**Soal Pertama**Jelaskan secara singkat dari ketiga jenis Constructor tersebut serta berikan contoh penerapan kodenya.

**Jawaban**

* Default Constructor

Merupakan Constructor yang disediakan secara otomatis serta dinisialisasi semua atribut tersebut menjadi memiliki nilai NULL.

**Source Code**

public class Mobil{

String warna;

int jmlRoda = 4;

public static void main(String[] args){

Mobil mobil = new Mobil();

System.out.println(mobil.warna);

mobil.warna = "hitam";

System.out.println(mobil.warna);

}

}

* Constructor tanpa Parameter

Merupakan Constructor yang telah didefinisikan. Akan tetapi, tidak menerima parameter/ argument apapun sehingga atribut dari suatu Class akan dinisialisasi sebagai nilai tetap yang didefinisilan untuk semua objek.

**Source Code**

public class Mobil{

String warna;

int jmlRoda = 4, cc;

public Mobil(){

warna = "hitam";

}

public static void main(String[] args){

Mobil mobil = new Mobil();

System.out.println(mobil.warna);

System.out.println(mobil.jmlRoda);

System.out.println(mobil.cc);

}

}

* Constructor dengan Parameter

Merupakan Constructor yang telah didefinisikan dan biasanya digunakan untuk memberikan nilai awal pada atribut Class bersamaan dengan object yang akan dibuat.

**Source Code**

public class Mobil{

String warna;

int jmlRoda = 4;

public Mobil(String warnaAwal, int jmlRoda) {

this.jmlRoda = jmlRoda;

warna = warnaAwal;

}

public static void main(String[] args){

Mobil mobil = new Mobil("hitam", 4);

System.out.println(mobil.warna);

System.out.println(mobil.jmlRoda);

}

}

**Soal Kedua**perhatikan kode berikut:

public class AdaYangSalah {

int a, b;

public AdaYangSalah() {

this.a = 1;

this.b = 2;

return true;

}

}

Cobalah kode diatas dan lihatlah apa yang terjadi? Jika ada error, coba jelaskan apa penyebab dari kesalahan kode diatas?

**Jawaban**

Maka akan terjadi error programnya. Penyebabmya adalah karena program tersebut muncul pesan dalam IDE”Java: incompatible types: unexpected return value” yang maknanya tidak bisa return value.

**Output Program**

|  |
| --- |
|  |

**Source Code**

public class AdaYangSalah {

int a, b;

public AdaYangSalah() {

this.a = 1;

this.b = 2;

}

}

**Soal Ketiga**Telah diberikan beberapa Class seperti dibawah ini,

* File Kucing.java

public class Kucing {

String jenisKucing;

int umur;

//Constructor

//...

}

* File Anjing.java

public class Anjing {

String jenisAnjing;

int umur;

//Constructor

//...

}

Buatlah “Constructor Tanpa Parameter” pada Class Anjing (isi juga atributnya) dan “Constructor Dengan Parameter” pada Class Kucing diatas!

**Source Code**

* File Kucing.java

public class Kucing {

String jenisKucing;

int umur;

public Kucing(String jenis, int usia) {

this.jenisKucing = jenis;

this.umur = usia;

}

}

* File Anjing.java

public class Anjing {

String jenisAnjing;

int umur;

public Anjing(){

jenisAnjing = "Cihuahua";

umur = 8;

}

}

**Soal Keempat**Cobalah kode dibawah ini

public class Mahasiswa{

String namaMhs, npm;

public Mahasiswa(){

namaMhs = "Agus";

npm = "06.2021.1.09876";

}

public Mahasiswa(String namaMhs, String npm){

this.namaMhs = namaMhs;

this.npm = npm;

}

}

Buat Object Mahasiswa mennggunakan 2 jenis Constructor diatas, kemudian coba print atribut namaMhs dan npm!

**Source Code**

public static void main(String[] args){

String nama = "Ryan", npm = "06.2021.1.07467";

Mahasiswa mahasiswa = new Mahasiswa();

System.out.println(mahasiswa.namaMhs);

System.out.println(mahasiswa.npm);

Mahasiswa mahasiswa1 = new Mahasiswa(nama, npm);

System.out.println(mahasiswa1.namaMhs);

System.out.println(mahasiswa1.npm);

}

**Output Program**

|  |
| --- |
|  |

**Soal Kelima**

**PROGRESS STUDI KASUS PROJECT AKHIR:** Terapkan Constructor yang telah kalian buat. Serta berikan penjelasan maksud dan tujuan dari Constructor yang kalian buat!

**Source Code**

public class Transaksi

{

int transfer, tarik;

public Transaksi(int transfer, int tarik){

this.transfer = transfer;

this.tarik = tarik;

}

}

**Jawaban**

Maksud dan tujuan dari diberikannya Constructor tersebut pada Class Transaksi adalah untuk menginisialisasi atribut transfer, tarik.

**Source Code**

public class Saldo

{

int jumlahSaldo;

public Saldo(int saldo){

jumlahSaldo = saldo;

}

}

**Jawaban**

Maksud dan tujuan dari diberikannya Constructor pada Class Saldo adalah untuk menginisialisasi atribut jumlahSaldo.

**Source Code**

public class Rekening

{

String nomor;

int pin;

Saldo saldo;

public Rekening(String no, int pin, Saldo saldo) {

nomor = no;

this.pin = pin;

this.saldo = saldo;

}

}

**Jawaban**

Maksud dan tujuan dari diberikannya Constructor pada Class Rekening adalah untuk menginisialisasi atribut nomor, pin, dan saldo.

**Source Code**

public class User

{

String nama, noTelpon, ttl, nik;

Rekening rekening;

public User (String fullName, String phone, String birth, String nik, Rekening rek)

{

this.nama = fullName;

this.noTelpon = phone;

this.ttl = birth;

this.nik = nik;

this.rekening = rek;

}

}

**Jawaban**

Maksud dan tujuan dari diberikannya Constructor pada Class User adalah untuk menginisialisasi atribut nama, noTelepon, ttl, nik, dan rekening.