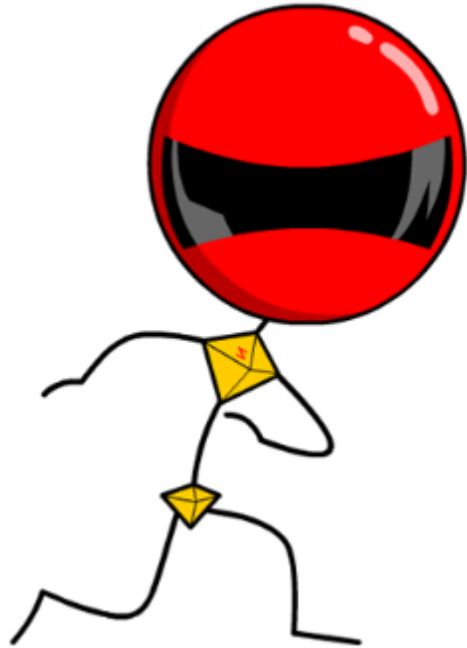


Character Class

Sungchul Lee

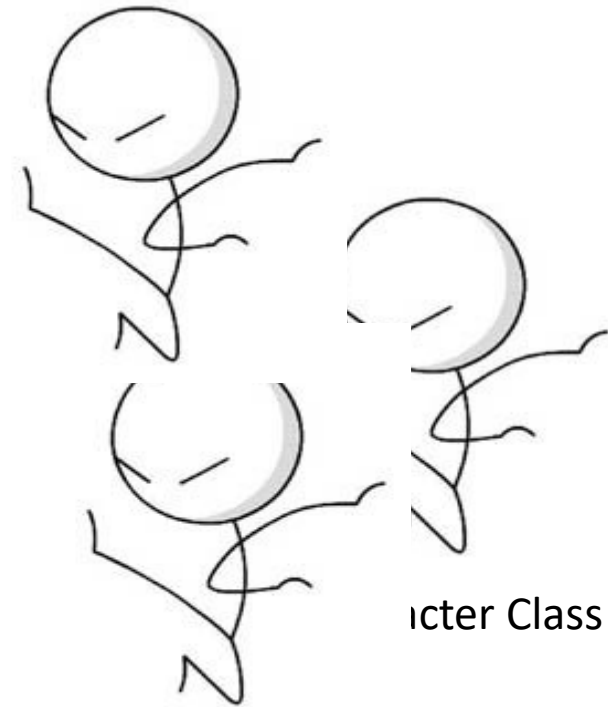
Learning Object

Main Character



Character Class

Enemy



Character Class

V.S.

Array of Object

Class Members

- A class have two kinds of members:
 - ❑ **Instance Variable (Attributes)**: data variables which determine the **status** of the class or an object
 - ❖ E.g.: HP, moving speed
 - ❑ **Methods**: executable code used to **manipulate /change the status** of an object or access the value of the data member
 - ❖ Similar to functions in procedural languages
 - ❖ E.g.: when your character hit an enemy, the enemy's HP will be reduced based on your hit points.

Character Class

➤ ***Instance Variable***

- ❑ Name, hp, power, defense, mp,
- ❑ server_name (static) – Next lecture

➤ ***Methods***

- ❑ Constructor for initializing character
- ❑ Getter and setter for instance variable
- ❑ Get_damage

➤ **Built-in Package**

- ❑ Random

Instance Variable in Character Class

- private String Name
 - ❑ Character's name
 - ❖ Main character and enemies
- private int hp
 - ❑ Character's health point
 - ❑ If hp is below '0', the character is dead
- private int power and private int defense
 - ❑ Power: amount of damage, defense: reducing damage
- private int mp
 - ❑ Mana Point for special skills

Methods in Character Class

- Getter and Setter
 - ❑ Hp, mp, power and defense
- Getter (Access control)
 - ❑ Name
- Constructor and Overloading
 - ❑ `Character(String name){}`
 - ❑ `Character(int hp, int power, String name){}`
- Damage
 - ❑ Reduce hp and return damage

Constructor and Overloading

Mostly, use capital
letter for class name

➤ Constructor

- ❑ Character name = **new** Character();

- ❑ Special **method**

 - ❖ **Not required return type**

- ❑ All class have **at least one constructor**

 - ❖ if you don't make constructor in the class

 - Classname () { } // is automatically created in the class

- ❑ Why we need the constructor?

 - ❖ Initializing necessary elements (variables)

 - ❖ **Overloading**

Built-in Package Random()

➤ Create random value

➤ **import java.util.Random;**

❑ Random random = new Random(); // generate object

❑ random.nextInt(100); //0~100

```
package testRandom;
import java.util.Random;
public class testRandom {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Random random = new Random();
        System.out.println(random.nextInt(100)); //0~100
    }
}
```


Practice

1. Make a new project (Reference: Create Project and Class File)
 - ☐ Project name: Game
2. Create a new Class File
 - ☐ Class name: Main
 - ☐ Class name: Character
3. Coding:
 - ☐ `import java.util.Random;` // for Character class
 - ☐ `import java.util.Scanner;` // for Main class

Practice – code (Character)

```
Character.java
1 import java.util.Random;
2 public class Character {
3     private int hp;
4     private int power;
5     private String name;
6     private int defense;
7     private int mp;
8     Random rnd = new Random();
9
10    public Character(String name) {
11        this.hp = rnd.nextInt(100);
12        this.power = rnd.nextInt(100);
13        this.name = name;
14        this.defense = rnd.nextInt(10);
15        this.mp = rnd.nextInt(100);
16    }
17    public Character(int hp, int power, String name) {
18        this.hp = hp;
19        this.power = power;
20        this.name = name;
21        this.defense = rnd.nextInt(10);
22        this.mp = rnd.nextInt(100);
23    }
```

Instance Variable
and Constructor

```
24    public int getHp() {
25        return hp;
26    }
27    public void setHp(int hp) {
28        this.hp = hp;
29    }
30    public int getPower() {
31        return power;
32    }
33    public void setPower(int power) {
34        this.power = power;
35    }
36    public String getName() {
37        return name;
38    }
39    //no setter for name
40    public int getDefense() {
41        return defense;
42    }
43    public void setDefense(int defense) {
44        this.defense = defense;
45    }
46    public int getMp() {
47        return mp;
48    }
49    public void setMp(int mp) {
50        this.mp = mp;
51    }
```

Getter and Setter

Damage method

```
53    public int damage(int enemy_power) {
54        int damage = enemy_power - this.defense;
55        if (damage < 0) { // avoid healing by damage
56            damage = 0;
57        }
58        this.hp = this.hp - damage;
59        if (this.hp < 0) { // avoid minus hp
60            this.hp = 0;
61        }
62        return damage;
63    }
64 } //End Class
```

Practice – code (Main)

Generate Characters

Static method: For
check status of
character

Main.java

```
1 import java.util.Scanner;
2 public class Main {
3     public static void main(String[] args) {
4         // TODO Auto-generated method stub
5         Scanner scanner = new Scanner(System.in);
6         System.out.println("Input Main Character Name:");
7         String main_name=scanner.nextLine();
8         System.out.println("Input Main Character Power:");
9         int main_power=scanner.nextInt();
10        System.out.println("Input Main Character HP:");
11        int main_hp=scanner.nextInt();
12
13        //Generate Main Character Object (Declare and initialize)
14        Character main_ch = new Character(main_hp,main_power,main_name);
15        show_status(main_ch);
16
17        Character enemies[] = new Character[5]; //Generate objects
18        for(int i =0 ; i<enemies.length;i++){
19            enemies[i]=new Character("enemy " + i); //initialize
20            show_status(enemies[i]);
21        }
22    } // end main method
23    public static void show_status(Character character){
24        System.out.println("=====Character Status=====");
25        System.out.println("Character Name:"+character.getName());
26        System.out.println("Character hp:"+character.getHp());
27        System.out.println("Character Name:"+character.getPower());
28        System.out.println("Character Name:"+character.getDefense());
29        System.out.println("Character hp:"+character.getMp());
30    }
31 } //End Class
```

Practice –Result

➤ Main Character Object

- ❑ Character Class

➤ Enemies Object

- ❑ Character Class

- ❑ Object Array

- ❑ For loop

Result

```
Markers Properties Servers Data Source Explorer Snippets Console
<terminated> Main [Java Application] C:\Program Files\Java\jre1.8.0_211\bin
Input Main Character Name:
Sung
Input Main Character Power:
10
Input Main Character HP:
1
=====Character Status=====
Character Name:Sung
Character hp:1
Character Name:10
Character Name:5
Character hp:49
=====Character Status=====
Character Name:enemy 0
Character hp:95
Character Name:24
Character Name:2
Character hp:84
=====Character Status=====
Character Name:enemy 1
Character hp:61
Character Name:92
Character Name:0
Character hp:81
=====Character Status=====
Character Name:enemy 2
Character hp:93
Character Name:62
Character Name:5
Character hp:36
```

Summary

Main Character



Character(int hp, int power "Name")

Character Class

Enemy

Character("enemy 2")

Character("enemy 1")

V.S.

Character Class

Character("enemy 3")