

Einsendeaufgaben

Aufgabe 1: (4 Punkte)

Was sind Modelle? Welche Arten von Modellen gibt es? Aus welchen Teilen bestehen Modelle?

Aufgabe 2: (7 Punkte)

An einer Hochschule gibt es viele Vorlesungen, die in Hörsälen stattfinden. Jede Vorlesung hat eine Bezeichnung, jeder Hörsaal eine Nummer.

Erstellen Sie das Domänenmodell inklusive Attributen und Multiplizitäten.

Aufgabe 3: (9 Punkte)

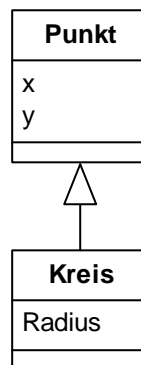
Manche Personen können an Vorlesungen teilnehmen, manche können Vorlesungen halten und manche können beides. An einer Vorlesung können viele Personen teilnehmen, es gibt aber immer nur einen Dozent.

Erstellen Sie das Domänenmodell inklusive Attributen und Multiplizitäten.

Welches Muster haben Sie wo und wie verwendet?

Aufgabe 4: (2 Punkte)

Welchen semantischen Mangel besitzt das folgende Klassendiagramm?



Aufgabe 5: (2 Punkte)

Mit welchen UML-Diagrammen können welche dynamischen Zusammenhänge im Entwurf am besten modelliert werden?

Aufgabe 6: (3 Punkte)

Wie wird sinnvollerweise beim Entwurf einer graphischen Benutzeroberfläche vorgegangen? Beschreiben Sie die wichtigsten Punkte des Ablaufs.

Aufgabe 7: (2 Punkte)

Gegen welche Grundsätze der Ergonomie wurde im folgenden Dialog wo verstoßen?



Aufgabe 8: (2 Punkte)

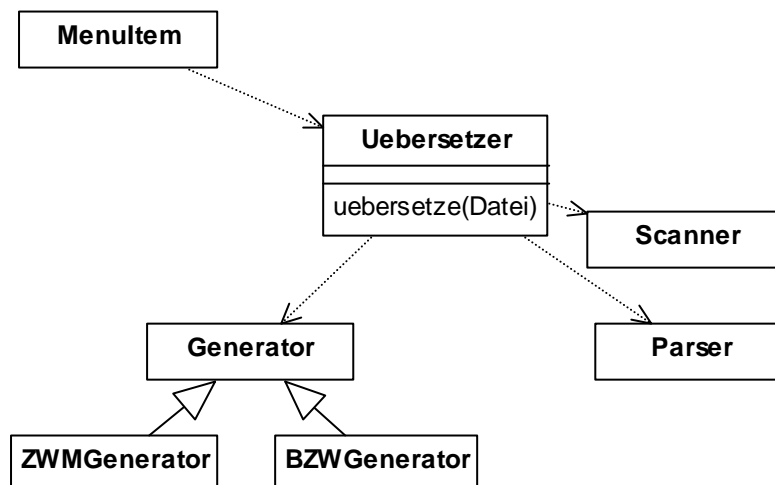
Welche Informationen werden für die Erstellung einer Software-Architektur mindestens benötigt?

Aufgabe 9: (2 Punkte)

Nennen Sie zwei Architekturmuster.

Aufgabe 10: (2 Punkte)

Welches Entwurfsmuster wurde hier wie verwendet?



Aufgabe 11: (3 Punkte)

```
public class Suche {
    List<Tier> tiere;
    //...
    /**
     * @return Instanzen von Tieren mit alter>=alter aus tiere
     */
    public List<Tier> mindestAlter(int alter){
        // wie vorgehen?
    }
}
```

Welches Muster setzen Sie in der Methode `mindestAlter()` ein und warum?

Aufgabe 12: (2 Punkte)

Was versteht man unter dem Begriff Kopplung?

Aufgabe 13: (4 Punkte)

Beurteilen Sie folgendes Code-Fragment aus Sicht des Software Engineerings. Wie könnte man die Probleme lösen?

```
public class Storage {
    private ArrayList<Integer> numbers;
    private Display display;
    public ArrayList<Integer> getNumbers() {
        return numbers;
    }
    public void addNumber(int i) {
        numbers.add(i);
        display.changed();
    }
    // ...
}

public class Display {
    private Storage storage;

    public void display() {
        Iterator<Integer> iter = storage.getNumbers().iterator();
        while (iter.hasNext()) {
            System.out.println(iter.next());
        }
    }
    public void changed() {
        //...
    }
    // ...
}
```

Aufgabe 14: (2 Punkte)

Was versteht man unter Refactoring?

Aufgabe 15: (4 Punkte)

Welche Typen von Zusammenhalt von Teilsystemen gibt es im Entwurf?