

**SPECIAL
Answer****เฉลยตัวอย่างโจทย์ที่ยังไม่ได้เฉลยในห้องติว****Chapter 05 - การเขียนโปรแกรมแบบวนซ้ำด้วยคำสั่ง for และ ข้อมูลชนิดอาเรย์****โจทย์ข้อที่ 6**

```
int n = JLabIO.readInt("Enter n: ");
for(int line=1; line<=n; line++) {
    for(int col=1; col<=line; col++)
        System.out.print(" ");
    System.out.print("*");
    for(int col=line+1; col<=2*n-line; col++)
        System.out.print(" ");
    System.out.println("");
}
```

โจทย์ข้อที่ 9

```
public static int diffByK(int[] array, int k) {
    int count = 0;
    for (int i = 0; i < array.length; i++) {
        for (int j = i+1; j < array.length; j++) {
            if (Math.abs(array[i] - array[j]) == k) {
                count++;
            }
        } //end of for j
    } //end of for i
    return count;
}
```

Chapter 06 - เมทีอดและขอบเขตการมองเห็นของตัวแปร**โจทย์ข้อที่ 8**

```
public class CompoundAnnual {
    public static void main(String[] args) {
        double money = 10000;
        double rate = 0.03;
        int year = 10;
        System.out.println(compoundAnnual(money, rate, year));
        System.out.println(compoundAnnual(money, year, rate));
    }
}
```

```
public static double compoundAnnual(double money, double rate,int year) {
    return money * Math.pow(1 + rate, year);
}

public static double compoundAnnual(double money, int year, double rate) {
    if (year == 1) return money + money * rate;
    return compoundAnnual(money, year - 1, rate) * (1 + rate);
}
```

```
}//End of class
```

Chapter 08 - การประกอบและการสืบทอดคลาส**โจทย์ข้อที่ 6**

```
public class Point3D extends Point{
    int z;
    public Point3D(int x, int y, int z) {
        super(x,y);
        this.z=z;
    }
    public String toString(){
        return ("+x+", "+y+", "+z+");
    }
    public boolean equals(Point3D p){
        if( p.x==this.x && p.y==this.y&&p.z==this.z ) return true;
        return false;
    }
    double distanceTo(Point3D p){
        int dX = Math.abs(this.x - p.x);
        int dY = Math.abs(this.y - p.y);
        int dZ = Math.abs(this.z - p.z);
        return Math.sqrt(Math.pow(dX,2)+Math.pow(dY,2)+Math.pow(dZ,2));
    }
}
```

โจทย์ข้อที่ 7

```
public class Rectangle {
    Point pLT; // point for Left and Top
    Point pRD; // point for Right and Down
    public Rectangle(Point pLT, Point pRD) {
        // Check Rectangle
        if( pLT.x == pRD.x || pLT.y == pRD.y ){
            System.out.println("Not A Rectangle");
        }
        else{
            this.pLT = pLT; this.pRD = pRD;
        }
    }
    double area(){
        return Math.abs(pLT.x-pRD.x)*Math.abs(pLT.y-pRD.y);
    }
    public boolean equals(Object obj){
        Rectangle rec = (Rectangle) obj;
        if( Math.abs(this.pLT.x-this.pRD.x) ==
            Math.abs(rec.pLT.x-rec.pRD.x) &&
            Math.abs(this.pLT.y-this.pRD.y) ==
            Math.abs(rec.pLT.y-rec.pRD.y) )
            return true;
        return false;
    }
    boolean contrains(Point p){
        int iL,iR,iT,iD;
        if( pLT.x > pRD.x ){
            iR = pLT.x;
            iL = pRD.x;
        }else{
            iR = pRD.x;
            iL = pLT.x;
        }
    }
}
```

```

    if( pLT.y > pRD.y ){
        iT = pLT.y;
        iD = pRD.y;
    }else{
        iT = pRD.y;
        iD = pLT.y;
    }
    if( (p.x>=iL)&&(p.x<=iR)&&(p.y>=iD)&&(p.y<=iT) )
        return true;
    return false;
}
}
}

```

Chapter 09 - การโปรแกรมแบบเวียนเกิด การเรียงลำดับ และการค้นข้อมูล

โจทย์ข้อที่ 2

```

public class RecursiveExample1 {
    static int aVar = 0;
    int bVar;

    RecursiveExample1() {
        bVar = 10;
    }

    public void aMethod() {
        if (bVar % 3 == 0) bVar--;
        int b = bVar--;
        if (bVar > 0) {
            aMethod();
        }
        aVar += b;
    }

    public static void main(String[] args) {
        RecursiveExample1 ex1 = new RecursiveExample1();
        System.out.println(ex1.aVar);
        System.out.println(ex1.bVar);
        ex1.aMethod();
        System.out.println(ex1.aVar);
        System.out.println(ex1.bVar);
    }
}

```

ผลลัพธ์

```

0
10
37
0

```

ค่าเริ่มต้น

```

aVar = 0
bVar = 10
b = No Define

```

รอบที่	ค่า aVar	ค่า bVar	ค่า b	ค่า aVar (เมื่อมีการคืนค่า)
1	0	9	10	aVar += 10 → 37 (7)
2	0	7	8	aVar += 8 → 27 (6)
3	0	6	7	aVar += 7 → 19 (5)
4	0	4	5	aVar += 5 → 12 (4)
5	0	3	4	aVar += 4 → 7 (3)
6	0	1	2	aVar += 2 → 3 (2)
7	0	0	1	aVar += 1 → 1 (1)

โจทย์ข้อที่ 7

pass = 1

23	2	3	14	5	99	9
2	23	3	14	5	99	9
2	3	23	14	5	99	9
2	3	14	23	5	99	9
2	3	14	5	23	99	9
2	3	14	5	23	99	9
2	3	14	5	23	9	99

pass = 2

2	3	14	5	23	9	99
2	3	14	5	23	9	99
2	3	14	5	23	9	99
2	3	5	14	23	9	99
2	3	5	14	23	9	99
2	3	5	14	9	23	99

โจทย์ข้อที่ 8

```
public static void selectionSort(int[] data) {
    selectionSort(data, data.length);
}

private static void selectionSort(int[] data, int n) {
    int i, t;
    if (n < 2) return;
    i = maxI(data, n);
    t = data[i];
    data[i] = data[n-1];
    data[n-1] = t;
    selectionSort(data, n-1);
}
```