

PEMROGRAMAN BERORIENTASI OBYEK D3IF-46-03 [MIU]

[Dashboard](#) / [My courses](#) / [FAKULTAS ILMU TERAPAN \(FIT\)](#) / [PRODI D3 REKAYASA PERANGKAT LUNAK APLIKASI \(FIT\)](#) / [2324/1](#)
/ [VII2C4-D3IF-46-03](#) / [Assessment 1](#) / [Assessment 1 Teori](#)

Started on Thursday, 26 October 2023, 12:48 PM

State Finished

Completed on Thursday, 26 October 2023, 1:22 PM

Time taken 34 mins 10 secs

Marks 201.25/265.00

Summary

Grade 75.94 out of 100.00

Question 1

Complete

Mark 10.00 out of 10.00

Sebuah method static hanya dapat mengakses property static dan method static lainnya. Sedangkan method non-static dapat mengakses property dan method, baik bersifat static ataupun non-static

Select one:

- ☒ True
☐ False

Question 2

Complete

Mark 10.00 out of 10.00

Diketahui kode sebagai berikut:

```
public class NewClass4 {  
    public static void main(String[] args) {  
        int[] b = {7,0,6,9,5};  
        try {  
            int hasil = b[4]/b[1];  
            System.out.println(hasil);  
        }catch (Exception e) {  
            System.out.println("eksepsi");  
        }catch (ArrayIndexOutOfBoundsException e) {  
            System.out.println("error akses array");  
        }catch (ArithmeticException e) {  
            System.out.println("error aritmatika");  
        }  
    }  
}
```

Hasil dari kode di atas adalah

Summary

Select one:

- ☒ a.
Compile Error
- ☐ b.
error aritmatika
- ☐ c.
eksepsi
- ☐ d.
error akses array

Question 3

Complete

Mark 10.00 out of 10.00

Mata kuliah Pemrograman Berorientasi Objek merupakan mata kuliah 4 SKS, yang terdiri atas 2 SKS teori (100 menit) dan 2 SKS praktik (200 menit)

Select one:

- ☒ True
- ☐ False

Question 4

Complete

Mark 10.00 out of 10.00

Pilihlah yang termasuk ke dalam keyword bahasa Java (jawaban lebih dari satu)

Select one or more:

- ☒ a.
do
- ☐ b.
main1
- ☒ c.
public
- ☒ d.
final

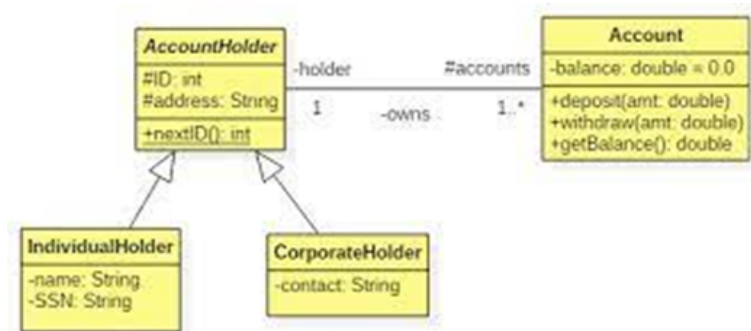
Summary

Question 5

Complete

Mark 0.00 out of 10.00

Perhatikan gambar berikut!



Sumber gambar: <https://www.cs.sjsu.edu/~pearce/modules/lectures/uml2/class/index.htm>

Berdasarkan class diagram di atas, maka kelas **IndividualHolder** dan **CorporateHolder** merupakan part (bagian) dari kelas **AccountHolder**

Select one:

- ☒ True
- ☐ False

Question 6

Complete

Mark 10.00 out of 10.00

Enkapsulasi didefinisikan sebagai “pembungkusan” data ke dalam sebuah unit.

Select one:

- ☒ True
- ☐ False

Question 7

Complete

Mark 10.00 out of 10.00

Kode mata kuliah Pemrograman Berorientasi Obyek di prodi D3 Rekayasa Perangkat Lunak Aplikasi tahun ajaran gasal 2023-2024 adalah VII2C4.

Summary

Select one:

- ☒ True
- ☐ False

Question 8

Complete

Mark 0.00 out of 10.00

Perhatikan kode berikut:

```
public void Kendaraan(){  
  
}
```

Kode di atas merupakan konstruktor kosong dari class Kendaraan.

Select one:

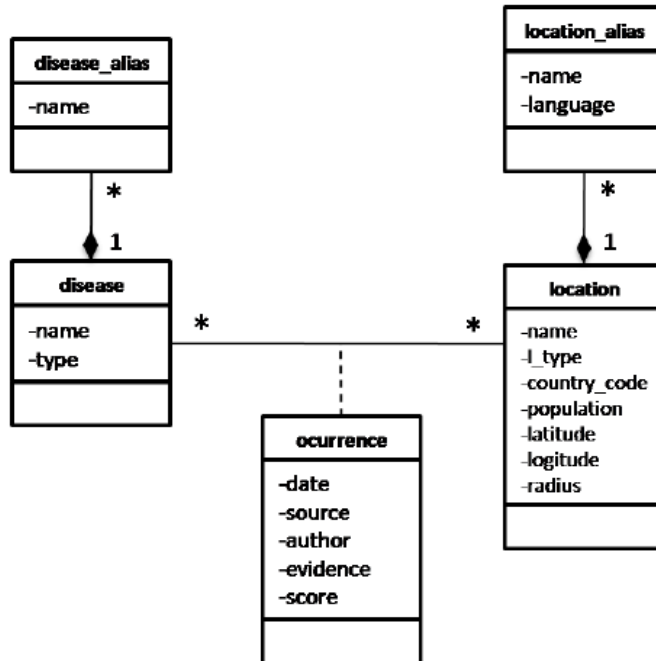
- ☒ True
- ☐ False

Question 9

Complete

Mark 11.25 out of 15.00

Berdasarkan gambar berikut, pilihlah pasangan yang sesuai



Summary

Sumber

gambar: https://www.researchgate.net/publication/252067754_Automated_Social_Network_Epidemic_Data_Collector/figures?lo=1

Kardinalitas antara kelas Disease dan Location

Many to many

Jenis hubungan kelas antara kelas Disease dan Location

Asosiasi

Pada hubungan antara kelas Location dan Location_alias, kelas yang menjadi kelas whole adalah...

Location_alias

Jenis hubungan kelas antara kelas Disease_Alias dan Disease

Komposisi

Question 10

Complete

Mark 10.00 out of 10.00

cocokkanlah access modifier berikut dengan sifatnya

atribut/method yang dikenai modifier ini hanya bisa diakses dalam package yang sama, atau melalui child class walaupun package berbeda.

protected

atribut/method yang dikenai modifier ini dapat diakses dari mana saja termasuk dari class dengan package yang berbeda

public

atribut/method yang dikenai modifier ini hanya bisa diakses dalam class yang sama.

private

Question 11

Complete

Mark 10.00 out of 10.00

Jika diketahui pada class diagram pada level/bagian bawah tertulis

"+x(a: int): void"

pilihlah hal yang benar tentang informasi berikut:

Select one or more:

- ☒ a.
Terdapat method x berupa procedure
- ☐ b.
Terdapat konstruktor dengan 1 parameter pada class X
- ☒ c.
Terdapat method x dengan 1 parameter bertipe integer
- ☐ d.
Terdapat method a dengan 1 parameter bernama x

Summary

Question 12

Complete

Mark 10.00 out of 10.00

Berikut adalah cara yang sering dilakukan untuk menerapkan enkapsulasi, kecuali:

Select one:

- ☒ a.
Memberikan access modifier private pada constructor
- ☐ b.
memberikan method "get" untuk mengambil value property
- ☐ c.
memberikan method "set" untuk memberi nilai/value property
- ☐ d.
menggunakan access modifier private pada property

Question 13

Complete

Mark 10.00 out of 10.00

Cocokkanlah aturan penulisan bahasa pemrograman java dan peruntukannya

pascal case

nama class

camel case

atribut

penggunaan ALL CAPS (dituliskan dengan huruf kapital semua)

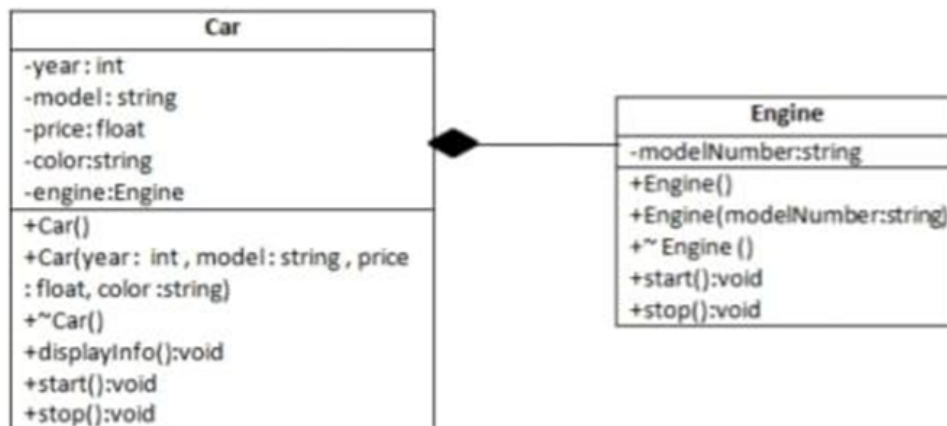
konstanta

Question 14

Complete

Mark 0.00 out of 10.00

Berdasarkan gambar berikut, pilihlah pernyataan yang tepat



Sumber: <https://www.chegg.com/homework-help/questions-and-answers/assignment-08-problem-01-implement-following-uml-diagram-except-destructors-car-engine-jav-q79232419>

Summary

Select one:

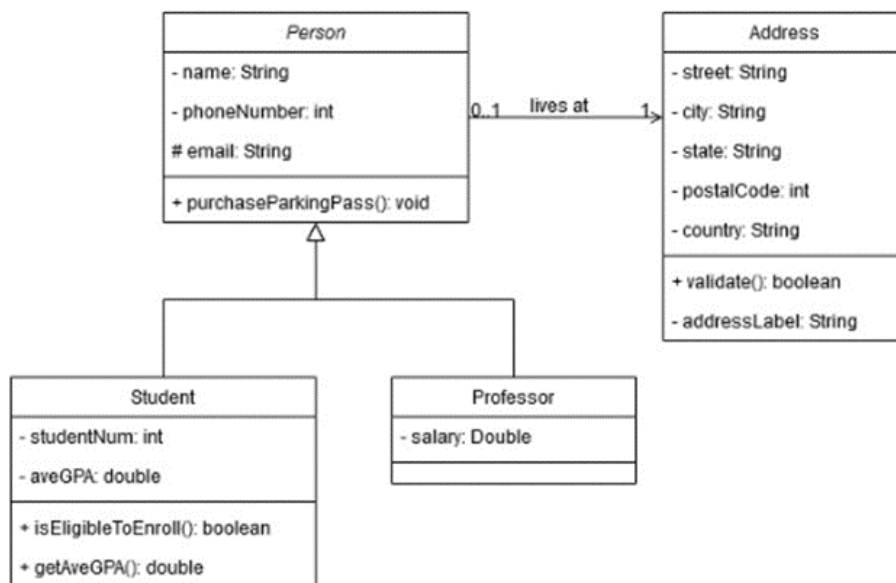
- ☐ a.
Hubungan antara kelas Car dan Engine merupakan hubungan pewarisan
- ☐ b.
Hubungan antara kelas Car dan Engine merupakan hubungan asosiasi
- ☒ c.
Hubungan antara kelas Car dan Engine merupakan hubungan komposisi, dengan kelas Engine merupakan kelas "utama", sehingga jika objek dari kelas Engine dihilangkan, maka objek kelas Car juga akan hilang
- ☐ d.
Hubungan antara kelas Car dan Engine merupakan hubungan komposisi, dengan kelas Car merupakan kelas "utama", sehingga jika objek dari kelas Car dihilangkan, maka objek kelas Engine juga akan hilang

Question 15

Complete

Mark 10.00 out of 10.00

Berdasarkan gambar berikut, tuliskan access modifier bagi atribut email (tuliskan dalam huruf kecil)



Sumber: <https://www.chegg.com/homework-help/questions-and-answers/ql-consider-following-uml-class-diagram-address-person-streetnumber-integer-name-string-ci-q75218193>

Answer: protected

Question 16

Complete

Mark 10.00 out of 10.00

Perbedaan utama antara checked and unchecked exception adalah: checked exceptions diperiksa (di-cek) saat program di-compile (compile-time), sementara unchecked exceptions diperiksa saat runtime.

Select one:

- ☒ True
- ☐ False

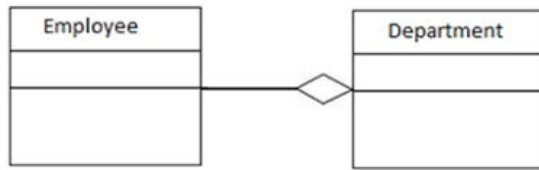
Summary

Question 17

Complete

Mark 5.00 out of 10.00

Berdasarkan gambar kelas diagram berikut, pilihlah pilihan yang sesuai...



Sumber: <https://www.c-sharpcorner.com/UploadFile/ff2f08/association-aggregation-and-composition/>

Select one or more:

- ☐ a. Pada kelas main, instansiasi objek bagi kelas Department maupun kelas Employee dilakukan
- ☒ b. Pada kelas main, instansiasi objek hanya dilakukan bagi kelas Department saja
- ☐ c. Pada kelas Employee akan terdapat atribut yang menerima objek Department
- ☒ d. Pada kelas Department akan terdapat atribut yang menerima objek Employee

Summary

Question 18

Complete

Mark 10.00 out of 10.00

Lengkapilah pernyataan berikut.

Berdasarkan kesepakatan yang digunakan pada perkuliahan PBO di D3 RPLA, terdapat 4 konsep object oriented programming, yaitu abstraction, polymorphism, dan encapsulation.

Noted: perhatikan, penulisan konvensi adalah dalam Bahasa Inggris

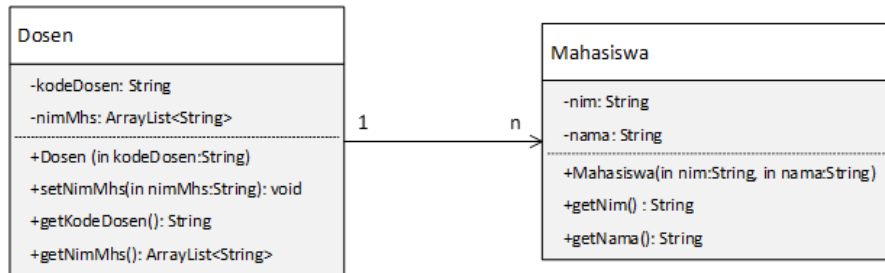
Answer: inheritance

Question 19

Complete

Mark 10.00 out of 10.00

Berdasarkan kelas diagram berikut, pilihlah pernyataan yang paling tepat!



Select one:

- ☒ a. Kelas diagram tersebut menunjukkan hubungan asosiasi navigable antara kelas Dosen dan Mahasiswa
- ☐ b. Kelas diagram tersebut menunjukkan hubungan komposisi antara kelas Dosen dan Mahasiswa
- ☐ c. Kelas diagram tersebut menunjukkan hubungan asosiasi non navigable antara kelas Dosen dan Mahasiswa
- ☐ d. Kelas diagram tersebut menunjukkan hubungan agregasi antara kelas Dosen dan Mahasiswa

Summary

Question 20

Complete

Mark 0.00 out of 10.00

Diketahui kode berikut:

```

import java.util.Scanner;
public class NewClass1 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        int x=8; int y = s.nextInt();
        if(y>x){
            System.out.println(y+x*2);
        }else{
            System.out.println(y*y/x);
        }
    }
}
  
```

Jika masukan dari pengguna adalah -4, keluaran aplikasi adalah

Answer: 16

Question 21

Complete

Mark 10.00 out of 10.00

Diketahui kode sebagai berikut:

```
public class NewClass4 {  
    public static void main(String[] args) {  
        int[] b = {7,0,6,9,5};  
        try {  
            int hasil = b[4]/b[1];  
            System.out.println(hasil);  
        } catch (ArrayIndexOutOfBoundsException e) {  
            System.out.println("error akses array");  
        } catch (ArithmeticException e) {  
            System.out.println("error aritmatika");  
        } catch (Exception e) {  
            System.out.println("eksepsi");  
        }  
    }  
}
```

Keluaran dari kode di atas adalah

Select one:

- ☐ a.
Compile Error
- ☐ b.
error akses array
- ☒ c.
error aritmatika
- ☐ d.
eksepsi

Summary

Question **22**

Complete

Mark 0.00 out of 10.00

Diketahui kode berikut:

```
import java.util.ArrayList;

public class NewClass2 {
    public static void main(String[] args) {
        ArrayList<Integer> al = new ArrayList<>();
        al.add(9);al.add(10);al.add(-1);al.add(2);
        al.add(-7);
        int sum;
        for(int x:al){
            sum = sum+x;
        }
        System.out.println(sum);
    }
}
```

Keluaran dari kode program di atas adalah

Select one:

- ☐ a.
Error Compile
- ☒ b.
Muncul exception ketika di-run
- ☐ c.
Tidak ada jawaban yang benar
- ☐ d.
-7
- ☐ e.
13

Summary

Question **23**

Complete

Mark 10.00 out of 10.00

Simbol yang digunakan pada agregasi di class diagram adalah hollow diamond

Select one:

- ☒ True
- ☐ False

Question **24**

Complete

Mark 10.00 out of 15.00

Diketahui kode bahasa pemrograman Java berikut:

```
public class Pegawai {  
    private String nama;  
  
    public Pegawai(String nama, double salary) {  
        this.nama = nama;  
    }  
}
```

Kode yang dapat membentuk objek dari class Pegawai adalah

Select one or more:

- ☐ a.
Pegawai p = new Pegawai("Peg 1", 3 millions);
- ☒ b.
Pegawai p = new Pegawai("Peg 1", 3000000);
- ☒ c.
Pegawai p = new Pegawai("Peg 1", 2500.00);
- ☐ d.
Pegawai p = new Pegawai("Peg 1", 2500.00f);
- ☐ e.
Pegawai p = new Pegawai("Peg 1");

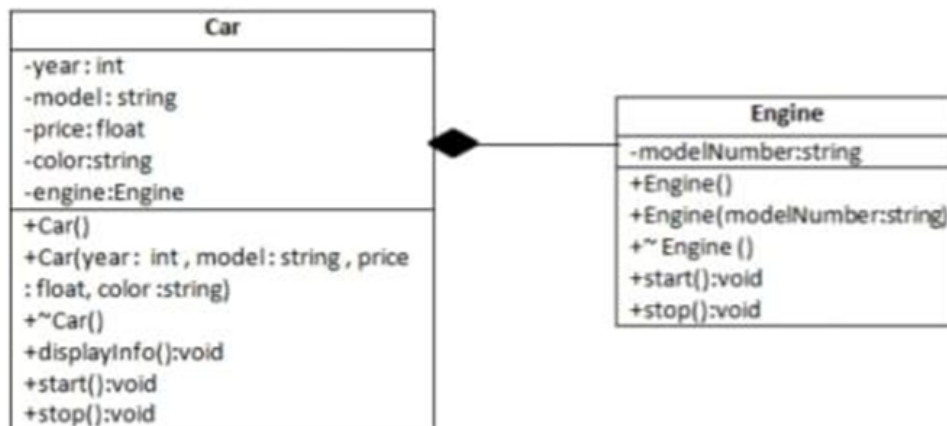
Summary

Question 25

Complete

Mark 15.00 out of 15.00

Berdasarkan gambar berikut, pilihlah implementasi kode yang sesuai



Select one:

Summary

☒ a.

```

public class Car {
    private int year;
    private String model;
    private float price;
    private String colour;
    private Engine engine;

    public Car(int year, String model, float price, String colour) {
        this.year = year;
        this.model = model;
        this.price = price;
        this.colour = colour;
    }

    public void addEngine(Engine engine) {
        this.engine = engine;
    }
}
  
```

```

public class Car {
    private int year;
    private String model;
    private float price;
    private String colour;
    private Engine engine;
}
  
```

```

public class Ass:
    public static Car daihatsu;
    public static Car daihatsu;
}
  
```

☐ b.

```

public class Car {
    private int year;
    private String model;
    private float price;
    private String colour;
    private Engine engine;

    public Car(int year, String model, float price, String colour) {
        this.year = year;
        this.model = model;
        this.price = price;
        this.colour = colour;
    }

    public void addEngine(Engine engine) {
        this.engine = engine;
    }
}

```

```

public class Engi
    private Strin

    public Engine
        this.mode
    }

```

```

public class AssessmentTe
{
    public static void ma
    Car daihatsu = ne
    Engine = new Engi
    daihatsu.addEngin
}

```

☐ c.

Summary

```

public class Engine {
    private String modelNumber;

    public Engine(String modelNumber) {
        this.modelNumber = modelNumber;
    }
}

```

```

public class Car {
    private int year;
    private String model;
    private float price;
    private String colour;
    private Engine engine;

    public Car(int year, String model, float price, S
        this.year = year;
        this.model = model;
        this.price = price;
        this.colour = colour;
        this.engine = engine;
    }
}

```

```

public class AssessmentTeori {

    public static void main(String[] args) {
        Car daihatsu = new Car(2010, "A1", 1500000000
    }
}

```

☐ d.

```
public class Car {  
    private int year;  
    private String model;  
    private float price;  
    private String colour;  
    private String engineNumb;  
  
    public Car(int year, String model, float price, String colour) {  
        this.year = year;  
        this.model = model;  
        this.price = price;  
        this.colour = colour;  
    }  
  
    public void setEngineNumb(String engineNumb) {  
        this.engineNumb = engineNumb;  
    }  
}
```

```
public class Engine {  
    private String modelN  
  
    public Engine(String  
        this.modelNumber  
    )  
  
    public String getMode  
        return modelNumbe  
    }  
  
}
```

```
public class AssessmentTeori {  
  
    public static void main(String[] args) {  
        Car daihatsu = new Car(2010, "A1", 1500000000, "Red");  
        Engine engine1 = new Engine ("21345432");  
  
        daihatsu.setEngineNumb(engine1.getModelNumber());  
    }  
}
```

Summary

[Previous Activity](#)[Jump to...](#)[Next Activity](#)