

LAPORAN Pengerjaan Teori Pertemuan 3

TEKNIK PEMOGRAMAN

Disusun untuk memenuhi salah satu tugas pertemuan ketiga mata kuliah

Teknik Pemograman



Oleh:

Fadilah Akbar (231524041)

Kelas :

Teknik Informatika D4 – 1B

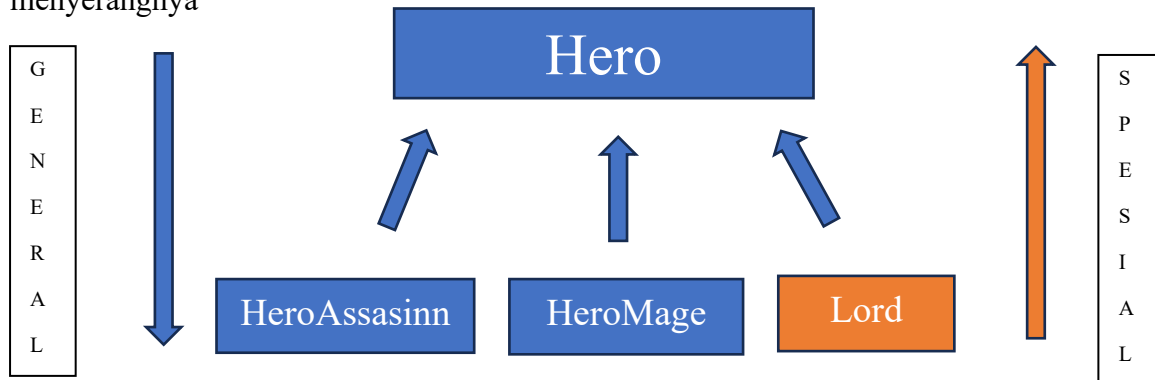
SARJANA TERAPAN PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA

POLITEKNIK NEGERI BANDUNG

CODE PROGRAM GAME HERO

Program ini bertujuan untuk menyediakan representasi hirarki dalam sebuah game, dengan hero-hero yang memiliki atribut dan perilaku yang berbeda, setiap hero memiliki nama, kekuatan serangan, dan kesehatan/darah yang berbeda. Ketika sebuah hero menyerang hero lain maka kesehatan pahlawan yang diserang akan berkurang sesuai dengan kekuatan serangan hero yang menyeranginya



TERDAPAT EMPAT CLASS DIANTARANYA

- **Class Hero**(Kelas Abstrak), kelas ini yang menyediakan kerangka dasar untuk semua jenis hero dalam game ini. Serta memiliki atribut untuk nama, kekuatan serangan, dan kesehatan/darah, serta metode untuk menampilkan informasi hero, method menyerang hero lain dan method menerima serangan musuh

```
1 package GameHero;
2
3 abstract class Hero {
4     // attributes
5     private String name;
6     private double attackPower;
7     private double health;
8
9     // superClass Constructor
10    public Hero(String name, double attackPower, double health) {
11        this.name = name;
12        this.attackPower = attackPower;
13        this.health = health;
14    }
15
16    // Methods
17    public void display() {
18        System.out.println("\nName: " + name);
19        System.out.println("Attack: " + attackPower);
20        System.out.println("Health: " + health);
21    }
22
23    public void attack(Hero enemy) {
24        System.out.println("\n" + name + " Menyerang Ke " + enemy.getName());
25        enemy.takeDamage(attackPower);
26    }
27
28    public void takeDamage(double damage) {
29        System.out.println(name + " Menerima Serangan Sebesar " + damage);
30        health -= damage;
31    }
32
33    // Setters
34    public void setHealth(double health) {
35        this.health = health;
36    }
37
38    // Getters
39    public String getName() {
40        return name;
41    }
42
43    public double getAttackPower() {
44        return attackPower;
45    }
46
47    public double getHealth() {
48        return health;
49    }
50 }
51
```

Program pada class ini memiliki sebuah superclass konstruktor yang nantinya akan dimiliki oleh setiap turunannya, konstruktor ini menerima tiga parameter yaitu 'name'(nama hero), 'attackPower'(kekuatan serangan) dan 'health'(kesehatan/darah).

Memiliki method 'display' untuk menampilkan informasi tentang pahlawan dari sebelum menerima serangan dan setelah menerima serangan dari hero lainya.

Terdapat sebuah method 'attack(Hero enemy)' digunakan untuk menyerang hero lain dan menerima objek 'Hero' sebagai parameter dan mencetak informasi yang menyatakan bahwa pada saat ini menyerang hero lain.

Dan method 'takeDamage(double damage)' method ini merupakan abstrak da harus diimplementasikan oleh kelas turunannya, digunakan untuk menerima jumlah kerusakan yang diterima dari kekuatan serangan sebagai parameternya, serta mengurangi kesehatan hero sesuai dengan jumlah kerusakan yang diterima.

- **HeroAssasin, HeroMage, dan Lord** (Kelas Konkreat) ketiga kelas ini adalah turunan dari kelas "Hero" dan merepresntasikan jenis hero yang berbeda dalam game. Setiap kelas ini memberikan implemtasi khusu untuk method 'takeDamage' yang menentukan seberapa banyak kesehatan/darah yang dikurangi saat menerima serangan.

```
1 package GameHero;
2
3 public class HeroAssasin extends Hero {
4     private String type = "Assassin";
5
6     // Constructor
7     public HeroAssasin(String name, double attackPower, double health) {
8         super(name, attackPower, health);
9     }
10
11    // Methods
12    @Override
13    public void display() {
14        super.display();
15        System.out.println("Type: " + type);
16    }
17
18    @Override
19    public void takeDamage(double damage) {
20        System.out.println(getName() + " Menerima Serangan Sebesar " + damage);
21        setHealth(getHealth() - 0.5 * damage);
22    }
23 }
24
```

Mengimplementasikan metode takeDamage(double damage) dengan mengurangi kesehatan hanya sebesar 50% dari jumlah kerusakan yang diterima.

```
1 package GameHero;
2
3 public class HeroMage extends Hero {
4     private String type = "Mage";
5
6     // Constructor
7     public HeroMage(String name, double attackPower, double health) {
8         super(name, attackPower, health);
9     }
10
11    // Methods
12    @Override
13    public void display() {
14        super.display();
15        System.out.println("Type: " + type);
16    }
17
18    @Override
19    public void takeDamage(double damage) {
20        System.out.println(getName() + " Menerima Serangan Sebesar " + damage);
21        setHealth(getHealth() - 0.3 * damage);
22    }
23 }
24
```

Mengimplementasikan metode takeDamage(double damage) dengan mengurangi kesehatan hanya sebesar 30% dari jumlah kerusakan yang diterima.

```
1 package GameHero;
2
3 public class Lord extends Hero {
4     private String type = "LORD";
5
6     // Constructor
7     public Lord(String name, double attackPower, double health) {
8         super(name, attackPower, health);
9     }
10
11    // Methods
12    @Override
13    public void display() {
14        super.display();
15        System.out.println("Type: " + type);
16    }
17
18    @Override
19    public void takeDamage(double damage) {
20        System.out.println(getName() + " Menerima Serangan Sebesar " + damage);
21        setHealth(getHealth() - 0.2 * damage); // Mengurangi hanya 30% dari damage
22    }
23 }
24
```

Mengimplementasikan metode takeDamage(double damage) dengan mengurangi kesehatan hanya sebesar 20% dari jumlah kerusakan yang diterima.

Meskipun ketiga kelas turunan(HeroAssassin, HeroMage, Lord) menggunakan method yang sama dari kelas Hero, implementasi dari metode takeDamage pada setiap kelas turunan dapat berbeda sesuai dengan karakteristik unik dari masing-masing tipe pahlawan. Hal ini memungkinkan pahlawan dengan tipe yang berbeda untuk memiliki respon yang berbeda terhadap serangan.

MAIN PROGRAM

```
1 package GameHero;
2
3 public class Main {
4     public static void main(String[] args) {
5         HeroAssasin hero1 = new HeroAssasin("Gusion", 25, 100);
6         HeroMage hero2 = new HeroMage("Lunox", 20, 100);
7         Lord lord = new Lord("Lord", 80, 1500);
8
9         // tampilan sebelum penyerangan
10        hero1.display();
11        hero2.display();
12        lord.display();
13
14        hero1.attack(lord);
15        hero2.attack(hero1);
16        lord.attack(hero2);
17
18        // tampilan setelah menerima serangan
19        hero1.display();
20        hero2.display();
21        lord.display();
22    }
23 }
24
```

Dalam program 'main' kita membuat sebuah objek hero dari berbagai jenis dengan menggunakan sebuah konsturktor masing masing dari kelas turunan dari 'Hero', setelah membuat objek hero dan lord selanjutnya memanggil method 'display' dari masing masing objek untuk menampilkan informasi hero sebelum melakukan penyerangan.

Selanjutnya mensimulasikan serangan antara hero dengan hero atau hero dengan lord yang telah dibuat, sebagai contoh 'hero1' menyerang 'lord', 'hero2' menyerang 'hero1', dan 'lord' menyerang 'hero2'. Dengan menggunakan method 'attack()' untuk menyerang.

Setelah simulasi serangan selesai, memanggil kembali method 'display()' dari masing masing objek untuk menampilkan informasi setelah serangan, informasi terdiri dari kesehatan hero yang akan diperbaharui

HASIL PROGRAM

sebelum serangan

```
Name: Gusion
Attack: 25.0
Health: 100.0
Type: Assasin
```

```
Name: Lunox
Attack: 20.0
Health: 100.0
Type: Mage
```

```
Name: Lord
Attack: 80.0
Health: 1500.0
Type: LORD
```

simulasi serangan

```
Gusion Menyerang Ke Lord
Lord Menerima Serangan Sebesar 25.0
```

```
Lunox Menyerang Ke Gusion
Gusion Menerima Serangan Sebesar 20.0
```

```
Lord Menyerang Ke Lunox
Lunox Menerima Serangan Sebesar 80.0
```

kondisi setelah serangan

```
Name: Gusion
Attack: 25.0
Health: 90.0
Type: Assasin
```

```
Name: Lunox
Attack: 20.0
Health: 76.0
Type: Mage
```

```
Name: Lord
Attack: 80.0
Health: 1495.0
Type: LORD
```