**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 bertujaun untuk menyediakan resentasi 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 menyerangnya

Hero

S

P

E

S

I

A

L

G

E

N

E

R

A

L

HeroAssasinn

HeroMage

Lord

# 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 methode untuk menampilkan informasi hero, method menyerang hero lain dan method menerima serangan musuh



Program pada class ini memiliki sebuah superclass construktor yang nantinya akan dimiliki oleh setiap turunanya, construktor 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 ‘takeDemage(double demage)’ method ini merupakan abstrak da harus diimplementasikan oleh kelas turunanya, 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 ‘takeDemage’ yang menentukan seberapa banyak kesehatan/darah yang dikurangi saat menerima serangan.



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



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



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

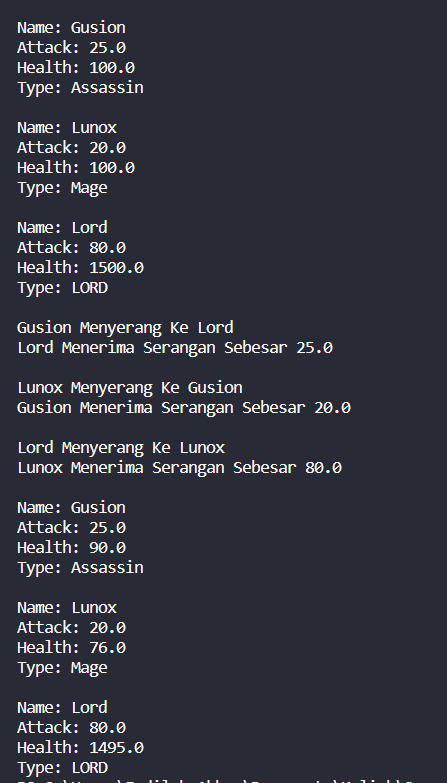
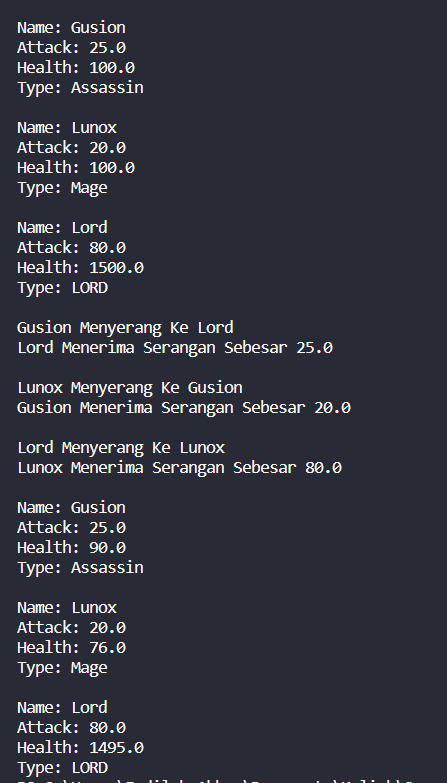
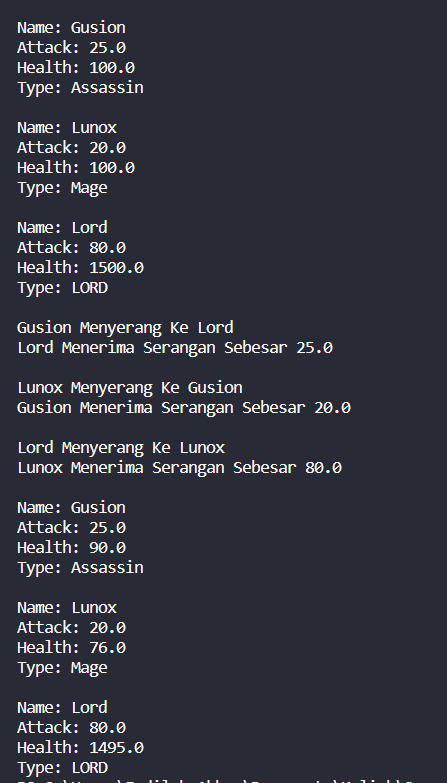


Dalam program ‘main’ kita membuat sebuah objek hero dari berbagau jenis dengan menggunakan sebuah konsturktor masing masing dari kelas turunan dari ‘Hero’, setelah membuat objek hero dan lord selnajutnya memanggil method ‘display’ dari masing masing objek untuk menampilkan informasi hero sebelum melakukan 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 simulasi serangan kondisi setelah serangan