

Umfrage Lernerfolge 2018

Liebe Lerngemeinschaft, wir freuen uns sehr über euer Feedback zur Verbesserung unserer zukünftigen Kurse. Der Fragebogen besteht aus 16 Fragen (6 allgemeine Fragen zur Einordnung und 10 Wiederholungsfragen zum Kursinhalt) und hilft uns dabei festzustellen, welche Themen wir in zukünftigen Kursen vertiefen sollten. Zudem hilft euer Feedback uns bei unserer Forschung, die sich mit Lernerfolgen in MOOCs beschäftigt. Wir hoffen, dass der Fragebogen einen kleinen Wiederholungseffekt hat und auch Spaß beim Knobeln bietet. Vielen Dank für euer Engagement und tatkräftige Mithilfe!

Frage 1

Hattest du vor dem Kurs bereits Programmiererfahrung?

Ja

Ja, aber sie lag so weit zurück, dass ich vieles wieder neu gelernt habe

Nein

Frage 2

Falls du vor dem Kurs bereits Programmiererfahrung hattest, wo hast du programmieren gelernt?

In einem online Programmierkurs (z.B. openHPI, coursera, udacity, edX)

Uni- bzw. Schulkurse

Durch meine Arbeit

Selbststudium (Bücher und anderes...)

Frage 3

Wie schätzt du deine aktuellen Programmierfertigkeiten ein?

Geringe bis keine Programmierkenntnisse (mehr) vorhanden

Grundlegende Programmierkenntnisse

Gute Programmierkenntnisse

Sehr gute Programmierkenntnisse

Ausgezeichnete Programmierkenntnisse

Frage 4

Hast du seit Kursende programmiert?

Nein

Ja, in meiner Freizeit (Hobby/Projekte)

Ja, beruflich (Arbeit)

Ja, in meiner Ausbildung (Schule, Studium)

Ja, im Selbststudium (online Kurse, Bücher, ...)

Frage 5

Was hat dir rückblickend besonders gut im Kurs gefallen?

Die Videos

Die praktischen Programmieraufgaben

Die Selbsttests

Die Story

Das Peer Assessment

Die Diskussionen im Forum

Frage 6

Was sind gültige Schlüsselwörter in Java?

Hinweis: Es können mehrer Antworten korrekt sein.

public

common

output

void

back

Frage 7

Welche der folgenden Logischen Ausdrücke wird zu wahr (true) ausgewertet? Dies könnte beispielsweise in einer if-Bedingung geschehen. Gegeben sei weiterhin die Variable i vom Typ int mit dem Wert 3:

`int i = 3;`

true

true && false

true || false

i == 3

(i <= 3 && true) || (i >= 3 && false)

Frage 8

Wie oft gibt das folgende Programm "I love Java!" aus?

```
1 public class ILoveJava {
2
3     public static void main(String[] args) {
4         for(int i=0; i<=3;i++){
5             System.out.println("I love Java!");
6         }
7     }
8 }
```

0

1

2

3

4

5

unendlich oft

Frage 9

Welche (Haupt-) Aufgabe hat ein Interface?

Es ist ein Framework zur Erstellung von Nutzeroberflächen

Es verbessert die Performance durch die bessere Nutzung von Compileroptionen

Es erlaubt die Deklaration von Methoden die implementiert werden müssen

Es ermöglicht zwei verschiedene Objekte miteinander zu verbinden; Aufrufe auf einem Objekt werden auch auf dem anderen Objekt ausgeführt

Es erlaubt die Einschränkung von Sichtbarkeiten von Methoden

Frage 10

Was ist Polymorphie in der objektorientierten Programmierung?

Das Umbenennen eines Objekts

Das dynamische Auswählen der aufzurufenden Implementierung (dynamisches Binden) in einer Vererbungshierarchie bei mehreren Methodenimplementierungen mit identischer Methodensignatur

Ein Aufruf einer statischen Methode innerhalb einer Vererbungshierarchie mit undefinierter Methodensignatur

Das Erzeugen (Instanzieren) von mehreren Objekten zur Laufzeit

Frage 11

Wie kann man eine Vererbungsbeziehung im Code anlegen?

Indem man das Schlüsselwort `extends` bei der Klassendeklaration nutzt.

Indem man Kind- und Elternklasse mit `child` und `parent` kennzeichnet.

Indem man `superclass` anstelle von `class` schreibt.

Es muss zunächst eine Stammbuch-Klasse (`class Register`) angelegt werden. In dieser wird Java mitgeteilt, welche Klassen von welchen erben.

Gar nicht. Java entscheidet dies auf Grund der Semantik (sprachliche Bedeutung).

Frage 12

Welche der folgenden Aussagen über abstrakte Klassen sind korrekt?

Hinweis: Es können mehrere Antworten korrekt sein.

Jede Klasse ist abstrakt, da sie nur einen Bauplan für die Objekte die aus ihr instanziiert werden sollen darstellt.

Man legt eine abstrakte Klasse an, indem man das Schlüsselwort `abstract` bei der Klassendefinition vor das Schlüsselwort `class` schreibt.

Abstrakte Klassen können nicht instanziiert werden.

Man legt eine abstrakte Klasse an, indem man statt `class` `ClassName` `abstractClass` `ClassName` schreibt.

Frage 13

Füge die fehlende if-Verzweigung im unten stehenden Code Beispiel ein. In dem Beispiel sollen alle positiven Zahlen aufsummiert werden. Das Ergebnis, dass bei korrekter Berechnung ausgegeben werden soll, ist 42.

```
1  public class FillInTheMissingLine {
2
3      public static void main(String[] args) {
4          int[] array = {2, 7, -3, 20, 13, -12, 0};
5          int arraySum = 0;
6          for(int i = 0; i < array.length; i++){
7              //fill in the correct statement here
8              arraySum = arraySum + array[i];
9          }
10     }
11     System.out.println(arraySum);
12 }
13
14 }
```

if(i >= 0){

if(i < 0){

if(array[i] >= 0){

if(array[i] < 0){

if(array[i+1] >= 0){

if(array[i+1] < 0){

Frage 14

Im folgenden Programm sollen exakt alle ungeraden Zahlen von 1 bis 10 ausgegeben werden. Füge die fehlende Zeile Code ein (for-Schleife). Die Ausgabe bei korrekter Ausführung wäre also die Zahlen: 1, 3, 5, 7 und 9.

Hinweis: Es können mehrere Antwortmöglichkeiten korrekt sein.

```

1  public class OddNumbers {
2
3      public static void main(String[] args) {
4          //fill in the correct statement here
5              if(i%2==1){
6                  System.out.println(i);
7              }
8          }
9      }
10 }

```

for(int i = 0; i < 10; i++){

for(int i = 1; i < 10; i+=2){

for(int i = 0; i < 10; i+=2){

for(int i = 0; i / 2 == 1; i++){

for(int i = 0; i%2 < 10; i++){

for(int i = 0; i%2==1; i++){

Frage 15

Gegeben sei das folgende kleine Programm. Wie lautet die Ausgabe?

(Aus Gründen der Übersicht sind die Zahlen per Kommata getrennt statt per Zeilenumbruch)

```

1  public class WhatDoesItDo {
2
3      static int result = 0;
4
5      public static void main(String args[]){
6          for (int i = 0; i < 5; i++){
7              System.out.println(count(i));
8          }
9      }
10
11     static int count(int number){
12         return result += number;
13     }
14 }

```

0, 0, 0, 0, 0

0, 1, 2, 3, 4, 5

0, 1, 2, 3, 4

1, 1, 2, 3, 5. Die ersten 5 Zahlen der Fibonacci Folge.

0, 1, 3, 6, 10

Das Programm gibt nichts aus. Es wirft einen Fehler.

0, 1, 3, 6, 10, 15

Frage 16

Was können wir noch verbessern?

Welche Lernmaterialien haben euch gefehlt?

Was für Kurse hättet ihr gerne auf openHPI?

Was wolltet ihr uns immer schon mitteilen?