LAPORAN JOBSHEET 9

MATKUL PRAKTIKUM PEMOGRAMAN BERBASIS OBJEK

DOSEN PEMBIMBING

Dian Wahyuningsih, S.Kom., MMSI.

NAMA MAHASISWA

Ahmad Dzul Fadhli Hannan

2341720106

KELAS TI-2E

POLITEKNIK NEGERI MALANG

JURUSAN TEKNOLOGI INFORMASI

PRODI D4-TEKNIK INFORMATIKA

LINK GITHUB KODE PROGRAM :
https://github.com/ahmaddzulfadhlihannan/Praktikum-PBO-Semester-3/tree/main/minggu9/src/Tugas/src

Class Interface Destroyable

```java
/**
 *
 * @author ahmad
 */
public interface Destroyable {
    public void destroyed();
}
```

Class Zombie

```java
 * @author ahmad
 */
public class Zombie implements Destroyable{
    protected int health;
    protected int level;

    public void heal(){
        switch (level) {
            case 1:
                health += health * 0;
                break;
            case 2:
                health += health * 0;
                break;
            case 3:
                health += health * 0;
                break;
        }
    }

    public void destroyed(){
        health = 0;
    }

    public String getZombieInfo(){
        String info = "Health = " + health + "\n"+
                "Level = " + level;
        return info;
    }
}
```

Class WalkingZombie

```java
/**
 *
 * @author ahmad
 */
public class WalkingZombie extends Zombie implements Destroyable {

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

    @Override
    public void heal(){
        switch (level) {
            case 1:
                health += health * 0.1;
                break;
            case 2:
                health += health * 0.3;
                break;
            case 3:
                health += health * 0.4;
                break;
        }
    }

    @Override
    public void destroyed(){
        health = 0;
    }

    @Override
    public String getZombieInfo(){
        String info = "Walking Zombie Data = \n" +
                super.getZombieInfo();
        return info;
    }

}
```

Class JumpingZombie

```java
/**
 *
 * @author ahmad
 */
public class JumpingZombie extends Zombie implements Destroyable {
    public JumpingZombie(int health, int level) {
        this.health = health;
        this.level = level;
    }

    @Override
    public void heal(){
        switch (level) {
            case 1:
                health += health * 0.3;
                break;
            case 2:
                health += health * 0.4;
                break;
            case 3:
                health += health * 0.5;
                break;
        }
    }

    @Override
    public void destroyed(){
        health = 0;
    }


    @Override
    public String getZombieInfo(){
        String info = "Jumping Zombie Data = \n" +
                super.getZombieInfo();
        return info;
    }
}
```

Class Barrier

```java
/**
 *
 * @author ahmad
 */
public 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() {
        strength = 0;
    }

    public String getBarrierInfo(){
        String info = "Barrier Strength = " + strength;
        return info;
    }
}
```

Class Plant

Versi 1 (Hasil running sama seperti di jobsheet dimana *health WalkingZombie* menjadi 42 dan *JumpingZombie* menjadi 66 setelah dilakukan penghancuran 4 kali)

```java
/**
 *
 * @author ahmad
 */
public class Plant {
    public void doDestroy(Destroyable d){
        if(d instanceof WalkingZombie){
            ((WalkingZombie) d).health -= (((WalkingZombie) d).health * 0.19);
        } else if(d instanceof JumpingZombie){
            ((JumpingZombie) d).health -= (((JumpingZombie) d).health * 0.095);
        } else if(d instanceof Barrier){
            int temp = ((Barrier) d).getStrength();
            temp -= (temp * 0.1);
            ((Barrier) d).setStrength(temp);
        }else{
            System.out.println("Musuh tidak diketahui");
        }
    }
}
```

Versi 2 (Hasil sesuai dengan deskripsi pada tugas jobsheet dimana *health WalkingZombie* Berkurang 2% dan *JumpingZombie* berkurang 1%)

```
/**
 *
 * @author ahmad
 */
public class Plant {
    public void doDestroy(Destroyable d){
        if(d instanceof WalkingZombie){
            ((WalkingZombie) d).health -= (((WalkingZombie) d).health * 0.02);
        } else if(d instanceof JumpingZombie){
            ((JumpingZombie) d).health -= (((JumpingZombie) d).health * 0.01);
        } else if(d instanceof Barrier){
            int temp = ((Barrier) d).getStrength();
            temp -= (temp * 0.1);
            ((Barrier) d).setStrength(temp);
        }else{
            System.out.println("Musuh tidak diketahui");
        }
    }
}
```

Untuk *Barrier* dibuat 10% agar sama seperti hasil running di jobsheet.

Class Main

```
/**
 *
 * @author ahmad
 */
public class Tugas {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        WalkingZombie wz = new WalkingZombie(100, 1);
        JumpingZombie jz = new JumpingZombie(100, 2);
        Barrier b = new Barrier(100);
        Plant p =new Plant();
        System.out.println(wz.getZombieInfo());
        System.out.println(jz.getZombieInfo());
        System.out.println(b.getBarrierInfo());
        System.out.println("--------------------");
        for(int i = 0; i < 4; i++){
            p.doDestroy(wz);
            p.doDestroy(jz);
            p.doDestroy(b);
        }
        System.out.println(wz.getZombieInfo());
        System.out.println(jz.getZombieInfo());
        System.out.println(b.getBarrierInfo());
    }
}
```

Hasil Running

Class Plant Versi 1

```
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: 0 seconds)
```

Class Plant Versi 2

```
run:
Walking Zombie Data =
Health = 100
Level = 1
Jumping Zombie Data =
Health = 100
Level = 2
Barrier Strength = 100
--------------------
Walking Zombie Data =
Health = 92
Level = 1
Jumping Zombie Data =
Health = 96
Level = 2
Barrier Strength = 64
BUILD SUCCESSFUL (total time: 0 seconds)
```