

**Nama : Taufik Hidayat**

**NIM : 301200032**

**Kelas : IF Pagi**

**Mata Kuliah : Pemrograman Berorientasi Objek**

**Soal 1 : Concept PBO**

1. Jelaskan perbedaan Pemrograman Berorientasi Objek dengan Aplikasi Hybrid? Beri contoh masing – masing!

**Jawaban:**

- Pemrograman berorientasi objek adalah paradigma pemrograman yang didasarkan pada konsep "objek", yang dapat berisi data dalam bentuk field atau disebut juga atribut; dan kode, dalam bentuk fungsi/prosedur atau method.

Contoh:

- Java
  - Java Spring
- PHP
  - CodeIgniter
  - Laravel
- Aplikasi Hybrid adalah aplikasi web yang kemudian ditransformasikan menjadi kode *native*.

Contoh:

- React Native
  - Electron.js
  - Apache Cordova
2. Jelaskan komponen dasar Pemrograman Berorientasi Objek! Berikan contohnya!!

**Jawaban:**

- a. Class

Di dalam objek, kelas terdapat prototipe atau desain, yang digunakan untuk menjelaskan variabel dan berbagai method pada objek tertentu.

b. Object

Object merupakan wujud nyata dari prototype atau rancangan yang disebutkan dalam class tadi.

c. Property

Property adalah karakteristik yang dimiliki oleh sebuah object yang terdiri dari variabel-variabel.

d. Method

Method atau metode merupakan suatu operasi berupa fungsi-fungsi yang dapat dikerjakan oleh suatu objek.

e. Encapsulation

Ini adalah suatu bentuk dimana pengguna dari sebuah objek tidak dapat mengganti atau mengubah (memodifikasi) suatu keadaan dari sebuah objek dengan cara yang tidak layak, oleh karena itu keadaan ini hanya dapat dimodifikasi sesuai dengan ketentuan yang terdapat di dalam objek tersebut.

f. Inheritance

Inheritance atau pewarisan, suatu class dapat diwariskan atau diturunkan kepada class lainnya di dalam satu program.

g. Polymorphism

Polimorfisme merupakan sebuah konsep OOP di mana class memiliki banyak “bentuk” method yang berbeda, meskipun namanya sama. Maksud dari “bentuk” adalah isinya yang berbeda, namun tipe data dan parameternya berbeda.

h. Interface

Interface adalah template untuk class dimana semua method di dalamnya harus diimplementasikan ulang oleh class yang memakainya (mengimplementasikannya).

i. Abstract

Abstraksi merupakan kemampuan dari sebuah program untuk melewati aspek informasi yang diproses olehnya, atau bisa juga disebut sebagai kemampuan untuk memfokuskan pada intinya.

## Soal 2 : Class & Object

Diketahui :

- Hewan
- Memakan
- Buah
- Sayuran

1. Buatlah sifat / atribut / properti / field
2. Buatlah tingkah laku / behavior / method
3. Buatlah script/ code untuk pemrograman berorientasi objek menggunakan bahasa pemrograman Java, beserta atribut dan methodnya

### Jawaban:

1. Hewan:

- Nama
- Jumlah Kaki
- Suara

Buah:

- Nama
- Warna
- Bentuk

Sayuran:

- Nama
- Warna
- Bentuk

2. Hewan:

- Bersuara
- Memakan

Buah:

- getName

Sayuran:

- getNama

3.

### Source Code

- Hewan.java

```
package UTSClassAndObject;

public class Hewan {
    private String nama, jumlahKaki, suara;
    public Hewan(String nama, String jumlahKaki, String suara) {
        this.nama = nama;
        this.jumlahKaki = jumlahKaki;
        this.suara = suara;
    }
    public String getNama() {
        return "Nama hewan ini adalah " + this.nama;
    }
    public String getJumlahKaki() {
        return "Kaki " + this.nama + " berjumlah " + this.jumlahKaki;
    }
    public String bersuara() {
        return this.nama + " bersuara " + this.suara;
    }

    public String memakan(String makanan) {
        return this.nama + " memakan " + makanan;
    }
}
```

- Buah.java

```
package UTSClassAndObject;

public class Buah {
    private String nama, warna, bentuk;
    public Buah(String nama, String warna, String bentuk) {
        this.nama = nama;
        this.warna = warna;
        this.bentuk = bentuk;
    }
    public String getNama() {
        return this.nama;
    }
    public void rincian() {
        System.out.println(
            "Nama : " + this.nama + "\n" +
            "Warna : " + this.warna + "\n" +
            "Bentuk: " + this.bentuk
        );
    }
}
```

- Sayuran.java

```
package UTSClassAndObject;

public class Sayuran {

    private String nama, warna, bentuk;

    public Sayuran(String nama, String warna, String bentuk) {

        this.nama = nama;

        this.warna = warna;

        this.bentuk = bentuk;

    }

    public String getNama() {

        return this.nama;

    }

    public void rincian() {

        System.out.println(

            "Nama : " + this.nama + "\n" +

            "Warna : " + this.warna + "\n" +

            "Bentuk: " + this.bentuk

        );

    }

}
```

- Main.java

```
package UTSClassAndObject;

public class Main {

    public static void main(String[] args) {

        Hewan monyet = new Hewan("Monyet", "2", "Uu aa");

        Hewan kelinci = new Hewan("Kelinci", "2", "Citcitcit");

        Buah pisang = new Buah("Pisang", "Kuning", "Panjang dan melengkung");

        Sayuran wortel = new Sayuran("Wortel", "Oranye", "Panjang dan ujungnya melancip");

    }

}
```

```
System.out.println(monyet.memakan(pisang.getNama()));  
    System.out.println(kelinci.memakan(wortel.getNama()));  
}  
}
```

### Output

```
Monyet memakan Pisang  
Kelinci memakan Wortel
```

### Soal 3 : Inheritance & Polymorphism

1. Untuk mewarisi sebuah class, anda hanya perlu menggunakan kata kunci extends.  
Tulis/buatlah sebuah program sederhana yang mendemostrasikan sebuah pewarisan!  
Beserta outputnya!
2. Tulis/buatlah sebuah program menggunakan pewarisan untuk mewarisi kotak yang membuat lebar, tinggi, dalam dan berat !

### Jawaban:

#### 1. Source Code

```
package UTSInheritancePolymorphism;  
  
public class Komputer {  
    private String merk;  
    public Komputer(String merk) {  
        this.merk = merk;  
    }  
    public String getMerk() {  
        return this.merk;  
    }  
}
```

Komputer.java

```
    }  
}
```

### Laptop.java

```
package UTSPolymorphism;  
  
public class Laptop extends Komputer {  
    public Laptop(String merk) {  
        super(merk);  
    }  
}
```

### Main.java

```
package UTSPolymorphism;  
  
public class Main {  
    public static void main(String[] args) {  
        Laptop lenovo = new Laptop("Lenovo");  
        System.out.println("Merk laptop ini adalah " + lenovo.getMerk());  
    }  
}
```

### Output

```
Merk laptop ini adalah Lenovo
```



## 2. Source Code

BangunRuang.java

```
package UTSInheritancePolymorphism;

public class BangunRuang {
    private double panjang, lebar, tinggi;

    public BangunRuang(double panjang, double lebar, double tinggi) {
        this.panjang = panjang;
        this.lebar = lebar;
        this.tinggi = tinggi;
    }

    public double hitungLuas() {
        return this.panjang * this.lebar * this.tinggi;
    }
}
```

Kotak.java

```
package UTSInheritancePolymorphism;

public class Kotak extends BangunRuang {
    private double berat;

    public Kotak(double panjang, double lebar, double tinggi, double
berat) {
        super(panjang, lebar, tinggi);
        this.berat = berat;
    }

    public double getBerat() {
        return this.berat;
    }
}
```

```
package UTSInheritancePolymorphism;

public class Main {

    public static void main(String[] args) {

        Kotak kotak = new Kotak(7.2, 13.6, 23.7, 47.3);

        System.out.println(

            "Rincian kotak \n" +

            "Luas : " + kotak.hitungLuas() + " cm³\n" +

            "Berat: " + kotak.getBerat() + " kg"

        );

    }

}
```

Main.java

### Output

```
Rincian kotak
Luas : 2320.704 cm³
Berat: 47.3 kg
```

### Soal 4 : Encapsulation

1. Tulis/buatlah sebuah program menggunakan pengkapsulan untuk menampilkan nama dan usia ! Beserta outputnya!
2. Mengapa pengkapsulan penting dalam sebuah pemrograman/ aplikasi? Jelaskan dan beri contohnya !

### Jawaban

1. **Source Code**

```
package UTSEncapsulation;  
  
public class Mahasiswa {  
    private String nama;
```

Mahasiswa.java

```
        private int usia;  
        public Mahasiswa(String nama, int usia) {  
            this.nama = nama;  
            this.usia = usia;  
        }  
        public String getNama() {  
            return this.nama;  
        }  
        public int getUsia() {  
            return this.usia;  
        }  
    }  
}
```

## Main.java

```
package UTSEncapsulation;

public class Main {

    public static void main(String[] args) {

        Mahasiswa mhs = new Mahasiswa("Taufik Hidayat", 19);

        System.out.println(

            "==== DATA MAHASISWA =====\n" +

            "Nama: " + mhs.getNama() + "\n" +

            "Usia: " + mhs.getUsia() + " tahun"

        );

    }

}
```

## Output

```
==== DATA MAHASISWA =====
Nama: Taufik Hidayat
Usia: 19 tahun
|
```

2. Encapsulation penting karena dapat menghilangkan akses publik ke atribut-atribut yang terdapat di dalam “kapsul” tersebut. Metode ini dapat memudahkan anda dalam mendefinisikan atribut apa saja yang dapat dibaca dan diperbarui

Contoh command encapsulation yaitu:

- Public

Atribut data diakses oleh semuanya

- Protected  
Atribut hanya dapat diakses oleh class itu sendiri dan class keturunanya
- Private  
Atribut hanya dapat diakses oleh class itu sendiri

### Soal 5 : Interface & Abstract

1. Tulis/buatlah sebuah program untuk mengilustrasikan antarmuka bersarang ! Beserta outputnya!
2. Tulis/buatlah sebuah program untuk mengilustrasikan class abstract dari sebuah bentuk (lingkaran, kotak) ! Beserta outputnya!

### Jawaban

#### 1. Source Code

Munculkan.java

```
package UTSInterfaceAbstract;  
  
interface Munculkan {  
    interface Pesan {  
        void msg(String pesan);  
    }  
}
```

```
package UTSInterfaceAbstract;  
  
class NestedInterface implements Munculkan.Pesan {  
    public void msg(String pesan) {  
        System.out.println(pesan);  
    }  
}
```

NestedInterface.java

```
package UTSInterfaceAbstract;

public class Main {

    public static void main(String args[]) {

        Munculkan.Pesan pesan = new NestedInterface();

        pesan.msg("Assalamu'alaikum");

    }

}
```

Main.java

**Output:**

Assalamu'alaikum

## 2. Source Code

### BangunDatar.java

```
package UTSTInterfaceAbstract;  
  
public abstract class BangunDatar {  
    public abstract double getKeliling();  
}
```

### Kotak.java

```
package UTSTInterfaceAbstract;  
  
public class Kotak extends BangunDatar {  
    private double sisi;  
  
    public Kotak(double sisi) {  
        this.sisi = sisi;  
    }  
  
    @Override  
    public double getKeliling() {  
        return this.sisi * 4;  
    }  
}
```

### Lingkaran.java

```
package UTSInterfaceAbstract;

public class Lingkaran extends BangunDatar {

    private double diameter;

    public Lingkaran(double diameter) {

        this.diameter = diameter;

    }

    @Override

    public double getKeliling() {

        return this.diameter * 3.14;

    }

}
```

### Main.java

```
package UTSInterfaceAbstract;

import java.text.NumberFormat;

public class Main {

    public static void main(String args[]) {

        NumberFormat nf = NumberFormat.getInstance();

        nf.setMaximumFractionDigits(2);

        double sisi = 9.1;

        BangunDatar kotak = new Kotak(sisi);

        double diameter = 2.5;

        BangunDatar lingkaran = new Lingkaran(diameter);

        System.out.println(

            "Sisi kotak          : " + sisi + " cm\n" +

            "Keliling kotak       : " + nf.format(kotak.getKeliling()) + " cm\n"

        );

        System.out.println(

            "Diameter lingkaran: " + diameter + " cm\n" +


```



```
        "Keliling lingkaran: " + nf.format(lingkaran.getKeliling()) + "  
cm"  
    );  
}  
,
```

### Output

```
Sisi kotak      : 9.1 cm  
Keliling kotak  : 36.4 cm  
  
Diameter lingkaran: 2.5 cm  
Keliling lingkaran: 7.85 cm  
|
```

### Soal 6 : Package

1. Tulis/buatlah sebuah program untuk mendemostrasikan teknik dalam mengimpor paket pada Java ! Beserta outputnya!

### Jawaban:

```
package UTSPackage;
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.print("Siapa nama Anda? ")
        String nama = in.nextLine();
        System.out.printf("Selamat datang, %s", nama);
        in.close();
    }
}
```

#### 1. Source Code

##### Output

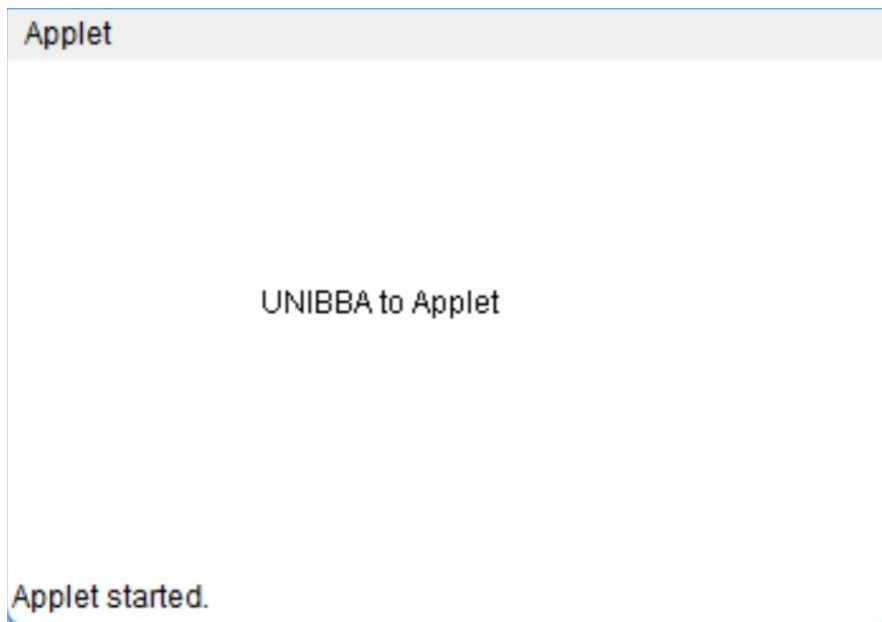
```
Siapa nama Anda? Taufik Hidayat
Selamat datang, Taufik Hidayat|
```

#### Soal 7: Applet

#### 1. Source Code

```
import java.applet.Applet;
import java.awt.Graphics;
public class UNIBBATOApplet extends Applet {
    public void paint (Graphics g) {
        g.drawString("UNIBBA to Applet", 100, 100);
    }
}
```

## Output



## 2. Source Code

```
import java.applet.Applet;
import java.awt.Graphics;
import java.awt.Color;
public class Garis extends Applet {
    public void paint(Graphics g) {
        setBackground(Color.black);
        g.setColor(Color.green);
        int awal = 0;
        for (int i = 0; i < 10; i++) {
            g.drawLine(awal, 0, 250, 250);
            awal += 20;
        }
    }
}
```

## Output

