

File

# อ่านไฟล์

```
File inFile = new File("input.txt");  
Scanner in = new Scanner(inFile);
```

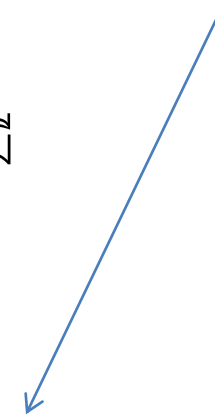
แสดงนเนอร์นี้จะอ่าน  
ไฟล์จาก **input.txt**

ต่อจากนี้ก็ใช้ **next,**  
**nextLine,**  
**nextInt** ของ  
**Scanner** อ่าน  
ไฟล์ได้ตามใจ

# เขียนไฟล์

```
PrintWriter out = new PrintWriter("output.txt");
```

ถ้ามีไฟล์อยู่แล้ว ไฟล์จะถูกล้างทิ้ง แต่ถ้าไม่มี  
ไฟล์ ไฟล์เปล่าๆจะถูกสร้างขึ้นให้เรา



ถ้ามี **File object** อยู่  
แล้ว ก็ใช้ **File object**  
เป็นพารามิเตอร์ก็ได้

`out.println(.....)` ก็จะถูกเรียกใช้ได้

เวลาจะเลิกเขียนอะไรลงไฟล์แล้ว อย่าลืม `out.close();`

# ตัวอย่าง โปรแกรม

อ่านไฟล์หนึ่ง แล้วไปเขียนลงอีกไฟล์ โดยเติมหมายเลขบรรทัดลงในด้าน  
หน้าข้อความทุกบรรทัด

I am Sam.

I am a man.

```
/* 1 */ I am Sam.
```

```
/* 2 */ I am a man.
```

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

```
public class LineNumberer {  
public static void main(String[] args) throws  
FileNotFoundException {
```

```
Scanner console = new Scanner(System.in);
```

```
System.out.print("Input File: ");
```

```
String inputFileName = console.next();
```

```
System.out.print("Output file: ");
```

```
String outputFileName = console.next();
```

```
File inputFile = new File(inputFileName);  
Scanner in = new Scanner(inputFile); // May not find the  
    specified file.
```

```
PrintWriter out = new PrintWriter(outputFileName);  
int lineNumber = 1;
```

```
while (in.hasNextLine()) {  
    String line = in.nextLine();  
    out.println("/ * " + lineNumber + " */ " + line);  
    lineNumber++;  
}
```

```
in.close();  
out.close();
```

จะมีประโยชน์ตอนใช้กับ  
in.nextDouble()

```
}
```

```
}
```

ซึ่งอ่านทีละบรรทัดเราจะสามารถเอาทั้งบรรทัดมาใช้ต่อได้  
เช่น

```
int i=0;  
While (!Character.isDigit(line.charAt(i))){  
    i++;  
}
```

เช่น ใช้กับไฟล์ที่บนหนึ่งบรรทัด มี ชื่อประเทศ ตามด้วยตัวเลขแสดงจำนวนประชากร

```
String countryName = line.substring(0,i);  
String population = substring(i);  
int populationValue = Integer.parseInt(population.trim());
```

# Reading one word at a time

```
while(in.hasNext()){  
    String input = in.next();  
    System.out.println(input);  
}
```

อ่านทีละคำ แต่คำๆหนึ่งคั่นด้วย เว้นวรรค  
tab หรือ new line เท่านั้น ดังนั้น

Snow.

22234

C++

ก็ถือเป็นหนึ่งคำเช่นกัน



ถ้าจะเอาส่วนที่ไม่ใช่คำทิ้งไปล่ะ

```
Scanner in = new Scanner(...);
```

```
In.useDelimiter("[^A-Za-z]+");
```

# นิดหน่อย กับ `nextDouble`

- มันไม่เอา **white space** ข้างหลังออกให้นะ
- สมมติเรียก `nextInt()` กับ `1729\nHarry`
  - จะเหลือ `\nHarry`

# Reading 1 character at a time

```
Scanner in = new Scanner(...);
```

```
In.useDelimiter("");
```

→ บังคับตรงนี้นั่นเอง

```
while (in.hasNext()){
```

```
    char ch = in.next().charAt(0);
```

```
    // คราวนี้ก็เล่นกับ ch ได้ตามใจ
```

```
}
```