Stellen Sie sicher, dass Sie die aufgelisteten Vokabeln aktiv verwenden und folgende Fragen richtig beantworten können.

**Liste der Fachbegriffe:**

|  |  |
| --- | --- |
| **Aufruf** | Aufrufen bezeichnet das benutzen von Methoden. Zum Beispiel ist println() der Aufruf einer Methode vom Objekt out. |
| **Parameter** | Parameter sind Werte die beim Schreiben einer Methode mit angegeben werden können. Sie werden im Methoden Kopf in den Klammern definiert. |
| **Argument** | Argumente sind im Gegensatz zu Parametern richtige Werte und werden beim Aufruf der Methode übergeben. Man schreibt sie in die Klammer beim Methodenaufruf |
| **Rückgabetyp** | Der Rückgabetyp einer Methode wird im Kopf definiert und kann entweder ein Objekt, ein primitiver Datentyp oder void (nichts) sein. |
| **Signatur** | Die Zusammensetzung aus Methodenname und Parameterliste. |
| **Scope** | Der Scope ist der Gültigkeitsbereich eine Methode und/oder Variable |
| **call-by-value** | Parameterübergabe per Wert |
| **call-by-reference** |  |

|  |  |
| --- | --- |
| **Klasse** | Eine Klasse ist eine Kompilationseinheit wo methoden und variablen definiert werden können |
| **Objekt** | Ein Objekt ist eine Instanz einer Klasse und hat eigene werte die unterschiedlich zu den werten von objekten der gleichen Klasse sind |
| **Instanz** | Erzeugung eines konkreten Objektes |
| **Eigenschaft** |  |
| **Attribut** | Die Eigenschaften eines Objekts |
| **Methode** | Unterprogramme in Form von Funktionen oder Prozeduren |
| **private** | Methode oder Variable ist nur in der Klasse sichtbar |
| **public** | Methode oder Variabel ist überall sichtbar |
| **Sichtbarkeit** | Wo man auf die Methode oder Variable zugreifen kann |
| **Datenkapselung** | Bezeichnung für das Verbergen von Daten oder Infromationen vor Zugriff von außen |
| **this** | Referenzierung der momentanen klasse |

**Kontrollfragen**

**Kontrollfragen zu Methoden:**

1. Wofür sind Methoden gut?
2. Was ist der Unterschied zwischen einem Parameter und einem Argument?
3. Was ist der Unterschied zwischen der Implementation (Definition) einer Methode und einem Aufruf?
4. Wozu dient eine Rückgabe?
5. Mit was für einem Buchstaben sollten Methoden beginnen?

**Kontrollfragen zu Objektorientierung:**

1. Was ist ein anderer Name für einen selbst definierten Datentyp?
2. Auf welche Variablen kann man in Methoden zugreifen, ohne, dass sie lokal deklariert werden oder als Parameter zur Verfügung stehen?
3. Was beginnt immer mit einem Großbuchstaben?
4. Welche OO-Begriffe entsprechen:  
   - einem Backrezept?  
   - einem Kuchen?  
   - dem Zuckergehalt des Kuchens?
5. Wie viele Objekte kann man mit Hilfe einer Klasse erzeugen?
6. Wieso soll man Eigenschaften mit private deklarieren?
7. Wo kann man die Eigenschaften einer Klasse überall verwenden?
8. Warum benötigt man setter?
9. Warum benötigt man getter?
10. Was benötigt man in main, damit man eine Methode verwenden kann?
11. Wie realisiert man Datenkapselung?