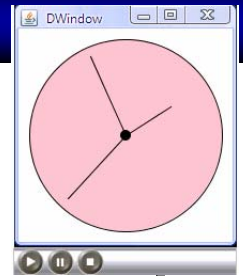
ตามองลูกบอลแดง ลูกบอลแดงหัวลูกศรเมาส์ ลูกบอลแดงบนพื้นโลก

จงเขียนโปรแกรมให้ทำงานตามที่ตัวอย่างที่แสดงข้างบนนี้

โจทย์ : นาฬิกาแบบเข็ม

```
import jlab.graphics.DWindow;
import java.util.Date;
public class Clock {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        Date now = new Date();
        int h = now.getHours();
        int m = now.getMinutes();
        int s = now.getSeconds();
        double x0 = 100, y0 = 100;
        while (true) {
            // เติมนาทีที่จำเป็นตรงนี้

            w.clearBackground();
            w.fillEllipse(w.PINK, 100, 100, 180, 180);
            w.drawEllipse(w.BLACK, 100, 100, 180, 180);
            w.fillEllipse(w.BLACK, 100, 100, 10, 10);
            w.drawLine(x0, y0, xh, yh); // วาดเข็มชั่วโมง
            w.drawLine(x0, y0, xm, ym); // วาดเข็มนาฬิกา
            w.drawLine(x0, y0, xs, ys); // วาดเข็มวินาที
            w.sleep(1000); // หยุด 1 วินาที
        }
    }
}
```



สิ่งที่พบว่าผิดกันบ่อย

```
if (0 <= x <= 10)
if x == 0
if (x >= 0 && <= 10)
if (x == 0 || == 1)
if (x = 0)
if (x >= 0)
```



```
if (0 <= x && x <= 10)
if (x == 0)
if (x >= 0 && x <= 10)
if (x == 0 || x == 1)
if (x == 0)
if (x >= 0)
```

```
if (x == 0)
    if (y == 0) z = 0;
else if (a == 1)
    z = 1;
```



```
if (x == 0) {
    if (y == 0) z = 0;
} else if (a == 1)
    z = 1;
```

เขียนไปไรค่า

```
if (x != 0 || x != 5)
if (x < 3 && x > 9)
if (x == 0 && x != 0)
if (x == 0 || x != 0)
if (success = true)
```

ต้องระวังเมื่อใช้ == กับจำนวนแบบ floating point

- **double** มีข้อจำกัดเรื่องความแม่นยำ
- การคำนวณจึงมักทำให้ผลลัพธ์มีความคลาดเคลื่อน
- $0.1 + 0.1 + 0.1 \neq 0.3$

```
public class Imprecision {
    public static void main(String [] args) {
        double a = 0.1 + 0.1 + 0.1;
        if (a == 0.3) {
            System.out.println( "ok" );
        }
        else
            System.out.println( "!!!" );
    }
}
```

JLab>java Imprecision
!!!

เมื่อต้องการเปรียบเทียบจำนวนจริง $x == y$?

- ถ้าผ่านการทดสอบข้างขว
นี้ ก็แสดงว่า x และ y มีค่า
ใกล้เคียงกันมาก

$$\frac{|x-y|}{\max(|x|, |y|)} \leq 10^{-10}$$

```
public class RelativeDiff {  
    public static void main(String [] args) {  
        double x = 0.1 + 0.1 + 0.1;  
        double y = 0.3;  
        if ( Math.abs(x-y) /  
            Math.max(Math.abs(x), Math.abs(y)) <= 1e-10 )  
            System.out.println( "x == y" );  
    }  
}
```

```
JLab>java RelativeDiff  
x == y
```

ใช้ `==` `<` `>` `!=` `<=` `>=` กับ String ไม่ได้ (โปรดติดตาม)

String และ การอ่านแฟ้มข้อมูล

ภาควิชาวิศวกรรมคอมพิวเตอร์
จุฬาลงกรณ์มหาวิทยาลัย

- สตริง + สตริง คือการต่อสตริง "AB" + "C" ได้ "ABC"
- สตริง + จำนวน หรือ จำนวน + สตริง
จะแปลงจำนวนเป็นสตริงก่อน แล้วต่อ
 - "A" + 2 ได้ "A2"
 - 2.5 + "A" ได้ "2.5A"
 - "A" + (4+2) ได้ "A6"
 - "A" + 4 + 2 ได้ "A42"
 - 2 + "A" + 4 ได้ "2A4"
 - 2 + 4 + A ได้ "6A"
- * / - % ใช้กับสตริงไม่ได้
 - "2" * 1 ผิด

- String คือข้อมูลแบบข้อความ

- "Hello สมชาย 你好"
- รหัสควบคุมการแสดง
 - \n - ขึ้นบรรทัดใหม่
 - \t - รหัส tab
 - \\ - ตัว \
 - \" - ตัว "

```
public class Test1 {
    public static void main(String[] args) {
        String s = "สวัสดี\t\"John\\\"\\n\tไม่โทษมา";
        System.out.println(s);
    }
}
```

Jlab>java Test

```
JLab>java Test
สวัสดี "John"
ไปไหนมา
```

```
public class Test2 {
    public static void main(String[] args) {
        System.out.println("_____");
        System.out.println("(_____)");
        System.out.println("____|_|");
        System.out.println("_ _|_|/ _ _)( _ _)/ _ _");
        System.out.println("( _ )| |( _|_| \/_ |( _|_|");
        System.out.println(" _/_/_/_/_/_/_/_/_/_/_/_/_/_/_");
    }
}
```

ผิด ต้องใส่ \

```
import java.util.Scanner;

public class ReadString1 {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("ชื่อ = ");
        String name = kb.nextLine();
        System.out.println("สวัสดีจ๊ะ " + name);
    }
}
```

```
nextDouble()  
nextInt()  
nextLine()
```

```
JLab>java ReadString1
ชื่อ = จาวา แซ่ซัน
สวัสดีจ้ะ จาวา แซ่ซัน
```

```
import java.util.Scanner;

public class ReadString2 {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("ID = ");
        String id = kb.nextLine();
        System.out.println(id + 9);
    }
}
```

JLab>java

```
JLab>java ReadString2
ID = 123
1239
```

เปลี่ยน 9 เป็นสตริงแล้วต่อ

บริการของสตริง

- String ไม่ใช่ข้อมูลพื้นฐาน มีบริการให้เรียกมากมาย (อ่านเพิ่มเติมใน help file)

```
//      012345678901234567890123456789
String s = "  Computer Programming  ";
int n = s.length();      // n = 30
String s1;
s1 = s.toUpperCase();    // s1 = "  COMPUTER PROGRAMMING  "
s1 = s.toLowerCase();    // s1 = "  computer programming  "
s1 = s.substring(11, 15); // s1 = "Prog"
s1 = s.trim();           // s1 = "Computer Programming"
n = s.indexOf("Prog");   // n = 11
n = s.indexOf("JAVA");   // n = -1
s1 = s + s;
n = s1.indexOf("Comp", 3); // n = 32
```

บริการของสตริงไม่เปลี่ยนแปลงสตริง
toUpperCase, toLowerCase, substring, trim คืนสตริงใหม่

ใช้ == < > เพื่อเปรียบเทียบสตริงไม่ได้

- ต้องใช้เมทอด **compareTo** ในการเปรียบเทียบ

s1.compareTo(s2) {

- < 0 ถ้า s1 มาก่อน s2 ตามลำดับตัวอักษร
- = 0 ถ้า s1 เหมือนกับ s2
- > 0 ถ้า s1 มาหลัง s2 ตามลำดับตัวอักษร

```
public class StringCompare {
    public static void main(String [] args) {
        System.out.print("Computer".compareTo("Computer") + ",");
        System.out.print("Computer".compareTo("Engineer") + ",");
        System.out.print("Computer".compareTo("Comp") + ",");
        System.out.println("Computer".compareTo("Computing"));
    }
}
```

สนใจผลลัพธ์ว่าบวก, ลบ หรือศูนย์ก็พอ

JLab>java StringCompare
0, -2, 4, -4

ตัวอย่าง : สตริงที่รับมามี "A" กี่ตัว

- ใช้ indexOf ค้น แล้วนับ

```
import java.util.Scanner;

public class CountA {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("สตริง = ");
        String t = kb.nextLine();
        t = t.toUpperCase();
        int k = 0, count = 0;
        while (true) {
            int j = t.indexOf("A", k);
            if (j < 0) break;
            count++;
            k = j + 1;
        }
        System.out.println("มี A " + count + " ตัว");
    }
}
```

JLab>java CountA
สตริง = JAVA Programming
มี A 3 ตัว

ถ้าเขียน k = j, โปรแกรมจะทำงานไม่หยุด

ตัวอย่าง : วนรับสตริงมาประกอบเป็นประโยค

- จนกว่าจะรับคำว่า "end" ก็ให้เลิก

```
import java.util.Scanner;

public class Word2Sentence {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        String sentence = "";
        while (true) {
            System.out.print("word = ");
            String word = kb.nextLine();
            if (word.compareTo("end") == 0) break;
            sentence = sentence + " " + word;
        }
        sentence = sentence + ".";
        System.out.println(sentence);
    }
}
```

สตริงที่ไม่มีอะไรเลย
ความยาวเป็นศูนย์

JLab>java Words2Sentence
word = Today
word = is
word = my
word = day
word = end
→ Today is my day.

จงแก้ไขโปรแกรมไม่ให้มีช่องว่างตรงนี้

ตัวอย่าง : แยกสตริงที่รับมา ออกเป็นคำ ๆ

- ให้คำแยกกันด้วยช่องว่าง ใช้ indexOf ค้น แล้ว substring

```
import java.util.Scanner;

public class Token {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("สตริง = ");
        String t = kb.nextLine();
        int i = 0, j = 0;
        while (true) {
            j = t.indexOf(" ", i);
            if (j < 0) break;
            System.out.print(t.substring(i, j) + " ");
            i = j + 1;
        }
        System.out.println(t.substring(i, t.length()));
    }
}
```

JLab>java Token
สตริง = Am I right ?
Am, I, right, ?

JLab>java Token
สตริง = Am I right ?
, , Am, I, , , right, ?

จงแก้ไขให้ถูกต้อง

ตัวอย่าง : Bar code มาตรฐาน EAN-13

- เป็นมาตรฐาน bar code ประกอบด้วยตัวเลข 13 ตัว
- ตัวทางซ้ายสุดคือตัวเลขพิเศษ (check digit) ซึ่งถูกเพิ่มเพื่อใช้ตรวจสอบความถูกต้อง 12 ตัวทางขวา

$$c = \left(10 - \left(3 \times \sum_{\substack{0 \leq k \leq 11 \\ k \text{ is even}}} d_k + \sum_{\substack{0 \leq k \leq 11 \\ k \text{ is odd}}} d_k \right) \bmod 10 \right) \bmod 10$$

$$(3(0+2+4+6+8+0) + (1+3+5+7+9+8)) = 93$$

$$(10 - (93 \% 10)) \% 10 = 7$$



ตัวอย่าง : การหา check digit ของ EAN-13

- รับตัวเลข 12 หลักทางแป้นพิมพ์
- ใช้สูตรการหา check digit ของ EAN-13
- แสดง check digit ทางจอภาพ
- รับเลข 12 หลักเก็บใส่ int ด้วย nextInt ไม่ได้ (เพราะมีค่าเกิน สองพันล้าน int รับไม่ไหว)
- ต้องรับเป็นสตริงแล้วค่อยใช้ substring ดึงทีละหลัก
- อย่าลืมนำสตริงมาบวกคูณหารไม่ได้
- ต้องเปลี่ยนเป็นจำนวนก่อน ใช้
 - Integer.parseInt(...), Double.parseDouble(...)

```
int v = Integer.parseInt("12");
double d = Double.parseDouble("12.5");
```

โปรแกรมหา check digit ของ EAN-13

```
import java.util.Scanner;

public class EAN13 {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("ตัวเลข 12 หลัก = ");
        String d = kb.nextLine();
        int s = 0;
        s += Integer.parseInt(d.substring(0, 1));
        s += Integer.parseInt(d.substring(2, 3));
        s += Integer.parseInt(d.substring(4, 5));
        s += Integer.parseInt(d.substring(6, 7));
        s += Integer.parseInt(d.substring(8, 9));
        s += Integer.parseInt(d.substring(10, 11));
        s *= 3;
        s += Integer.parseInt(d.substring(1, 2));
        s += Integer.parseInt(d.substring(3, 4));
        s += Integer.parseInt(d.substring(5, 6));
        s += Integer.parseInt(d.substring(7, 8));
        s += Integer.parseInt(d.substring(9, 10));
        s += Integer.parseInt(d.substring(11, 12));
        System.out.println("Check Digit คือ " + (10 - (s % 10)) % 10);
    }
}
```

โปรแกรมหา check digit ของ EAN-13 (ปรับปรุง)

```
import java.util.Scanner;

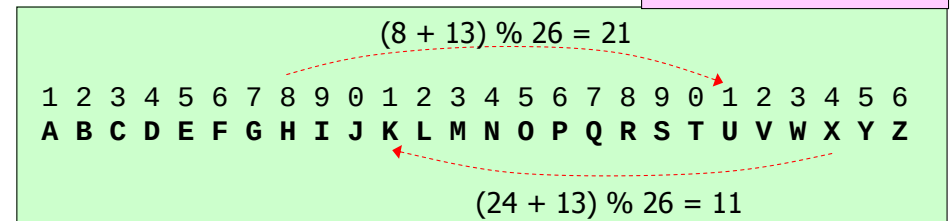
public class EAN13 {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("ตัวเลข 12 หลัก = ");
        String d = kb.nextLine();
        int s = 0, k = 0;
        while (true) {
            int v = Integer.parseInt(d.substring(k, k + 1));
            if (k % 2 == 0) {
                s = s + 3 * v;
            } else {
                s = s + v;
            }
            if (++k > 11) break;
        }
        System.out.println("Check Digit คือ " + (10 - (s % 10)) % 10);
    }
}
```

$$c = \left(10 - \left(3 \times \sum_{\substack{0 \leq k \leq 11 \\ k \text{ is even}}} d_k + \sum_{\substack{0 \leq k \leq 11 \\ k \text{ is odd}}} d_k \right) \bmod 10 \right) \bmod 10$$

โจทย์ : Rot13

- คือการเข้ารหัสข้อมูลง่าย ๆ
 - แทนตัวอักษรหนึ่ง ด้วยอีกตัวที่ห่างจากตัวนั้นไป 13 ตำแหน่ง
 - แทน H (ตัวที่ 8) ด้วย U (ตัวที่ 21 = 8 + 13)
 - แทน X (ตัวที่ 24) ด้วย K (ตัวที่ 11 = (24+13)%26 ต้องวน)
 - ตัวอย่าง : bar ↔ one, purely ↔ cheryl, Somchai ↔ Fbzipunv, ...
Wnin ebpxf ↔ ?
- ในเน็ตมักใช้ rot13 กับข้อความที่ไม่ต้องการให้คนอ่านเข้าใจทันที เช่น เฉลยปริศนา, คำไม่สุภาพ, ...

Julius Caesar ใช้วิธีนี้มานานแล้ว แต่เป็น rot-3



โจทย์ : เขียนโปรแกรม Rot13

```
import java.util.Scanner;

public class Rot13 {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("text = ");
        String text = kb.nextLine();
        String rot13 = "";

        System.out.println("rot13 = " + rot13);
    }
}
```

การอ่านแฟ้มข้อมูล

- ขอแนะนำเฉพาะการอ่านแฟ้มข้อความ (text file)
- ใช้ Scanner ตัวเดียวกับที่ใช้อ่านจากแป้นพิมพ์
- สามารถอ่าน int, double หรืออ่านทั้งบรรทัดเป็นสตริง

```
import java.util.Scanner;
import java.io.*;

public class ReadingTextFile {
    public static void main(String[] args) throws IOException {
        Scanner sc = new Scanner(new File("ReadingTextFile.java"));
        int k = 1, n = 4;
        while (true) {
            String s = sc.nextLine();
            System.out.println(s);
            if (++k > n) break;
        }
    }
}
```

ตรงนั้นบอกว่าถ้าเปิดแฟ้มไม่ได้ ไม่ขอรับผิดชอบ (นอกขอบเขตวิชานี้)

อ่านแค่ 4 บรรทัด

JLab>java ReadingTextFile
import java.util.Scanner;
import java.io.*;

ต้องการอ่านจนหมดแฟ้ม

- ใช้เมทอด `hasNext` เพื่อถามว่ายังมีข้อมูลเหลือหรือไม่
 - `hasNext` คืน `true` ถ้ายังมีข้อมูลเหลือให้อ่าน
 - `hasNext` คืน `false` ถ้าไม่มีข้อมูลเหลือ

```
import java.util.Scanner;
import java.io.*;

public class ReadingTextFile {
    public static void main(String[] args) throws IOException {
        Scanner sc = new Scanner(new File("ReadingTextFile.java"));
        while (true) {
            if (!sc.hasNext()) break;
            String s = sc.nextLine();
            System.out.println(s);
        }
    }
}
```

JLab>java ReadingTextFile
import java.util.Scanner;
import java.io.*;

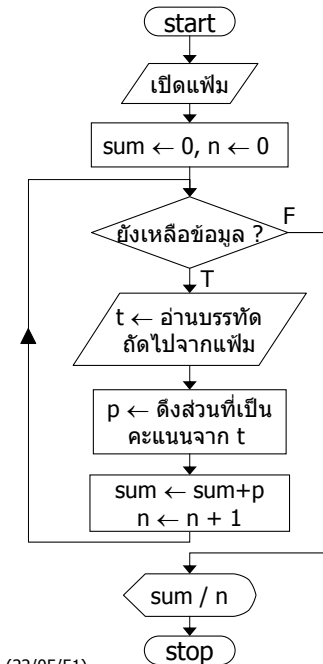
```
public class ReadingTextFile {
    public static void main(String[] args) throws IOException {
        Scanner sc = new Scanner(new File("ReadingTextFile.java"));
        while (true) {
            if (!sc.hasNext()) break;
            String s = sc.nextLine();
            System.out.println(s);
        }
    }
}
```

ถ้าหมดแล้วก็หลุดออกจากวงวน

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

17

ตัวอย่าง : หาค่าเฉลี่ยคะแนนที่เก็บในแฟ้ม



5130120321	74.0
5130293921	85.5
5130294121	58.0
5130338421	90.2
...	

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

18

ตัวอย่าง : หาค่าเฉลี่ยคะแนนที่เก็บในแฟ้ม

```
import java.util.Scanner;
import java.io.*;

public class AverageScore {
    public static void main(String[] args) throws IOException {
        Scanner sc = new Scanner(new File("scores.txt"));
        double sum = 0;
        int n = 0;
        while (true) {
            if (!sc.hasNext()) break;
            String t = sc.nextLine();
            String t1 = t.substring(10, t.length());
            double p = Double.parseDouble(t1.trim());
            sum = sum + p;
            n++;
        }
        System.out.println("คะแนนเฉลี่ย = " + (sum / n));
    }
}
```

5130120321	74.0
5130293921	85.5
5130294121	58.0
5130338421	90.2
...	

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

19

ตัวอย่าง : หาค่าเฉลี่ยคะแนนที่เก็บในแฟ้ม (อีกแบบ)

```
import java.util.Scanner;
import java.io.*;

public class AverageScore2 {
    public static void main(String[] args) throws IOException {
        Scanner sc = new Scanner(new File("scores.txt"));
        double sum = 0;
        int n = 0;
        while (true) {
            if (!sc.hasNext()) break;
            sc.nextDouble(); // อ่านรหัสนิสิต แต่ไม่สนใจ
            double p = sc.nextDouble(); // อ่านคะแนน
            sum = sum + p;
            n++;
        }
        System.out.println("คะแนนเฉลี่ย = " + (sum / n));
    }
}
```

5130120321	74.0
5130293921	85.5
5130294121	58.0
5130338421	90.2
...	

เนื่องจากข้อมูลมีแต่จำนวน
ใช้ `nextDouble` อ่านก็ได้

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

20

โจทย์ : ให้เกรด และคำนวณ class GPA

- อ่านคะแนนของนักเรียนจากแฟ้ม เพื่อให้เกรดแต่ละคนตามเกณฑ์ที่แสดงในตาราง
 - ให้คำนวณน้ำหนักเกรดเฉลี่ยของทั้งชั้น (class GPA) จากสูตร $4A + 3.5B^+ + 3B + 2.5C^+ + 2C + 1.5D^+ + D$
 - A คือจำนวนนักเรียนที่ได้เกรด A
 - B⁺ คือ " " B⁺
 - B คือ " " B
 - ...
- | คะแนน | เกรด | น้ำหนัก |
|-------------|------|---------|
| $x \geq 80$ | A | 4 |

input		output	
5130120321	74.0	5130120321	B
5130293921	85.5	5130293921	A
5130294121	58.0	5130294121	D+
5130338421	90.2	5130338421	A
...		...	
		class GPA = ...	

WordScanner : การอ่านจากแฟ้มทีละคำ

```
import java.util.Scanner;
import java.util.Scanner;

public class WordSeparation {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("ข้อความ = ");
        String t = kb.nextLine();
        WordScanner wsc = new WordScanner(t);
        while (true) {
            if (!wsc.hasNext()) break;
            System.out.print(wsc.nextWord() + ",");
        }
        System.out.println();
    }
}
```

JLab>java WordSeparation
ข้อความ = จงคิดดีทำดีคบคนดีและเลือกไปสถานที่ดี
จ,ง,ค,ิด,ดี,ทำ,ดี,ค,บ,ค,น,ดี,แ,ล,แ,ว,เลือก,ไป,ส,ถ,า,น,ที่,ดี,

ตัวอย่าง : คำใดในเพลงชาติไทยไม่มีในพจนานุกรม

แผ่นดินสยามนามประเทืองว่าเมืองทอง
ไทยเข้าครองตั้งประเทศเขตต์แดนสง่า
สืบเผ่าไทยดึกดำบรรพ์โบราณลงมา
รวมรักษาสามัคคีทวีไทย

บางสมัยสัตว์จุใจมดี
ไทยพลีชีพร่วมรวมรักไล่
เข้าลุยเลือดหมายมุ่งผดุงผะไท
สยามสมัยปรากฏรอดตลอดมา

อันดินสยามคือว่าเนื้อของเชื้อไทย
 น้ำรินไหลคือว่าเลือดของเชื้อข้า
 เอกราษฎร์คือเจดีย์ที่เราบูชา
 เราจะสามัคคีร่วมใจ

รักษาชาติประเทศเอกราชจงดี
ใครย่ำยีเราจะไม่ละให้
เอาเลือดล้างให้สิ้นแผ่นดินของไทย
สถาปนาสยามให้**เทอดไทย**ไชโย

ฉบับ พ.ศ. 2477

เหล่าเราทั้งหลายขอน้อมกายถวายชีวิต
รักษาสีทธีอิสระ ณ แดนสยาม
ที่พ่อแม่สื้อมม้วยด้วยพยายาม
ปราบเสี้ยนหนามให้พินาศสืบชาติมา

แม้ถึงภัยไทยด้อยจนย่อยยับ
ยังถูกลับคงคืนได้ขึ้นหน้า
ควรแก่นามงามสุดอยุธยา
นั่นมิใช่ว่าจะขัดสนหมดคนดี

เหล่าเราทั้งหลายเลือดและเนื้อเชื้อชาติไทย
มิให้ใครเข้าเหยียบย่ำขยาด
ประดับประดาป้องสิทธิ์อิสริ
เมื่อภัยมีช่วยกันจนวันตาย

จะสิ้นชีพไว้อธิษฐานให้
ว่าไทยมันรักชาติไม่
มีไมตรีดียิ่งทั้งหญิง
สยามนิเวศน์ผู้มั่งคั่ง

ประเทศไทยรวมเลือดเนื้อชาติเชื้อไทย
เป็นประชารัฐ ไผทของไทยทุกส่วน
อยู่ดำรงคงไว้ได้ทั้งมวล
ด้วยไทยล้วนหมาย รักสามัคคี
ไทยนี้รักสงบ แต่ถึงรบไม่ขลาด
เอกราชจะไม่ให้ใครข่มขี่
สละเลือดไทยหยาดเป็นชาติพลี
เถลิงประเทศชาติไทยทวี มีชัย ชโย

ฉบับปัจจุบัน

ตัวอย่าง : คำใดในเพลงชาติไทยไม่มีในพจนานุกรม

- ใช้แฟ้ม "thaidict.txt" ซึ่งเก็บคำไทยต่าง ๆ จำนวน 32896 คำในพจนานุกรม (ฉบับราชบัณฑิตฯ)
- คำทั้งหลายถูกนำมาต่อกันยาว 250KB คั่นแต่ละคำด้วย \$ เก็บในแฟ้ม (ทั้งแฟ้มมีเพียงหนึ่งบรรทัด !)

\$ก\$ก็\$กก\$กัก\$กข\$นก\$กข\$ข้าง\$กกร\$รูป\$... \$ไฮโดรมิเตอร์\$ไฮไฟ\$ไฮโล\$ไฮอี\$

- วิธีทำ
 - อ่าน thaidict.txt หนึ่งบรรทัด เก็บใส่สตริง dict
 - ใช้ WordScanner อ่านเนื้อเพลงทีละคำมาค้นใน dict ด้วย indexOf ก็สามารถตรวจสอบได้

ตัวอย่าง : โปรแกรมตรวจเพลงชาติ

จงแก้ไขให้ตรวจสอบเฉพาะคำไทย

```
JLab>java SpellChecking
ด้แดน
เทอดไทย
อยุธยา
```

\$ก\$ก็\$กก\$ก็ก\$กขนก\$กค\$ข้าง\$กค\$รูป\$... \$ไฮโดรมิเตอร์\$ไฮโฟ\$ไฮโล\$ไฮฮิ\$

โจทย์ : "ฉัน" และ "เธอ" ใครมากกว่ากัน

- มีแฟ้ม thaisong.txt ที่เก็บเนื้อเพลงไทย 800 เพลง
- อยากรู้ว่าในเนื้อเพลงไทย มีคำว่า "ฉัน" กับ "เธอ" ปรากฏกี่ครั้ง
- วิธีทำ
 - ใช้ WordScanner เปิดแฟ้ม thaisong.txt
 - มีตัวแปร ใวนับค่าที่สนใจ
 - อ่านมาทีละคำ เปรียบเทียบกับคำที่ต้องการนับ เมื่อพบค่าที่สนใจ เพิ่มค่าตัวแปรอีกหนึ่ง
- แล้วถ้าอยากรู้ว่าคำใดปรากฏมากที่สุด ? (รอเรียนเรื่อง array)

โจทย์ : โปรแกรม YouAndI

```
import jlab.WordScanner;
import java.io.*;

public class YouAndI {
    public static void main(String[] args) throws IOException {
        String thaisong = "c:/temp/thaisong.txt";
        WordScanner wsc = new WordScanner(new File(thaisong));
        String you = "เธอ", I = "ฉัน";
        int countYou = 0, countI = 0;
        while (true) {

        }
        System.out.println("มี \" " + I + "\" " + countI + " คำ");
        System.out.println("มี \" " + you + "\" " + countYou + " คำ");
    }
}
```


2110101 : การทำโปรแกรมคอมพิวเตอร์

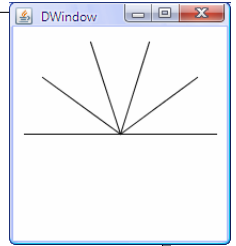
Methods

ภาควิชาวิศวกรรมคอมพิวเตอร์
จุฬาลงกรณ์มหาวิทยาลัย

ตัวอย่าง : โปรแกรมวาดก้านพัด (copy & paste)

```
import jlab.graphics.DWindow;

public class LineDrawing6 {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double x0 = 100, y0 = 100, len = 90;
        double x1, y1, angle = 0, delta = 36;
        //-----
        x1 = x0 + len * Math.cos(Math.toRadians(angle));
        y1 = y0 - len * Math.sin(Math.toRadians(angle));
        w.drawLine(x0, y0, x1, y1);
        angle = angle + delta;
        //-----
        //ทำซ้ำ ๆ แบบ 4 คำสั่งบนนี้อีก 4 ครั้ง
        //-----
        x1 = x0 + len * Math.cos(Math.toRadians(angle));
        y1 = y0 - len * Math.sin(Math.toRadians(angle));
        w.drawLine(x0, y0, x1, y1);
    }
}
```

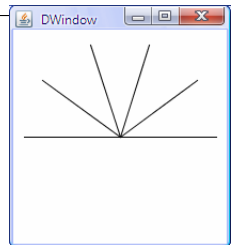


```
import jlab.graphics.DWindow;
public class LineDrawing6 {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double x0 = 100, y0 = 100, len = 90;
        double x1, y1, angle = 0, delta = 36;
        x1 = x0 + len * Math.cos(Math.toRadians(angle));
        y1 = y0 - len * Math.sin(Math.toRadians(angle));
        w.drawLine(x0, y0, x1, y1);
        angle = angle + delta;
        x1 = x0 + len * Math.cos(Math.toRadians(angle));
        y1 = y0 - len * Math.sin(Math.toRadians(angle));
        w.drawLine(x0, y0, x1, y1);
        angle = angle + delta;
        x1 = x0 + len * Math.cos(Math.toRadians(angle));
        y1 = y0 - len * Math.sin(Math.toRadians(angle));
        w.drawLine(x0, y0, x1, y1);
        angle = angle + delta;
        x1 = x0 + len * Math.cos(Math.toRadians(angle));
        y1 = y0 - len * Math.sin(Math.toRadians(angle));
        w.drawLine(x0, y0, x1, y1);
        angle = angle + delta;
        x1 = x0 + len * Math.cos(Math.toRadians(angle));
        y1 = y0 - len * Math.sin(Math.toRadians(angle));
        w.drawLine(x0, y0, x1, y1);
        angle = angle + delta;
        x1 = x0 + len * Math.cos(Math.toRadians(angle));
        y1 = y0 - len * Math.sin(Math.toRadians(angle));
        w.drawLine(x0, y0, x1, y1);
    }
}
```

ตัวอย่าง : โปรแกรมวาดก้านพัด (ใช้วงวน)

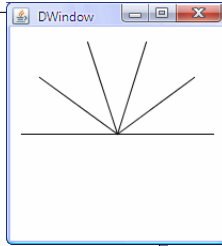
```
import jlab.graphics.DWindow;

public class LineDrawing6L {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double x0 = 100, y0 = 100, len = 90;
        double x1, y1, angle = 0, delta = 36;
        int k = 1, n = 6;
        while (true) {
            x1 = x0 + len * Math.cos(Math.toRadians(angle));
            y1 = y0 - len * Math.sin(Math.toRadians(angle));
            w.drawLine(x0, y0, x1, y1);
            angle = angle + delta;
            if (++k > n) break;
        }
    }
}
```



ตัวอย่าง : โปรแกรมวาดก้านพัด (ใช้เมทอด)

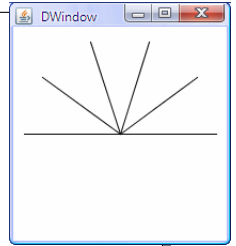
```
import jlab.graphics.DWindow;
public class LineDrawing6M {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double x0 = 100, y0 = 100, len = 90;
        double angle = 0, delta = 36;
        //-----
        drawALine(w, x0, y0, len, angle);
        drawALine(w, x0, y0, len, (angle = angle+delta));
        drawALine(w, x0, y0, len, (angle = angle+delta));
        drawALine(w, x0, y0, len, (angle = angle+delta));
        drawALine(w, x0, y0, len, (angle = angle+delta));
        drawALine(w, x0, y0, len, (angle = angle+delta));
        //-----
    }
    public static void drawALine(DWindow w, double x, double y,
                                double length, double a) {
        double x1 = x + length * Math.cos(Math.toRadians(a));
        double y1 = y - length * Math.sin(Math.toRadians(a));
        w.drawLine(x, y, x1, y1);
    }
}
```



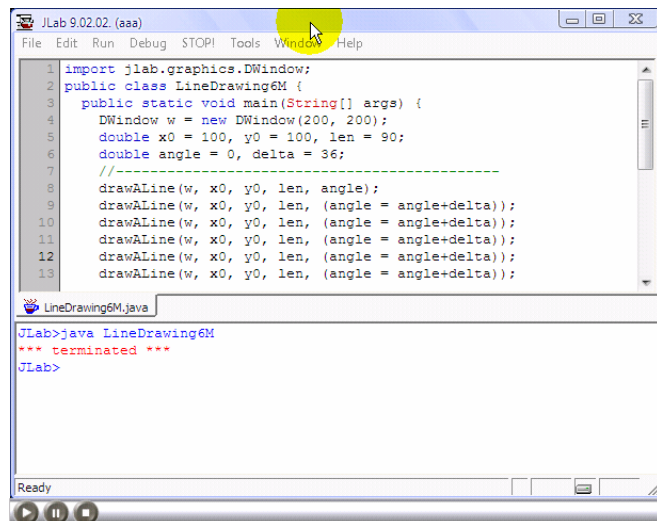
ตัวอย่าง : โปรแกรมวาดก้านพัด (ใช้วงวน+เมทอด)

```
import jlab.graphics.DWindow;

public class LineDrawing6LM {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double x0 = 100, y0 = 100, len = 90;
        double x1, y1, angle = 0, delta = 36;
        int k = 1, n = 6;
        while (true) {
            drawALine(w, x0, y0, len, angle);
            if (++k > n) break;
            angle = angle + delta;
        }
    }
    public static void drawALine(DWindow w, double x, double y,
                                double length, double a) {
        double x1 = x + length * Math.cos(Math.toRadians(a));
        double y1 = y - length * Math.sin(Math.toRadians(a));
        w.drawLine(x, y, x1, y1);
    }
}
```



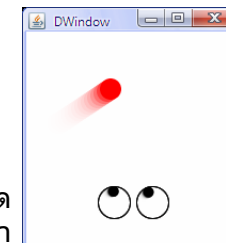
ตัวอย่าง : การทำงานระหว่างการเรียกเมทอด



การเขียนโปรแกรม

- แบ่งโปรแกรมเป็นส่วนย่อยๆ
- แยกส่วนของโปรแกรมที่ซ้ำๆ คล้ายๆ กันออกมา
- เรียกส่วนย่อยเหล่านี้ว่า subroutine, subprogram, function หรือ method
- แต่ละ method มีหน้าที่ในตัวเองอย่างเด่นชัด

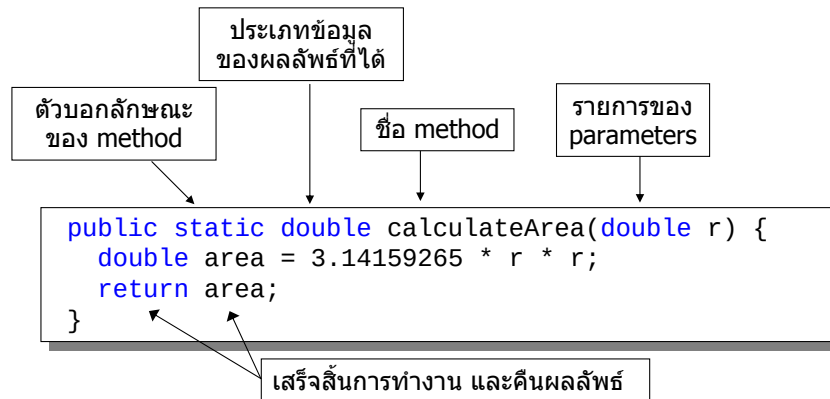
```
class Game {
    main(...) {
        ...
    }
    demo(...) {
    }
    option(...) {
    }
    begin(...) {
    }
    ending(...) {
    }
    ...
}
```



น่าจะเมทอด
วาดลูกตา

องค์ประกอบของเมทอด

- หัวเมทอด : ลักษณะการใช้งานของเมทอด
- ตัวเมทอด : ตัวแปรที่ใช้, การทำงาน, การคืนผลลัพธ์



หัวเมทอด : ประเภทข้อมูลของผลลัพธ์

- ระบุว่าเรียกเมทอดแล้วจะได้ข้อมูลประเภทใด (int, double, boolean, String, ...)
- ถ้าเป็น method ที่ไม่คืนผลลัพธ์ ให้ใส่คำว่า void

```
int m = min(a, 0);

public static int min(int a, int b) {
    ...
}
```

```
double a = calculateArea(5);

public static double calculateArea(double r) {
    ...
}
```

```
w.fade(0.2);

public void fade(double level) {
    ...
}

String t = word.trim();

public String trim() {
    ...
}
```

หัวเมทอด : ชื่อ

- การตั้งชื่อใช้กฎเหมือนกับตัวแปร
 - ประกอบด้วยตัวอักษร ตัวเลข ตัว \$ หรือ _ ก็ได้
 - ห้ามขึ้นต้นด้วยตัวเลข
 - ไม่มีข้อจำกัดเรื่องความยาวของชื่อ
 - ตัวอักษรตัวใหญ่ไม่เหมือนตัวเล็ก
 - ต้องไม่ซ้ำกับคำสงวนของภาษาจาวา
- โดยทั่วไปชื่อเมทอดขึ้นต้นด้วยตัวเล็ก
- มักตั้งชื่อเมทอดให้เป็นกริยา

```
fillEllipse drawLine fade turnLeft println start run
```

หัวเมทอด : รายการของพารามิเตอร์

- parameter เป็นตัวแปรสำหรับรับข้อมูลเข้ามาประมวลผล
- parameter แต่ละตัว เขียนเหมือนการประกาศตัวแปร (แต่ไม่ต้องปิดท้ายด้วย semi-colon)
- คั่น parameter แต่ละตัว ด้วยจุลภาค (comma)
- รายการของ parameters อยู่ภายในวงเล็บ (ถ้าไม่รับ parameter ใดๆ ก็ไม่ต้องใส่อะไรในวงเล็บ)

```
public void turnLeft(double angle) { ... }
```

```
public static double pow(double a, double b) { ... }
```

```
public String nextLine( ) { ... }
```

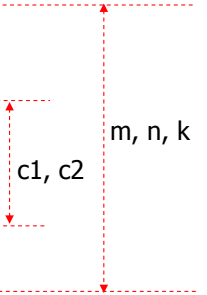
ตัวเมทอด : การคืนการทำงานสู่ผู้เรียก

- เมทอดทำงานเสร็จและคืนการทำงานกลับ เมื่อ
 - ทำคำสั่ง `return` หรือ
 - ทำจนถึง `}` ที่ปิดตัวเมทอด
- เมทอดที่มีการคืนผล
 - หลังคำสั่ง `return` ต้องตามด้วยนิพจน์ที่คืนผล
 - คืนผลได้เพียงค่าเดียวเท่านั้น

ตัวเมทอด : Local Variables

- สร้างใหม่ทุกครั้งที่ทำงานถึงจุดที่ใช้ครั้งแรก
- ถูกทำลายหลังจบ block ที่ประกาศไว้
- ตัวแปรที่ประกาศใน block ใดใช้ได้เฉพาะใน block นั้น

```
public static boolean isPalindrome(String w) {  
    int m = w.length() - 1, n = m / 2, k = 0;  
    while (true) {  
        if (k > n) break;  
        String c1 = w.substring(k, k + 1);  
        String c2 = w.substring(m - k, m - k + 1);  
        if (c1.compareTo(c2) != 0) return false;  
        k++;  
    }  
    return true;  
}
```



สิ่งที่มักผิดพลาด

```
public static printError(String msg) {  
    System.out.println( msg );  
}
```

```
public static int abs(int a) {  
    if ( a < 0 ) a = -a;  
}
```

```
public static double max(double x, y) {  
    if ( x > y ) return x; else return y;  
}
```

```
public static double length(double dx, double dy) {  
    double dx = Math.abs(dx);  
    return Math.sqrt( dx*dx + dy*dy );  
}
```

```
public static int getBlackColor() {  
    return 0.0;  
}
```

ตัวอย่าง : เมทอดแปลงหน่วย

```
public static double toRadian(double angle) {  
    double r = 3.141592653589793 / 180 * angle;  
    return r;  
}
```

```
public static double toAngle(double radian) {  
  
}
```

```
public static double toFahrenheit(double c) {  
    return c * 9.0 / 5.0 + 32;  
}
```

```
public static double toCelsius(double f) {  
  
}
```

โจทย์ : เดือน - วัน

```
public static int dayOfWeek(int d, int m, int y) {  
  
}
```

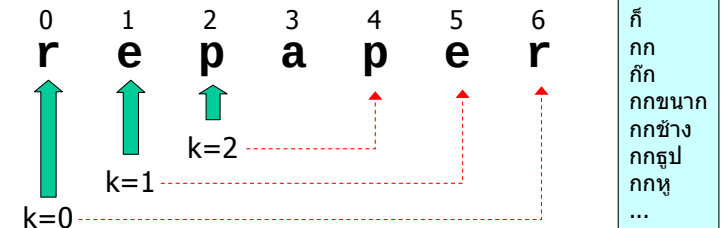
วันอะไร อาทิตย์ หรือจันทร์ หรือ ... หรือเสาร์

```
public static int daysInMonth(int m, int y) {  
  
}
```

เดือนนี้มีกี่วัน

ตัวอย่าง : โปรแกรมหาพาลินโดรมในพจนานุกรมไทย

- พาลินโดรม (Palindrome) คือ
 - คำที่มีตัวอักษรที่ปรากฏจากซ้ายไปขวาเหมือนขวามาซ้าย
 - เช่น boob, mem, oho, repaper, ..., กก, ทายาท, ...
- วิธีหาพาลินโดรมในคำไทย
 - มี thaidic.txt ที่เก็บคำไทยในพจนานุกรม 32,896 คำ
 - แฟ้มนี้เก็บบรรทัดละคำ
 - ค่อย ๆ อ่านมาทีละคำ แล้วเรียกเมทอด isPalindrome(w) เพื่อตรวจสอบว่า w เป็นพาลินโดรมหรือไม่



เมทอดหาพาลินโดรมในพจนานุกรม

```
import java.util.Scanner;  
import java.io.*;  
public class Palindrome {  
    public static void main(String[] args) throws IOException {  
        Scanner sc = new Scanner(new File("c:/temp/thaidic.txt"), "UTF-8");  
        int k = 4;  
        while (true) {  
            if (!sc.hasNext()) break;  
            String w = sc.nextLine();  
            w = w.trim();  
            if (w.length() > k && isPalindrome(w)) System.out.print(w + ",");  
            System.out.println();  
        }  
        public static boolean isPalindrome(String w) {  
            int m = w.length(), n = m / 2 - 1, k = 0;  
            while (true) {  
                if (k > n) break;  
                String c1 = w.substring(k, k + 1);  
                String c2 = w.substring(m - k - 1, m - k);  
                if (c1.compareTo(c2) != 0) return false;  
                k++;  
            }  
            return true;  
        }  
    }  
}
```

พจนานุกรมฉบับราชบัณฑิตยฯ บรรทัดละคำ

สองเมทอดนี้มีตัวแปรชื่อเดียวกัน แต่เป็นคนละตัว

พบคู่ที่ไม่เท่า คินเท็จ

เท่าทุกคู่ คินจริง

JLab>java Palindrome
กอดอก,กาฝาก,ทายาท,

โจทย์ : abecedarian

- คำที่เป็น abecedarian (อ่านว่า a-b-c-darian) คือคำที่แต่ละตัวอักษรเรียงตามลำดับจากซ้ายไปขวา เช่น

G H O S T
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

BOY ใช่ แต่ GIRL ไม่ใช่

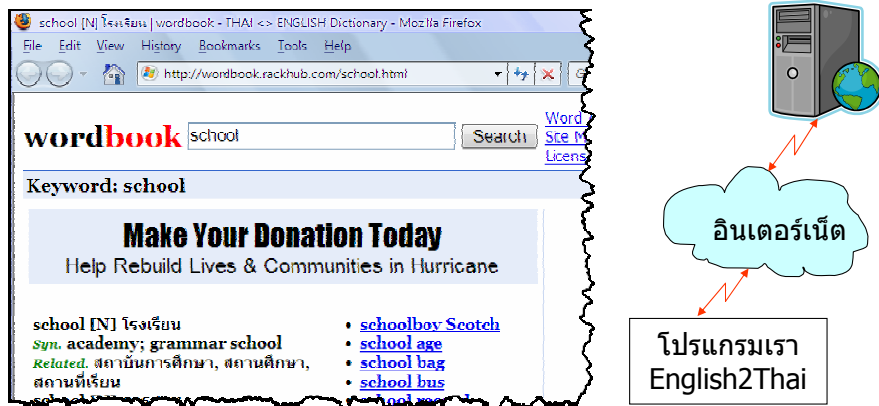
- จงเขียนเมทอด isAbecedarian ที่ตรวจสอบว่าสตริง w ที่ได้รับ เป็น abecedarian หรือไม่

```
public static boolean isAbecedarian(String w) {  
  
}
```

อย่าลืมใช้ compareTo ในการเปรียบเทียบสตริง

ตัวอย่าง : โปรแกรมแปลคำอังกฤษเป็นคำไทย

- ใช้บริการเปิดพจนานุกรมที่ <http://wordbook.rackhub.com>



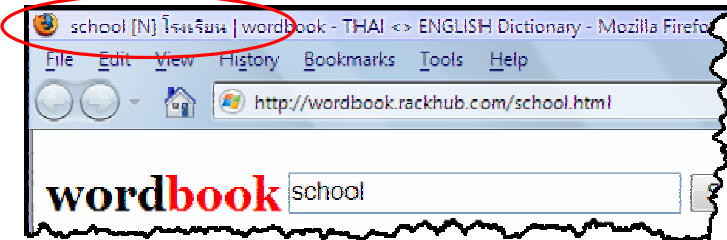
school แปลว่าอะไร ไปที่ <http://wordbook.rackhub.com/school.html>
java แปลว่าอะไร ไปที่ <http://wordbook.rackhub.com/java.html>
program แปลว่าอะไร ไปที่ <http://wordbook.rackhub.com/program.html>
...

ตัวอย่าง : ดึงข้อมูลจาก web page

แฟ้ม html ที่ได้รับเมื่อไปที่ <http://wordbook.rackhub.com/school.html>

```
<!DOCTYPE html ...  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head>  
<base href="http://wordbook.rackhub.com/" />  
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />  
<title>school [N] โรงเรียน | wordbook = THAI <> ENGLISH Dictionary - ...
```

เพื่อความง่าย จะขอดึงค่าแปลจาก title



ภาพที่ web browser แสดง

ตัวอย่าง : โปรแกรมแปลคำอังกฤษเป็นคำไทย

```
import java.util.Scanner;  
import jlab.URLStream;  
  
public class English2Thai {  
    public static void main(String[] args) {  
        Scanner kb = new Scanner(System.in);  
        System.out.print("english word = ");  
        String english = kb.nextLine().trim();  
        String thai = english2thai(english);  
        if (thai.length() > 0) {  
            System.out.println(english + " --> " + thai);  
        } else {  
            System.out.println("ไม่พบคำแปล");  
        }  
    }  
    //-----  
    public static String english2thai(String english) {  
        ...  
    }  
}
```

```
public static String english2thai(String english) {  
    english = english.toLowerCase();  
    String url = "http://wordbook.rackhub.com/" + english + ".html";  
    Scanner sc = new Scanner(new URLStream(url), "UTF-8");  
    while (true) {  
        if (!sc.hasNext()) break;  
        String t = sc.nextLine();  
        t = t.toLowerCase();  
        int j = t.indexOf("<title>");  
        if (j >= 0) {  
            j = t.indexOf("]", j+7);  
            if (j >= 0) {  
                int k = t.indexOf("|", j+1);  
                if (k >= 0) {  
                    String thai = t.substring(j+1, k);  
                    return thai.trim();  
                }  
            }  
        }  
    }  
    return "";  
}
```

http://wordbook.rackhub.com/school.html

<title>school [N] โรงเรียน | ...

```
<!DOCTYPE html ...  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head>  
<base href="http://wordbook.rackhub.com/" />  
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />  
<title>school [N] โรงเรียน | ...  
...
```

การเรียกใช้เมทอด

- ส่งข้อมูลแต่ละตัวให้ครบและถูกประเภท
- เตรียมตัวแปรรับผลลัพธ์ให้ถูกต้อง (หรือไม่ก็เปลี่ยนประเภทให้ถูกต้อง)

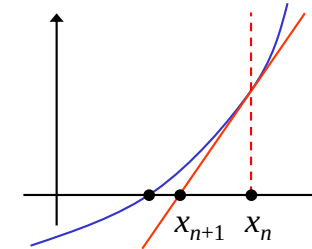
```
double p = distance(10, 10, 20, 20.5);
int q = distance(10, 10, 20, 30);
int r = (int) distance(20, 30, 45, 45);
double s = distance(20, 30, 45, (int) p);
```

บรรทัดใดผิด ?

```
public static double distance(int x1, int y1, int x2, int y2) {
    double dx = (x1 - x2);
    double dy = (y1 - y2);
    double d = Math.sqrt(dx*dx + dy*dy);
    return d;
}
```

ตัวอย่าง : การหารากโดยวิธีของนิวตัน

- เช่นต้องการหารากของสมการ $f(x) = 2x^2 - 11x + 5$
- หาอนุพันธ์เตรียมไว้ก่อน $f'(x) = 4x - 11$
- เริ่มด้วยการให้ค่าเริ่มต้นกับ x_0
- คำนวณ $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$
- วนคำนวณจนค่า x_n และ x_{n+1} เปลี่ยนแปลงน้อยมาก ๆ



ตัวอย่าง : โปรแกรมการหารากโดยวิธีของนิวตัน

```
import java.util.Scanner;
public class Newton {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("initial x = ");
        double x0 = kb.nextDouble();
        double err = 0;
        do {
            System.out.println(x0);
            double x1 = x0 - f(x0) / df(x0);
            err = Math.abs(x1 - x0);
            x0 = x1;
        } while (err > 1e-10);
        System.out.println(x0);
    }
    public static double f(double x) {
        return 2 * x * x - 11 * x + 5;
    }
    public static double df(double x) {
        return 4 * x - 11;
    }
}
```

```
JLab>java Newton
initial x = 2
2.0
-1.0
0.19999999
0.48235294
0.49993133
0.49999999
0.5
0.5
```

```
JLab>java Newton
initial x = 10
10.0
6.724137931
5.373999551
5.026653141
5.000156016
5.000000005
5.0
5.0
```

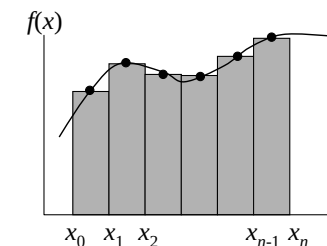
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

หมายเหตุ โปรแกรมนี้
หาเฉพาะรากจริง

$$f(x) = 2x^2 - 11x + 5 = (2x - 1)(x - 5)$$

ตัวอย่าง : Integration

- แบ่งพื้นที่ใต้กราฟเป็นแถบๆ
- หาพื้นที่ของแต่ละแถบ แล้วหาผลรวม
- ใช้ midpoint rule ประมาณพื้นที่ใต้กราฟดังนี้



$$h = \frac{b - a}{n}$$

$$\int_a^b f(x) dx \approx \sum_{i=1}^n hf \left(\frac{x_{i-1} + x_i}{2} \right)$$

Integration using Midpoint Rule

```
public class Integration {
    public static void main(String[] args) {
        System.out.println(midpoint(0, 10, 1000));
    }
    public static double midpoint(double a, double b, int n) {
        double h, s1, x;
        h = (b - a) / n;
        s1 = 0.0;
        x = a + h / 2.0;
        for (int i = 1; i <= n; i++) {
            s1 += h*f(x);
            x += h;
        }
        return s1;
    }
    public static double f(double x) {
        return 2 * x * x;
    }
}
```

$$\int_a^b f(x) dx \approx \sum_{i=1}^n hf \left(\frac{x_{i-1} + x_i}{2} \right)$$

Method Overloading

- ชื่อเมทอดซ้ำได้ ถ้ารายการพารามิเตอร์ไม่เหมือนกัน
- ระบบจะเรียกเมทอดที่เหมาะสม

```
int a, b;
double x, y;
...
int m = max(a, b, 0);
double z = max(0, x);

public static int max(int x, int y) {
    if (x > y) return x; else return y;
}
public static int max(int x, int y, int z) {
    return max(max(x, y), z);
}
public static int max(int w, int x, int y, int z) {
    return max(max(w, x), max(y, z));
}
public static double max(double x, double y) {
    if (x > y) return x; else return y;
}
public static double max(double x, double y, double z)
{
    return max(x, max(y, z));
}
```

Method Overloading

- ไม่สามารถตั้งชื่อเมทอดซ้ำ โดยที่รายการพารามิเตอร์เหมือน แต่ประเภทข้อมูลที่คืนไม่เหมือน
- ความเหมือนไม่เหมือนของพารามิเตอร์ อยู่ที่ประเภทข้อมูล ไม่เกี่ยวกับชื่อที่ใช้

```
public static int max(int x, int y) {
    ...
}

public static double max(int x, int y) {
    ...
}

public static int max(int a, int b) {
    ...
}
```

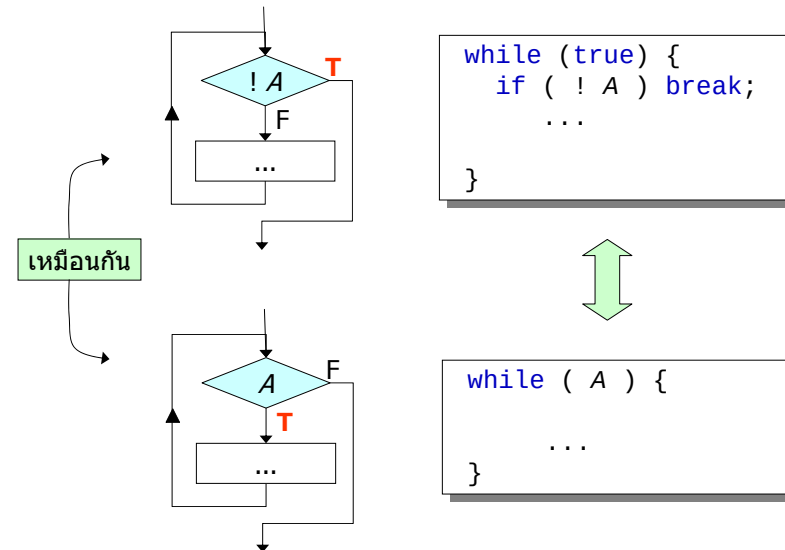
ผิด

2110101 : การทำโปรแกรมคอมพิวเตอร์

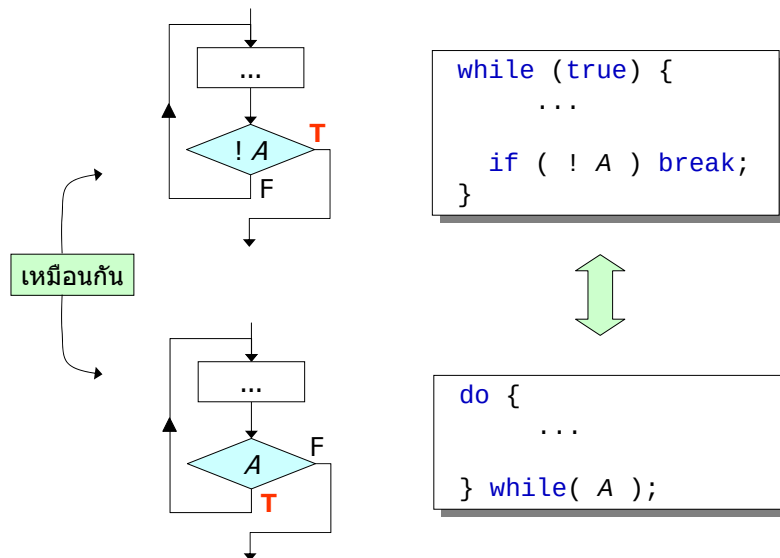
Iteration : while, do-while, for

ภาควิชาวิศวกรรมคอมพิวเตอร์
จุฬาลงกรณ์มหาวิทยาลัย

วงวนแบบ while



วงวนแบบ do-while



ตัวอย่าง : โปรแกรมหาคะแนนเฉลี่ย

```
Scanner sc = new Scanner(new File("scores.txt"));  
double sum = 0; int n = 0;  
while (true) {  
    if ( !sc.hasNext() ) break;  
    sc.nextDouble(); // อ่านรหัสนิสิต แต่ไม่สนใจ  
    double p = sc.nextDouble(); // อ่านคะแนน  
    sum = sum + p;  
    n++;  
}  
System.out.println("คะแนนเฉลี่ย = " + (sum / n));
```

5130120321	24.0
5130293921	25.5
5130294121	28.0
5130338421	30.2
...	

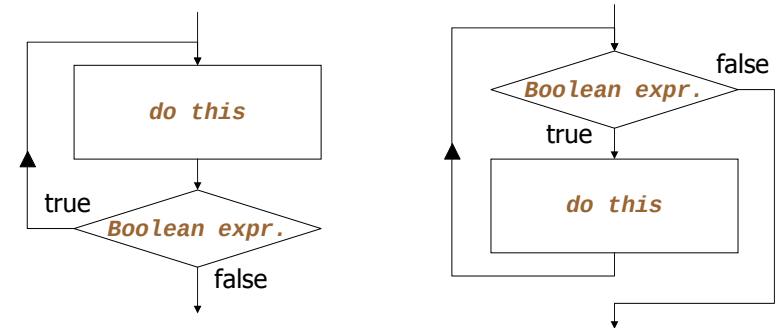
```
Scanner sc = new Scanner(new File("scores.txt"));  
double sum = 0; int n = 0;  
while ( sc.hasNext() ) {  
    sc.nextDouble(); // อ่านรหัสนิสิต แต่ไม่สนใจ  
    double p = sc.nextDouble(); // อ่านคะแนน  
    sum = sum + p;  
    n++;  
}  
System.out.println("คะแนนเฉลี่ย = " + (sum / n));
```

ตัวอย่าง : โปรแกรมหารากที่สอง

```
Scanner kb = new Scanner(System.in);
System.out.print("a = ");
double a = kb.nextDouble();
double x = 1.0;
while ( true ) {
    x = (x + (a / x)) / 2.0;
    if ( (x*x - a) < 1e-5 ) break;
}
System.out.println("รากที่สองของ " + a + " = " + x);
```

```
Scanner kb = new Scanner(System.in);
System.out.print("a = ");
double a = kb.nextDouble();
double x = 1.0;
do {
    x = (x + (a / x)) / 2.0;
} while ( (x*x - a) >= 1e-5 );
System.out.println("รากที่สองของ " + a + " = " + x);
```

วงวนแบบ do-while และ while



```
do {
    do this
} while ( Boolean expr );
```

```
while ( Boolean expr ) {
    do this
}
```

ตัวอย่าง : ใครได้คะแนนมากที่สุด

```
import java.util.Scanner;
import java.io.*;

public class MaxScore {
    public static void main(String[] args) throws IOException {
        Scanner sc = new Scanner(new File("scores.txt"));
        double max = 0;
        String maxID = "";
        while ( sc.hasNext() ) {
            String t = sc.nextLine();
            String id = t.substring(0, 10);
            String t1 = t.substring(11, t.length());
            double p = Double.parseDouble(t1.trim());
            if ( p > max ) {
                max = p;
                maxID = id;
            }
        }
        System.out.println( maxID + " ได้คะแนนมากที่สุด = " + max );
    }
}
```

```
5130120321 24.0
5130293921 25.5
5130294121 28.0
5130338421 20.2
...
```

พบคะแนนใหม่มากกว่า
คะแนนมากที่สุดที่ได้พบมา

วงวนที่ทำเป็นจำนวนครั้งตามที่กำหนด

```
int i = 1;
while (true) {
    if (i > n) break;
    // ...
    i++;
}
```

i = 1, 2, 3, ..., n

```
int i = 1;
while (i <= n) {
    // ...
    i++;
}
```

i = 1, 2, 3, ..., n

```
int i = 0;
while (i < n) {
    // ...
    i++;
}
```

i = 0, 1, 2, ..., n-1

```
int i = n;
while (true) {
    if (i <= 0) break;
    // ...
    i--;
}
```

i = n, ..., 3, 2, 1

```
int i = n;
while (i > 0) {
    // ...
    i--;
}
```

i = n, ..., 3, 2, 1

```
int i = n-1;
while (i >= 0) {
    // ...
    i--;
}
```

i = n-1, ..., 2, 1, 0

จำนวนรอบอาจขาดหรือเกิน ต้องระวังการกำหนดค่าเริ่มต้น และเงื่อนไข

วงวนแบบ for

- เหมาะสำหรับวงวนที่ทำเป็นจำนวนครั้งที่กำหนด

```
int i = 0;
while (i < n) {
    // ...
    i++;
}
```

$i = 0, 1, 2, \dots, n-1$

```
int i;
for (i = 0; i < n; i++) {
    // ...
}
```

$i = 0, 1, 2, \dots, n-1$

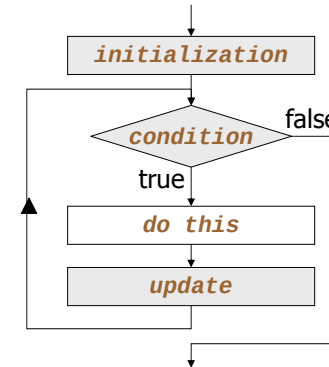
```
int i = n;
while (i > 0) {
    // ...
    i--;
}
```

$i = n, \dots, 3, 2, 1$

```
int i;
for (i = n; i > 0; i--) {
    // ...
}
```

$i = n, \dots, 3, 2, 1$

วงวนแบบ for



```
for ( initialization ; condition ; update ) {
    do this
}
```

ประกาศตัวแปรที่ใช้ภายใน for เท่านั้น

```
int i;
for (i = 0; i < n; i++) {
    // ...
}
```

```
for (int i = 0; i < n; i++) {
    // ใช้ i ได้เฉพาะในวงวน
}
```

```
for (int i = 0; i < 10; i = i+2) {
    System.out.println(i);
}

System.out.println(i);

for (int i = 1; i < 10; i = i+2) {
    System.out.println(i);
}
```

0, 2, 4, 6, 8



1, 3, 5, 7, 9

หนึ่ง block หรือ หนึ่ง statement

```
for ( initialization ; condition ; update ) {
    statements
}
```

```
for ( initialization ; condition ; update )
    a statement ;
```

```
do {
    statements
} while ( Boolean expr );
```

```
while ( Boolean expr ) {
    statements
}
```

```
do
    a statement ;
while ( Boolean expr );
```

```
while ( Boolean expr )
    a statement ;
```

ตัวอย่าง : เลขฟีโบนัชชี (Fibonacci)



Fibonacci
(1175-1250)

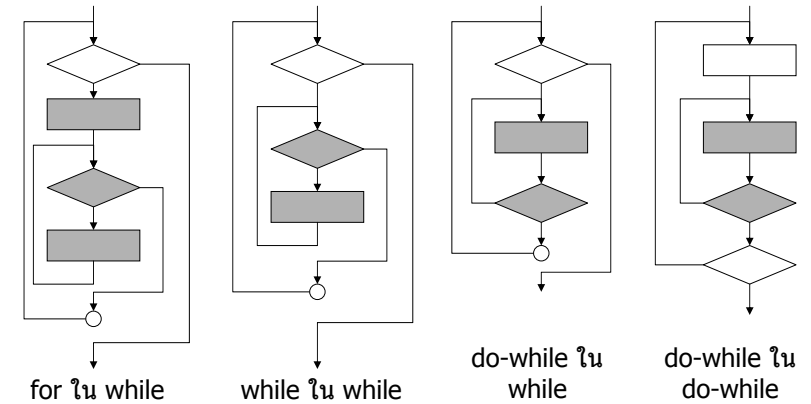
- 0, 1, 1, 2, 3, 5, 8, 13, 21, ...
- $f_n = f_{n-1} + f_{n-2}$ เมื่อ $n \geq 2$, $f_0 = 0$, $f_1 = 1$

```
import java.util.Scanner;
public class Fibonacci {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("n = ");
        int n = kb.nextInt();
        int fn = n, fn1 = 1, fn2 = 0;
        for (int i = 2; i <= n; i++) {
            fn = fn1 + fn2;
            fn2 = fn1;
            fn1 = fn;
        }
        System.out.println("f(" + n + ") = " + fn);
    }
}
```

```
JLab>java Fib
n = 30
f(30) = 832040
```

Nested Loops

- ภายในวงวนหนึ่งจะมีวงวนอื่นก็ได้



ตัวอย่าง : ตารางสูตรคูณ

```
public class MTable {
    public static void main(String[] args) {
        for (int i = 2; i <= 3; i++) {
            for (int j = 1; j <= 12; j++) {
                System.out.println(i + "x" + j +
                    "\t= " + (i * j));
            }
            System.out.println("-----");
        }
    }
}
```

```
JLab>java MTable
2x1 = 2
2x2 = 4
2x3 = 6
2x4 = 8
2x5 = 10
2x6 = 12
2x7 = 14
2x8 = 16
2x9 = 18
2x10 = 20
2x11 = 22
2x12 = 24
-----
3x1 = 3
3x2 = 6
3x3 = 9
3x4 = 12
3x5 = 15
3x6 = 18
3x7 = 21
3x8 = 24
3x9 = 27
3x10 = 30
3x11 = 33
3x12 = 36
-----
```

ตัวอย่าง : การหาค่า cosine

- ใช้ $\text{Math.cos}(x)$, x เป็นเรเดียน
- แต่เราจะใช้สูตร

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \frac{x^8}{8!} - \frac{x^{10}}{10!} + \dots + (-1)^k \frac{x^{2k}}{(2k)!} + \dots$$

$$\cos(x) = \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k}}{(2k)!}$$

- วิธีทำ
 - คำนวณค่า $(-1)^k \frac{x^{2k}}{(2k)!}$?

โปรแกรม : แบบที่ 1

```
import java.util.Scanner;
public class Cosine {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("มุม (องศา) = ");
        double a = kb.nextDouble();
        double x = Math.toRadians(a);
        double cos = 1.0;
        for (int k = 1; k < 10; k++) {
            double s = 1.0;
            for (int j = 1; j <= k; j++) s = s * (-1);
            double b = 1.0;
            for (int j = 1; j <= (2 * k); j++) b = b * x;
            double c = 1.0;
            for (int j = 1; j <= (2 * k); j++) c = c * j;
            cos = cos + s * b / c;
        }
        System.out.println("Math.cos = " + Math.cos(x));
        System.out.println("cos = " + cos);
    }
}
```

$$\cos(x) = \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k}}{(2k)!}$$

อาจต้องหมุนมากหรือน้อยกว่า 10 รอบ

$(-1)^k$

x^{2k}

$(2k)!$

JLab>java Cosine1

มุม (องศา) = 400

Math.cos = 0.7660444431189781

cos = 0.7379649659255636

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

17

โปรแกรม : แบบที่ 2

```
public static void main(String[] args) {
    Scanner kb = new Scanner(System.in);
    System.out.print("มุม (องศา) = ");
    double a = kb.nextDouble();
    double x = Math.toRadians(a);
    double cos = 1.0;
    double term = 0;
    int k = 1;
    do {
        double s = 1.0;
        for (int j = 1; j <= k; j++) s = s * (-1);
        double b = 1.0;
        for (int j = 1; j <= (2 * k); j++) b = b * x;
        double c = 1.0;
        for (int j = 1; j <= (2 * k); j++) c = c * j;
        term = s * b / c;
        cos = cos + term;
        k++;
    } while (Math.abs(term) > 1e-10);
    System.out.println("Math.cos = " + Math.cos(x));
    System.out.println("cos = " + cos);
}
```

$$\cos(x) = \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k}}{(2k)!}$$

วนต่อ ตราบเท่าที่พจน์ใหม่ยังไม่เล็กพอ

JLab>java Cosine2

มุม (องศา) = 400

Math.cos = 0.7660444431189781

cos = 0.766044443119205

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

18

โปรแกรม : แบบที่ 3

```
public static void main(String[] args) {
    Scanner kb = new Scanner(System.in);
    System.out.print("มุม (องศา) = ");
    double a = kb.nextDouble();
    double x = Math.toRadians(a);
    double cos = 1.0;
    double term = 0;
    int k = 1;
    do {
        term = 1.0;
        for (int j = 1; j <= k; j++) {
            term = term * (-1) * x * x / ((2 * j - 1) * (2 * j));
        }
        cos = cos + term;
        k++;
    } while (Math.abs(term) > 1e-10);
    System.out.println("cos(" + a + ") = " + Math.cos(x));
    System.out.println("cos(" + a + ") = " + cos);
}
```

$$\cos(x) = \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k}}{(2k)!}$$

วงวนคำนวณพจน์ที่ k

JLab>java Cosine3

มุม (องศา) = 400

Math.cos = 0.7660444431189781

cos = 0.7660444431192086

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

19

โปรแกรม : แบบสุดท้าย

```
public static void main(String[] args) {
    Scanner kb = new Scanner(System.in);
    System.out.print("มุม (องศา) = ");
    double a = kb.nextDouble();
    double x = Math.toRadians(a);
    double cos = 1.0;
    double term = 1.0;
    int k = 1;
    do {
        term = term * (-1) * x * x / ((2 * k - 1) * (2 * k));
        System.out.println(term);
        cos = cos + term;
        k++;
    } while (Math.abs(term) > 1e-10);
    System.out.println("Math.cos = " + Math.cos(x));
    System.out.println("cos = " + cos);
}
```

คำนวณพจน์ที่ k จากพจน์ที่ k-1

$$(-1)^k \frac{x^{2k}}{(2k)!} = \left((-1)^{k-1} \frac{x^{2(k-1)}}{(2(k-1))!} \right) (-1) \frac{x^2}{(2k-1)(2k)}$$

JLab>java Cosine4

มุม (องศา) = 400

Math.cos = 0.7660444431189781

cos = 0.7660444431192086

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

20

โจทย์

เขียนโปรแกรมเพื่อคำนวณค่า $\sin(x)$, e^x ด้วยสูตร

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots + (-1)^k \frac{x^{2k+1}}{(2k+1)!} + \dots$$

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \dots + \frac{x^k}{k!} + \dots$$

โจทย์ : วงกลมหลายวง

```
import jlab.graphics.DWindow;

public class Circles {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        int i = 1;
        for (int x = 20; x < 200; x = x + 40) {
            for (int y = 20; y < 200; y = y + 40) {
                w.fillEllipse(w.RED, x, y, 30, 30);
                w.drawString(w.WHITE, ""+(i++), 18, x-8, y-10);
            }
        }
    }
}
```

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

สตริงที่จะแสดง

ขนาด

จงแก้ไขโปรแกรม
ให้แสดงรูปทางขวานี้

1
2 3
4 5 6
7 8 9 10
11 12 13 14 15

●	●	●	●	●
●	●	●	●	●
●	●	●	●	●
●	●	●	●	●
●	●	●	●	●

●	●	●	●	●
●	●	●	●	●
●	●	●	●	●
●	●	●	●	●
●	●	●	●	●

2110101 : การทำโปรแกรมคอมพิวเตอร์

อาเรย์ 1 มิติ

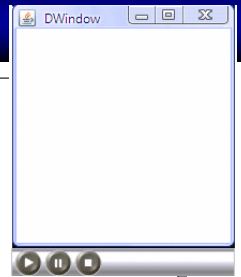
ภาควิชาวิศวกรรมคอมพิวเตอร์
จุฬาลงกรณ์มหาวิทยาลัย

ลูกบอลแดงผนัง (อีกแล้ว)

```
import jlab.graphics.DWindow;

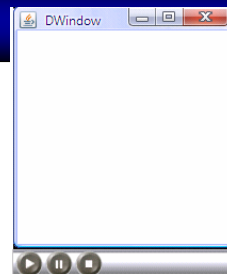
public class OneBouncingBall {

    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double x1 = 100, y1 = 100, r1 = 5;
        double dx1 = -5 + 10 * Math.random();
        double dy1 = -5 + 10 * Math.random();
        while (true) {
            if ((x1 - r1 <= 0) || (x1 + r1 >= 200)) dx1 = -dx1;
            if ((y1 - r1 <= 0) || (y1 + r1 >= 200)) dy1 = -dy1;
            x1 = x1 + dx1; y1 = y1 + dy1;
            w.clearBackground();
            w.fillEllipse(w.RED, x1, y1, 2 * r1, 2 * r1);
            w.sleep(50);
        }
    }
}
```



ลูกบอล 2 ลูกแดงผนัง

```
import jlab.graphics.DWindow;
public class TwoBouncingBalls {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        double x1 = 100, y1 = 100, r1 = 5;
        double dx1 = -5 + 10 * Math.random();
        double dy1 = -5 + 10 * Math.random();
        double x2 = 100, y2 = 100, r2 = 5;
        double dx2 = -5 + 10 * Math.random();
        double dy2 = -5 + 10 * Math.random();
        while (true) {
            if ((x1 - r1 <= 0) || (x1 + r1 >= 200)) dx1 = -dx1;
            if ((y1 - r1 <= 0) || (y1 + r1 >= 200)) dy1 = -dy1;
            x1 = x1 + dx1; y1 = y1 + dy1;
            if ((x2 - r2 <= 0) || (x2 + r2 >= 200)) dx2 = -dx2;
            if ((y2 - r2 <= 0) || (y2 + r2 >= 200)) dy2 = -dy2;
            x2 = x2 + dx2; y2 = y2 + dy2;
            w.clearBackground();
            w.fillEllipse(w.RED, x1, y1, 2 * r1, 2 * r1);
            w.fillEllipse(w.RED, x2, y2, 2 * r2, 2 * r2);
            w.sleep(50);
        }
    }
}
```



ถ้าต้องลูกบอลแดงไปมา 10 ลูก

ลูกบอลหลายลูกแดงผนัง (ใช้อาเรย์)

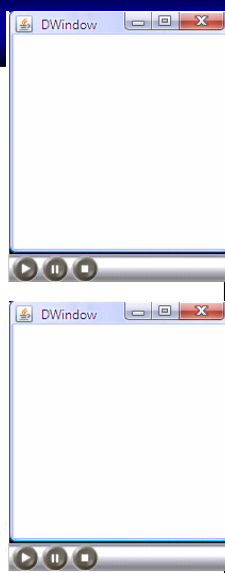
- อาเรย์เป็นชุดข้อมูลประเภทเดียวกันเก็บเรียงเป็นแถว
- อาเรย์มีชื่อกำกับ
- แต่ละช่องในอาเรย์มีหลายเลขกำกับเริ่มที่ 0
- อ้างอิงด้วยชื่อ + หมายเลขช่อง เช่น x[3], y[k]

	0	1	2	3	4	5	6	7	8	9
x	100	10	15	30	45	96	173	40	45	180
y	30	110	115	83	145	52	30	40	20	80
dx	-2	-1	3	-2	3	5	0	-1	0	2
dy	-1	1	2	1	3	3	-5	0	0	1

ลูกบอลเต่งผนัง (ใช้อาเรย์)

```
import jlab.graphics.DWindow;
public class NBouncingBalls {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        int n = 10;
        double[] x = new double[n], y = new double[n];
        double[] dx = new double[n], dy = new double[n];
        double[] r = new double[n];
        for (int i = 0; i < n; i++) {
            x[i] = 100; y[i] = 100; r[i] = 5;
            dx[i] = -5 + 10 * Math.random();
            dy[i] = -5 + 10 * Math.random();
        }
        while (true) {
            w.clearBackground();
            for (int i = 0; i < n; i++) {
                if ((x[i]-r[i] <= 0) || (x[i]+r[i] >= 200))
                    if ((y[i]-r[i] <= 0) || (y[i]+r[i] >= 200))
                        x[i] = x[i] + dx[i]; y[i] = y[i] + dy[i];
                w.fillEllipse(w.RED, x[i], y[i], 2 * r[i], 2 * r[i]);
            }
            w.sleep(50);
        }
    }
}
```

เปลี่ยน
n = 50



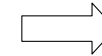
2110

5

ตัวอย่าง : โปรแกรมเก็บสถิติคะแนนสอบ

- มีแฟ้มข้อมูลเก็บรหัสนักเรียน + คะแนน
- คะแนนมีตั้งแต่ 0 ถึง 4 (จำนวนเต็ม)
- อยากทราบว่านักเรียนได้ 0 ก็คน 1 ก็คน ... 4 ก็คน

4531001321	3
4531003621	3
4531004221	4
4531005921	3
4531006521	3
4531101421	4
4531102021	4
4531102221	0



คะแนน	จำนวนนักเรียน
0	1
1	0
2	0
3	4
4	3

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

6

โปรแกรมเก็บสถิติคะแนนสอบ (ใช้ตัวแปร 5 ตัว)

```
import java.util.Scanner;
import java.io.*;

public class Stat1 {
    public static void main(String[] args) throws IOException {
        Scanner sc = new Scanner(new File("points.txt"));
        int p0 = 0, p1 = 0, p2 = 0, p3 = 0, p4 = 0;
        while (sc.hasNext()) {
            sc.nextDouble(); // อ่านรหัส
            int v = sc.nextInt(); // อ่านคะแนน
            if (v == 0) p0++;
            else if (v == 1) p1++;
            else if (v == 2) p2++;
            else if (v == 3) p3++;
            else if (v == 4) p4++;
        }
        System.out.println(p0 + " คนได้ 0 คะแนน");
        System.out.println(p1 + " คนได้ 1 คะแนน");
        System.out.println(p2 + " คนได้ 2 คะแนน");
        System.out.println(p3 + " คนได้ 3 คะแนน");
        System.out.println(p4 + " คนได้ 4 คะแนน");
    }
}
```

JLab>java Stat1
1 คนได้ 0 คะแนน
0 คนได้ 1 คะแนน
0 คนได้ 2 คะแนน
4 คนได้ 3 คะแนน
3 คนได้ 4 คะแนน

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

7

โปรแกรมเก็บสถิติคะแนนสอบ (ใช้อาเรย์ 5 ช่อง)

```
import java.util.Scanner;
import java.io.*;

public class Stat2 {
    public static void main(String[] args) throws IOException {
        Scanner sc = new Scanner(new File("points.txt"));
        int[] p = new int[5];
        while (sc.hasNext()) {
            sc.nextDouble();
            int v = sc.nextInt();
            if (0 <= v && v <= 4) {
                p[v]++;
            }
        }
        for (int i = 0; i < 5; i++) {
            System.out.println(p[i] + " คนได้ " + i + " คะแนน");
        }
    }
}
```

JLab>java Stat2
1 คนได้ 0 คะแนน
0 คนได้ 1 คะแนน
0 คนได้ 2 คะแนน
4 คนได้ 3 คะแนน
3 คนได้ 4 คะแนน

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

8

การประกาศและการสร้างอาเรย์

- ก่อนใช้อาเรย์ ต้อง
 - ประกาศตัวแปรอาเรย์
 - สร้างตัวอาเรย์

```
int [] data;
```

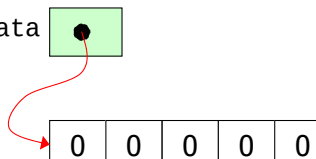
ประกาศตัวแปรอาเรย์แบบ int ชื่อว่า data

```
data = new int[5];
```

สร้างอาเรย์ของ int จำนวน 5 ช่อง แล้วให้ตัวแปร data อ้างอิง

```
int [] data = new int[5];
```

ประกาศตัวแปรและสร้างอาเรย์พร้อมกันเลยก็ได้



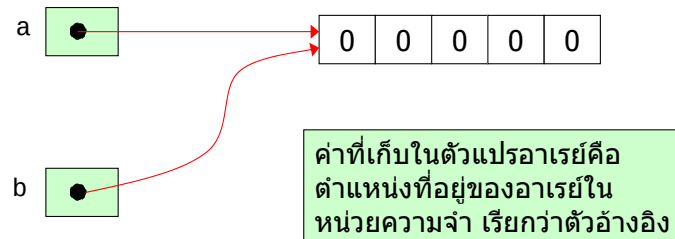
ข้อสังเกต : การประกาศ+การสร้าง

- การประกาศตัวแปรอาเรย์ ไม่ได้เป็นการสร้างตัวอาเรย์
- ขนาดของอาเรย์ถูกกำหนดตอนสร้างตัวอาเรย์
- ขนาดของอาเรย์เป็นค่าของนิพจน์ก็ได้
(data = new int[3*n + 1];)
- อาเรย์ที่สร้างขึ้น ไม่สามารถเพิ่มหรือลดขนาดได้
- ระบบจะตั้งค่าเริ่มต้นของข้อมูลในอาเรย์โดยอัตโนมัติ
(ให้ค่าศูนย์กับอาเรย์ของจำนวน และ ให้ค่า false กับอาเรย์ของ boolean)

ข้อสังเกต : การอ้างอิงอาเรย์

- อาเรย์แถวหนึ่ง อาจมีตัวแปรหลายตัวอ้างอิงได้

```
int [] a = new int[5];
```



ค่าที่เก็บในตัวแปรอาเรย์คือ ตำแหน่งที่อยู่ของอาเรย์ในหน่วยความจำ เรียกว่าตัวอ้างอิง

```
int [] b = a;
```

a กับ b มีค่าเท่ากัน หมายความว่าอ้างอิงอาเรย์เดียวกัน
ดังนั้น a == b ให้ค่า true และ
a[i] กับ b[i] อ้างอิงที่เก็บข้อมูลที่เดียวกัน

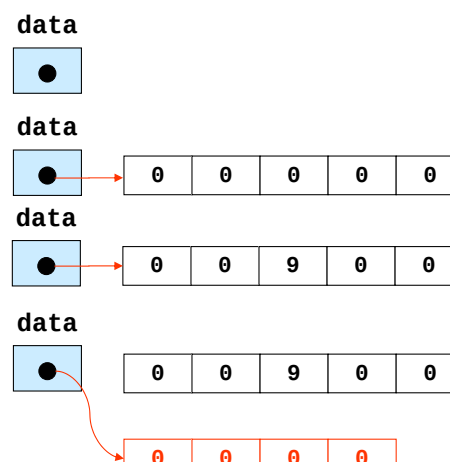
ตัวแปรอาเรย์เปลี่ยนการอ้างอิงได้

```
int[] data;
```

```
data = new int[5];
```

```
data[2] = 9;
```

```
data = new int[4];
```



ตัวอาเรย์ที่ไม่มีใครอ้างอิง เปรียบเหมือนขยะ ระบบจะนำเนื้อที่นี้ไปใช้ใหม่เมื่อจำเป็น

ประกาศ + สร้าง + ตั้งค่าเริ่มต้น

```
int[] data = new int[4];
data[0] = 23;
data[1] = 3;
data[2] = 21;
data[3] = 47;
```

```
int[] data = new int[]{23, 3, 21, 47};
```

ไม่ต้องบอกจำนวนช่อง

```
int[] data = {23, 3, 21, 47};
```

เขียนแบบสั้น



การใช้ข้อมูลในอาเรย์

- ใช้หมายเลขช่อง (เรียกว่า index) ของอาเรย์เป็นตัวอ้างอิงข้อมูลในอาเรย์ เช่น data[2]
- index ของอาเรย์เริ่มที่ 0 เสมอ
- ถ้าอาเรย์มีขนาด n ช่อง index ต้องเป็นจำนวนเต็มมีค่าได้ตั้งแต่ 0 ถึง n-1

```
int n = 10;
double[] x = new double[n], y = new double[n];
double[] dx = new double[n], dy = new double[n];
double[] r = new double[n];
for (int i = 0; i < n; i++) {
    x[i] = 100; y[i] = 100; r[i] = 5;
    dx[i] = -5 + 10 * Math.random();
    dy[i] = -5 + 10 * Math.random();
}
```

ข้อควรระวัง

- index ของอาเรย์ต้องเป็นจำนวนเต็ม int
- index ห้ามเป็นจำนวนลบ
- ถ้าอาเรย์มี n ช่อง, $0 \leq \text{index} \leq (n - 1)$

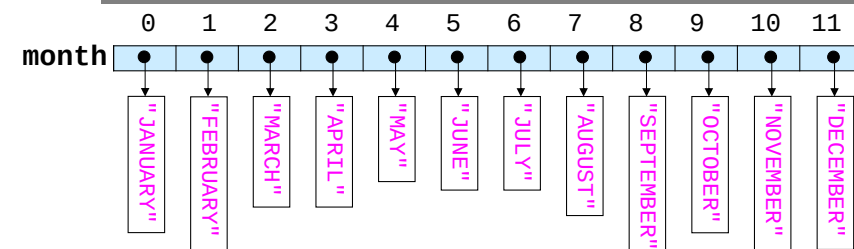
```
public class Array3 {
    public static void main(String [] args) {
        int [] data = new int[100];
        System.out.println( data[100] );
    }
}
```

```
JLab>java Array3
java.lang.ArrayIndexOutOfBoundsException
    at Array3.main(Array3.java:4)
Exception in thread "main"
JLab>
```

โปรแกรมแสดงชื่อเดือน

```
import java.util.Scanner;

public class MonthName {
    public static void main(String[] args) {
        String month[] =
            { "JANUARY", "FEBRUARY", "MARCH", "APRIL",
              "MAY", "JUNE", "JULY", "AUGUST", "SEPTEMBER",
              "OCTOBER", "NOVEMBER", "DECEMBER" };
        Scanner kb = new Scanner(System.in);
        System.out.print("หมายเลขเดือน = ");
        int m = kb.nextInt();
        if (1<=m && m<=12) System.out.println(month[m-1]);
    }
}
```



การส่งอาร์เรย์ไปยังเมทอดอื่น

- ต้องการส่งอาร์เรย์ใด ก็ส่งชื่อตัวแปรอาร์เรย์นั้น ไม่ต้องมี []
- การส่งอาร์เรย์จะไม่ได้ส่งข้อมูลของทั้งอาร์เรย์ไป แต่จะส่งค่าของตัวแปรอาร์เรย์ (ตำแหน่งอ้างอิงอาร์เรย์) ไปแทน

```
public static void main(String[] args) {
    double [] data = new double[5];
    ...
    double a = average( data );
}
```

data

```
public static double average(double[] x) {
    ...
}
```

x

1 23 9 -19 5

การคืนอาร์เรย์จากเมทอด

- เมทอดคืนผลลัพธ์แบบอาร์เรย์ก็ได้
- สิ่งที่คืนคือตำแหน่งอ้างอิงอาร์เรย์ จึงนำไปเก็บในตัวแปรอาร์เรย์

```
public static void main(String[] args) {
    int[] data = createRandomIntArray(20, 100, 200);
    ...
}
```

อาร์เรย์ที่สร้างในเมทอดถูกอ้างอิงโดย data

```
public static int[] randomIntArray(int n, int min, int max) {
    int[] d = new int[n];
    for (int i = 0; i < n; i++) {
        d[i] = min + (int)((max - min + 1) * Math.random());
    }
    return d;
}
```

ถ้าอยากรู้ขนาดของอาร์เรย์

- เติม .length ตามหลังชื่ออาร์เรย์

```
public class Array2 {
    public static void main(String [] args) {
        int [] data = new int[100];
        ...
        System.out.println( sum(data) );
    }

    public static int sum(int [] d) {
        int sum = 0;
        for(int i = 0; i < d.length; i++)
            sum += d[i];
        return sum;
    }
}
```

อย่าจำสลับกับความยาว String ซึ่งใช้ .length()

รูปแบบการประมวลผลอาร์เรย์

```
public static double sum(double[] d) {
    double sum = 0.0;
    for (int i = 0; i < d.length; i++) {
        sum = sum + d[i];
    }
    return sum;
}
```

วิ่งทั้งอาร์เรย์

```
public static int search(String[] t, String x) {
    for (int i = 0; i < t.length; i++) {
        if (t[i].compareTo(x) == 0) return i;
    }
    return -1;
}
```

```
public static int bsearch(int[] d, int x) {
    int left = 0, right = d.length - 1;
    while (left <= right) {
        int m = (left + right) / 2;
        if (x == d[m]) return m;
        if (x < d[m])
            right = m - 1;
        else
            left = m + 1;
    }
    return -1;
}
```

วิ่งให้พบเงื่อนไข
บางอย่าง

วิ่งเฉพาะ
บางช่วง
ของอาร์เรย์

(อธิบายภายหลัง)

ตัวอย่าง : การแสดงข้อมูลทั้งอาเรย์

```
public class TestArray {
    public static void main(String[] args) {
        int[] data = { 1, 2, 3, 4, 5, 3 };
        print(data);
    }
    public static void print(int[] data) {
        System.out.print("[");
        for (int i = 0; i < data.length; i++) {
            System.out.print(data[i]);
            if (i < data.length - 1) System.out.print(", ");
        }
        System.out.println("]");
    }
}
```

รูปแบบที่พบบ่อยมากในการวิ่งทั้งอาเรย์

JLab>java TestArray
[1, 2, 3, 4, 5, 3]

ตัวอย่าง : การทำสำเนาอาเรย์

```
public class TestArray {
    public static void main(String[] args) {
        int[] data = { 1, 2, 3, 4, 5, 3 };
        int[] copy = createCloneInt(data);
        print(copy);
    }
    public static int[] createCloneInt(int[] d) {
        int[] c = new int[d.length];
        for (int i = 0; i < d.length; i++) {
            c[i] = d[i];
        }
        return c;
    }
    public static void print(int[] d) {
        ...
    }
}
```

JLab>java TestArray
[1, 2, 3, 4, 5, 3]

ตัวอย่าง : การค้นข้อมูลในอาเรย์ (แบบลำดับ)

```
public static int seqSearch1(int[] d, int x) {
    for (int i = 0; i < d.length; i++) {
        if (d[i] == x) return i;
    }
    return -1;
}
// ค้นจากซ้ายไปขวา

public static int seqSearch2(int[] d, int x) {
    for (int i = d.length - 1; i >= 0; i--) {
        if (d[i] == x) return i;
    }
    return -1;
}
// ค้นจากขวามาซ้าย
```

	0	1	2	3	4	5	6	7	8	9	10	11
d	23	3	21	47	53	20	2	7	63	37	55	11

→

x 7

ตัวอย่าง : การค้นข้อมูลในอาเรย์ที่เรียงลำดับ

```
public static int sortedSearch1(int[] d, int x) {
    for (int i = 0; i < d.length; i++) {
        if (x == d[i]) return i;
        if (x > d[i]) break;
    }
    return -1;
}
// ค้นจากซ้ายไปขวา

public static int sortedSearch2(int[] d, int x) {
    for (int i = d.length - 1; i >= 0; i--) {
        if (x == d[i]) return i;
        if (x < d[i]) break;
    }
    return -1;
}
// ค้นจากขวามาซ้าย
```

	0	1	2	3	4	5	6	7	8	9	10	11
d	2	3	7	11	23	37	45	55	58	63	87	95

→

x 13

ตัวอย่าง : การค้นข้อมูลแบบทวิภาค

```
public static int bsearch(int[] d, int x) {
    int left = 0, right = d.length - 1;
    while (left <= right) {
        int m = (left + right) / 2;
        if (x == d[m]) return m;
        if (x < d[m])
            right = m - 1;
        else
            left = m + 1;
    }
    return -1;
}
```

รอบที่	จำนวนข้อมูล
1	n
2	$n/2$
3	$n/2^2$
...	...
k	$n/2^{k-1}$
ถ้า $n/2^{k-1} = 1$ จะได้ $k = 1 + \lfloor \log_2 n \rfloor$	

x 45

	0	1	2	3	4	5	6	7	8	9	10	11
d	2	3	7	11	23	37	45	55	58	63	87	95
d	2	3	7	11	23	37	45	55	58	63	87	95
d	2	3	7	11	23	37	45	55	58	63	87	95

ตัวอย่าง : เขียน indexOf ให้เรา

```
int[] s = { 1, 2, 3 }, t = { 9, 1, 2, 3, 4};
int j = indexOf(t, s); // j มีค่าเท่ากับ 3
```

ค้นใน t ว่ามีข้อมูลตามลำดับที่ปรากฏใน s หรือไม่
ถ้ามีก็คืน index เริ่มต้นของ t ที่พบว่าเหมือน s, ถ้าไม่มี ก็คืน -1

```
public static int indexOf(int[] t, int[] s) {
    if (t.length < s.length) return -1;
    for (int i = 0; i < t.length; i++) {
        int k = i;
        boolean match = true;
        for (int j = 0; match && j < s.length; j++) {
            if (t[k] != s[j]) match = false;
            k++;
        }
        if (match) return i;
    }
    return -1;
}
```

ตัวอย่าง : การหาหมายเลขช่องที่เก็บค่ามากที่สุด

	0	1	2	3	4	5	6	7
d	23	3	21	47	53	20	2	7

maxIndex(d) ได้ค่า 4

```
public static int maxIndex(int[] d) {
    int maxI = 0;
    for (int i = 1; i < d.length; i++) {
        if (d[i] > d[maxI]) maxI = i;
    }
    return maxI;
}
```

	0	1	2	3	4	5	6	7
d	23	3	21	47	53	20	2	7
i =	0	1	2	3	4	5	6	7
maxI =	0	0	0	3	4	4	4	4

ตัวอย่าง : การเรียงลำดับข้อมูล (แบบเลือก)

```
public static void selectionSort( int [] data )
```

	0	1	2	3	4	5	6	k	maxI
data	23	2	3	14	5	99	9	6	5
	23	2	3	14	5	9	99	5	0
	9	2	3	14	5	23	99	4	3
	9	2	3	5	14	23	99	3	0
	5	2	3	9	14	23	99	2	0
	3	2	5	9	14	23	99	1	0
	2	3	5	9	14	23	99		

ตัวอย่าง : เมท็อดการเรียงลำดับแบบเลือก

```
public static void selectionSort(int[] data) {
    for (int k = data.length - 1; k >= 1; k--) {
        int i = maxIndex(data, k);
        int t = data[i];
        data[i] = data[k];
        data[k] = t;
    }
}

public static int maxIndex(int[] d, int k) {
    int maxI = 0;
    for (int i = 1; i <= k; i++) {
        if (d[i] > d[maxI]) maxI = i;
    }
    return maxI;
}
```

สลับข้อมูลของที่ i และ k

หา index ของอาร์เรย์ d
ที่เก็บค่ามากที่สุดจาก
ช่องที่ 0 ถึง k

โจทย์ : โปรแกรมเขียนคำร้อง

และในวันนี้ที่ฉันได้เจอเธอ

ข้อมูลขาเข้า

ผลลัพธ์

และในวันนี้
ในวันนี้
วันนี้ที่
วันนี้ที่
ฉันได้
ฉันได้
เจอเธอ
เจอเธอ

	0	1	2	3	4	5	6	7	8
words	และ	ใน	วัน	นี้	ที่	ฉัน	ได้	เจอ	เธอ

• วิธีทำ

- ใช้ WordScanner เพื่อแยกสตริงออกเป็นคำ
- อ่านมาทุกคำรอบแรก เพื่อนับว่ามีกี่คำ
- สร้างอาร์เรย์ของสตริงเพื่อเก็บคำ
- อ่านใหม่อีกรอบ คราวนี้นำไปเก็บเรียงในอาร์เรย์
- วนหียบค่าในอาร์เรย์มาพิมพ์ตามต้องการ

โจทย์ : เขียนโปรแกรมให้สมบูรณ์

```
import jlab.WordScanner;
import java.util.Scanner;
public class Lyric {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("ข้อความ = ");
        String t = kb.nextLine();
        int n = numberOfWords(t);
        String[] words = new String[k];
    }

    public static int numberOfWords(String t) {
        WordScanner sc = new WordScanner("และในวันนี้ที่ฉันได้เจอเธอ");
        int k = 0;
        while (sc.hasNext()) { sc.nextWord(); k++; }
        return k;
    }
}
```

และในวันนี้ที่ฉันได้เจอเธอ

และในวันนี้
ในวันนี้
วันนี้ที่
วันนี้ที่
ฉันได้
ฉันได้
เจอเธอ
เจอเธอ

โจทย์เกี่ยวกับอาร์เรย์ : 1

จงเขียนเมท็อดต่อไปนี้

- **public static int count(int [] d, int x)**
มี x ปรากฏในอาร์เรย์ d กี่ครั้ง
- **public static int mode(int [] d)**
คืนข้อมูลที่ปรากฏมากสุดใน d
- **public static boolean majority(int [] d)**
มีข้อมูลที่ปรากฏเป็นจำนวนเกินครึ่งใน d หรือไม่

โจทย์เกี่ยวกับอาเรย์ : 2

- ตรวจสอบว่าอาเรย์ a มีข้อมูลเหมือนของอาเรย์ b หรือไม่
`boolean equals(int[] a, int[] b)`

p

23	3	21
----	---	----

 q

23	3	21
----	---	----

 r

3	23	21
---	----	----

equals(p, q) ได้ค่า true แต่ equals(p, r) ได้ค่า false

- คืนอาเรย์ที่ได้มาจากการผสานข้อมูลอาเรย์ a และ b โดยที่ข้อมูลในอาเรย์ a และ อาเรย์ b เรียงลำดับจากน้อยไปมากแล้ว (ข้อมูลในอาเรย์ที่คืนมาก็ต้องเรียงจากน้อยไปมากด้วย)

`int[] merge(int[] a, int[] b)`

a

3	13	21
---	----	----

 c

2	3	4	13	21	29
---	---	---	----	----	----

b

2	4	29
---	---	----

`int[] c = merge(a, b)`

โจทย์เกี่ยวกับอาเรย์ : 3

- `boolean isSorted(int[] d)`
คืน true ถ้าข้อมูลใน d เรียงจากน้อยไปมาก (ซ้ายไปขวา)
- `int[] inRange(int[] d, int a, int b)`
คืนอาเรย์ที่เก็บข้อมูลใน d ที่อยู่ระหว่าง a กับ b
- `int[] longestIncreasingSequence(int[] d)`
คืนอาเรย์ที่เก็บข้อมูลที่ติดกันใน d ที่มีค่าเพิ่มขึ้นเรื่อยๆ เป็นช่วงที่ยาวที่สุด

d

21	3	4	13	2	9
----	---	---	----	---	---

isSorted(d) ได้ false

inRange(d, 1, 10) ได้

3	4	2	9
---	---	---	---

longestIncreasingSequence(d) ได้

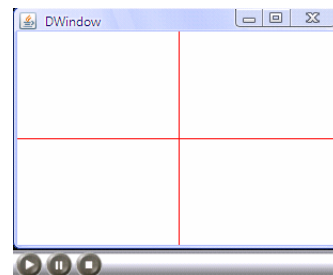
3	4	13
---	---	----

ตัวอย่าง : โปรแกรมวาดกราฟเส้น

- เขียน `void plot(double[] x, double[] y)`
- ขนาดของวินโดว์อาจใหญ่หรือแคบกว่าช่วงของค่า x และ y จึงต้องคำนวณตำแหน่งการวาดไม่ให้ออกนอกวินโดว์
- ให้วาดเส้นสีแดง x = 0 และ y = 0 ด้วย
(จะวาดเมื่อ x=0 หรือ y=0 อยู่ในช่วง x และ y ที่ได้รับ)

```
public static void main(String[] args) {
    int n = 100;
    double[] x = new double[n];
    double[] y = new double[n];
    double xmin = -10, xmax = 10;
    double dx = (xmax - xmin) / x.length;
    for (int i = 0; i < x.length; i++) {
        x[i] = xmin + i * dx;
        for (int k = 1; k < 2; k++) {
            y[i] += Math.sin(k * x[i]);
        }
    }
    plot(x, y);
}
```

$$y[i] = \sum_{k=1}^4 \sin(kx[i])$$



x[] อยู่ในช่วง [-10, 10]

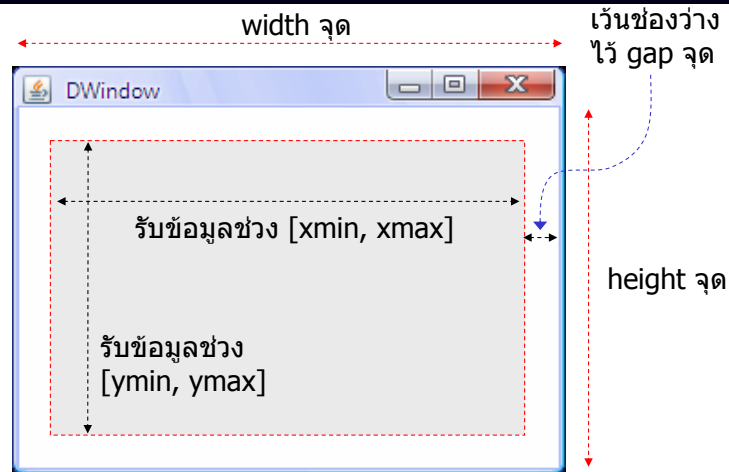
หา xmin, xmax, ymin, ymax

```
public static void plot(double[] x, double[] y) {
    DWindow w = new DWindow(300, 200);
    double xmin = min(x), xmax = max(x);
    double ymin = min(y), ymax = max(y);
    ...
}

public static double max(double[] d) {
    double max = d[0];
    for (int i = 1; i < d.length; i++)
        if (d[i] > max) max = d[i];
    return max;
}

public static double min(double[] d) {
    double min = d[0];
    for (int i = 1; i < d.length; i++)
        if (d[i] < min) min = d[i];
    return min;
}
```

การเปลี่ยนค่า x และ y ไปเป็นพิกัดของจุดภาพ



$$p_x = gap + (x - x_{min}) \left(\frac{width - 2gap}{x_{max} - x_{min}} \right)$$

$$p_y = gap + (y - y_{min}) \left(\frac{height - 2gap}{y_{max} - y_{min}} \right)$$

เปลี่ยนจาก (x, y) ไปเป็น
พิกัดของจุดภาพ (p_x, p_y)

วาดเส้น x = 0, y = 0

- ถ้า xmin ≤ 0 ≤ xmax, วาดเส้นแนวนอน
- ถ้า ymin ≤ 0 ≤ ymax, วาดเส้นแนวตั้ง

```
public static void plot(double[] x, double[] y) {
    DWindow w = new DWindow(300, 200);
    double xmin = min(x), xmax = max(x);
    double ymin = min(y), ymax = max(y);
    double gap = 20;
    double width = w.getWidth(), height = w.getHeight();
    if (xmin <= 0 && 0 <= xmax) {
        double px = toScreen(0, xmin, xmax, width, gap);
        w.drawLine(w.RED, px, 0, px, height); // เส้นแนวนอน
    }
    if (ymin <= 0 && 0 <= ymax) {
        double py = toScreen(0, ymin, ymax, height, gap);
        w.drawLine(w.RED, 0, py, width, py); // เส้นแนวตั้ง
    }
    ...
}

public static double toScreen(double v, double min,
    double max, double d, double g) {
    return g + (v - min) * (d - 2 * g) / (max - min);
}
```

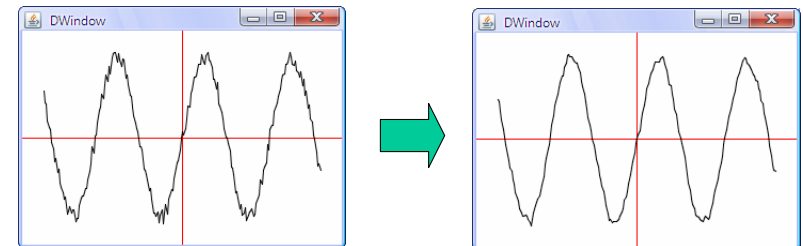
$$p_x = g + (v - \min) \left(\frac{d - 2g}{\max - \min} \right)$$

สุดท้าย : วาดเส้นกราฟ

```
public static void plot(double[] x, double[] y) {
    DWindow w = new DWindow(300, 200);
    double xmin = min(x), xmax = max(x);
    double ymin = min(y), ymax = max(y);
    double gap = 20;
    double width = w.getWidth(), height = w.getHeight();
    if (xmin <= 0 && 0 <= xmax) {
        double px = toScreen(0, xmin, xmax, width, gap);
        w.drawLine(w.RED, px, 0, px, height); // เส้นแนวนอน
    }
    if (ymin <= 0 && 0 <= ymax) {
        double py = toScreen(0, ymin, ymax, height, gap);
        w.drawLine(w.RED, 0, py, width, py); // เส้นแนวตั้ง
    }
    for (int i = 1; i < x.length; i++) {
        double px0 = toScreen(x[i-1], xmin, xmax, width, gap);
        double px1 = toScreen(x[i], xmin, xmax, width, gap);
        double py0 = height - toScreen(y[i-1], ymin, ymax, height, gap);
        double py1 = height - toScreen(y[i], ymin, ymax, height, gap);
        w.drawLine(px0, py0, px1, py1);
        w.sleep(50);
    }
}
```

โจทย์ : Moving Average

- ถ้าข้อมูลในอาร์เรย์ x และ y ที่รับมามี "สัญญาณรบกวน"
- ต้องการทำให้ "เรียบ" ขึ้น



- วิธีง่ายสุด : ให้ค่า y ใหม่เป็นค่าเฉลี่ยของ y เดิมกับตัวข้างเคียงหน้าหลัง $y'_i = (y_{i-1} + y_i + y_{i+1}) / 3$
- จงเขียนเมทอด double[] smooth(double[] y) ที่คืน อาร์เรย์ของ y ใหม่ที่ "เรียบ" กว่าอาร์เรย์ y ที่ได้รับ

2110101 : การทำโปรแกรมคอมพิวเตอร์

อาร์เรย์ 2 มิติ

ภาควิชาวิศวกรรมคอมพิวเตอร์
จุฬาลงกรณ์มหาวิทยาลัย

อาร์เรย์หลายมิติ (Multidimensional Array)

หนึ่งมิติ

```
int[] a = new int[7];  
a[2] = 3;
```

	0	1	2	3
a	0	0	3	0

สองมิติ

```
int[][] b = new int[3][4];  
b[1][3] = 4;
```

	0	1	2	3
0	0	0	0	0
1	0	0	0	4
2	0	0	0	0

สามมิติ

```
int[][][] c = new int[2][3][4];  
c[0][1][2] = 9;
```

	c[1]	0	0	0	0
c[0]	0	0	0	0	0
	0	0	9	0	0
	0	0	0	0	0

การตั้งค่าเริ่ม

```
int[] a = { 1, 2, 3, 4 };
```

	0	1	2	3
a	1	2	3	4

```
int[][] b = { {1, 2, 3, 4},  
              {5, 6, 7, 8},  
              {9, 10, 11, 12} };
```

	0	1	2	3
0	1	2	3	4
1	5	6	7	8
2	9	10	11	12

ตัวอย่าง : การแสดงค่าในอาร์เรย์สองมิติ

a.length = จำนวนแถว, a[i].length = จำนวนคอลัมน์ของแถวที่ i

```
public class TestPrintArray2 {  
    public static void main(String[] args) {  
        int[][] b = {{1, 2, 3, 4, 5},  
                     {5, 4, 3, 2, 1},  
                     {10, 20, 30, 40, 50}};  
        printArray2(b);  
    }  
    public static void printArray2(int[][] a) {  
        for (int i = 0; i < a.length; i++) {  
            for (int j = 0; j < a[i].length; j++) {  
                System.out.print(a[i][j] + "\t");  
                System.out.println();  
            }  
        }  
    }  
}
```

```
JLab>java TestPrintArray2  
1      2      3      4      5  
5      4      3      2      1  
10     20     30     40     50
```

ตัวอย่าง : การบวกเมทริกซ์

- แทนเมทริกซ์ C ด้วยอาร์เรย์สองมิติ c
- สมาชิก c_{ij} ของเมทริกซ์ C คือ $c[i][j]$
- ให้ $C = A + B$, จะได้ว่า $c_{ij} = a_{ij} + b_{ij}$

$$\begin{bmatrix} 2 & 3 & 4 \\ 5 & 6 & 7 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 2+0 & 3+1 & 4+0 \\ 5+1 & 6+0 & 7+1 \end{bmatrix} = \begin{bmatrix} 2 & 4 & 4 \\ 6 & 6 & 8 \end{bmatrix}$$

```
public static int[][] addMatrices(int[][] a, int[][] b) {
    int row = a.length, col = a[0].length;
    int[][] c = new int[row][col];
    for (int i = 0; i < row; i++) {
        for (int j = 0; j < col; j++) {
            c[i][j] = a[i][j] + b[i][j];
        }
    }
    return c;
}
```

ตัวอย่าง : การคูณเวกเตอร์กับเมทริกซ์

$$\begin{bmatrix} a_0 & a_1 \end{bmatrix} \begin{bmatrix} b_{0,0} & b_{0,1} & b_{0,2} \\ b_{1,0} & b_{1,1} & b_{1,2} \end{bmatrix} = \begin{bmatrix} a_0 b_{0,0} + a_1 b_{1,0} & a_0 b_{0,1} + a_1 b_{1,1} & a_0 b_{0,2} + a_1 b_{1,2} \end{bmatrix}$$

$$= \begin{bmatrix} c_0 & c_1 & c_2 \end{bmatrix}$$

$$c_i = \sum_{k=0}^{n-1} a_k b_{k,i}$$

```
public static int[] multiply(int[] a, int[][] b) {
    int n = b.length; // == a.length
    int m = b[0].length;
    int[] c = new int[m];
    for (int i = 0; i < m; i++) {
        for (int k = 0; k < n; k++) {
            c[i] = c[i] + a[k] * b[k][i];
        }
    }
    return c;
}
```

ตัวอย่าง : การคูณเมทริกซ์กับเมทริกซ์

$$\begin{bmatrix} a_{0,0} & a_{0,1} \\ a_{1,0} & a_{1,1} \end{bmatrix} \begin{bmatrix} b_{0,0} & b_{0,1} & b_{0,2} \\ b_{1,0} & b_{1,1} & b_{1,2} \end{bmatrix} = \begin{bmatrix} a_{0,0}b_{0,0} + a_{0,1}b_{1,0} & a_{0,0}b_{0,1} + a_{0,1}b_{1,1} & a_{0,0}b_{0,2} + a_{0,1}b_{1,2} \\ a_{1,0}b_{0,0} + a_{1,1}b_{1,0} & a_{1,0}b_{0,1} + a_{1,1}b_{1,1} & a_{1,0}b_{0,2} + a_{1,1}b_{1,2} \end{bmatrix}$$

$$= \begin{bmatrix} c_{0,0} & c_{0,1} & c_{0,2} \\ c_{1,0} & c_{1,1} & c_{1,2} \end{bmatrix} \quad c_{i,j} = \sum_{k=0}^{n-1} a_{i,k} b_{k,j}$$

```
public static int[][] multiply(int[][] a, int[][] b) {
    int[][] c = new int[a.length][b[0].length];
    int n = b.length;
    for (int i = 0; i < a.length; i++) {
        for (int j = 0; j < b[0].length; j++) {
            for (int k = 0; k < n; k++) {
                c[i][j] = c[i][j] + a[i][k] * b[k][j];
            }
        }
    }
    return c;
}
```

โจทย์เกี่ยวกับอาร์เรย์

จงเขียนเมทอด

- **void printEqualRows(int[][] d)**
แสดงแถวใน d ที่มีค่าเหมือนกัน

1	2	4	3
9	6	5	7
1	2	4	3

- **double saddlePoint(double[][] d)**
คืนข้อมูลใน d ที่น้อยสุดในแถว
แต่มากที่สุดในคอลัมน์, ถ้าไม่มีคืน -1

1	2	4	3
9	6	5	7
1	2	4	3

- **boolean isMagicSquare(int[][] d)**
คืนจริงถ้า d (ซึ่งเป็น square matrix) เก็บข้อมูล
ที่มีผลรวมของทุกแถว ทุกคอลัมน์ และตามแนว
ทแยงมุม เท่ากันหมด

16	3	2	13
5	10	11	8
9	6	7	12
4	15	14	1

นี้เป็น magic square

อาเรย์ 2 มิติกับตารางจุดภาพ : getPixmap, setPixmap



```
public static void main(String[] args) {
    DWindow w = new DWindow(200, 200);
    int[][] p = w.getPixmap();
    p[10][10] = DWindow.mixRGB(0, 0, 0);
    p[10][11] = DWindow.mixRGB(255, 0, 0);
    p[10][12] = DWindow.mixRGB(0, 255, 0);
    p[10][13] = DWindow.mixRGB(0, 0, 255);
    p[11][10] = DWindow.mixRGB(128, 128, 128);
    p[11][11] = DWindow.mixRGB(128, 0, 0);
    p[11][12] = DWindow.mixRGB(0, 128, 0);
    p[11][13] = DWindow.mixRGB(0, 0, 128);
    p[12][10] = DWindow.mixRGB(255, 255, 255);
    p[12][11] = DWindow.mixRGB(255, 255, 0);
    p[12][12] = DWindow.mixRGB(255, 0, 255);
    p[12][13] = DWindow.mixRGB(0, 255, 255);
    w.setPixmap(p);
}
```

pixmap = pixel map
pixel = picture element

p[x][y]

มีค่าตั้งแต่ 0 -255

mixRGB เป็นเมทอดที่เขียนในคลาส DWindow

DWindow.mixRGB(r, g, b)

- แมสแต่ละสีมีค่าได้ตั้งแต่ 0 ถึง 255 (ซึ่งคือ 00000000₂ ถึง 11111111₂)

mixRGB(200, 100, 70)

200₁₀ = 11001000₂

100₁₀ = 01100100₂

70₁₀ = 01000110₂

1 1 1 1 1 1 1 1	1 1 0 0 1 0 0 0	0 1 1 0 0 1 0 0	0 1 0 0 0 1 1 0
-----------------	-----------------	-----------------	-----------------

w.WHITE = mixRGB(255, 255, 255)

1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1
-----------------	-----------------	-----------------	-----------------

w.BLACK = mixRGB(0, 0, 0)

1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0
-----------------	-----------------	-----------------	-----------------

A ₇ A ₆ A ₅ A ₄ A ₃ A ₂ A ₁ A ₀	R ₇ R ₆ R ₅ R ₄ R ₃ R ₂ R ₁ R ₀	G ₇ G ₆ G ₅ G ₄ G ₃ G ₂ G ₁ G ₀	B ₇ B ₆ B ₅ B ₄ B ₃ B ₂ B ₁ B ₀
---	---	---	---

32 บิต = 1 int

กำหนดระดับความทึบ : 0 คือโปร่งใส, 255 คือทึบ (เรียกกันว่าค่า alpha)

```
import jlab.graphics.DWindow;
```

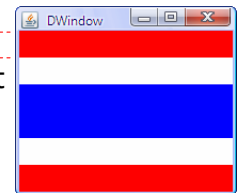
```
public class StdColor {
    public static void main(String[] args) {
        DWindow w = new DWindow(450, 180);
        int[] c = { w.BLACK, w.BLUE, w.CYAN, w.GRAY, w.GREEN, w.MAGENTA,
            w.ORANGE, w.PINK, w.RED, w.WHITE, w.YELLOW };
        String[] n = { "BLACK", "BLUE", "CYAN", "GRAY", "GREEN", "MAGENTA",
            "ORANGE", "PINK", "RED", "WHITE", "YELLOW" };
        w.setBackground(w.BLACK); w.setForeground(w.WHITE);
        int k = 0;
        for (int i = 0; i < 4; i++) {
            for (int j = 0; j < 3; j++) {
                if (k >= c.length) break;
                w.fillRect(c[k], 10 + i * 120, 10 + j * 60, 75, 15);
                w.drawString(n[k], 18, 10 + i * 120, 30 + j * 60);
                k++;
            }
        }
    }
}
```

ค่าคงตัวที่ได้ตั้งค่าไว้
แล้วคลาส DWindow
เรียกใช้ได้เลย



ตัวอย่าง : ธงชาติ

```
import jlab.graphics.DWindow;
public class NationalFlag {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 150);
        int[][] p = w.getPixmap();
        int height = p[0].length / 6;
        for (int y = 0; y < height; y++) {
            fillColor(p, y, w.RED);
        }
        for (int y = 2*height; y < 4*height; y++) {
            fillColor(p, y, w.BLUE);
        }
        for (int y = 5*height; y < 6*height; y++) {
            fillColor(p, y, w.RED);
        }
        w.setPixmap(p);
    }
    public static void fillColor(int[][] p, int y, int color) {
        for (int x = 0; x < p.length; x++) {
            p[x][y] = color;
        }
    }
}
```



โจทย์ : เติมสีหลายรูปแบบ

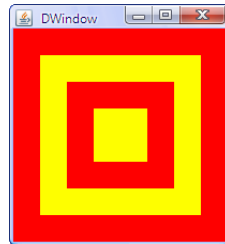
```
import jlab.graphics.DWindow;

public class FillColor {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        int[][] p = w.getPixmap();
        for (int x = 0; x < p.length; x++) {
            for (int y = 0; y < p[0].length; y++) {
                if (x > y)
                    p[x][y] = w.RED;
                else
                    p[x][y] = w.YELLOW;
            }
        }
        w.setPixmap(p);
    }
}
```

ได้รูปนี้



จงเขียน
โปรแกรมที่ได้
รูปทางขวานี้



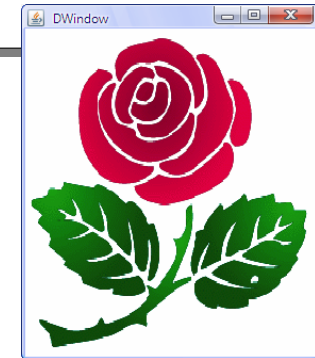
ตัวอย่าง : การแสดงแฟ้มภาพ

```
import jlab.graphics.DWindow;
import jlab.graphics.DImage;

public class ImageDisplay {
    public static void main(String[] args) {
        DWindow w1 = new DWindow(new DImage("baby.jpg"));
        int w = w1.getWidth(), h = w1.getHeight();
        DWindow w2 = new DWindow(new DImage("rose.gif"), w, h);
    }
}
```

สร้างวินโดว์ขนาดเท่ากับที่กำหนด

สร้างวินโดว์ขนาดเท่ากับรูป



ตัวอย่าง : การพับภาพ

```
import jlab.graphics.*;

public class MirrorFlip {
    public static void main(String[] args) {
        DWindow w = new DWindow(new DImage("baby.jpg"));
        int[][] p = w.getPixmap();
        int width = p.length, height = p[0].length;
        for (int x = 0; x < width/2; x++) {
            for (int y = 0; y < height; y++) {
                p[width - x - 1][y] = p[x][y];
            }
        }
        w.setPixmap(p);
    }
}
```



ตัวอย่าง : การทำภาพ sepia



$$\begin{aligned} R' &= (R \times 0.393 + G \times 0.769 + B \times 0.189) \\ G' &= (R \times 0.349 + G \times 0.686 + B \times 0.168) \\ B' &= (R \times 0.272 + G \times 0.534 + B \times 0.131) \end{aligned}$$

http://en.wikipedia.org/wiki/Sepia_tone

ตัวอย่าง : เมท็อดทำภาพ sepia

```
import jlab.graphics.*;

public class ImageSepia {
    public static void main(String[] args) {
        DWindow w1 = new DWindow(new DImage("baby.jpg"));
        DWindow w2 = new DWindow(w1.getWidth(), w1.getHeight());
        w2.setPixmap(sepia(w1.getPixmap()));
    }
    public static int[][] sepia(int[][] p) {
        int[][] out = new int[p.length][p[0].length];
        for (int x = 0; x < p.length; x++) {
            for (int y = 0; y < p[x].length; y++) {
                int r = DWindow.getR(p[x][y]);
                int g = DWindow.getG(p[x][y]);
                int b = DWindow.getB(p[x][y]);
                int r1 = (int)(r * 0.393 + g * 0.769 + b * 0.189);
                int g1 = (int)(r * 0.349 + g * 0.686 + b * 0.168);
                int b1 = (int)(r * 0.272 + g * 0.534 + b * 0.131);
                out[x][y] = DWindow.mixRGB(r1, g1, b1);
            }
        }
        return out;
    }
}
```

DWindow.getR(c) คำนวณสีแดงของสี c ที่ได้รับ
DWindow.getG(c) คำนวณสีเขียวของสี c ที่ได้รับ
DWindow.getB(c) คำนวณสีน้ำเงินของสี c ที่ได้รับ

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

17

โจทย์ : การทำภาพเทา



อ่าน <http://en.wikipedia.org/wiki/Grayscale> แล้วเขียนเมท็อด

```
public static int[][] gray(int[][] p) {
    ...
}
```

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

18

โจทย์ : การทำภาพ negative



ลองคิดเองดูว่าจะทำอย่างไร ?

```
public static int[][] negative(int[][] p) {
    ...
}
```

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

19

โจทย์ : การเปลี่ยนภาพแบบ fading

```
import jlab.graphics.*;
public class Fade {
    public static void main(String[] args) {
        int width = 200, height = 200;
        DWindow w1 = new DWindow(new DImage("baby.jpg"), width, height);
        DWindow w2 = new DWindow(new DImage("rose.gif"), width, height);
        int[][] p1 = w1.getPixmap();
        int[][] p2 = w2.getPixmap();
        int[][] p3 = w3.getPixmap();

        ...

        while (true) {
            ...
            w3.setPixmap(p3);
            w1.sleep(100);
        }
    }
}
```

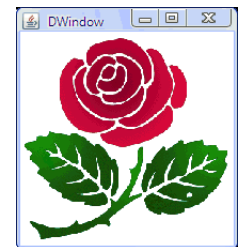
สร้างวินโดว์และปรับภาพ
ในมีขนาด width และ
height ตามที่กำหนด

$$r_3 = \alpha r_1 + (1 - \alpha) r_2$$

$$g_3 = \alpha g_1 + (1 - \alpha) g_2$$

$$b_3 = \alpha b_1 + (1 - \alpha) b_2$$

รวมแม่สีของแต่ละจุดในภาพทั้งสองด้วยสูตร โดย
ให้ค่า α ค่อยๆ เปลี่ยนจาก 0 $\rightarrow$ 1 และจาก 1 $\rightarrow$ 0



2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย (22/05/51)

20

โจทย์ : การเปลี่ยนภาพแบบ Blinding

```
import jlab.graphics.*;
public class Blind {
    public static void main(String[] args) {
        int width = 200, height = 200;
        DWindow w1 = new DWindow(new DImage("baby.jpg"), width, height);
        DWindow w2 = new DWindow(new DImage("rose.gif"), width, height);
        int[][] p1 = w1.getPixmap();
        int[][] p2 = w2.getPixmap();
        int[][] p3 = w3.getPixmap();

        ...

        while (true) {

            ...

            w3.setPixmap(p3);
            w1.sleep(100);
        }
    }
}
```



2110101 : การทำโปรแกรมคอมพิวเตอร์

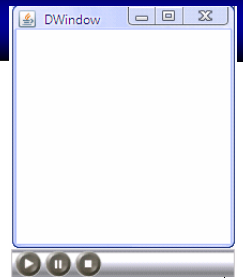
Classes & Objects

ภาควิชาวิศวกรรมคอมพิวเตอร์

จุฬาลงกรณ์มหาวิทยาลัย

ลูกบอลแดงได้ (อีกแล้ว)

```
DWindow w = new DWindow(200, 200);
int n = 50;
double[] x = new double[n], y = new double[n];
double[] dx = new double[n], dy = new double[n];
double[] r = new double[n];
int[] color = new int[n];
for (int i = 0; i < n; i++) {
    x[i] = 100; y[i] = 100; r[i] = 5;
    dx[i] = -5 + 10 * Math.random();
    dy[i] = -5 + 10 * Math.random();
    color[i] = DWindow.randomColor();
}
w.setRepaintDuringSleep(true);
while (true) {
    w.clearBackground();
    for (int i = 0; i < n; i++) {
        if ((x[i]-r[i] <= 0) || (x[i]+r[i] >= 200)) dx[i] = -dx[i];
        if ((y[i]-r[i] <= 0) || (y[i]+r[i] >= 200)) dy[i] = -dy[i];
        x[i] = x[i] + dx[i]; y[i] = y[i] + dy[i];
        w.fillEllipse(color[i], x[i], y[i], 2 * r[i], 2 * r[i]);
    }
    w.sleep(50);
}
```



2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

2

ลูกบอลแดงได้ : เขียนแบบแยกเมทอด

```
public static void main(String[] args) {
    ...
    while (true) {
        moveBalls(w, x, y, r, dx, dy);
        paintBalls(w, x, y, color, r, dx, dy);
        w.sleep(50);
    }
}
public static void moveBalls(DWindow w, double[] x, double[] y,
    double[] r, double[] dx, double[] dy) {
    int width = w.getWidth(), height = w.getHeight();
    for (int i = 0; i < x.length; i++) {
        if ((x[i]-r[i] <= 0) || (x[i]+r[i] >= width)) dx[i] = -dx[i];
        if ((y[i]-r[i] <= 0) || (y[i]+r[i] >= height)) dy[i] = -dy[i];
        x[i] = x[i] + dx[i];
        y[i] = y[i] + dy[i];
    }
}
public static void paintBalls(DWindow w, double[] x, double[] y,
    int[] c, double[] r, double[] dx, double[] dy) {
    w.clearBackground();
    for (int i = 0; i < x.length; i++) {
        w.fillEllipse(c[i], x[i], y[i], 2 * r[i], 2 * r[i]);
    }
}
```

ต้องส่งพารามิเตอร์
มากมายที่บรรยาย
ลักษณะลูกบอล
ไม่สวย

สร้างประเภทข้อมูลใหม่ชื่อ Ball

- เขียนคลาสชื่อ **Ball**
 - สร้างลูกบอลด้วยคำสั่ง **new Ball()**
- เขียนประกาศตัวแปรในคลาส ซึ่งคือตัวแปรประจำ object ที่จะถูกสร้างขึ้น

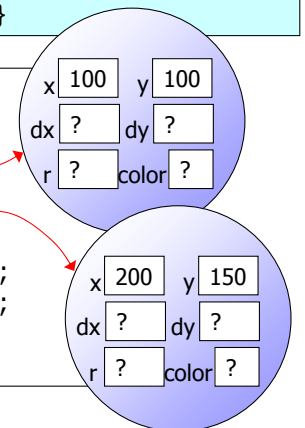
การบรรยายคลาส Ball

```
public class Ball {
    double x, y;
    double dx, dy;
    double r;
    int color;
}
```

เขียนโปรแกรมทดสอบการสร้าง object ของ Ball

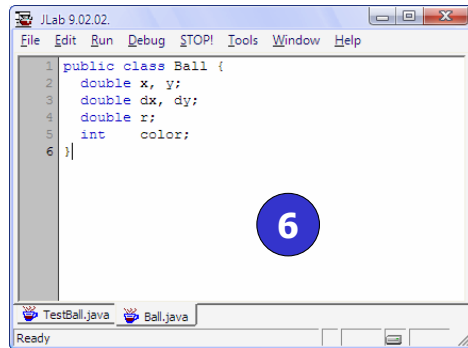
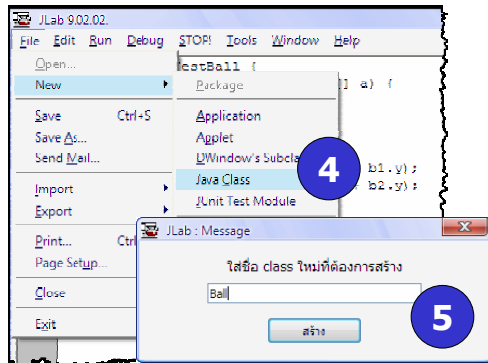
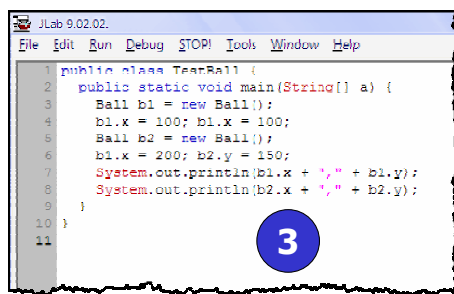
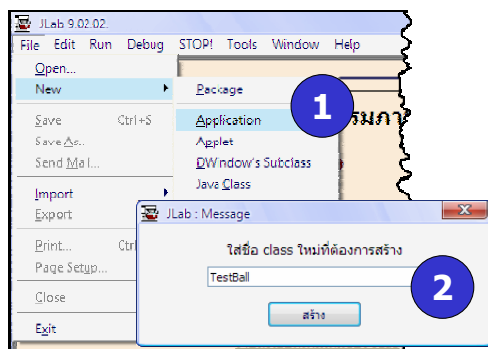
```
public class TestBall {
    public static void main(String[] a) {
        Ball b1 = new Ball();
        b1.x = 100; b1.y = 100;
        Ball b2 = new Ball();
        b2.x = 200; b2.y = 150;
        System.out.println(b1.x + " ; " + b1.y);
        System.out.println(b2.x + " ; " + b2.y);
    }
}
```

b1.x คือตัวแปร x ของ b1



2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

4



ลูกบอลแดงได้ : เขียนแบบใช้ object

```
public static void main(String[] args) {
    DWindow w = new DWindow(200, 200);
    Ball[] b = new Ball[50];
    for (int i = 0; i < b.length; i++) {
        b[i] = new Ball();
        b[i].x = 100; b[i].y = 100;
        b[i].r = 5;
        b[i].dx = -5 + 10 * Math.random();
        b[i].dy = -5 + 10 * Math.random();
        b[i].color = DWindow.randomColor();
    }
    w.setRepaintDuringSleep(true);
    while (true) {
        moveBalls(w, b);
        paintBalls(w, b);
        w.sleep(50);
    }
}
```

สร้างอาร์เรย์ที่แต่ละ
ช่องเป็นที่เก็บ Ball

สร้าง object ของ Ball
แล้วนำไปเก็บในอาร์เรย์

ตั้งค่าภายในลูกบอล

สั้น + สวย

ลูกบอลแต่ละลูกเก็บข้อมูล
บรรยายลักษณะของตัวเอง

```
public class Ball {
    double x, y;
    double dx, dy;
    double r;
    int color;
}
```

ลูกบอลแดงได้ : เขียนแบบใช้ object

```
public static void moveBalls(DWindow w, Ball[] b) {
    int width = w.getWidth();
    int height = w.getHeight();
    for (int i = 0; i < b.length; i++) {
        if ((b[i].x - b[i].r <= 0) || (b[i].x + b[i].r >= width))
            b[i].dx = -b[i].dx;
        if ((b[i].y - b[i].r <= 0) || (b[i].y + b[i].r >= height))
            b[i].dy = -b[i].dy;
        b[i].x = b[i].x + b[i].dx;
        b[i].y = b[i].y + b[i].dy;
    }
}

public static void paintBalls(DWindow w, Ball[] b) {
    w.clearBackground();
    for (int i = 0; i < b.length; i++) {
        double d = 2 * b[i].r;
        w.fillEllipse(b[i].color, b[i].x, b[i].y, d, d);
    }
}
```

ยังไม่สวยพอ รอสักครู่ !

ที่ผ่านมา

- เราเขียนคลาส เพื่อเขียน
 - public static void main(String[] args) เพื่อกำหนดจุดเริ่มต้นการทำงานของโปรแกรม
 - เมทอดอื่น ๆ ที่คำสั่งใน main จะมีการเรียกใช้

```
public class BouncingBallDemo {
    public static void main(String[] args) {
        ...
    }
    public static void moveBalls( ... ) {
        ...
    }
    public static void paintBalls( ... ) {
        ...
    }
}
```

จากนี้ไป

- เราจะเขียนคลาส เพื่อกำหนดประเภทข้อมูลใหม่ ๆ
 - เป็นที่เก็บกลุ่มของข้อมูลย่อยภายในที่มีความสัมพันธ์กัน
 - และวิธีการ (เมทอด) ต่าง ๆ ที่คลาสนี้มีให้ผู้อื่นใช้

```
public class Ball {
    double x, y;
    double dx, dy;
    double r;
    int color;
    void move(...) { ... }
    void paint(...) { ... }
}
```

```
public class Complex {
    double real;
    double imag;
    Complex add(Complex c) {...}
    Complex mult(Complex c) {...}
    double abs() {...}
}
```

```
public class BankAccount {
    String id;
    String name;
    double balance;
    void deposit(double amt) {...}
    void withdraw(double amt) {...}
    void transfer(BankAccount a, double amt) {...}
}
```

ให้คลาส Ball มีบริการ paint

```
public static void paintBalls(DWindow w, Ball[] b) {
    w.clearBackground();
    for (int i = 0; i < b.length; i++) {
        int d = 2 * b[i].r;
        w.fillEllipse(b[i].color, b[i].x, b[i].y, d, d);
    }
}
```

แบบเดิม

```
public static void paintBalls(DWindow w, Ball[] b) {
    w.clearBackground();
    for (int i = 0; i < b.length; i++) {
        b[i].paint(w);
    }
}
```

แบบใหม่

```
public class Ball {
    public double x, y;
    public double dx, dy;
    public double r;
    public int color;
}
```

b[i] เป็น Ball
เรียก paint
ในคลาส Ball

ข้อสังเกต :
ไม่มีคำว่า static

```
public void paint(DWindow w) {
    double d = 2 * r;
    w.fillEllipse(color, x, y, d, d);
}
```

ย้ายการวาดบอลมาอยู่ในคลาส Ball

ตัวแปรและเมทอดประจำอ็อบเจกต์

```
public class Ball {
    public double x, y;
    public double dx, dy;
    public double r;
    public int color;
}
```

object variables

```
public void paint(DWindow w) {
    double d = 2 * r;
    w.fillEllipse(color, x, y, d, d);
}
```

object methods

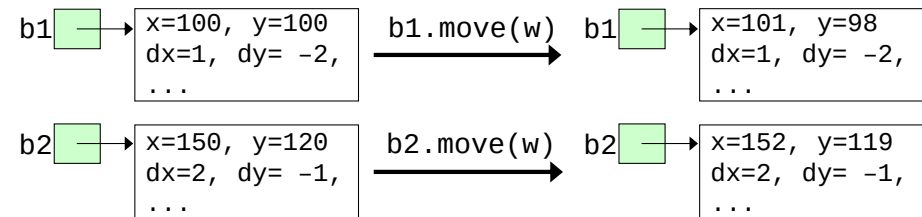
```
public void move(DWindow w) {
    int width = w.getWidth();
    int height = w.getHeight();
    if ((x - r <= 0) || (x + r >= width)) dx = -dx;
    if ((y - r <= 0) || (y + r >= height)) dy = -dy;
    x = x + dx;
    y = y + dy;
}
```

ตัวแปรอ้างอิงอ็อบเจกต์ (object reference)

```
public class TestBall {
    public static void main(String[] a) {
        Ball b1 = new Ball();
        Ball b2 = new Ball();
        b1.x = 100; b1.y = 100; b1.dx = 1; b1.dy = -2;
        b2.x = 150; b2.y = 120; b2.dx = 2; b2.dy = -1;
        DWindow w = new DWindow(200, 200);
        b1.move(w);
        b2.move(w);
        System.out.println(b1.x + ", " + b1.y);
        System.out.println(b2.x + ", " + b2.y);
    }
}
```

การใช้ตัวแปร
ประจำอ็อบเจกต์

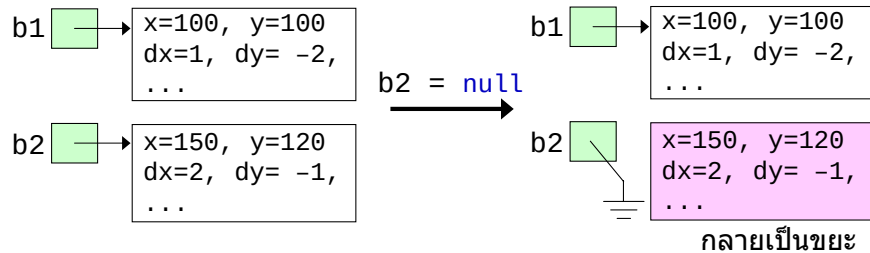
การเรียกเมทอด
ประจำอ็อบเจกต์



null

- ถ้าไม่ต้องการอ้างอิงอ็อบเจกต์ใด ๆ , ให้เก็บ **null**

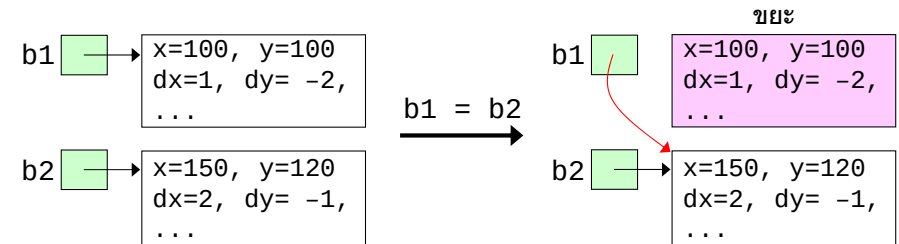
```
public class TestBall {
    public static void main(String[] a) {
        Ball b1 = new Ball();
        Ball b2 = new Ball();
        b1.x = 100; b1.y = 100; b1.dx = 1; b1.dy = -2;
        b2.x = 150; b2.y = 120; b2.dx = 2; b2.dy = -1;
        b2 = null;
        ...
    }
}
```



อ็อบเจกต์หนึ่งตัวมีตัวอ้างอิงหลายตัวได้

```
public class TestBall {
    public static void main(String[] a) {
        Ball b1 = new Ball();
        Ball b2 = new Ball();
        b1.x = 100; b1.y = 100; b1.dx = 1; b1.dy = -2;
        b2.x = 150; b2.y = 120; b2.dx = 2; b2.dy = -1;
        b1 = b2;
        System.out.println(b1.x + ", " + b2.x);
    }
}
```

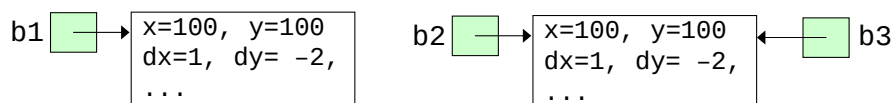
ได้ค่า 150, 150



การใช้ == กับอ็อบเจกต์

- ถ้า a และ b เป็นตัวแปรอ้างอิงอ็อบเจกต์
- a == b
 - เป็นการตรวจว่า a และ b มีค่าเดียวกันหรือไม่
 - เป็นการตรวจว่า a และ b อ้างอิงตำแหน่งเดียวกันหรือไม่
 - เป็นการตรวจว่า a และ b อ้างอิงอ็อบเจกต์ตัวเดียวกันหรือไม่

```
Ball b1 = new Ball();
Ball b2 = new Ball();
b1.x = 100; b1.y = 100; b1.dx = 1; b1.dy = -2;
b2.x = 100; b2.y = 100; b2.dx = 1; b2.dy = -2;
Ball b3 = b2;
System.out.println( (b1 == b2) ); // false
System.out.println( (b2 == b3) ); // true
```



การเรียกใช้ตัวแปรและเมทอดประจำอ็อบเจกต์

objectRef . variableName

objectRef . methodName(params)

ที่เคยพบมา ทำไมไม่เห็นแบบใช้ตัวแปรประจำอ็อบเจกต์ ?

```
DWindow w = new DWindow(200, 200);
w.drawLine(0, 0, 100, 100);
w.fade(0.2);
int width = w.getWidth();
int height = w.getHeight();
Scanner kb = new Scanner(System.in);
int a = kb.nextInt();
double b = kb.nextDouble();
String c = kb.nextLine();
String s = c.substring(0, 5);
String t = c.toUpperCase();
```

ทำไมไม่ให้ใช้
w.width
w.height

เพราะมันอันตราย !!!

w.width = -100;
w.height = 0;

จะป้องกันอย่างไร ?

การปกปิดตัวแปรประจำอ็อบเจกต์ด้วย private

```
public class Ball {
    private double x, y;
    private double dx, dy;
    private double r;
    private int color;

    public void setPosition(double x1, double y1) {
        if (x1 < 0 || y1 < 0) return;
        x = x1; y = y1;
    }
    public void setSize(double size) {
        if (size > 0) r = size;
    }
    public double getX() {
        return x;
    }
    public double getY() {
        return y;
    }
    public double getSize() {
        return r;
    }
    ...
}
```

ตัวแปรเป็น private
ใช้ได้เฉพาะในคลาสนี้

เขียนเมทอดให้บริการตั้งค่า และ
คืนค่าตัวแปรภายในที่เป็น private

ใช้ private ปกปิดเมทอดประจำอ็อบเจกต์ก็ได้

```
public class Ball {
    private double x, y;
    private double dx, dy;
    private double r;
    private int color;

    ...
    public void move(DWindow w) {
        if (hitWall(x, w.getWidth()) ) dx = -dx;
        if (hitWall(y, w.getHeight()) dy = -dy;
        x = x + dx;
        y = y + dy;
    }
    private boolean hitWall(double center, double length) {
        if ((center <= r) || (center + r >= length))
            return true;
        else
            return false;
    }
    ...
}
```

ใช้ภายใน

การตั้งค่าเริ่มต้นของตัวแปรประจำอ็อบเจกต์

- สามารถตั้งค่าเริ่มให้กับตัวแปร ณ จุดที่ประกาศตัวแปร
- ถ้าไม่ตั้งค่าเริ่มต้น ระบบจะใส่ค่าให้อัตโนมัติ
 - ให้ค่า 0 ถ้าเป็นตัวแปรประเภทจำนวน (int, double)
 - ให้ค่า false ถ้าเป็นตัวแปรประเภท boolean
 - ให้ค่า null ถ้าเป็นตัวแปรอ้างอิงอ็อบเจกต์

```
public class Ball {
    private double x = 100, y = 100;
    private double dx = 2, dy = -2;
    private double r = 5;
    ...
}
```

การส่งอ็อบเจกต์ไปยังเมทอด

- การส่งอาร์เรย์ไปยังเมทอด
 - ไม่ได้ส่งทั้งอาร์เรย์ไป
 - แต่ส่งตำแหน่งอ้างอิงอาร์เรย์
 - เมทอดที่รับตัวอ้างอิง สามารถแก้ไขข้อมูลในอาร์เรย์ได้
- การส่งอ็อบเจกต์ไปยังเมทอด
 - ไม่ได้ส่งตัวอ็อบเจกต์ไป
 - แต่ส่งตำแหน่งอ้างอิงอ็อบเจกต์
 - เมทอดที่รับตัวอ้างอิง สามารถใช้ตัวแปรและเมทอดประจำอ็อบเจกต์นั้นได้

```
int[] a = { 1, 2, 3 };
clear(a);
// ทุกช่องใน a เป็น 0
```

a

d

```
void clear(int[] d) {
    for(int i=0; i<d.length; i++)
        d[i] = 0;
}
```

```
Ball b1 = new Ball();
init(b1);
// b1 ถูกตั้งค่าเรียบร้อยแล้ว
```

b1

b

```
void init(Ball b) {
    b.x = 100;
    b.y = 100;
    b.dx = 2;
    b.dy = 2;
    b.r = 10;
    b.color = DWindow.BLACK;
}
```


ตัวอย่าง : BankAccount

```
public class BankAccount {
    private double balance = 0;
    public void deposit(double amt) {
        if (amt > 0) balance = balance + amt;
    }
    public void withdraw(double amt) {
        if (amt > 0 && balance >= amt)
            balance = balance - amt;
    }
    public double getBalance() {
        return balance;
    }
    public void transfer(BankAccount a, double amt) {
        if (amt > 0 && a.getBalance() >= amt) {
            a.withdraw(amt);
            deposit(amt);
        }
    }
}
```

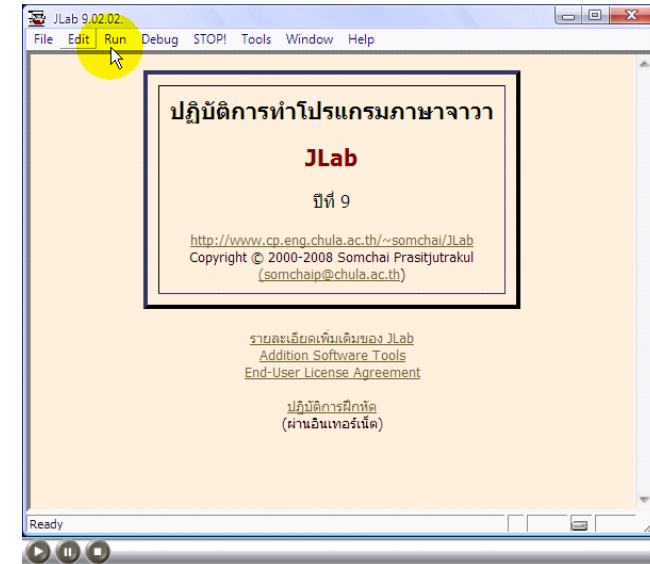
ต้อง private ไม่ให้ใครมาแก้ไข

ถ้าบัญชี a มีเงินพอ

ถอนเงินจากบัญชี a

ฝากเข้าบัญชีเรา

สาธิตการใช้งานอีอบเจกต์



มารู้จักกับตัวแปร this

```
public void transfer(BankAccount a, double amt) {
    if (amt > 0 && a.getBalance() >= amt) {
        a.withdraw(amt);
        deposit(amt);
    }
}

public void transfer(BankAccount a, double amt) {
    if (amt > 0 && a.balance >= amt) {
        a.balance = a.balance - amt;
        this.balance = this.balance + amt;
    }
}
```



ba2.transfer(...) คือการเรียกให้เมทอด transfer ประจําอีอบเจกต์ ba2 ทำงาน เมื่อ transfer เริ่มทำงานระบบจะเตรียมตัวแปรชื่อ this ให้อ้างอิงอีอบเจกต์เดียวกับที่ ba2 อ้างอิง

เขียนแบบเต็มยศ : this

```
public class BankAccount {
    private double balance = 0;
    public void deposit(double amt) {
        if (amt > 0) this.balance = this.balance + amt;
    }
    public void withdraw(double amt) {
        if (amt > 0 && this.balance >= amt)
            this.balance = this.balance - amt;
    }
    public double getBalance() {
        return this.balance;
    }
    public void transfer(BankAccount a, double amt) {
        if (amt > 0 && a.getBalance() >= amt) {
            a.withdraw(amt);
            this.deposit(amt);
        }
    }
}
```

เพิ่ม balance ของบัญชีนี้ อีก amt

ลด balance ของบัญชีนี้ ออก amt

คืนยอดเงินของบัญชีนี้

ย้ายเงินจากบัญชี a มาบัญชีนี้

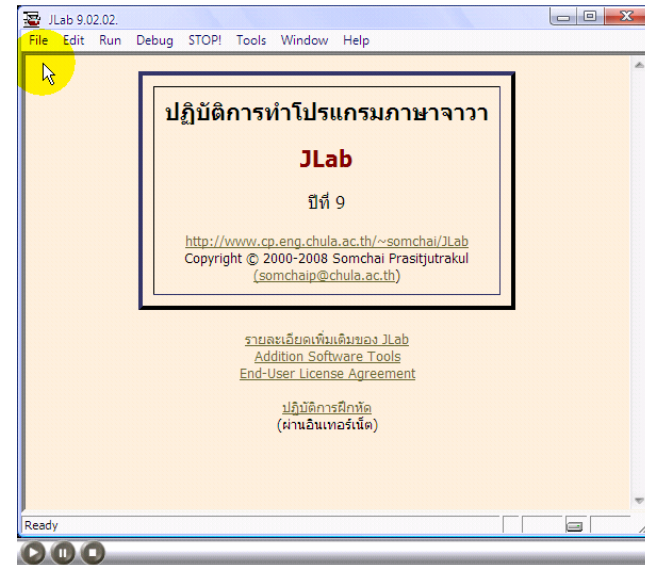
Constructor

- constructor เป็นเมทอดพิเศษของคลาสที่
 - มีชื่อเดียวกับชื่อคลาส
 - ไม่มี return type
- constructor ถูกเรียกเมื่อมีการ **new** เพื่อสร้างอ็อบเจกต์
- มีหน้าที่ตั้งค่าเริ่มต้นให้กับตัวแปรประจำอ็อบเจกต์หลังเกิดใหม่

```
Point p1 = new Point();  
Point p2 = new Point(12);  
Point p3 = new Point(3, 4);
```

```
public class Point {  
    int x, y;  
  
    public Point() {  
        x = 0; y = 0;  
    }  
    public Point(int x0) {  
        x = x0; y = 0;  
    }  
    public Point(int x0, int y0) {  
        x = x0; y = y0;  
    }  
    ...  
}
```

สาธิตการเรียก constructor



No-argument Constructor

- ถ้าไม่ได้เขียน constructor ในคลาสเลย
 - ตัวแปลภาษาจะเติม no-argument constructor
- ถ้าเราเขียน constructor ในคลาสสักหนึ่งแบบ
 - ตัวแปลภาษาจะไม่เติมให้อีก

```
public class Point1 {  
    int x, y;  
}
```

ไม่มี constructor เลย
ระบบเติมให้

```
public class Point1 {  
    int x, y;  
    public Point() {  
    }  
}
```

```
public class Point2 {  
    int x, y;  
    public Point(int x0, int y0) {  
        x = x0; y = y0;  
    }  
}
```

มี constructor แล้ว
ระบบไม่เติมให้

```
Point1 p1 = new Point1(); // ถูก  
Point2 p2 = new Point2(); // ผิด
```

ตัวอย่าง : Complex Number

- จาวาไม่มีจำนวนเชิงซ้อนให้ใช้, ต้องเขียนเอง

$$(a + bi) + (c + di) = (a + c) + (b + d)i \quad i = \sqrt{-1}$$

$$(a + bi)(c + di) = (ac - bd) + (bc + ad)i$$

$$\frac{(a + bi)}{(c + di)} = \left(\frac{ac + bd}{c^2 + d^2} \right) + \left(\frac{bc - ad}{c^2 + d^2} \right)i$$

```
public class Complex {  
    private double re; // real part  
    private double im; // imaginary part  
  
    public Complex(double r, double i) {  
        re = r;  
        im = i;  
    }  
    ...  
}
```

นอกเรื่อง : เมทอด toString

- การ print อ็อบเจกต์ a :
 - ระบบจะ print ผลที่ได้จาก a.toString()
- การ + สตริงกับอ็อบเจกต์ a :
 - ระบบจะ + สตริงกับผลที่ได้จาก a.toString()

```
public class Complex {
    private double re; // real part
    private double im; // imaginary part

    public Complex(double r, double i) {
        re = r;
        im = i;
    }

    public String toString() {
        return "(" + re + ", " + im + "i)";
    }
}
```

ควรเขียนเมทอด toString ให้กับคลาส จะได้แสดงข้อมูลภายในได้ง่าย เหมาะสำหรับการ debug โปรแกรม

```
public static void main(String[] args) {
    Complex a = new Complex(1, 2);
    System.out.println("a = " + a.toString());
    System.out.println("a = " + a);
}
```

a = (1.0, 2.0i)

Complex : การใช้งาน

```
public static void main(String[] args) {
    Complex a = new Complex(1, 2);
    Complex b = new Complex(3, 4);
    Complex c = a.add(b);
    Complex d = a.multiply(b);
    Complex e = a.divide(b);
    System.out.println("a = " + a);
    System.out.println("b = " + b);
    System.out.println("c = " + c);
    System.out.println("d = " + d);
    System.out.println("e = " + e);
    a = a.add(b);
    System.out.println("a = " + a);
}
```

a = (1.0, 2.0i)
b = (3.0, 4.0i)
c = (4.0, 6.0i)
d = (-5.0, 10.0i)
e = (0.44, 0.08i)
a = (4.0, 6.0i)

c = a.add(b);

คือ a บวกกับ b ได้ผลเก็บไว้ที่ c
ผลที่ได้เป็นอ็อบเจกต์ใหม่ที่ add ผลลัพธ์
ค่าของ a และ b ไม่เปลี่ยนแปลง

Complex : add

```
public static void main(String[] args) {
    Complex a = new Complex(1, 2);
    Complex b = new Complex(3, 4);
    Complex c = a.add(b);
}
```

```
public class Complex {
    ...
    public Complex add(Complex z) {
        double r = this.re + z.re;
        double i = this.im + z.im;
        return new Complex(r, i);
    }
}
```

ละ this. ได้

```
public class Complex {
    ...
    public Complex add(Complex z) {
        double r = re + z.re;
        double i = im + z.im;
        return new Complex(r, i);
    }
}
```

```
public class Complex {
    ...
    public Complex add(Complex z) {
        return new Complex(re + z.re, im + z.im);
    }
}
```

หาผลบวกแล้วสร้างอ็อบเจกต์
ในบรรทัดเดียวกันเลย

Complex : บวก ลบ คูณ หาร

```
public class Complex {
    ...
    public Complex add(Complex z) {
        return new Complex(re + z.re, im + z.im);
    }
    public Complex subtract(Complex z) {
        return new Complex(re - z.re, im - z.im);
    }
    public Complex multiply(Complex z) {
        return new Complex(re*z.re - im*z.im, re*z.im + im*z.re);
    }
    public Complex divide(Complex z) {
        double d = z.re*z.re + z.im*z.im;
        return new Complex((re*z.re + im*z.im)/d,
            (im*z.re - re*z.im)/d);
    }
}
```

$$(a + bi) + (c + di) = (a + c) + (b + d)i$$

$$(a + bi)(c + di) = (ac - bd) + (bc + ad)i$$

$$\frac{(a + bi)}{(c + di)} = \left(\frac{ac + bd}{c^2 + d^2} \right) + \left(\frac{bc - ad}{c^2 + d^2} \right)i$$

Complex : บวก ลบ คูณ หาร กับจำนวนจริง

```
public class Complex {
    ...
    public Complex add(double c) {
        return new Complex(re + c, im);
    }
    public Complex subtract(double c) {
        return new Complex(re - c, im);
    }
    public Complex multiply(double c) {
        return new Complex(c * re, c * im);
    }
    public Complex divide(double c) {
        return new Complex(re / c, im / c);
    }
    public double abs() {
        return Math.sqrt(re * re + im * im);
    }
}

public static Complex f(Complex x) {
    return x.multiply(x).multiply(x).subtract(1);
}
```

$$f(x) = x^3 - 1$$

ตัวอย่าง : กลับไปเรื่อง Newton's Method

```
import java.util.Scanner;
public class Newton {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("initial x = ");
        double x0 = kb.nextDouble();
        double err = 0;
        do {
            System.out.println(x0);
            double x1 = x0 - f(x0) / df(x0);
            err = Math.abs(x1 - x0);
            x0 = x1;
        } while (err > 1e-10);
        System.out.println(x0);
    }
    public static double f(double x) {
        return x * x * x - 1;
    }
    public static double df(double x) {
        return 3 * x * x;
    }
}
```

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

$$f(x) = x^3 - 1$$

$$f'(x) = 3x^2$$

```
JLab>java Newton
initial x = 2
2.0
1.41666666
1.11053440
1.01063676
1.00011155
1.00000001
1.0
```

```
JLab>java Newton
initial x = -2
-2.0
-1.25
-0.62
0.453818938
1.921048977
1.371023049
1.091348232
1.007432716
1.000054702
1.000000002
1.0
```

โปรแกรมนี้หาเฉพาะรากจริง

ตัวอย่าง : Newton's Method ใช้กับจำนวนเชิงซ้อน

```
import java.util.Scanner;
public class Newton {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("initial real = ");
        double r = kb.nextDouble();
        System.out.print("initial imag = ");
        double i = kb.nextDouble();
        Complex x0 = new Complex(r, i);
        double err = 0;
        do {
            System.out.println(x0);
            Complex x1 = x0.subtract(f(x0).divide(df(x0)));
            err = x1.subtract(x0).abs();
            x0 = x1;
        } while (err > 1e-10);
        System.out.println(x0);
    }
    public static Complex f(Complex x) {
        return x.multiply(x).multiply(x).subtract(1);
    }
    public static Complex df(Complex x) {
        return x.multiply(x).multiply(3);
    }
}
```

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

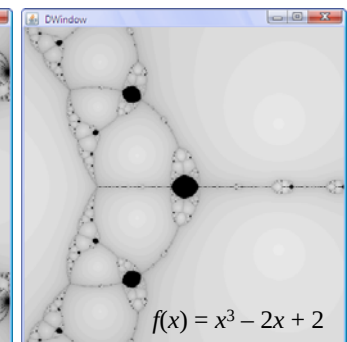
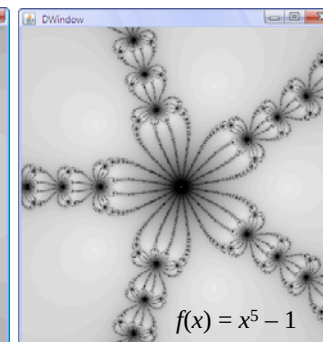
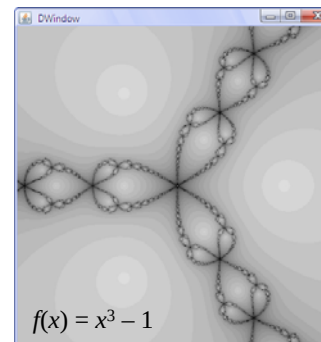
$$f(x) = x^3 - 1$$

$$f'(x) = 3x^2$$

```
JLab>java Newton
initial real = -3
initial imag = -3
(-3.0, -3.0i)
(-2.0, -2.018i)
(-1.333, -1.386i)
(-0.892, -1.014i)
(-0.618, -0.857i)
(-0.506, -0.854i)
(-0.499, -0.866i)
(-0.500, -0.866i)
(-0.499, -0.866i)
(-0.5, -0.866i)
```

นอกเรื่อง : Newton's Fractal

- ลองใช้วิธีของนิวตันหารากของ $f(x)$ โดยเริ่มที่จุดเริ่มต้นต่าง ๆ กันบนระนาบจำนวนเชิงซ้อน
- แล้วนับจำนวนรอบที่ต้องทำงานจน converge
- วาดรูปโดยให้สีขึ้นกับจำนวนรอบที่ใช้



```

import java.awt.*;
public class NewtonFractal {
    public static void main(String[] args) {
        DWindow w = new DWindow(400, 400);
        double width = w.getWidth(), height = w.getHeight();
        int[][] p = w.getPixmap();
        for (int x = 0; x < width; x++) {
            for (int y = 0; y < height; y++) {
                double re = -1.5 + 3 * x / width;
                double im = -1.5 + 3 * y / height;
                int k = numLoops(new Complex(re, im));
                int g = 255 - k * 255/50;
                p[x][y] = DWindow.mixRGB(g, g, g);
            }
        }
        w.setPixmap(p);
    }
    private static int numLoops(Complex z) {
        int k = 0;
        double diff = 0;
        do {
            Complex z1 = z.subtract(f(z).divide(df(z)));
            diff = z.subtract(z1).abs();
            z = z1;
            k++;
        } while (k < 50 && diff > 1e-10);
        return k;
    }
    private static Complex f(Complex z) { ... }
    private static Complex df(Complex z) { ... }
}

```

โปรแกรมวาด Newton Fractal

เดิม f และ df ที่สนใจ
แล้วลองสังเกตงานดู

2110101 จุฬ

37

ตัวแปรและเมทอดแบบ static

```

public class A {
    public static int v1;
    public static void m1() { ... }

    public int v2;
    public void m2() { ... }
}

```

ตัวแปรประจำคลาส

เมทอดประจำคลาส

ตัวแปรประจำอ็อบเจกต์

เมทอดประจำอ็อบเจกต์

```

public class TestA {
    public static void main(String[] a) {
        A.v1++;
        A.m1();

        A obj = new A();
        obj.v2++;
        obj.m2();

        obj.v1++;
        obj.m1();
    }
}

```

เรียกใช้ตัวแปรและเมทอดประจำคลาส

ต้องระบุชื่อคลาส

เรียกใช้ตัวแปรและเมทอดประจำอ็อบเจกต์

ต้องระบุอ็อบเจกต์

เรียกใช้ตัวแปรและเมทอดประจำคลาส

โดยการระบุอ็อบเจกต์ก็ได้

2110101 วิศวกรรมคอมพิวเตอร์ จุฬฯ (22/05/51)

38

ตัวแปรแบบ static มีชุดเดียวประจำคลาส

```

public class A {
    public static int v1;
    public static void m1() { ... }

    public int v2;
    public void m2() { ... }
}

```

ตัวแปรประจำคลาส

เมทอดประจำคลาส

ตัวแปรประจำอ็อบเจกต์

เมทอดประจำอ็อบเจกต์

```

public class TestA {
    public static void main(String[] a) {
        A obj1 = new A();
        A obj2 = new A();
        A.v1++; obj1.v2++;
        A.v1++; obj2.v2++;
    }
}

```

class A

class A

class A

class A

obj1

obj1

obj1

obj1

v2

v2

v2

v2

obj2

obj2

obj2

2110101 วิศวกรรมคอมพิวเตอร์ จุฬฯ (22/05/51)

39

ตัวอย่าง : ใช้ตัวแปร static นับจำนวนอ็อบเจกต์

```

public class A {
    private static int objCount = 0;
    ...
    public A() {
        objCount++;
    }
    public static int getObjectCount() {
        return objCount;
    }
    ...
}

```

```

public class TestA {
    public static void main(String[] a) {
        for (int i = 0; i < 10; i++) {
            new A();
        }
        System.out.println( A.getObjectCount() );
    }
}

```

มีค่าเท่าไร ?

2110101 วิศวกรรมคอมพิวเตอร์ จุฬฯ (22/05/51)

40

เมทอดแบบ static ใน DWindow

```
public static int[][] sepia(int[][] p) {
    int[][] out = new int[p.length][p[0].length];
    for (int x = 0; x < p.length; x++) {
        for (int y = 0; y < p[x].length; y++) {
            int r = DWindow.getR(p[x][y]);
            int g = DWindow.getG(p[x][y]);
            int b = DWindow.getB(p[x][y]);
            int r1 = (int)(r * 0.393 + g * 0.769 + b * 0.189);
            int g1 = (int)(r * 0.349 + g * 0.686 + b * 0.168);
            int b1 = (int)(r * 0.272 + g * 0.534 + b * 0.131);
            out[x][y] = DWindow.mixRGB(r1, g1, b1);
        }
    }
    return out;
}
```

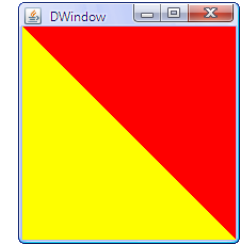
static methods

mixRGB, getR, getG, getB เป็นบริการเสริมของ DWindow
ไม่ต้องมีอ็อบเจกต์ของ DWindow ก็เรียกใช้เมทอดเหล่านี้ได้

ตัวแปรแบบ static ใน DWindow

```
DWindow w = new DWindow(200, 200);
int[][] p = w.getPixmap();
for (int x = 0; x < p.length; x++) {
    for (int y = 0; y < p[0].length; y++) {
        if (x > y)
            p[x][y] = w.RED;
        else
            p[x][y] = w.YELLOW;
    }
}
w.setPixmap(p);
```

จริง ๆ แล้วเป็น
static variables



```
public class DWindow {
    public static int BLACK = mixRGB( 0, 0, 0);
    public static int BLUE = mixRGB( 0, 0, 255);
    public static int GREEN = mixRGB( 0, 255, 0);
    public static int RED = mixRGB(255, 0, 0);
    public static int WHITE = mixRGB(255, 255, 255);
    ...

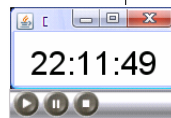
    public static int mixRGB(int r, int g, int b) {...}
    ...
}
```

เป็น static
จะได้มีเพียง
ชุดเดียว

ตัวอย่าง : โปรแกรมนาฬิกาตัวเลข

```
import jlab.graphics.DWindow;
import java.util.Date;
```

```
public class DigitalClock {
    public static void main(String[] args) {
        DWindow w = new DWindow(150, 50);
        Date now = new Date();
        int h = now.getHours();
        int m = now.getMinutes();
        int s = now.getSeconds();
        Time t = new Time(h, m, s);
        while (true) {
            w.clearBackground();
            w.drawString(t.toString(), 30, 10, 10);
            w.sleep(1000);
            t.increment();
        }
    }
}
```



คลาสมาตรฐานของจาวา

object methods ของ Date

ต้องการสร้างคลาส Time

แปลงเวลาเป็นสตริงเพื่อแสดงผล

เพิ่มเวลาอีก 1 วินาที

ตัวอย่าง : คลาส Time

```
public class Time {
    private int hour;
    private int minute;
    private int second;
    public Time(int h, int m, int s) {
        hour = h; minute = m; second = s;
    }
    public void increment() { // เพิ่ม 1 วินาที
        second = (second + 1) % 60;
        if (second == 0) {
            minute = (minute + 1) % 60;
            if (minute == 0) hour = (hour + 1) % 24;
        }
    }
    public String toString() {
        return twoDigits(hour) + ":" +
            twoDigits(minute) + ":" + twoDigits(second);
    }
    private String twoDigits(int d) {
        String s = "00" + d;
        return s.substring(s.length() - 2, s.length());
    }
}
```

โจทย์ : คลาส Time

จงเขียนเมทอดเพิ่มให้กับคลาส Time

- **public boolean isAM()**
 - คืน true ถ้าเป็นเวลาก่อนเที่ยงวัน, ไม่เช่นนั้นคืน false
- **public int toSeconds()**
 - แปลงเวลานี้ให้เป็นจำนวนวินาทีตั้งแต่เที่ยงคืน
- **public void decrement()**
 - ลดเวลานี้ลง 1 วินาที
- **public void add(Time t)**
 - เพิ่มเวลานี้ไปอีกเท่ากับปริมาณเวลาที่กำหนดให้ใน t
- **public boolean equals(Time t)**
 - คืน true ถ้าเวลานี้เป็นเวลาเดียวกับเวลา t, ไม่เช่นนั้นคืน false

โจทย์ : จงเขียนคลาส "Rational"

- Rational เป็นคลาสเก็บเลขตรรกยะ
 - ด.ย. $\frac{3}{4}$, $\frac{15}{7}$, $\frac{8}{1}$
- มีตัวแปรภายในสองตัว :
 - เศษ (numerator) และ
 - ส่วน (denominator)
 - ด.ย. $\frac{15}{7}$ มี numerator เป็น 15 และ denominator เป็น 7
- มีเมทอดดังนี้
 - add, subtract, multiply, divide, toString, และ toDouble
- ควรเก็บแบบลดรูป เช่น $\frac{4}{8}$ ควรเก็บเป็น $\frac{1}{2}$
- อาจต้องมี private เมทอดเพื่อหา ห.ร.ม. ไว้ใช้ภายใน

2110101 : การทำโปรแกรมคอมพิวเตอร์

Inheritance

ภาควิชาวิศวกรรมคอมพิวเตอร์
จุฬาลงกรณ์มหาวิทยาลัย

DotMatrixLED : วิธีใช้

```
import jlab.graphics.DWindow;

public class TestDotMatrixLED {
    public static void main(String[] args) {
        boolean[][] J = {{ false, false, true, true, true },
                          { false, false, false, true, false },
                          { false, false, false, true, false },
                          { false, false, false, true, false },
                          { false, false, false, true, false },
                          { true, false, false, true, false },
                          { false, true, true, false, false } };

        DWindow w = new DWindow(120, 170);
        w.setBackground(w.BLACK);
        DotMatrixLED led = new DotMatrixLED(5, 7);
        led.setWidth(100);
        led.setDots(J);
        led.setColor(w.GREEN);
        led.paint(w, 10, 10);
    }
}
```

5 x 7

7 แถว
แถวละ 5 จุด

พิกัด (x,y) มุมซ้ายบน

100

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

2

DotMatrixLED

```
import jlab.graphics.DWindow;

public class DotMatrixLED {
    private boolean[][] dots;
    private double width;
    private int color;

    public DotMatrixLED(int w, int h) {
        width = 50;
        color = DWindow.GREEN;
        dots = new boolean[h][w];
    }

    public void setDots(boolean[][] s) {
        if (s.length == dots.length && s[0].length == dots[0].length) {
            for (int i = 0; i < s.length; i++)
                for (int j = 0; j < s[0].length; j++)
                    dots[i][j] = s[i][j];
        }
    }

    public void setWidth(double w) {
        if (w > 0) width = w;
    }

    public int getWidth() {
        return (int) width;
    }
}
```

5 x 7

7 แถว
แถวละ 5 จุด

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

3

DotMatrixLED

```
public class DotMatrixLED {
    private boolean[][] dots;
    private double width;
    private int color;
    ...

    public void setColor(int c) {
        color = c;
    }

    public void paint(DWindow w, double x, double y) {
        double wd = width / dots[0].length;
        x = x + wd / 2;
        y = y + wd / 2;
        for (int i = 0; i < dots.length; i++) {
            for (int j = 0; j < dots[0].length; j++) {
                if (dots[i][j]) {
                    w.fillEllipse(color, x + j * wd, y + i * wd, wd, wd);
                }
            }
        }
    }
}
```

5 x 7

7 แถว
แถวละ 5 จุด

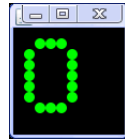
2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

4

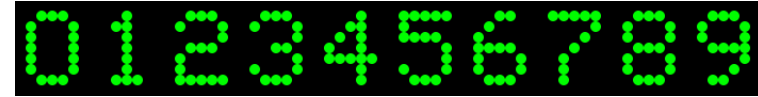
อยากได้ DecimalDotMatrixLED

```
import jlab.graphics.DWindow;

public class TestDecimalDotMatrixLED {
    public static void main(String[] args) {
        DWindow w = new DWindow(100, 100);
        w.setBackground(w.BLACK);
        DecimalDotMatrixLED led = new DecimalDotMatrixLED();
        led.setColor(w.GREEN);
        led.setWidth(50);
        led.setDigit(0);
        while (true) {
            w.clearBackground();
            led.paint(w, 10, 10);
            w.sleep(1000);
            led.setDigit((s.getDigit() + 1) % 10);
        }
    }
}
```



ออกแบบเลข 0 ถึง 9



```
public class DecimalDotMatrixLED {
    private static boolean[][] zero = {
        { false, true, true, true, false },
        { true, false, false, false, true },
        { true, false, false, false, true },
        { true, false, false, false, true },
        { true, false, false, false, true },
        { true, false, false, false, true },
        { false, true, true, true, false }
    };

    private static boolean[][] one = {
        { false, false, true, false, false },
        { false, true, true, false, false },
        { false, false, true, false, false },
        { false, false, true, false, false },
        { false, false, true, false, false },
        { false, false, true, false, false },
        { false, true, true, true, false }
    };

    ...

    private static boolean[][] nine = {
        { false, true, true, true, false },
        ...
    };
}
```

เขียน two, three, four, ... nine เอง

เขียน DecimalDotMatrixLED จาก DotMatrixLED

```
import jlab.graphics.DWindow;
public class DecimalDotMatrixLED extends DotMatrixLED {
    ...
    private static boolean[][][] dots = { zero, one, two,
        three, four, five, six, seven, eight, nine };

    private int digit;

    public DecimalDotMatrixLED() {
        super(5, 7);
        super.setDots(zero);
    }

    public void setDigit(int n) {
        if (0 <= n && n <= 9) {
            digit = n;
            super.setDots(dots[digit]);
        }
    }

    public int getDigit() {
        return digit;
    }
}
```

constructor ของคลาส
ใหม่เรียก constructor
ของ superclass

เรียกเมทอดของ superclass ได้

คลาส DecimalDotMatrixLED ขยาย
คุณสมบัติจากคลาส DotMatrixLED
(superclass) คลาสใหม่ใช้ของเดิม
ของ superclass ได้ และเขียนตัวแปร
และเมทอดใหม่เพิ่มเติม

Inheritance

- class B extends A
- B ว่าเป็น subclass ของ A
- A ว่าเป็น superclass ของ B
- B เสมือนมีทุกๆ ตัวแปร ของ A
- B เสมือนมีทุกๆ เมทอด ของ A (ยกเว้น constructors)

```
public class A {
    private int a0;
    public int a1;

    public A() {}
    public A(int a) {}

    private void m0() {}
    public void m1() {}
}
```

```
public class Test {
    public static void main(String[] args) {
        B b = new B();
        int c = b.b2 + b.a1;
        b.m1();
        b.m2();
        int d = b.a0; // ผิด
        b.m0(); // ผิด
        B obj = new B(5); // ผิด
    }
}
```

```
public class B extends A {
    {
        public int b2;

        public B() {}
        public void m2() {}
    }
}
```

ของที่ไม่ private ของ superclass
เปรียบเหมือนเป็นของ subclass

DecimalDotMatrixLED extends DotMatrixLED

```
public class DotMatrixLED {
    public DotMatrixLED(int w, int h) {...}
    public void setDots(boolean[][] s) {...}
    public void setWidth(double w) {...}
    public int getWidth() {...}
    public void setColor(int c) {...}
    public void paint(DWindow w, double x, double y) {...}
}
```

```
public class DecimalDotMatrixLED extends DotMatrixLED {
    public DecimalDotMatrixLED() {...}
    public void setDigit(int n) {...}
    public int getDigit() {...}
}
```

ด้วยเหตุที่ DecimalDotMatrixLED extends DotMatrixLED ทำให้ DecimalDotMatrixLED มีเมทอด

- setDigit
 - getDigit
 - setWidth, getWidth
 - setColor, paint
 - setDots
- ของ DecimalDotMatrixLED
- ได้รับมรดกมาจาก DotMatrixLED

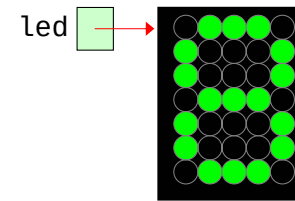
ข้อสังเกตของ DecimalDotMatrixLED คงไม่อยากให้ใครมาเรียก setDots

มรดกที่จำต้องรับ

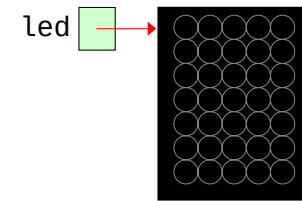
- เมทอดของ superclass ตกทอดให้ subclass
- subclass ไม่ต้องการรับ ไม่ได้

```
public class Test {
    public static void main(String[] args) {
        DecimalDotMatrixLED led = new DecimalDotMatrixLED();
        led.setDigit(8);
        led.setDots(new boolean[7][5]); // เป็น false หมด
    }
}
```

ไม่ควรเปิดให้บริการ setDots เลย



led.setDigit(8)



led.setDots(new boolean[7][5])

Method Overriding

- subclass เขียนพฤติกรรมใหม่ให้เมทอดที่เป็นมรดกได้
- เขียนหัวเมทอดที่ subclass ให้เหมือนของที่ superclass

```
public class DecimalDotMatrixLED extends DotMatrixLED {
    public DecimalDotMatrixLED() {...}
    public void setDigit(int n) {...}
    public int getDigit() {...}

    public void setDots(boolean[][] s) {
        return; // ไม่ต้องทำอะไร return เลย
    }
}
```

```
public class Test {
    public static void main(String[] args) {
        DecimalDotMatrixLED led = new DecimalDotMatrixLED();
        led.setDigit(8);
        led.setDots(new boolean[7][5]);
    }
}
```

ตัวอย่าง : FrameDecimalLED

```
import jlab.graphics.DWindow;

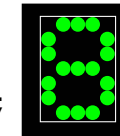
public class FrameDecimalLED extends DecimalDotMatrixLED {
    private int frameColor;

    public FrameDecimalLED() {
        frameColor = DWindow.WHITE;
    }

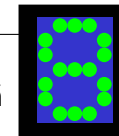
    public void setFrameColor(int c) {
        frameColor = c;
    }

    public void paint(DWindow w, int x, int y) {
        super.paint(w, x, y);
        double width = super.getWidth();
        w.drawRect(frameColor, x, y, width, 7 * width / 5);
    }
}
```

this. ของเรา super. ของบรรพบุรุษ

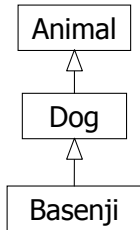


override เมทอด paint เริ่มด้วยการเรียก paint ของ superclass ตามด้วยการวาดกรอบสี่เหลี่ยม



โจทย์ : จงเขียน BackColorDecimalLED ที่ตั้งสีพื้นได้

A extends B : A เป็นประเภทพิเศษของ B



Basenji
"Barkless dog"

```

public class Animal {
    public void eat(Food f) {...}
    public void sleep() {
        System.out.println("Zzzz...");
    }
    ...
}
    
```

```

public class Dog extends Animal {
    public void bite(Animal a) {...}
    public void bark() {
        System.out.println("โห้ง ๆ");
    }
    ...
}
    
```

```

public class Basenji extends Dog {
    public void bark() {
        System.out.println("...");
    }
    ...
}
    
```

```

Dog lek = new Dog();
Basenji dang = new Basenji();
lek.bark(); // โห้ง ๆ
dang.bark(); // ...
    
```

BankAccount

```

public class BankAccount {
    public double getBalance() {...}
    public void deposit(double a) {...}
    public void withdraw(double a) {...}
    public void transfer(BankAccount ac, double a) {...}
}
    
```

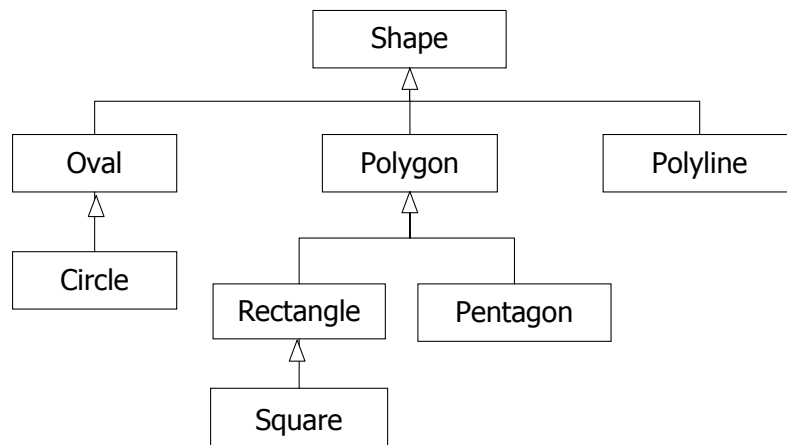
```

public class SavingAccount extends BankAccount {
    public void setInterestRate(double r) {...}
    public double getInterestRate() {...}
    public void addInterest() {...}
}
    
```

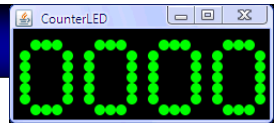
```

public class OverdraftAccount extends BankAccount {
    public double getOverdraft() {...}
    public double getOverdraftMax() {...}
    public void deposit(double a) {...}
    public void withdraw(double a) {...}
}
    
```

Inheritance Hierarchy



ตัวอย่าง : Counter



```

import jlab.graphics.DWindow;
public class CounterLED extends DWindow {
    private DecimalDotMatrixLED[] digits;
    private int value;
}
    
```

การ extends ไม่จำเป็นต้องมี
source code ของ superclass

```

public CounterLED(int numDigits, double digitSize) {
    super((numDigits * 1.2) * digitSize, digitSize * 7 / 5 * 1.2);
    super.setBackground(super.BLACK);
    digits = new DecimalDotMatrixLED[numDigits];
    for (int i = 0; i < digits.length; i++) {
        digits[i] = new DecimalDotMatrixLED();
        digits[i].setWidth(digitSize);
    }
}

public void setValue(int d) {
    value = d;
    double width = 1.2 * digits[0].getWidth();
    super.clearBackground();
    for (int i = digits.length - 1; i >= 0; i--) {
        digits[i].setDigit(d % 10);
        digits[i].paint(this, (i+0.1)*width, 10);
        d = d / 10;
    }
}

public void increment() { setValue(value + 1); }
}
    
```

```

CounterLED c =
    new CounterLED(4, 50);
c.setValue(0);
while (true) {
    c.sleep(1000);
    c.increment();
}
    
```

โจทย์ : Traffic Countdown Display

```
public class TrafficLED extends DWindow {
    private DecimalDotMatrixLED[] digits;
    public TrafficLED(double width) {
        super(width, width / 2);
        super.setBackground(super.BLACK);
        digits = new DecimalDotMatrixLED[3];
        for (int i = 0; i < digits.length; i++)
            digits[i] = new DecimalDotMatrixLED();
            digits[i].setWidth(width / 3.6);
        }
    }
    public void setValue(int v, int color) {
        ...
    }
}
```



จงเขียนเมทอด
ทั้งหลายให้สมบูรณ์

```
public class TrafficTimer {
    ...
    public TrafficTimer(int r, int g, int allR) {
        ...
    }
    public void start() {
        ...
    }
    public static void main(String[] args) {
        new TrafficTimer(20, 15, 2).start();
    }
}
```

บอกเฉพาะแยกแวนอนก็พอ
2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/01/2564)

แดง 20 วินาที, เขียว 15 วินาที, มี 2 วินาทีที่แดงทุกแยก

2110101 : การทำโปรแกรมคอมพิวเตอร์

การโปรแกรมแบบเวียนเกิด

ภาควิชาวิศวกรรมคอมพิวเตอร์

จุฬาลงกรณ์มหาวิทยาลัย

เมทอดหาผลรวม $0+1+2+3+...+n$

```
public static int sum(int n) {  
    return n * (n + 1) / 2;  
}
```

แบบที่ 0

```
public static int sum(int n) {  
    int sum = 0;  
    for (int i = 1; i <= n; i++) sum = sum + i;  
    return sum;  
}
```

แบบที่ 1

```
public static int sum(int n) {  
    if (n == 0) return 0;  
    return sum(n - 1) + n;  
}
```

กรณีเล็กสุด

ตามนิยามของผลบวก

แบบที่ 2

$$\sum_{i=0}^n i = \left(\sum_{i=0}^{n-1} i \right) + n$$

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

2

เมทอดหาค่า $n!$

```
public static int fac(int n) {  
    int fac = 1;  
    for (int i = 1; i <= n; i++) {  
        fac = fac * i;  
    }  
    return fac;  
}
```

แบบที่ 1

$$n! = n \times (n-1) \times (n-2) \times \dots \times 2 \times 1$$

```
public static int fac(int n) {  
    if (n == 0) return 1;  
    return n * fac(n-1);  
}
```

กรณีเล็กสุด

ตามนิยามของแฟกทอเรียล

แบบที่ 2

$$n! = n \times (n-1)!$$

$$0! = 1$$

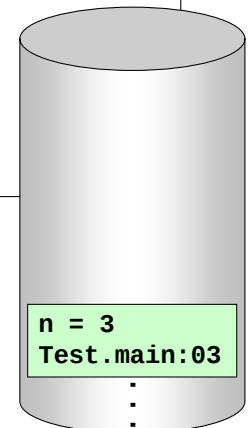
2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

3

การเก็บข้อมูลระหว่างการเรียกเมทอด

```
01: public class Test{  
02:     public static void main(String[] args) {  
03:         System.out.println(fac(3));  
04:     }  
05:     public static int fac(int n) {  
06:         if (n == 0) return 1;  
07:         int f = fac(n-1);  
08:         return n * f;  
09:     }  
10: }
```

เมื่อระบบเรียกเมทอด fac
ระบบจะสร้างตัวแปรของ fac



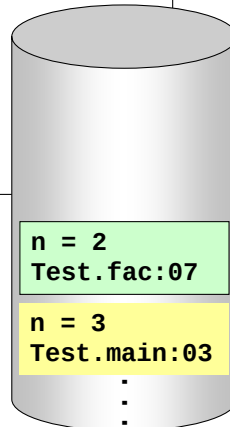
2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (22/05/51)

4

การเก็บข้อมูลระหว่างการเรียกเมทอด

```

01: public class Test{
02:   public static void main(String[] args) {
03:     System.out.println(fac(3));
04:   }
05:   public static int fac(int n) {
06:     if (n == 0) return 1;
07:     int f = fac(n-1);
08:     return n * f;
09:   }
10: }
    
```

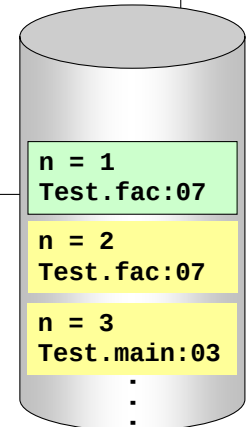


เมื่อเรียกเมทอด fac อีก
ระบบจะสร้างตัวแปรชุดใหม่ของ fac
ชุดใหม่ทับชุดเก่า
(จะนำตัวแปรชุดเก่ากลับมาใช้ใหม่
เมื่อ return)

การเก็บข้อมูลระหว่างการเรียกเมทอด

```

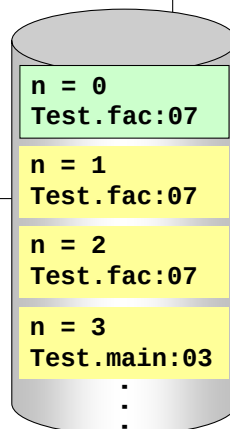
01: public class Test{
02:   public static void main(String[] args) {
03:     System.out.println(fac(3));
04:   }
05:   public static int fac(int n) {
06:     if (n == 0) return 1;
07:     int f = fac(n-1);
08:     return n * f;
09:   }
10: }
    
```



การเก็บข้อมูลระหว่างการเรียกเมทอด

```

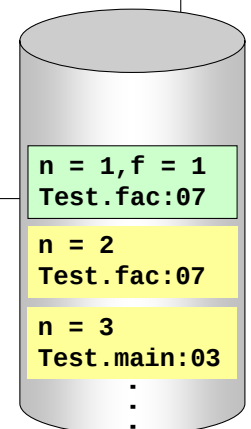
01: public class Test{
02:   public static void main(String[] args) {
03:     System.out.println(fac(3));
04:   }
05:   public static int fac(int n) {
06:     if (n == 0) return 1;
07:     int f = fac(n-1);
08:     return n * f;
09:   }
10: }
    
```



การเก็บข้อมูลระหว่างการเรียกเมทอด

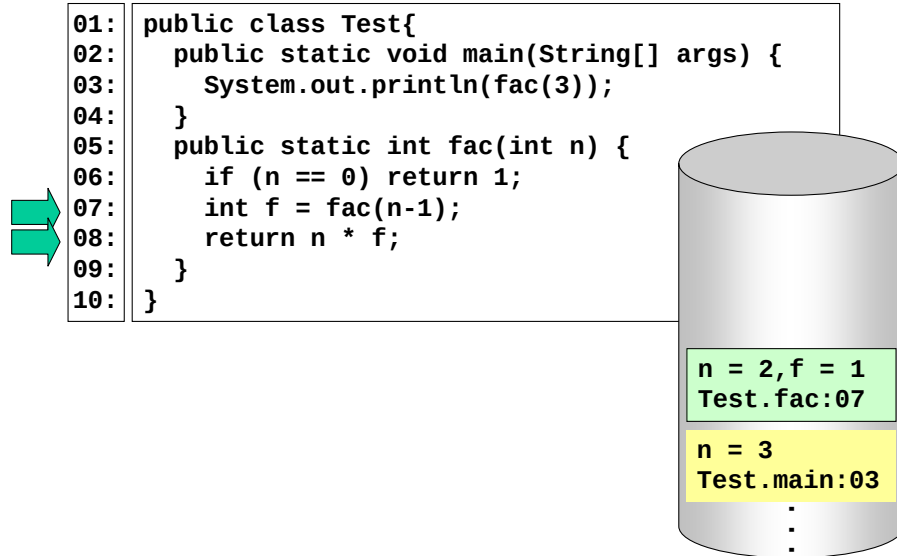
```

01: public class Test{
02:   public static void main(String[] args) {
03:     System.out.println(fac(3));
04:   }
05:   public static int fac(int n) {
06:     if (n == 0) return 1;
07:     int f = fac(n-1);
08:     return n * f;
09:   }
10: }
    
```

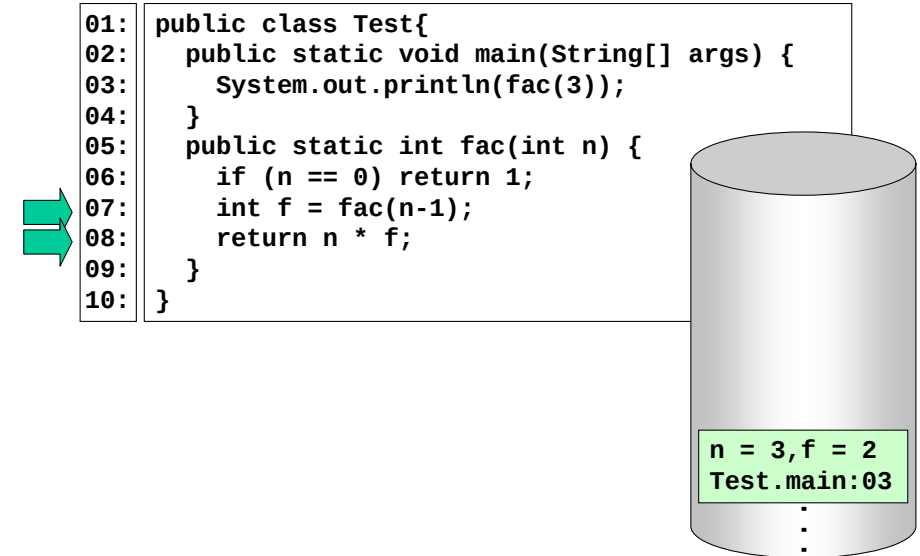


เมื่อ return
ระบบลบตัวแปรของเมทอดออก
และนำตัวแปรชุดล่าสุดมาใช้ต่อ

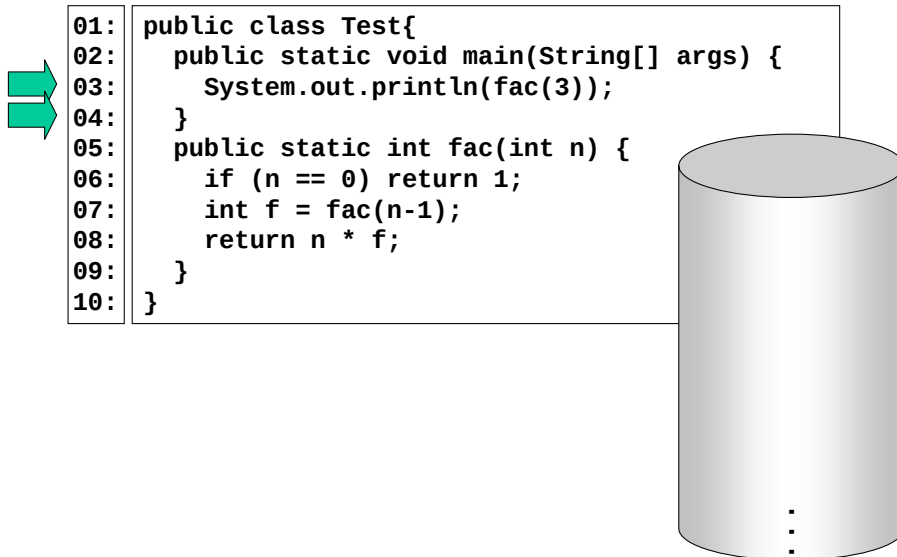
การเก็บข้อมูลระหว่างการเรียกเมทอด



การเก็บข้อมูลระหว่างการเรียกเมทอด



การเก็บข้อมูลระหว่างการเรียกเมทอด



เมทอดหาจำนวน Fibonacci ตัวที่ n

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

$F_n = F_{n-1} + F_{n-2}$ เมื่อ $n = 2, 3, 4, \dots$ โดยที่ $F_0 = 0, F_1 = 1$

```

public static int fib(int n) {
    int[] f = new int[n + 1];
    f[0] = 0; f[1] = 1;
    for (int i = 2; i <= n; i++) f[i] = f[i-1] + f[i-2];
    return f[n];
}
    
```

แบบที่ 1

ลองปรับปรุงแบบไม่ต้องใช้อาเรย์

```

public static int fib(int n) {
    if (n == 0) return 0;
    if (n == 1) return 1;
    return fib(n - 1) + fib(n - 2);
}
    
```

แบบที่ 2

ลองหา fib(20) ทั้งสองวิธี, ลองหา fib(50) ทั้งสองวิธี

เมทอดหาค่า a^k

```
public static int pow1(int a, int k) {
    int p = 1;
    for (int i = 1; i <= k; i++) {
        p = p * a;
    }
    return p;
}
```

แบบที่ 1

```
public static int pow2(int a, int k) {
    if (k == 0) return 1;
    return a * pow2(a, k - 1);
}
```

แบบที่ 2

กรณีเล็กสุด

$$a^k = a \times a^{k-1}$$

```
public static int pow3(int a, int k) {
    if (k == 0) return 1;
    int p = pow3(a, k / 2);
    p = p * p;
    if (k % 2 == 1) p = p * a;
    return p;
}
```

แบบที่ 3

$$a^k = \begin{cases} 1 & \text{if } k = 0 \\ \left(a^{\lfloor k/2 \rfloor}\right)^2 & \text{if } k \text{ is even} \\ a \left(a^{\lfloor k/2 \rfloor}\right)^2 & \text{if } k \text{ is odd} \end{cases}$$

เมทอดตรวจสอบว่าอาเรย์เรียงลำดับหรือไม่

```
public static boolean isSorted(int[] d) {
    for (int i = 1; i < d.length; i++) {
        if (d[i - 1] > d[i]) return false;
    }
    return true;
}
```

แบบที่ 1

```
public static boolean isSorted(int[] d) {
    return isSorted(d, 0, d.length - 1);
}
private static boolean isSorted(int[] d, int b, int e) {
    if (b == e) return true;
    if (d[b] > d[b + 1]) return false;
    return isSorted(d, b + 1, e);
}
```

แบบที่ 2

กรณีเล็กสุด มีตัวเดียวถือว่าเรียง

เมทอดค้นข้อมูลในอาเรย์

```
public static boolean search(int x, int[] d) {
    for (int i = 0; i < d.length; i++) {
        if (d[i] == x) return true;
    }
    return false;
}
```

แบบที่ 1

```
public static boolean search(int x, int[] d) {
    return search(x, d, 0);
}
private static boolean search(int x, int[] d, int b) {
    if (b >= d.length) return false;
    if (d[b] == x) return true;
    return search(x, d, b + 1);
}
```

แบบที่ 2

กรณีเล็กสุด ไม่มีข้อมูลคือหาไม่พบ

โจทย์ : เมทอดค้นข้อมูลในอาเรย์ (binary search)

```
public static int binarySearch(int x, int[] d) {
    int left = 0, right = d.length - 1;
    while (left <= right) {
        int mid = (left + right) / 2;
        if (x == d[mid]) return mid;
        if (x < d[mid])
            right = mid - 1;
        else
            left = mid + 1;
    }
    return - 1;
}
```

แบบที่ 1

```
public static int binarySearch(int x, int[] d) {
    ...
}
```

แบบที่ 2

จงเขียนแบบ recursive

เมท็อดกลับลำดับสตริง (แบบที่ 1)

```
public static String rev(String s) {
    String rev = "";
    for (int i = s.length()-1; i >= 0; i--) {
        rev = rev + s.substring(i, i + 1);
    }
    return rev;
}
```

แบบที่ 1

s = "ABCD"
 rev=""
 i= 0, rev="D"
 i= 1, rev="DC"
 i= 2, rev="DCB"
 i= 3, rev="DCBA"

เมท็อดกลับลำดับสตริง (แบบที่ 2)

```
public static String rev(String s) {
    if (s.length() == 0) return "";
    String r = rev(s.substring(1, s.length())) +
                s.substring(0, 1);
    return r;
}
```

แบบที่ 2

rev("ABCD") = rev("BCD") + "A" = "DCBA"
 rev("BCD") = rev("CD") + "B" = "DCB"
 rev("CD") = rev("D") + "C" = "DC"
 rev("D") = rev("") + "D" = "D"
 rev("") = ""

เมท็อดแปลงจำนวนเต็มเป็นสตริงฐานสอง

```
public static String toBinaryString(int x) {
    String out = "";
    while (x > 0) {
        out = (x % 2) + out;
        x = x / 2;
    }
    return "0" + out;
}
```

แบบที่ 1

1%2 3%2 6%2 13%2
 ↓ ↓ ↓ ↓
 "0" + "1" + "1" + "0" + "1" + ""

```
public static String toBinaryString(int x) {
    if (x == 0) return "0";
    String s = toBinaryString(x / 2);
    return s + (x % 2);
}
```

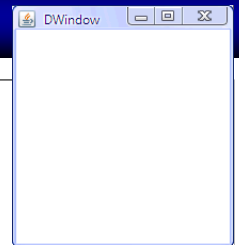
แบบที่ 2

toBinaryString(13) = toBinaryString(6) + (x%2)
 = "0110" + 1
 = "01101"

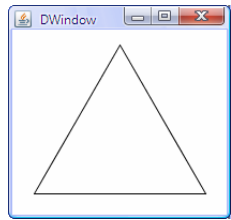
เมท็อดวาดต้นไม้

```
import jlab.graphics.DWindow;

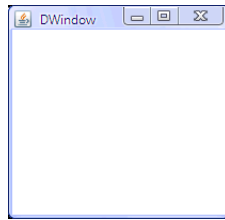
public class DrawTree {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        drawTree(w, 100, 190, 90, 80, 9);
    }
    private static void drawTree(DWindow w, double x0, double y0,
        double angle, double length, int level) {
        if (level <= 0 || length <= 0) return;
        double x1 = x0 + length * Math.cos(Math.toRadians(angle));
        double y1 = y0 - length * Math.sin(Math.toRadians(angle));
        w.drawLine(x0, y0, x1, y1);
        w.sleep(20);
        drawTree(w, x1, y1, angle + 25, 0.6 * length, level - 1);
        drawTree(w, x1, y1, angle - 25, 0.6 * length, level - 1);
    }
}
```



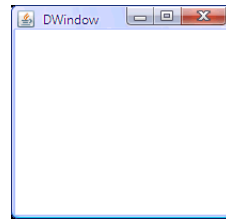
เมท็อดวาดลายสามเหลี่ยม



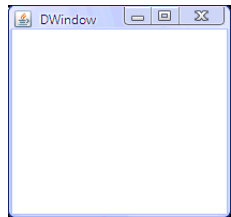
level = 0



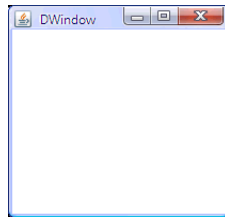
level = 1



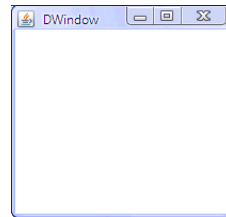
level = 2



level = 3



level = 4



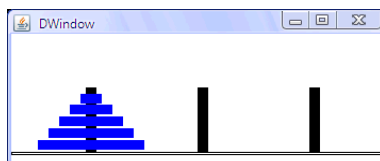
level = 5

เมท็อดวาดลายสามเหลี่ยม

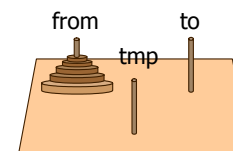
```
import jlab.graphics.DWindow;
public class Triangle {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200*Math.sin(Math.toRadians(60)));
        triangle(w, 20, w.getHeight() - 20, w.getWidth() - 40, 5);
    }
    private static void triangle(DWindow w, double x, double y,
                                double len, int level) {
        double dx = len * Math.cos(Math.toRadians(60));
        double dy = len * Math.sin(Math.toRadians(60));
        if (level == 0) {
            double x1 = x + dx, y1 = y - dy;
            w.drawLine(x, y, x + len, y);
            w.drawLine(x, y, x1, y1);
            w.drawLine(x1, y1, x + len, y);
            w.sleep(20);
        } else {
            double x1 = x + dx / 2, y1 = y - dy / 2;
            len = len / 2;
            triangle(w, x, y, len, level - 1);
            triangle(w, x1, y1, len, level - 1);
            triangle(w, x + len, y, len, level - 1);
        }
    }
}
```

ปัญหา Tower of Hanoi

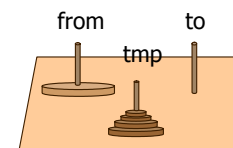
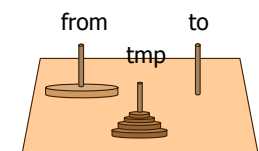
- มีเสา 3 ต้น
- ใส่จานกลมมีรูตรงกลางขนาดต่างกัน n ใบ ในเสาด้านซ้ายกองเรียงตามขนาด ใบใหญ่สุดอยู่ล่างสุด
- ต้องการ
 - ย้ายจานทุกใบจากเสาด้านซ้ายไปใส่ที่เสาด้านขวา
 - ย้ายได้ครั้งละใบ
 - อนุญาตให้ใบเล็กทับใบใหญ่ได้เท่านั้น
- ย้ายอย่างไร ?



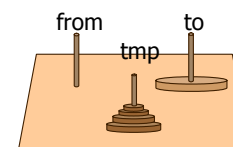
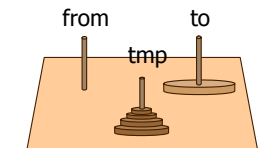
วิธีแก้ปัญหา Tower of Hanoi



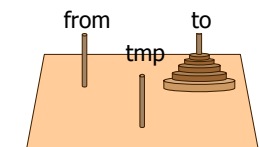
hanoi(n - 1, from, tmp, to)



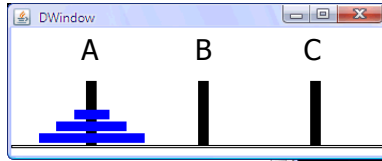
move n : from → to



hanoi(n - 1, tmp, to, from)



โปรแกรมแก้ปัญหา Tower of Hanoi



```
public class HanoiTower0 {
    public static void main(String[] args) {
        hanoi(3, "A", "C", "B");
    }
    private static void hanoi(int n, String from, String to, String tmp) {
        if (n == 0) return;
        hanoi(n - 1, from, tmp, to);
        System.out.println("move " + n + " : " + from + " -> " + to);
        hanoi(n - 1, tmp, to, from);
    }
}
```

```
move 1 : A -> C
move 2 : A -> B
move 1 : C -> B
move 3 : A -> C
move 1 : B -> A
move 2 : B -> C
move 1 : A -> C
```

นอกรื่อง : HanoiTower แบบ OO + Graphics

```
import jlab.graphics.*;
public class HanoiTower {
    public static void main(String[] args) {
        double w = 350, h = 120;
        DWindow win = new DWindow(w, h);
        Peg from = new Peg("A", w / 5, h / 2 - 10, 10, h / 2);
        Peg tmp = new Peg("B", w / 2, h / 2 - 10, 10, h / 2);
        Peg to = new Peg("C", 4 * w / 5, h / 2 - 10, 10, h / 2);
        win.add(from); win.add(tmp); win.add(to);
        win.add(new DRectangle(0, h-10, w, 2));
        int n = 5;
        for (int i = n; i >= 1; i--) {
            Disk d = new Disk(i, w/4/n);
            from.push(d);
            win.add(d.getXonPeg(from), d.getYonPeg(from), d);
        }
        hanoi(win, n, from, to, tmp);
    }
    private static void hanoi(DWindow w, int n, Peg from, Peg to, Peg tmp) {
        if (n == 0) return;
        hanoi(w, n - 1, from, tmp, to);
        from.peek().move(from, to);
        hanoi(w, n - 1, tmp, to, from);
    }
}
```

```
import jlab.graphics.*;
public class Peg extends DRectangle {
    private String name;
    private Disk[] stack = new Disk[16];
    private int numDisks = 0;
    public Peg(String name, double x, double y, double w, double h) {
        super(x, y, w, h);
        setColor(ColorUtil.BLACK);
        this.name = name;
    }
    public Disk pop() {
        Disk d = stack[--numDisks];
        return d;
    }
    public void push(Disk disk) {
        stack[numDisks++] = disk;
    }
    public Disk peek() {
        return stack[numDisks - 1];
    }
    public int size() {
        return numDisks;
    }
}
```

```
import jlab.graphics.*;
public class Disk extends DRectangle {
    private int id;
    public Disk(int id, double width) {
        super(id * width, 9);
        setColor(ColorUtil.BLUE);
        this.id = id;
    }
    public void move(Peg from, Peg to) {
        int x0 = getXonPeg(from);
        int y0 = getYonPeg(from);
        to.push(from.pop());
        int top = from.getY() - getHeight() - 2;
        moveDY(top - y0, 10);
        moveDX(getXonPeg(to) - x0, 10);
        moveDY(getYonPeg(to) - top, 10);
    }
    private void moveDY(int dy, int step) {
        int sign = dy / Math.abs(dy);
        dy = Math.abs(dy);
        while (dy != 0) {
            if (step > dy) step = dy;
            move(0, step*sign);
            dy = dy - step;
            Util.sleep(10);
        }
    }
}
```

```
...  
  
private void moveDX(int dx, int step) {  
    int sign = dx / Math.abs(dx);  
    dx = Math.abs(dx);  
    while (dx != 0) {  
        if (step > dx) step = dx;  
        move(step*sign, 0);  
        dx = dx - step;  
        Util.sleep(10);  
    }  
}  
  
public int getXonPeg(Peg p) {  
    return (int)(p.getX() + (p.getWidth() - getWidth()) / 2);  
}  
  
public int getYonPeg(Peg p) {  
    return (int)(p.getY() + p.getHeight() - p.size()*getHeight()*1.2);  
}  
}
```