**QUIZ 1**

**PEMROGRAMAN BERBASIS OBYERK**

Rabu, 31 August 2022

****

|  |
| --- |
| **Disusun Oleh:**  Nama: Bima Satria Erlangga  NIM: 210411100085  Kelas: Pemrograman Berbasis Obyek C |

**PRODI TEKNIK INFORMATIKA**

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS TEKNIK**

**UNIVERSITAS TRUNOJOYO MADURA**

**2022**

1. What is Class and Object in OOP….?

Jawab :

**Class** merupakan sebuah template/sturktur yang akan digunakan untuk membuat suatu obyek nantinya. Sedangkan obyek merupakan hasil dari suatu class yang dapat menjadi beberapa obyek yang bisa mengandung karakteristik yang berbeda.

1. What is different of Class and Object….?

Jawab :

Perbedaan class dan objek terletak pada kegunaannya, bila class digunakan sebagai struktur yang akan digunakan untuk membuat sebuah obyek. Sedangkan obyek merupakan hasil dari struktur class

1. How to create a new Object from a Class in Java OOP….?

Jawab :

Pertama buat class main terlebih dahulu, contoh:

public class Main {

public static void main(String[] args) {

Character jouza = new Character(200,60,45,Jouza,Human);

}

}

public class Character(){

int health, damage, defense;

String name, race;

public function Character(int healthCst, int damageCst, int defenseCst, String nameCst, String raceCst)

{

health = healthCst;

damage = damageCst;

defense = defenseCst;

name = nameCst;

race = raceCst;

}

}

1. What is Mutator and Accessor method…?

Jawab :

Mutator atau biasa disebut dengan setter method itu digunakan untuk mengset suatu nilai yang berada di dalam obyek. Sedangkan Acessor method atau biasa disebut sebagai getter method merupakan method yang digunakan untuk mendapatkan nilai dari obyek tersebut.

1. Create a Class with has minimal 5 attribute and 5 method.

Jawab :

public class Character(){

int health, damage, defense;

String name, race;

public Character(int healthCst, int damageCst, int defenseCst, String nameCst, String raceCst)

{

health = healthCst;

damage = damageCst;

defense = defenseCst;

name = nameCst;

race = raceCst;

}

public int attack()

{

health = damage-defense;

return health;

}

public int superAttack1()

{

health = (damage\*1.5)-defense;

return health;

}

public int finisherAttack()

{

health = 0;

return health;

}

public void characterTaunt()

{

System.out.println(“You will die in my hand”);

}

public int restoringHealth()

{

health = health+(health\*30/100)

}

}