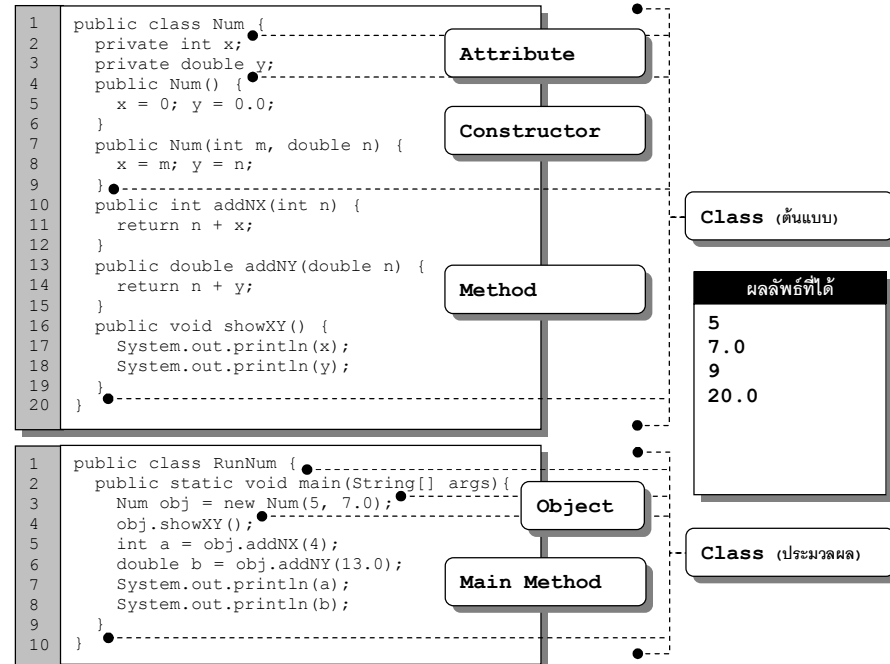


**CHAPTER**  
**ANS-12**
**คลาสและอ็อบเจกต์**  
**(Classes and Objects)**
**โจทย์ข้อที่ 1 [ระดับง่าย]**

**โจทย์ข้อที่ 2 [ระดับง่าย]**

- |  |   |   |
|--|---|---|
| 1) <input checked="" type="checkbox"/> | 6) <input checked="" type="checkbox"/>  | 11) <input checked="" type="checkbox"/> |
| 2) <input checked="" type="checkbox"/> | 7) <input checked="" type="checkbox"/>  | 12) <input checked="" type="checkbox"/> |
| 3) <input checked="" type="checkbox"/> | 8) <input checked="" type="checkbox"/>  | 13) <input checked="" type="checkbox"/> |
| 4) <input checked="" type="checkbox"/> | 9) <input checked="" type="checkbox"/>  | 14) <input checked="" type="checkbox"/> |
| 5) <input checked="" type="checkbox"/> | 10) <input checked="" type="checkbox"/> | 15) <input checked="" type="checkbox"/> |

**โจทย์ข้อที่ 3 [ระดับง่าย]**

```
public class Course {
```

```
//แอตทริบิวต์ 3 ตัว
```

```
private int id;
public String title;
protected double credit;
```

```
//ตัวสร้างแบบไม่มีพารามิเตอร์ใดๆ
```

```
public Course() {
    id = 0;
    title = "";
    credit = 0.0;
}
```

```
//ตัวสร้างแบบมีพารามิเตอร์ 3 ตัว
```

```
public Course(int i, String t, double c) {
    id = i;
    title = t;
    credit = c;
}
```

```
//เมธอด setID(...)
```

```
public void setID(int i) {
    id = i;
}
```

```
//เมธอด setTitle(...)
```

```
public void setTitle(String t) {
    title = t;
}
```

```
//เมธอด setCredit(...)
```

```
public void setCredit(double c) {
    credit = c;
}
```

```
//เมธอด getLevel(...)
```

```
public String getLevel() {
    if (id % 1000 / 100 <= 4) {
        return "Undergraduate";
    } else {
        return "Graduate";
    }
}
```

```
//เมื่อกด getFaculty(...)
```

```
public int getFaculty() {
    return id / 100000;
}
```

```
//เมื่อกด getDepartment(...)
```

```
public int getDepartment() {
    return id / 1000 % 100;
}
```

```
//เมื่อกด toString(...)
```

```
public String toString() {
    return id + " " + title + " (" + credit + ")";
}
```

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

โจทย์ข้อที่ 4 [ระดับง่าย]

```
9
5
7
1
1
12
11
1
12
```

- 1) แอดทริบิวต์ y หรือตัวแปร y
- 2) ตัวแปร y (ในบรรทัดที่ 13)

โจทย์ข้อที่ 5 [ระดับง่าย]

```
0,
101,Taksin
103,Apisit
Taksin:U
Apisit:S
```

โจทย์ข้อที่ 6 [ระดับง่าย]

//ประกาศหัตถศาสตร์ TestNumber

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

```
public class TestNumber {
```

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

```
        //อ็อบเจ็คชื่อ no1
```

```
        Number no1 = new Number();
```

```
        //อ็อบเจ็คชื่อ no2
```

```
        System.out.println("Enter x : ");
        double x = kb.nextDouble();
        System.out.println("Enter y : ");
        double y = kb.nextDouble();
        Number no2 = new Number(x, y);
```

```
        //แสดงผลการบวก การลบ การคูณ การหาร และการมีอด
```

```
        System.out.println("Add: " + no2.add());
        System.out.println("Sub: " + no2.sub());
        System.out.println("Mul: " + no2.mul());
        System.out.println("Div: " + no2.div());
        System.out.println("Mod: " + no2.mod());
```

```
    } //End of main
```

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

โจทย์ข้อที่ 7 [ระดับง่าย]

```
1) public double data;
```

```
2) private static int var = 10;
```

```
3) protected static boolean check;
```

```
4) private String stdName[] = new String[351];
```

```
5) public static int m[][] = new int[8][5];
```

```
6) private void show(String stdName, double grade)
```

```
7) protected static int search(int num[], int x)
```

```
8) public double[][] mulMatrix(double a[][], double b[][])
```

9) `public static int[] union(int x[], int y[])`

โจทย์ข้อที่ 8 [ระดับง่าย]

ข้อ	คำตอบ
1.	B
2.	D
3.	B
4.	C
5.	B

ข้อ	คำตอบ
6.	A
7.	D
8.	D
9.	A
10.	D

ข้อ	คำตอบ
11.	C
12.	B
13.	D
14.	A
15.	B

ข้อ	คำตอบ
16.	D
17.	C
18.	D
19.	B
20.	C

โจทย์ข้อที่ 9 [ระดับปานกลาง]

```
0,0
6,10
Good By A
See You F
0,0
5,7
5,7
9,7
9,7
I Love Java
```

โจทย์ข้อที่ 10 [ระดับปานกลาง]

```
public class Dice {
    //แอตทริบิวต์ประจำอ็อบเจ็คคือ face และ value
```

```
    public int face;
    public int value;
```

```
//ตัวสร้างแบบไม่มีพารามิเตอร์
```

```
    public Dice() {
        face = 2;
        roll();
    }
```

```
//ตัวสร้างแบบมีพารามิเตอร์หนึ่งตัว
```

```
    public Dice(int f) {
        face = f;
        roll();
    }
```

```
//ตัวสร้างแบบมีพารามิเตอร์สองตัว
```

```
    public Dice(int f, int v) {
        face = f;
        value = v;
    }
```

```
//เมทอดประจำอ็อบเจ็คคือ roll
```

```
    public void roll() {
        value = (int)(Math.random() * face) + 1;
    }
```

```
//เมทอดประจำอ็อบเจ็คคือ setValue
```

```
    public void setValue(int v) {
        value = v;
    }
```

```
//เมทอดประจำอ็อบเจ็คคือ getFace
```

```
    public int getFace() {
        return face;
    }
```

```
//เมทอดประจำอ็อบเจ็คคือ getValue
```

```
    public int getValue() {
        return value;
    }
```

```
//เมทอดประจำอ็อบเจ็คคือ show
```

```
    public void show() {
        System.out.println("Face: " + face);
        System.out.println("Value: " + value);
    }
```

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

```
public class TestDice {
    /* สร้างลูกเต๋า 3 ลูก ในรูปของอ็อบเจ็คที่ชื่อ d1, d2 และ d3 */
    public static void main(String[] args) {
        Dice d1 = new Dice();
        Dice d2 = new Dice(13);
        Dice d3 = new Dice(30, 9);
        d1.show();
        d2.show();
        d3.show();
        for (int i = 1; i <= 20; i++) {
            d1.roll();
            System.out.println(d1.getValue());
            d2.roll();
            System.out.println(d2.getValue());
            d3.roll();
            System.out.println(d3.getValue());
            System.out.println("-----");
        }
    }
} //End of class
```

โจทย์ข้อที่ 11 [ระดับปานกลาง]

```
public class RealNumber {
    //แอดทริบิวต์ประจำอ็อบเจ็คแบบ public ชื่อ num
```

```
    public double num;
```

//ตัวสร้างแบบไม่มีพารามิเตอร์

```
    public RealNumber() {
        num = 0.0;
    }
```

//ตัวสร้างแบบมีพารามิเตอร์

```
    public RealNumber(double n) {
        num = n;
    }
```

//เมทอดประจำอ็อบเจ็คชื่อ plus

```
    public double plus(double n) {
        return num + n;
    }
```

//เมทอดประจำอ็อบเจ็คชื่อ diff

```
    public double diff(double n) {
        double x = num - n;
        if (x >= 0) return x;
        else return -x;
    }
```

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

```
public class TestRealNumber {
    /* สร้างจำนวนจริง 1 จำนวนในรูปของอ็อบเจ็ค r */
    public static void main(String[] args) {
        RealNumber r = new RealNumber();
        System.out.println(r.plus(15));
        System.out.println(r.diff(50));
    }
} //End of class
```

โจทย์ข้อที่ 12 [ระดับปานกลาง]

```
public class Account {
    private double balance;
    public Account() {
        balance = 0.0;
    }
    public Account(double money) {
        balance = money;
    }
    public void deposit(double money) {
        balance += money;
    }
    public double withdraw(double money) {
        if (balance >= money) {
            balance -= money;
            return money;
        } else {
            return 0.0;
        }
    }
    public double getbalance() {
        return balance;
    }
} //End of Class
```

สร้างคลาส TestAccount

```
public class TestAccount {
    public static void main(String[] args) {
        Account acc = new Account(2000);
        System.out.println("New Account");
        System.out.println("Balance : " + acc.getbalance());
        double money = acc.withdraw(1500);
        System.out.println("Withdraw : " + money);
        System.out.println("Balance : " + acc.getbalance());
        acc.deposit(800);
        System.out.println("Deposit : " + 800);
        System.out.println("Balance : " + acc.getbalance());
    }
}
```

## โจทย์ข้อที่ 13 [ระดับปานกลาง]

```
public class PiggyBank {
```

(1) แอตทริบิวต์ one, two, five และ ten

(2) แอตทริบิวต์ size

```
public int one, two, five, ten;
public int size;
```

(3) ตัวสร้าง 2 แบบ

```
PiggyBank() {
    clear();
    size = 100;
}
PiggyBank(int s) {
    clear();
    size = s;
}
```

(4) เมทอดประจำอ็อบเจ็คชื่อ clear(...)

```
public void clear() {
    one = two = five = ten = 0;
}
```

(5) เมทอดประจำอ็อบเจ็คชื่อ getTotal(...)

```
public int getTotal() {
    return one + (2 * two) + (5 * five) + (10 * ten);
}
```

(6) เมทอดประจำอ็อบเจ็คชื่อ full(...)

```
public boolean full() {
    return getTotal() >= size;
}
```

```
public void addOne(int c) {
    if (getTotal() + c <= size) one = one + c;
    else System.out.println("Piggy Bank Full");
}
```

(8) เมทอดประจำอ็อบเจ็คชื่อ addTwo(...)

```
public void addTwo(int c) {
    if (getTotal() + 2 * c <= size) two = two + c;
    else System.out.println("Piggy Bank Full");
}
```

(9) เมทอดประจำอ็อบเจ็คชื่อ addFive(...)

```
public void addFive(int c) {
    if (getTotal() + 5 * c <= size) five = five + c;
    else System.out.println("Piggy Bank Full");
}
```

(10) เมทอดประจำอ็อบเจ็คชื่อ addTen(...)

```
public void addTen(int c) {
    if (getTotal() + 10 * c <= size) ten = ten + c;
    else System.out.println("Piggy Bank Full");
}
```

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

สร้างคลาส TestPiggyBank

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

```
PiggyBank pb = new PiggyBank(500);
System.out.println("Money Total : " + pb.getTotal());
pb.addTwo(34);
System.out.println("Add 2 Baht Money : " + 34);
System.out.println("Money Total : " + pb.getTotal());
pb.addTen(13);
System.out.println("Add 10 Baht Money : " + 13);
System.out.println("Money Total : " + pb.getTotal());
```

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

**CHAPTER**  
**ANS-13**
**คลาสและอ็อบเจ็คแบบซับซ้อน**  
**(Advanced Classes and Objects)**

โจทย์ข้อที่ 1 [ระดับปานกลาง]

เรียกตัวสร้างใด (ระบุหมายเลข)	ผลลัพธ์ที่แสดง ขึ้นบนจอภาพคือ
1	0,0
2	13,2
3	13,2
2 3	7,11
2	11,13
1 3	0,0

```
public class Num {
    ...
    public boolean isEqual(int x, int y) {
        return this.x == x && this.y == y;
    }
    public boolean isEqual(Num n) {
        return x == n.x && y == n.y;
    }
} //End of class
```

โจทย์ข้อที่ 2 [ระดับยาก]

```
public class Coin {
    //แอตทริบิวต์ side และ value
    public int value;
    public String side;
```

//ตัวสร้างแบบไม่มีพารามิเตอร์

```
public Coin() {
    side = "H";
    value = 1;
}
```

//ตัวสร้างแบบมีพารามิเตอร์สองตัว

```
public Coin(String s, int v) {
    side = s;
    value = v;
}
```

//ตัวสร้างแบบมีพารามิเตอร์หนึ่งตัว

```
public Coin(Coin c) {
    side = c.side;
    value = c.value;
}
```

//เมธอด flip(...)

```
public void flip() {
    side = list[(int)(Math.random() * 2)];
}
```

//เมธอด reverse(...)

```
public void reverse() {
    if (side.equals("H")) {
        side = "T";
    } else {
        side = "H";
    }
}
```

//เมธอด equals(...)

```
public boolean equals(Coin c) {
    return side == c.side && value == c.value;
}
```

} //End of class

```

/* คลาส De oCoin */
public class De oCoin {
    public static void main(String args) {
        Coin c1 = new Coin();
        Coin c2 = new Coin(" ", 1);
        Coin c3 = new Coin("T", 5);
        Coin c4 = new Coin(c3);
        for (int i = 1; i <= 20; i++) {
            c1.flip();
            c2.reverse();
            c3.flip();
            c4.reverse();
            System.out.println(c1.equals(c2));
            System.out.println(c3.equals(c4));
        }
    }
} // end of main
} // end of class

```

## โจทย์ข้อที่ 3 [ระดับยาก]

```

public class SetOfInteger {
    //แอตทริบิวต์ iSet
    public int iSet;

    //ตัวสร้างแบบที่รับพารามิเตอร์ 1 ตัวที่เป็นอาเรย์
    public SetOfInteger(int arr) {
        arr = sort(arr);
        iSet = new ArrayList<>();
        for (int i = 0; i < arr.length; i++) {
            iSet.add(arr[i]);
        }
    } // end of constructor

    //ตัวสร้างแบบที่รับพารามิเตอร์ 1 ตัวที่เป็นชนิด SetOfInteger
    public SetOfInteger(SetOfInteger s) {
        iSet = new ArrayList<>();
        for (int i = 0; i < s.iSet.size(); i++) {
            iSet.add(s.iSet.get(i));
        }
    } // end of constructor

    //เมธอด sort(...)
    public int[] sort(int arr) {
        ...
    } // end of method

```

```

//เมธอด removeDuplicatedMembers(...)
public int removeDuplicatedMembers(int arr) {
    ...
} // end of method

//เมธอด equals(...)
public boolean equals(SetOfInteger s) {
    if (iSet.size() == s.iSet.size()) {
        for (int i = 0; i < iSet.size(); i++) {
            if (iSet.get(i) != s.iSet.get(i)) return false;
        }
        return true;
    } else {
        return false;
    }
} // end of method

//เมธอด toString(...)
public String toString() {
    String s = "";
    for (int i = 0; i < iSet.size(); i++) {
        s = s + iSet.get(i) + " ";
    }
    return s;
} // end of method

//เมธอด addElement(...)
public void addElement(int a) {
    int arr = new ArrayList<>();
    for (int i = 0; i < iSet.size(); i++) {
        arr.add(iSet.get(i));
    }
    arr.add(a);
    arr = sort(arr);
    iSet = new ArrayList<>();
    for (int i = 0; i < arr.size(); i++) {
        iSet.add(arr.get(i));
    }
} // end of method

//เมธอด findElement(...)
public int findElement(int a) {
    for (int i = 0; i < iSet.size(); i++) {
        if (iSet.get(i) == a) return i;
    }
    return -1;
} // end of method

```

```
//เมื่อกัด isSubset(...)
public boolean isSubset(SetOfInteger s) {
    int count = 0;
    for (int i = 0; i < s.iSet.length; i++) {
        for (int j = 0; j < iSet.length; j++) {
            if (s.iSet[i] == iSet[j]) {
                count++;
                break;
            }
        }
    }
    return s.iSet.length == count;
} //End of method

//เมื่อกัด union(...)
public SetOfInteger union(SetOfInteger s) {
    int x[] = new int[s.iSet.length + iSet.length];
    for (int i = 0; i < x.length; i++) {
        if (i < iSet.length) x[i] = iSet[i];
        else x[i] = s.iSet[i - iSet.length];
    }
    SetOfInteger un = new SetOfInteger(x);
    return un;
} //End of method

//เมื่อกัด intersect(...)
public SetOfInteger intersect(SetOfInteger s) {
    int x[] = new int[s.iSet.length];
    int len = 0;
    for (int i = 0; i < s.iSet.length; i++) {
        for (int j = 0; j < this.iSet.length; j++) {
            if (s.iSet[i] == this.iSet[j]) {
                x[len] = s.iSet[i];
                len++;
                break;
            }
        }
    }
    int n[] = new int[len];
    for (int i = 0; i < n.length; i++)
        n[i] = x[i];
    SetOfInteger inter = new SetOfInteger(n);
    return inter;
} //End of method
} //End of class
```

โจทย์ข้อที่ 4 [ระดับยาก]

```
public class Student {
    //แอตทริบิวต์ประจำอ็อบเจ็ค 3 ตัว
    private int id;
    private String name;
    private double score[];

    //ตัวสร้างแบบไม่กำหนดพารามิเตอร์
    Student() {
        id = 0;
        name = "";
        score = new double[0];
    }

    //ตัวสร้างแบบ 3 พารามิเตอร์
    Student(int i, String n, int x) {
        id = i;
        name = n;
        score = new double[x];
    }

    //ตัวสร้างแบบ 1 พารามิเตอร์
    Student(Student s) {
        id = s.getID();
        name = s.getName();
        score = s.getScore();
    }

    //เมื่อกัดประจำอ็อบเจ็ค 4 เมื่อกัด getID (...), getName (...), getScore (...) และ setScore (...)
    public int getID() {
        return id;
    }
    public String getName() {
        return name;
    }
    public double[] getScore() {
        return score;
    }
    public void setScore(double scr[]) {
        for (int i = 0; i < scr.length; i++) {
            score[i] = scr[i];
        }
    }
} //End of class
```

คลาส Freshmen

```
import java.util.Scanner;
public class Freshmen {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
```

//กำหนดให้มีนิสิต 800 คน

```
Student s[] = new Student[800];
```

//กำหนดค่าเริ่มต้นให้กับเลขประจำตัวของนิสิต ชื่อ นิสิต จำนวนวิชา

```
for (int i = 0; i < s.length; i++) {
    int id = (53001 + i) * 100 + 21;
    System.out.print("Enter name : ");
    String name = kb.nextLine();
    s[i] = new Student(id, name, 4);
}
```

//เรียกใช้งานเมทอด setScore(...)

```
for (int i = 0; i < s.length; i++) {
    double scr[] = new double[4];
    for (int j = 0; j < scr.length; j++)
        scr[j] = Math.random() * 100;
    s[i].setScore(scr);
}
```

//แสดงเลขประจำตัว นิสิต ชื่อ นิสิต และคะแนนทั้ง 4 วิชาของนิสิตแต่ละคน

```
for (int i = 0; i < s.length; i++) {
    System.out.print(s[i].getID() + "\t" + s[i].getName() + "\t");
    for (int j = 0; j < s[i].getScore().length; j++)
        System.out.print(s[i].getScore()[j] + "\t");
    System.out.println();
}
```

//คำนวณหาคะแนนเฉลี่ยของแต่ละรายวิชา

```
for (int j = 0; j < s[0].getScore().length; j++) {
    double sum = 0.0;
    for (int i = 0; i < s.length; i++) {
        sum += s[i].getScore()[j];
    }
    System.out.println(sum / s.length);
}
```

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

โจทย์ข้อที่ 5 [ระดับยาก]

1. คลาส Owner

```
public class Owner {
    //แอตทริบิวต์ id และแอตทริบิวต์ name
```

```
private int id;
private String name;
```

//ตัวสร้างแบบไม่มีพารามิเตอร์

```
public Owner() {
    id = 0;
    name = "";
}
```

//ตัวสร้างแบบที่มีพารามิเตอร์ 2 ตัว

```
public Owner(int i, String n)
    id = i;
    name = n;
}
```

//ตัวสร้างแบบที่มีพารามิเตอร์ 1

```
public Owner(Owner own) {
    id = own.getID();
    name = own.getName();
}
```

//เมทอด getID(...)

```
public int getID() {
    return id;
}
```

//เมทอด getName(...)

```
public String getName() {
    return name;
}
```

//เมทอด setID(...)

```
public void setID(int i) {
    id = i;
}
```

//เมทอด setName(...)

```
public void setName(String n) {
    name = n;
}
```

```
//เมื่อกด show(...)
public void show() {
    System.out.println("ID : " + id);
    System.out.println("Name : " + name);
}

} //End of class

2. คลาส Land
public class Land {
//แอตทริบิวต์ n และแอตทริบิวต์ price
public Owner n;
public double price;

//ตัวสร้างแบบไม่มีพารามิเตอร์
public Land() {
    n = new Owner();
    price = 0.0;
}

//ตัวสร้างแบบมีพารามิเตอร์ 2 ตัว
public Land(Owner w, double p) {
    n = new Owner(w);
    price = p;
}

//เมื่อกด setOwner(...)
public void setOwner(Owner w) {
    n.setID(w.getID());
    n.setName(w.getName());
}

//เมื่อกด setPrice(...)
public void setPrice(double p) {
    price = p;
}

//เมื่อกด show(...)
public void show() {
    n.show();
    System.out.println("Price : " + price);
}

} //End of class
```

```
3. คลาส City
import java.util.Scanner;
public class City {
    public static void main(String[] args) {
        //อาเรียของอ็อบเจ็ค Land
        Land a[][] = new Land[50][60];

        //กำหนดค่าเริ่มต้นให้กับที่ดินทุกแปลง
        for (int i = 0; i < a.length; i++) {
            for (int j = 0; j < a[i].length; j++) {
                Owner n = new Owner();
                a[i][j] = new Land(n, (int)(Math.random() * 10000000) + 1);
            }
        }

        //อ็อบเจ็คชื่อ ow1
        Owner ow1 = new Owner(1001, "Tukie");
        ow1.show();

        //อ็อบเจ็คชื่อ ow2
        Owner ow2 = new Owner(1002, "Annie");
        ow2.show();

        //ให้ ow1 ครอบครองที่ดินในแปลงตำแหน่งที่ (4, 9)
        a[4][9].setOwner(ow1);
        a[4][9].show();

        //เปลี่ยนราคาที่ดินที่ ow1 ครอบครองอยู่
        a[4][9].setPrice(30000000.0);
        a[4][9].show();

        //เปลี่ยนให้ ow2 เข้าไปครอบครองที่ดินแปลงที่ ow1 ครอบครองอยู่
        a[4][9].setOwner(ow2);
        a[4][9].show();

        //กำหนดให้ ow1 เข้าครอบครองที่ดินแปลงใหม่ 50 แปลง
        for (int i = 0; i < a.length; i++) {
            a[i][0].setOwner(ow1);
            a[i][0].setPrice(50000000.0);
            a[i][0].show();
        }

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

## โจทย์ข้อที่ 6 [ระดับยาก]

คลาส SScanner

```
import java.util.Scanner;
import java.io.InputStream;
public class SScanner {
```

//แอตทริบิวต์ kb

```
public Scanner kb;
```

//ตัวสร้าง

```
public SScanner(InputStream in) {
    kb = new Scanner(in);
}
```

//เมทอด nextInt(...)

```
public int nextInt() {
    return kb.nextInt();
}
```

//เมทอด nextLine(...)

```
public String nextLine() {
    String s = "";
    while (true) {
        s = kb.nextLine();
        if (s.length() != 0) break;
    }
    return s;
}
```

} //End of class

คลาส TestSScanner

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

```
        SScanner kb = new SScanner(System.in);
        System.out.print("Enter Int: ");
        int a = kb.nextInt();
        System.out.print("Enter Str: ");
        String b = kb.nextLine();
        System.out.println(a + ", " + b);
```

} //End of main

} //End of class

## โจทย์ข้อที่ 7 [ระดับเทพ]

1. คลาส Fruit

```
public class Fruit {
```

//แอตทริบิวต์ประจำอ็อบเจ็คชื่อ name และ weigh

```
public String name;
public double weigh;
```

//ตัวสร้าง 4 แบบ ดังต่อไปนี้

```
public Fruit() {
    name = "";
    weigh = 0.0;
}

public Fruit(String n) {
    name = n;
    weigh = wh[(int)(Math.random() * 5)];
}

public Fruit(String n, double w) {
    name = n;
    weigh = w;
}

public Fruit(Fruit f) {
    name = f.name;
    weigh = f.weigh;
}
```

//เมทอดประจำอ็อบเจ็คชื่อ showFruit

```
public void showFruit() {
    System.out.println(name + " " + weigh + " kg.");
}
```

} //End of class

2. คลาส Basket

```
public class Basket {
```

//แอตทริบิวต์ประจำอ็อบเจ็คชื่อ fruit และ count

```
public Fruit fruit[];
public int count;
```

//ตัวสร้างแบบไม่มีพารามิเตอร์

```
public Basket() {
    fruit = new Fruit[10];
    count = 0;
}
```

//ตัวสร้างแบบหนึ่งพารามิเตอร์ชนิดจำนวนเต็ม

```
public Basket(int n) {
    fruit = new Fruit[n];
    count = 0;
}
```

//ตัวสร้างแบบหนึ่งพารามิเตอร์ชนิด Basket

```
public Basket(Basket b) {
    fruit = new Fruit[b.fruit.length];
    for (int i = 0; i < fruit.length; i++) {
        fruit[i] = b.fruit[i];
    }
    count = b.count;
}
```

//เมื่อกดประจำอับเจ็คชื่อ isFull

```
public boolean isFull() {
    return count == fruit.length;
}
```

//เมื่อกดประจำอับเจ็คชื่อ isEmpty

```
public boolean isEmpty() {
    return count == 0;
}
```

//เมื่อกดประจำอับเจ็คชื่อ takeIn

```
public void takeIn(Fruit f) {
    if (!isFull()) {
        fruit[count] = f;
        count++;
    } else {
        System.out.println("Basket Full");
    }
}
```

//เมื่อกดประจำอับเจ็คชื่อ takeOut

```
public Fruit takeOut() {
    if (!isEmpty()) {
        count--;
        Fruit f = fruit[count];
        fruit[count] = null;
        return f;
    } else {
        System.out.println("Basket Empty");
        return null;
    }
}
```

//เมื่อกดประจำอับเจ็คชื่อ getTotalWeigh

```
public double getTotalWeigh() {
    double weigh = 0.0;
    for (int i = 0; i < count; i++) {
        weigh += fruit[i].weigh;
    }
    return weigh;
}
```

//เมื่อกดประจำอับเจ็คชื่อ showBasket

```
public void showBasket() {
    for (int i = 0; i < count; i++) {
        fruit[i].showFruit();
    }
}
```

} //End of class

3. คลาส FruitBasket

```
import java.util.Scanner;
public class FruitBasket {
```

//เมื่อกดประจำคลาสชื่อ getFruitBasket

```
public static Basket getFruitBasket(String f, double w, int n) {
    Basket bsk = new Basket(n);
    Fruit frt[] = new Fruit[n];
    for (int i = 0; i < frt.length; i++) {
        if (w <= 0.0) frt[i] = new Fruit(f);
        else frt[i] = new Fruit(f, w);
        bsk.takeIn(frt[i]);
    }
    return bsk;
}
```

//เมื่อกดประจำคลาสชื่อ showFruitBasket

```
public static void showFruitBasket(Basket b) {
    System.out.println("Total Weigh: " + b.getTotalWeigh());
    System.out.println("Total Fruit: " + b.count);
    System.out.println("List of Fruit: ");
    b.showBasket();
}
```

//เมื่อกด main

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

//สร้างกระเช้าผลไม้ขึ้นมา 3 กระเช้า

```
Basket x[] = new Basket[3];
x[0] = getFruitBasket("Apple", 0.12, 25);
x[1] = getFruitBasket("Orange", 0.1, 30);
x[2] = getFruitBasket("Melon", 0.45, 15);
```

//สร้างกระเช้าผลไม้เปล่าๆ ขึ้นมาอีก 6 กระเช้า

```
Basket y[] = new Basket[6];
for(int i = 0; i < y.length; i++) {
    y[i] = new Basket();
    int j = 0;
    while (!y[i].isFull()) {
        if (!x[j].isEmpty()) y[i].takeIn(x[j].takeOut());
        if (y[i].getTotalWeigh() > 2.0) {
            Fruit tmp = y[i].takeOut();
            break;
        }
        j++;
        if (j > 2) j = 0;
    }
    System.out.println("-----Basket " + (i + 1) + "-----");
    showFruitBasket(y[i]);
}
```

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

โจทย์ข้อที่ 8 [ระดับเทพ]

คลาส Student

```
public class Student {
```

//แอตทริบิวต์ id แอตทริบิวต์ name และแอตทริบิวต์ noob

```
private int id;
private String name;
private int noob;
```

//ตัวสร้างแบบไม่มีพารามิเตอร์

```
Student() {
    id = 0;
    name = "";
    noob = 0;
}
```

//ตัวสร้างแบบที่รับพารามิเตอร์ 3 ตัว

```
Student(int i, String n, int x) {
    id = i;
    name = n;
    noob = x;
}
```

//ตัวสร้างแบบที่รับพารามิเตอร์เป็นชนิด Student

```
Student(Student s) {
    set(s);
}
```

//เม็ท็อด getID(...)

```
public int getID() {
    return id;
}
```

//เม็ท็อด getName(...)

```
public String getName() {
    return name;
}
```

//เม็ท็อด getNoob(...)

```
public int getNoob() {
    return noob;
}
```

//เม็ท็อด set(...)

```
public void set(Student s) {
    id = s.getID();
    name = s.getName();
    noob = s.getNoob();
}
```

//เม็ท็อด show(...)

```
public void show() {
    String s = "";
    if (noob == 1) s = "1-Noob";
    else if (noob == 2) s = "2-Father Noob";
    else if (noob == 3) s = "3-God Noob";
    else if (noob == 4) s = "4-Hof Noob";
    else s = "Unknown";
    System.out.println(id + " " + name + " [" + s + "]);
}
```

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

คลาส Room

```
public class Room {
```

//แอตทริบิวต์ s ที่เป็นอาร์เรย์ 2 มิติชนิด Student

```
public static Student s[][] = new Student[16][40];
```

//เมื่อกำหนด setSeat(...)

```
public static void setSeat(Student st) {
    int startRow = s.length - 1 - (s.length / 4) * (4 - st.getNoob());
    for (int i = startRow; i < startRow + 4; i++) {
        for (int j = 0; j < s[i].length; j++) {
            if (s[i][j].getName().equals("") &&
                j != s[i].length / 2 &&
                j != s[i].length / 2 - 1) {
                s[i][j].set(st);
                return;
            }
        }
    }
    System.out.println("No Reserved Seat");
}
```

//เมื่อกำหนด showroom(...)

```
public static void showRoom() {
    for (int i = 0; i < s.length; i++) {
        for (int j = 0; j < s[i].length; j++) {
            if (j == s[i].length / 2 || j == s[i].length / 2 - 1) {
                System.out.print(" ");
            } else {
                if (!s[i][j].getName().equals(""))
                    System.out.print(s[i][j].getName().substring(0, 1) + " ");
                else
                    System.out.print("# ");
            }
        }
        System.out.println();
    }
}
```

//เมื่อกำหนด showStudent(...)

```
public static void showStudent(int r, int c) {
    System.out.print("Seat Row " + r + " Column " + c + "\t");
    s[r + s.length - 2][c - 1].show();
}
```

//เมื่อกำหนด showStudent(...)

```
public static void showStudent() {
    for (int i = s.length - 1; i >= 0; i--) {
        for (int j = 0; j < s[i].length; j++) {
            if (!s[i][j].getName().equals("")) {
                System.out.print("Seat Row " + (s.length - i) + " Column " +
                    (j + 1) + "\t");
                s[i][j].show();
            }
        }
    }
}
```

//เมื่อกำหนด main(...)

```
public static void main(String[] args) {
    for (int i = 0; i < s.length; i++) {
        for (int j = 0; j < s[i].length; j++) {
            s[i][j] = new Student();
        }
    }
    Student s1 = new Student(53300121, "Akeudom", 1); setSeat(s1);
    Student s2 = new Student(53300221, "Phakhin", 1); setSeat(s2);
    Student s3 = new Student(53300321, "Taparb", 2); setSeat(s3);
    Student s4 = new Student(53300421, "Ong-ard", 3); setSeat(s4);
    Student s5 = new Student(53300521, "Wongyos", 4); setSeat(s5);
    showRoom();
    showStudent();
}
```

} //End of class