

Disclaimer !!

Jawabannya hanya untuk referensi saja. Jika sekiranya terdapat kesalahan, silahkan perbaiki sesuai preferensi masing-masing disesuaikan dengan materi pembelajaran kelas dan praktikum

Perangkat lunak bernama kuisus merupakan aplikasi untuk memfasilitasi seorang instruktur dan beberapa peserta dalam mempelajari suatu judul pelatihan. Instruktur mengunggah materi dan soal test, sedangkan peserta membaca materi dan mengerjakan test. Dalam satu judul pelatihan terdapat beberapa materi kuisus. Materi kuisus dapat berupa teori, praktikum, atau studi kasus. Aplikasi menghitung persentase nilai akhir peserta, menentukan tingkat kelulusan peserta, serta mencetak PDF sertifikat peserta.

1

1. Rancanglah sebuah kelas materi yang relevan berisi informasi id materi, konten, soal test disertai operasi yang relevan

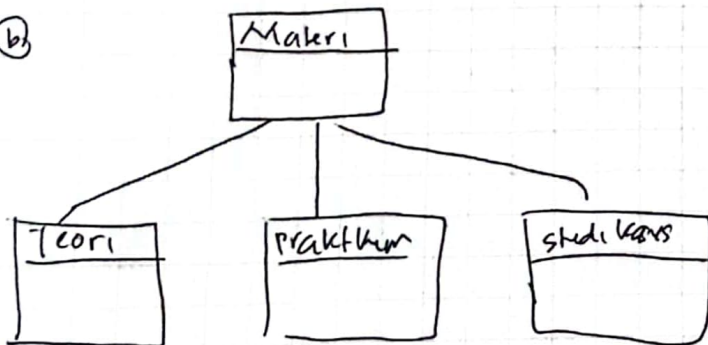
Materi

- idMateri : String
- konten : String
- soalTest : String

bisa dibuat sendiri dalam bahasa java nya untuk atribut dan metode

- + Materi ()
- + setIdMateri (idMateri : string)
- + setKonten (konten : string)
- + setSoalTest (soalTest : string)
- + getIdMateri () : string
- + getKonten () : string
- + getSoalTest () : string

2



```

public class Teori extends Materi {
    public Teori (String idMateri, String konten, String soalTest) {
        super (idMateri, konten, soalTest);
    }
}
    
```

```

Public class Praktikum extends Materi {
    Public Praktikum (String idMateri, String konten, String soalTest) {
        super (idMateri, konten, soalTest);
    }
}
    
```

```

public class studikasus extends Materi {
    public studikasus (string idmateri, string konten, string soaltest) {
        super (idmateri, konten, soaltest)
    }
}

```

}

② Buatkan deklarasi dan instansiasi setiap kelas anak dengan konsep Polimorfisme

```

public class main {
    public static void main (String[] args) {
        Materi teori = new teori ("kur101", "Analisa perangkat lunak dasar", "Apakah dasar pengembangan perangkat lunak");

        Materi praktikum = new praktikum ("kur201", "Modul 1 pembuatan DFD", "Buatlah DFD level 0");

        Materi studikasus = new studikasus ("kur301", "studikasus 1: implementasi aplikasi untuk proyek A", "Bagaimana flow yang digunakan berpengaruh terhadap proyek A");
    }
}

```

}

}

d) berikan contoh pengisian nilai dan cara akses informasi setiap instans kelas anak

- pengisian nilai

```
Teori.setIdMateri("KUP102");  
Praktikum.setMateri("KUP202");  
studiKasus.setIdMateri("KUP302");  
Teori.setKonken("...");  
Praktikum.setKonken("...");  
studiKasus.setKonken("...");  
Teori.setSoalTest("...");  
Praktikum.setSoalTest("...");  
studiKasus.setSoalTest("...");
```

- akses informasi

```
Teori.getIdMateri();  
Praktikum.getIdMateri();  
studiKasus.getIdMateri();  
Teori.getKonken();  
Praktikum.getKonken();  
studiKasus.getKonken();  
Teori.getSoalTest();  
Praktikum.getSoalTest();  
studiKasus.getSoalTest();
```



2. pada aplikasi kuis, seorang peserta yang mengikuti judul pelatihan memiliki koleksi nilai untuk setiap materi.
- a. rancanglah kelas list nilai dengan elemen yang berisi informasi id materi yang pernah dibaca, nilai list pada materi tersebut disertai operator yang relevan

```
public class Nilai {  
    private List<Nilai> nilaiList;
```

Asumsi sudah ada class  
nilai yang memiliki  
atribut idMateri dan  
nilaiTest.

```
    public List<Nilai>() {  
        this.nilaiList = new ArrayList<>();  
    }  
  
    public void addNilai(String idMateri, double nilaiTest) {  
        nilaiList.add(new Nilai(idMateri, nilaiTest));  
    }  
  
    public List<Nilai> getListNilai() {  
        return nilaiList;  
    }  
}
```

2

1. Buatlah deklarasi dan instantiasi

```
public class main {  
    public static void main(String[] args) {  
        ArrayList<String, real> listNilai = new ArrayList<String, real>();  
  
        listNilai.addNilai("KUR101", 87.3);  
        listNilai.addNilai("KUR201", 90.2);  
        listNilai.addNilai("KUR301", 92.3);  
    }  
}
```

2. Tuliskan algoritma fungsi untuk menghitung nilai total dari listNilai

```
public double hitungNilai() {  
    double total = 0;    double sum = 1  
    for (Nilai nilai : listNilai) {  
        total += nilai.getNilaiTest();  
        sum += 1;  
    }  
    return total / sum;  
}
```

3. Berikan contoh pengisian nilai dan cara akses informasi setiap elemen listNilai

- pengisian nilai

```
listNilai.addNilai("KUR103", 81.0);  
listNilai.addNilai("KUR203", 95.7);  
listNilai.addNilai("KUR303", 75.9);
```

- akses informasi tiap elemen

```
for (Nilai nilai : listNilai, getListNilai()) {  
    System.out.println("ID materi: " + nilai.getIDmateri() +  
        "Nilai test: " + nilai.getNilaiTest());  
}
```

⑥ Gambarkan pemetaan kelas listNilai ke dalam tabel relasi.

Setiap objek "nilai" dalam daftar listNilai akan menjadi sebuah record dalam tabel "Nilai".

contoh dalam SQL

```
CREATE TABLE Nilai (  
  idMateri VARCHAR(10) PRIMARY KEY,  
  nilaiTest FLOAT NOT NULL);
```

contoh data dalam Tabel Nilai

idMateri	nilaiTest
KUR103	81,0
KUR203	95,7
KUR303	75,9

3] SOLID

• SRP : Single Responsibility Principle

penerapan prinsip pertama dalam perancangan kelas-kelas dalam aplikasi KURSUS telah sesuai. Masing-masing kelas bertanggungjawab tunggal untuk mengerjakan task masing-masing. perhitungan nilai, kumpulan teori, praktikum dan studi kasus

• OCP : open closed principle

class materi terbuka untuk diwaniskan pada tiga kelas anaknya yaitu kelas teori, praktikum, dan studi kasus. Namun tertutup untuk dimodifikasi