

Programmieren 1

SW3 – Selbststudium

Kapitel 2.1

1. zu bearbeitende Aufgaben: 2.1 bis 2.5

→ TicketMachine vergleicht die Eingabe von Geld nicht mit dem Ticketpreis. Ein Ticket wird auch gedruckt, wenn zuwenig Geld eingelassen wurde. Rückgeld wird nicht erstattet.

Kapitel 2.3

2. zu bearbeitende Aufgabe: 2.10

3. Aus welchen wesentlichen 3 Komponenten ist eine Klasse aufgebaut?

→ Konstruktor

→ Variablen

→ Methoden

4. Wie ist deren Reihenfolge?

→ Die Reihenfolge spielt keine Rolle. Wichtig ist nur das Vorhandensein des Konstruktors – selbst wenn dieser leer ist.

5. Was ist deren Aufgabe/Zweck?

→ Der Konstruktor wird aufgerufen, wenn das Objekt instanziiert wird. Er übernimmt gewissermaßen die Initialisierung des Objekts.

→ Variablen können konkrete Werte (Eigenschaften) eines Objekts speichern.

→ Methoden implementieren Funktions-/Verhaltensweisen von Objekten.

Kapitel 2.3.1

6. zu bearbeitende Aufgaben: 2.11 bis 2.15

private int count; → Typ „int“

private Student representative; → Typ „Student“

private Server host; → Typ „Server“

→ Das Semikolon am Ende einer Zeile muss zwingend vorhanden sein!

7. Was versteht man unter einer Variablen?

→ Ein Name für einen Speicherbereich, welcher verschiedene Werte aufnehmen kann.

[http://de.wikipedia.org/wiki/Variable_\(Programmierung\)](http://de.wikipedia.org/wiki/Variable_(Programmierung))

8. Zählen Sie Synonyme für "Instanzvariable" auf?

→ Objektvariable

→ „Globale“ Variable

9. Worauf gründet wohl die Bezeichnung "Instanzvariable"?

→ Die Variable gehört zu einer bestimmten Instanz eines Objekts.

10. Wie kann man in Java Source-Code kommentieren?

→ Zeilenweise mit //

→ Abschnittsweise mit /* und */

11. Wichtige Frage! Mit welchem Zugriffsmodifizierer deklariert man normalerweise Instanzvariablen (mit public oder private)? Haben Sie eine Erklärung für Ihre Antwort?

→ Instanzvariablen werden in der Regel mit dem Modifizierer „private“ versehen. Mit dieser Massnahme wird verhindert, dass eine Instanzvariable eines Objekts von ausserhalb des Objekts unkontrolliert überschrieben werden kann. Das Objekt muss Methoden zur Verfügung stellen, welche das Setzen/Auslesen von Instanzvariablen erlauben.

Kapitel 2.3.2

12.

Erklären Sie den Zusammenhang zwischen Konstruktor und Objektzustand.

→ Ein Konstruktor initialisiert das Objekt bei der Instanzierung. Das Objekt wird also vom Konstruktor in einen Initialzustand versetzt.

13.

Wie benennt man Konstruktoren?

→ Modifier „public“ + kein Rückgabewert + Name der Klasse + Allfällige Parameter

→ Beispiel:

```
public class Student
{
    String inputString;
    public Student(String input)
    {
        inputString = input;
    }
}
```

14.

Wie lange sind die Instanzvariablen eines Objektes verfügbar/ansprechbar?

→ Solange das Objekt besteht.

15.

Wieso soll man Instanzvariablen nach Möglichkeit explizit initialisieren?

→ Weil man sich nie auf die Standardinitialisierung (int mit 0, float mit 0.0, bool mit false, usw...) verlassen sollte. Ein Programmierer muss sich auf sichere Werte verlassen können.

16.

Mit welchem Default-Wert wird eine Instanzvariable vom Datentyp int automatisch initialisiert?

→ 0

Kapitel 2.4

17.

zu bearbeitende Aufgaben: 2.16 bis 2.18

18.

Wozu gebraucht man Parameter?

→ Methoden (im Speziellen auch Konstruktoren) nehmen via Parameter die Eingabewerte für weitere Verarbeitungen entgegen.

19.

Erklären Sie den Unterschied zwischen einem formalen und einem aktuellen Parameter?

→ Mit „formaler Parameter“ ist die Variable des Parameters gemeint.

→ Als „aktueller Parameter“ wird der Wert dieser Variable bezeichnet.

20.

Stimmt folgende Aussage: "Formale Parameter sind spezielle Variablen"?

→ Innerhalb einer bestimmten Methode unterscheiden sich Parameter nicht von lokalen Variablen. Ihre Lebenszeit ist genau so beschränkt wie diejenige der lokalen Variablen.

21.

Was gilt betreffend der Sichtbarkeit von formalen Parametern?

→ Formale Parameter sind nur innerhalb der Methode sichtbar. Dasselbe gilt für methoden-lokale Variablen.

22.

Inwiefern ist dies anders bei Instanzvariablen?

→ Methoden können auf Instanzvariablen zugreifen – nicht aber auf Variablen von anderen Methoden.

23.

Wie verhält es sich analog mit der Lebensdauer von formalen Parametern bzw. von Instanzvariablen?
→ Verglichen mit Instanzvariablen leben Parametervariablen nur solange wie die Methode ausgeführt wird.

→ Instanzvariablen hingegen leben solange das Objekt besteht.

Kapitel 2.5

24.

zu bearbeitende Aufgaben: 2.19 bis 2.20

25.

Wie übersetzen Sie "Assignment" und "Expression" ins Deutsche?

→ Assignment bedeutet die Zuweisung. Sie erfolgt stets über das mathematische Gleichheitszeichen „=" . Beispielsweise die Zuweisung eines Wertes zu einer Variablen: `int i = 5;`

→ Unter Expression wird ein berechnender Ausdruck – auf der rechten Seite des Gleichheitszeichens – verstanden, welcher durch ein Assignment einer Variablen zugewiesen wird.

26.

Wie funktioniert die Assignment-Anweisung genau? Was ist betr. Datentypen zu beachten?

→ Beispiel: `i = 5 * (4 + i)`

→ Das Format des zugewiesenen Ausdrucks hängt vom Datentyp ab und muss zwingend korrekt sein.