Jobsheet 10

Oleh :

NAME : Maulana Dwi Cahyono

CLASS : 2I

NO.ABSENT: 14

Major : Information Technology

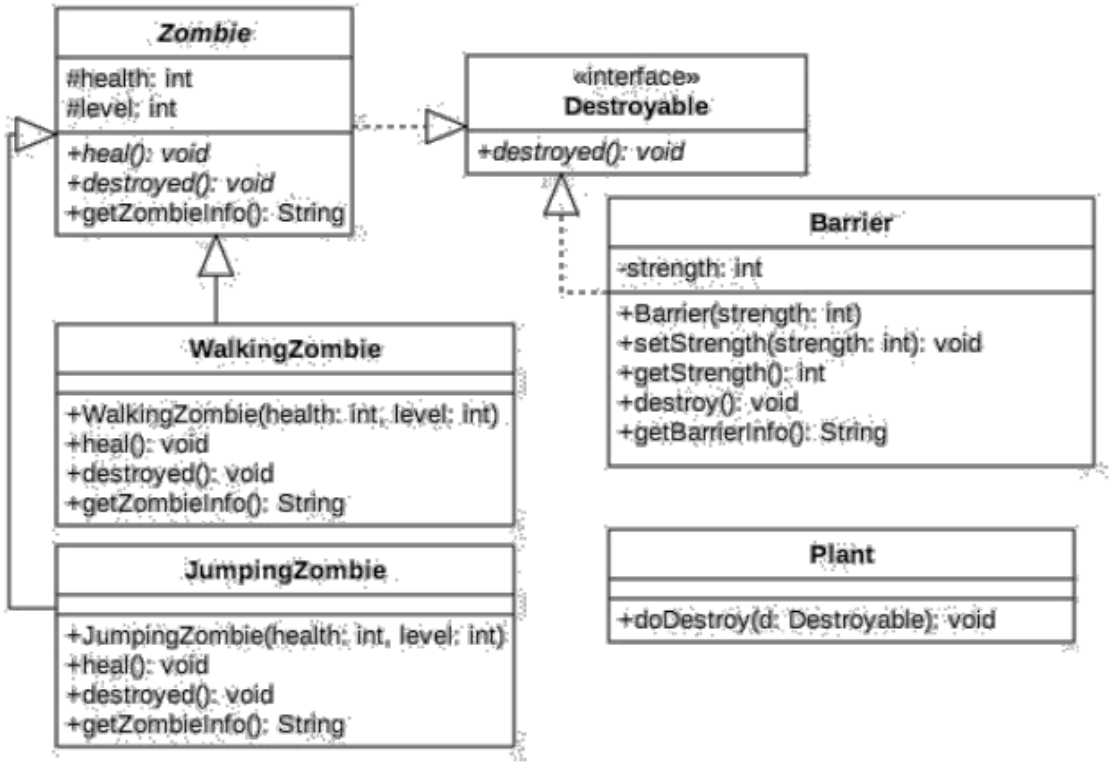
STUDY PROGRAM : Information Engineering

8. Tugas

Dalam suatu permainan, Zombie dan Barrier bisa dihancurkan oleh Plant dan bisa menyembuhkan diri. Terdapat dua jenis Zombie, yaitu Walking Zombie dan Jumping Zombie. Kedua Zombie tersebut memiliki cara penyembuhan yang berbeda, demikian juga cara penghancurannya, yaitu ditentukan oleh aturan berikut ini:

- Pada WalkingZombie
    - Penyembuhan : Penyembuhan ditentukan berdasar level zombie yang bersangkutan
        - Jika zombie level 1, maka setiap kali penyembuhan, health akan bertambah 10%
        - Jika zombie level 2, maka setiap kali penyembuhan, health akan bertambah 30%
        - Jika zombie level 3, maka setiap kali penyembuhan, health akan bertambah 40%
    - Penghancuran : setiap kali penghancuran, health akan berkurang 2%
- Pada Jumping Zombie
    - Penyembuhan : Penyembuhan ditentukan berdasar level zombie yang bersangkutan
        - Jika zombie level 1, maka setiap kali penyembuhan, health akan bertambah 30%
        - Jika zombie level 2, maka setiap kali penyembuhan, health akan bertambah 40%
        - Jika zombie level 3, maka setiap kali penyembuhan, health akan bertambah 50%
    - Penghancuran : setiap kali penghancuran, health akan berkurang 1%

Buat program dari class diagram di bawah ini!



Contoh: jika class Tester seperti di bawah ini:

```
3   public class Tester {
4       public static void main(String[] args) {
5           WalkingZombie wz = new WalkingZombie(100, 1);
6           JumpingZombie jz = new JumpingZombie(100, 2);
7           Barrier b = new Barrier(100);
8           Plant p = new Plant();
9           System.out.println(""+wz.getZombieInfo());
10          System.out.println(""+jz.getZombieInfo());
11          System.out.println(""+b.getBarrierInfo());
12          System.out.println("----------------------");
13          for(int i=0;i<4;i++){//Destroy the enemies 4 times
14              p.doDestroy(wz);
15              p.doDestroy(jz);
16              p.doDestroy(b);
17          }
18          System.out.println(""+wz.getZombieInfo());
19          System.out.println(""+jz.getZombieInfo());
20          System.out.println(""+b.getBarrierInfo());
21      }
22  }
```

Akan menghasilkan output:

```
run:
Walking Zombie Data =
Health = 100
Level = 1

Jumping Zombie Data =
Health = 100
Level = 2

Barrier Strength = 100

----------------------
Walking Zombie Data =
Health = 42
Level = 1

Jumping Zombie Data =
Health = 66
Level = 2

Barrier Strength = 64

BUILD SUCCESSFUL (total time: 2 seconds)
```

```java
package Jobsheet_12;

interface Destroyable {
    void destroyed();
}

class Zombie implements Destroyable {
    protected int health;
    protected int level;

    public Zombie(int health, int level) {
        this.health = health;
        this.level = level;
    }

    public void heal() {
        // Default implementation, bisa diubah sesuai kebutuhan
    }

    public void destroyed() {
        // Implement destroyed method logic here
    }

    public String getZombieInfo() {
        return "Health = " + health + "\nlevel = " + level;
    }
}


class WalkingZombie extends Zombie {
    public WalkingZombie(int health, int level) {
        super(health, level);
    }

    @Override
    public void heal() {
        // Penyembuhan ditentukan berdasarkan level zombie
        if (level == 1) {
            health += (int) (0.2 * health); // Bertambah 20%
        } else if (level == 2) {
            health += (int) (0.3 * health); // Bertambah 30%
        } else if (level == 3) {
            health += (int) (0.4 * health); // Bertambah 40%
        }
    }

    @Override
    public void destroyed() {
        // Penghancuran: health berkurang 2%
        health -= (int) (0.2 * health);
    }
}

class JumpingZombie extends Zombie {
    public JumpingZombie(int health, int level) {
        super(health, level);
    }

    @Override
    public void heal() {
        // Penyembuhan ditentukan berdasarkan level zombie
        if (level == 1) {
            health += (int) (0.3 * health); // Bertambah 30%
        } else if (level == 2) {
            health += (int) (0.4 * health); // Bertambah 40%
        } else if (level == 3) {
            health += (int) (0.5 * health); // Bertambah 50%
        }
    }

    @Override
    public void destroyed() {
        // Penghancuran: health berkurang 1%
        health -= (int) (0.1 * health);
    }
}


class Barrier implements Destroyable {
    private int strength;

    public Barrier(int strength) {
        this.strength = strength;
    }

    public void setStrength(int strength) {
        this.strength = strength;
    }

    public int getStrength() {
        return strength;
    }

    @Override
    public void destroyed() {
        this.strength -= 9;
        if (this.strength <= 0) {
            System.out.println("Barrier has been destroyed");
        }
    }

    public String getBarrierinfo() {
        return "" + strength;
    }
}

class Plant {
    public void doDestroy(Destroyable d) {
        d.destroyed();
    }
}

public class Tester {
    public static void main(String[] args) {
        WalkingZombie wz = new WalkingZombie(100, 1);
        JumpingZombie jz = new JumpingZombie(100, 2);
        Barrier b = new Barrier(100);
        Plant p = new Plant();

        System.out.println("Walking Zombie Data = \n" + wz.getZombieInfo());
        System.out.println("\nJumping Zombie Data = \n" + jz.getZombieInfo());
        System.out.println("\nBarrier Strength = " + b.getBarrierinfo());
        System.out.println("----------------------");

        for (int i = 0; i < 4; i++) {
            p.doDestroy(wz);
            p.doDestroy(jz);
            p.doDestroy(b);
        }

        System.out.println("\nWalking Zombie Data = \n" + wz.getZombieInfo());
        System.out.println("\nJumping Zombie Data = \n" + jz.getZombieInfo());
        System.out.println("\nBarrier Strength = " + b.getBarrierinfo());
    }
}

```

```
PS C:\Users\ASUS\Desktop\kuliah python\Kuliah Java_Sems 3>  c:;
121\bin\java.exe' '-cp' 'C:\Users\ASUS\AppData\Roaming\Code\Use
4a\bin' 'Jobsheet_12.Tester'
Walking Zombie Data =
Health = 100
level = 1

Jumping Zombie Data =
Health = 100
level = 2

Barrier Strength = 100
----------------------

Walking Zombie Data =
Health = 42
level = 1

Jumping Zombie Data =
Health = 66
level = 2

Barrier Strength = 64
PS C:\Users\ASUS\Desktop\kuliah python\Kuliah Java_Sems 3>
```