

```

public interface Measurable {
    double getMeasure();
}

public class DataSet {
    private double sum;
    private Measurable max,min;
    private int count;

    //เพิ่มค่าเข้ามาในผลรวม รวมทั้งตรวจค่า max / min
    public void add(Measurable x){
        sum = sum + x.getMeasure();
        if (count ==0)
            {max = x; min = x;}
        else{
            if (x.getMeasure() > max.getMeasure())
                max = x;
            else if (x.getMeasure() < min.getMeasure())
                min = x;
        }
        count++;
    }

    //คืน Measurable ที่มีค่ามากที่สุด
    public Measurable getMax(){
        return max;
    }

    //คืนค่าเฉลี่ยของ Object
    public double getAverage() {
        return (sum/count);
    }

    //คืน Measurable ที่มีค่าน้อยสุด
    public Measurable getMin() {
        return min;
    }

    public double getSum() {
        return sum;
    }

    public void setSum(double sum) {
        this.sum = sum;
    }

    public int getCount() {
        return count;
    }

    public void setCount(int count) {
        this.count = count;
    }
}

```

```

public class Quiz implements Measurable {
    private String name;
    private double score;
    private String grade;

    //Constructor ของ Quiz สำหรับใส่ค่าเริ่มต้น อันนี้จะเขียนยังไงก็ได้ ขอให้มีความโจทย
    public Quiz(String n, double s, String g) {
        name = n;
        score = s;
        grade = g;
    }

    //คืนค่าที่วัดได้สำหรับ Quiz ในที่นี้ใช้ คะแนน
    public double getMeasure() {
        return score;
    }
}

public interface Measurer {
    public double measure(Object o);
}

public class BetterDataSet{

    private double sum;
    private Object max,min;
    private int count;
    private Measurer m;

    //รับ Measurer เข้ามา โดยต้องสอดคล้องกับObject ที่จะส่งเข้ามาใน add
    public BetterDataSet(Measurer m) {
        this.m = m;
    }

    //เพิ่มค่าเข้ามาในผลรวม รวมทั้งตรวจค่า max / min
    public void add(Object x) {
        sum = sum + m.measure(x);
        if (count ==0)
            {max = x; min = x;}
        else{
            if (m.measure(x) > m.measure(max))
                max = x;
            else if (m.measure(x) < m.measure(min))
                min = x;
        }
        count++;
    }

    //คืน Object ที่มีค่ามากที่สุด
    public Object getMax() {
        return max;
    }

    //คืนค่าของ Object ที่มีค่ามากที่สุดเป็น double อันนี้ พิมพ์ไว้เผื่อขี้ขี้
    public double getMaxVal() {return m.measure(max); }

    //คืนค่าเฉลี่ยของ Object
    public double getAverage() {

```

```

        return (sum/count);
    }

    //คืน Object ที่มีค่าน้อยสุด
    public Object getMin() {
        return min;
    }

    //คืนค่าของ Object ที่มีค่าน้อยสุดเป็น double อันนี้ พิมพ์ไว้เผื่อนะ
    public double getMinVal() {return m.measure(min);}
}

```

ส่วนด้านล่างนี้ ไม่ได้สั่งให้ทำนะ แต่เป็นตัวช่วยอำนวยความสะดวก

```

import java.awt.Rectangle;

public class RectMeasurer implements Measurer {

    //ค่าจากการวัด สำหรับ class Rectangle
    public double measure(Object o) {
        Rectangle rect = (Rectangle)o;
        return (rect.getWidth()*rect.getHeight())/2;
    }

}

import java.awt.Rectangle;

public class Main {

    public static void main(String[] args) {
        Quiz q1 = new Quiz("AAA",10.5,"F");
        Quiz q2 = new Quiz("BBB",55.5,"D+");
        Quiz q3 = new Quiz("CCC",98.7,"A");
        DataSet dset = new DataSet();
        dset.add((Measurable)q1);
        dset.add((Measurable)q2);
        dset.add((Measurable)q3);
        System.out.println(dset.getMax().getMeasure());
        System.out.println(dset.getMin().getMeasure());
        System.out.println(dset.getAverage());
        //=====
        RectMeasurer rctM = new RectMeasurer();
        BetterDataSet bset = new BetterDataSet(rctM);
        Rectangle r1 = new Rectangle();r1.width=5;r1.height=4;
        Rectangle r2 = new Rectangle();r2.width=7;r2.height=5;
        Rectangle r3 = new Rectangle();r3.width=10;r3.height=3;
        bset.add(r1);
        bset.add(r2);
        bset.add(r3);
        System.out.println(bset.getMaxVal());
        System.out.println(bset.getMinVal());
        System.out.println(bset.getAverage());
    }

}

```