

# Übung 7 – SE Grundlagen

---

## Aufgabe 1 (0 Punkte):

- Erstellen Sie eine neue Klasse mit einer main Methode.
- Erstellen Sie ein Programm, das 2 ganze Zahlen vom Benutzer einliest.
- Dividieren Sie beide Zahlen.
- Beachten Sie mögliche Fehler und passen Sie Ihr Programm so an, dass es eine Fehlermeldung ausgibt, sollte eine Division durch 0 erfolgen. Nutzen Sie für die Fehlerabfrage eine if Anweisung. Geben Sie den Fehler mittels System.err aus.

## Aufgabe 2 (0 Punkte):

- Erstellen Sie nun eine weitere Methode analog zu Aufgabe 1 aber nutzen Sie diesmal das try/catch Konstrukt.

## Aufgabe 3 (0 Punkte):

- Recherchieren Sie gängige Java Exceptions und erweitern Sie Ihr Programm aus Aufgabe 2 sinnvoll (Mindestens 3 weitere Exception Typen).

## Aufgabe 4 (0 Punkte):

- Erstellen Sie ein Programm, dass vom Nutzer 5 konkrete Eingaben erfordert (z.B.: Geben Sie eine positive Zahl ohne Kommastelle ein; Geben Sie einen Text mit mindestens 10 Zeichen ein; ...)
- Prüfen Sie diese Eingaben mittels des try/catch Konstrukts und geben Sie am Ende des Programms eine Auswertung über falsche Eingaben aus.

# Leistungsbeurteilung

---

Die folgenden Aufgaben sind zu erledigen und werden in der folgenden Übung bewertet.

Aufgabe 1 (1 Punkt):

Schreiben Sie auf einem Din A4 Blatt Papier folgende Konstrukte (jeweils in einer main Methode):

- Eine for-Schleife von 0 bis 10 in der eine for-Schleife von 0 bis 5 läuft
- Das selbe mit while-Schleifen statt for-Schleifen
- Einen try/catch/finally Block
- Aufruf einer rekursiven Methode, sowie die rekursive Methode, die alle Zahlen von 1 bis zum übergebenen Wert addiert.

Aufgabe 2 (1 Punkt):

Erstellen Sie in IntelliJ IDEA ein neues Projekt Uebung7\_PKZ. Schreiben Sie ein Programm, das vom Nutzer solange Eingaben einliest, bis es sich um eine Zahl handelt. Nutzen Sie hierfür das try / catch Konstrukt.

Aufgabe 3 (1 Punkt):

Schreiben Sie eine Methode checkEmail(String email), die prüft ob ein String eine Email Adresse ist.

Hierzu können Sie das Pattern / Matcher Konstrukt verwenden:

```
Pattern pattern = Pattern.compile( "[\\w|-]+@\\w[\\w|-]*\\. [a-z]{2,3}" );  
Matcher m = pattern.matcher(email);  
m.find();
```

Diese Methode soll mittels throws Konstrukt (Im Methodenkopf von checkEmail) Exceptions an die Aufrufende Methode (main) weiterleiten.

Aufgabe 4 (1 Punkt):

Erstellen Sie eine eigene Exception die eine eigene Fehlermeldung ausgibt. Werfen Sie diese Exception mittels eines Testprogramms.

Aufgabe 5 (Exceptions) (1 Punkt):

Erarbeiten Sie den Unterschied zwischen checked und unchecked Exceptions. Erstellen Sie ein Beispiel, mit dem sich der Unterschied demonstrieren lässt und kommentieren Sie wo angebracht.