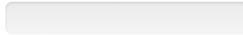
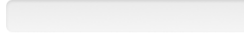


## Quizzies

Aktuelle Punktzahl: 0 / 8



Beantwortet: 0 / 9



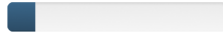
Welche der aufgeführten Aussagen sind korrekt?

- ☐ Eine hohe Kohäsion sollte möglichst vermieden werden.
- ☒ Kohäsion beschreibt, wie gut eine Programmeinheit (z.B. eine Klasse) eine logische Aufgabe oder Einheit abbildet.
- ☐ Lose Kopplung soll wenn immer möglich vermieden werden.
- ☒ Der Begriff Kopplung beschreibt den Grad der Abhängigkeiten zwischen Klassen.

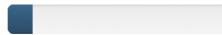
➔ 1 Punkt

## Quizzies

Aktuelle Punktzahl: 1 / 8



Beantwortet: 1 / 9



Eine Programmiererin hat eine Java-Anwendung mit folgenden Eigenschaften geschrieben:

- Jede Klasse ist weitgehend unabhängig von anderen Klassen
- Falls Klassen doch von anderen Klassen abhängig sind, kommunizieren sie über möglichst schmale, wohldefinierte Schnittstellen
- ☐ Über die Art der Kopplung kann keine Aussage gemacht werden
- ☐ Diese Java-Anwendung zeichnet sich durch eine starke Kopplung aus.
- ☒ Diese Java-Anwendung zeichnet sich durch eine lose Kopplung aus.

➔ 1 Punkt

Aktuelle Punktzahl: 2 / 8



Beantwortet: 2 / 9



Studieren Sie die Klassen A und B (Die Namensgebung entspricht nicht den Clean Code Regeln)

```
public class A {  
  
    private B b = null;  
  
    public A() {  
        b = new B(this);  
    }  
  
    public void run() {  
        b.run();  
    }  
  
    public void setzeErgebnisVonB(int ergebnis) {  
        // mache etwas  
    }  
}  
  
public class B {  
    private A a = null;  
  
    public B(A a) {  
        this.a = a;  
    }  
  
    public void run() {  
        int ergebnis = 1 + 1;  
        a.setzeErgebnisVonB(ergebnis);  
    }  
}
```

- ☐ Die Schnittstellen zwischen den Klassen A und B sind gut definiert.
- ☒ Die Klassen A und B sind stark gekoppelt.
- ☒ Der Grad der Abhängigkeiten der Klassen A und B ist hier nicht optimal gelöst.
- ☐ Über die Kopplung der Klassen A und B kann keine Aussage gemacht werden.
- ☐ Die Klassen A und B sind lose gekoppelt.

➔ 1 Punkt

## Quizzies

Aktuelle Punktzahl: 3 / 8



Beantwortet: 3 / 9



Welche der folgenden Antworten sind korrekt?

- ☒ Code-Duplizierung ist ein Hinweis für einen schlechten Entwurf.
- ☐ Code-Duplizierung hat keinen direkten Einfluss auf die Code-Qualität.
- ☒ Code-Duplizierung sollte unbedingt vermieden werden.
- ☐ Mit modernen Entwicklungsumgebungen wie Eclipse ist duplizierter Code kein Problem mehr.

➔ 1 Punkt

## Quizzies

Aktuelle Punktzahl: 4 / 8



Beantwortet: 4 / 9



Mit Code-Duplizierung erreicht man eine höhere Kohäsion.

- ☐ Wahr
- ☒ Falsch

➔ 1 Punkt

## Quizzies

Aktuelle Punktzahl: 5 / 8



Beantwortet: 5 / 9



Welche der folgenden Aussagen zur Klasse Brief sind korrekt?

```
public class Brief {  
  
    private String herr = "Sehr geehrter Herr ";  
    private String frau = "Sehr geehrte Frau ";  
  
    public String erstelleBriefFrau(String nachname, String briefText) {  
        return frau + nachname + "\n\n" + briefText;  
    }  
  
    public String erstelleBriefHerr(String nachname, String briefText) {  
        return herr + nachname + "\n\n" + briefText;  
    }  
  
}
```

- ☒ In der Klasse Brief hat es duplizierten Code.
- ☒ Die Klasse weist eine enge Kopplung auf.
- ☐ Wegen fehlender Code-Duplizierung kann hier die Kopplung gering gehalten werden.

➔ 0 Punkte

## Quizzies

Aktuelle Punktzahl: 5 / 8



Beantwortet: 6 / 9



Welche der folgenden Aussagen sind korrekt?

- ☒ Werden eine oder mehrere Instanzvariablen einer Klasse als public deklariert, verletzt dies das Prinzip der Kapselung.
- ☒ Sauber gekapselte Klassen reduzieren die Kopplung.
- ☐ Die Kapselung hat keinen Einfluss auf die Kopplung von Klassen.
- ☐ Sauber gekapselte Klassen erhöhen die Kopplung.
- ☐ Werden eine oder mehrere Instanzvariablen einer Klasse als private deklariert, so wird das Prinzip der Kapselung verletzt.

➔ 1 Punkt

Aktuelle Punktzahl: 6 / 8



Beantwortet: 7 / 9



Welche der folgenden Aussage sind korrekt?

```
public class Notenerfassung {  
  
    public String name;  
    public String vorname;  
    public String fach;  
    public int note;  
  
    public Notenerfassung(String name, String vorname,  
  
                           String fach, int note) {  
        this.name = name;  
        this.vorname = vorname;  
        this.fach = fach;  
        this.note = note;  
    }  
  
    // Weitere Methoden  
  
}
```

- ☐ Die Klasse Notenerfassung ist gut gekapselt.
- ☐ Die Klasse Notenerfassung ist teilweise gekapselt.
- ☒ Die Klasse Notenerfassung ist nicht gekapselt.

➔ 1 Punkt