



Wiederholungs-Montag

09. Dezember 2024

Ankündigungen



Kurztest 20 Minuten am **05.12.2024**



Tests a 60 Minuten über alle
bisherigen Inhalte:

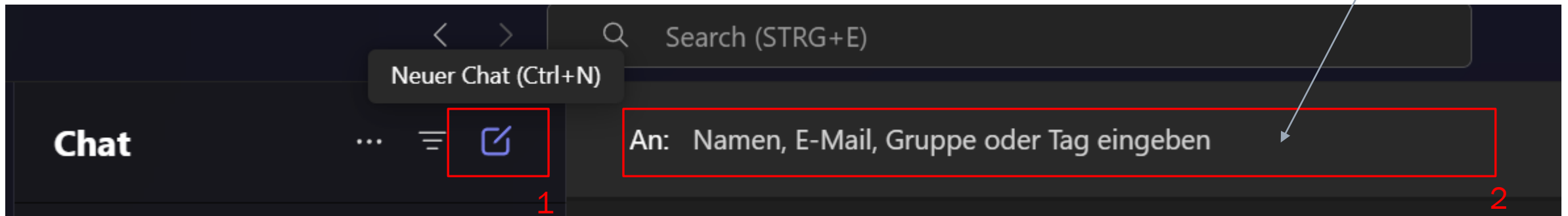
10.12.2024

17.12. 2024

Ankündigungen

Was tun, wenn ich krank bin?

Lisa Klimczak UND Alexandra
Bobenhausen





Menti

Plan für heute

- static
- Konstanten
- Ternäre Operator
- Schleifen
- Wrapper und Casten



Wieso wird das Schlüsselwort „static“ verwendet?

FRAGE/DISKUSSION



Quiz

Which of the following statements are true about Java static methods?

Choose 2 correct answers:

- a) Can be invoked using the class name, without having to create an instance of the class
- b) Can call other static methods directly
- c) Can use the this keyword to refer to instance methods and member variables
- d) Must be declared as public

static

- Arbeiten auf Klassenebene statt auf Instanzebene
- Statische **Variablen**:
 - Gehört zur Klasse nicht zu Objekten
- Statische **Methoden**:
 - Können ohne ein Objekt zu erstellen über die Klasse abgerufen werden
 - Haben keinen Zugriff auf Instanzvariablen oder -methoden, außer über ein Objekt.
- Statische **Blöcke**:
 - Ein statischer Block wird einmal beim Laden der Klasse ausgeführt und dient zur Initialisierung statischer Variablen.

Konstanten (constant)

- **Unveränderliche** Variable, deren Wert nach der Initialisierung nicht mehr geändert werden kann
- Häufig verwendet, um feste Werte wie PI, Maximalgrößen oder Konfigurationsdaten zu speichern
- Benennung i.d.R. in GROSSBUCHSTABEN und Unterstrich geschrieben
- mit **static**: Die Konstante gehört zur Klasse und wird nicht für jede Instanz separat gespeichert.
- mit **final**: Der Wert der Variablen kann nach der Zuweisung nicht mehr geändert werden.

Konstanten (constant)

```
public class Circle { 1 usage
    public static final double PI = 3.14159; 1 usage

    public double calculateCircumference(double radius) { 1 usage
        return 2 * PI * radius;
    }
}
```

Aufgabe



Erstelle ein Programm, das die Fläche eines Kreises und eines Rechtecks berechnet. Definiere die folgenden **Konstanten in einer eigenen Klasse**:

- Der Wert von PI für die Berechnung der Kreisfläche.
- Die Länge und Breite für die Berechnung der Fläche eines Rechtecks.

Verlagere die **Berechnungsmethoden in eine separate Klasse** und stelle sicher, dass die Konstanten in einer eigenen Klasse zentral verwaltet werden. Die Berechnungen sollen mit den Konstanten aus der separaten Klasse erfolgen.

- Die Formel für die Fläche des Kreises ist: $A = PI * r^2$ (wobei r der Radius des Kreises ist).
- Die Formel für die Fläche des Rechtecks ist: $A = \text{Länge} * \text{Breite}$.

Verwende die static final Deklaration für die Konstanten, um sicherzustellen, dass ihre Werte nach der Initialisierung nicht mehr verändert werden können.

Bedingungsoperator

Ternärer Operator (ternary conditional operator) ?:

– Verkürzung des if-else-Statements

Bedingung ? Ausdruck1 : Ausdruck2;

-> Ausdruck1: Falls die Bedingung true

-> Ausdruck2: Falls die Bedingung false

Vergleich if-else

```
if (Bedingung) {
```

```
    // Code, wenn die Bedingung wahr ist
```

```
} else {
```

```
    // Code, wenn die Bedingung falsch ist
```

```
}
```

```
Bedingung ? Ausdruck1 : Ausdruck2;
```



Aufgabe

Schreibe ein Programm, das eine Zahl überprüft und angibt, ob sie gerade oder ungerade ist. Verwende sowohl den ternären Operator als auch die if-Anweisung, um dies zu tun.

Die Ausgabe soll folgendermaßen aussehen:

„Die Zahl [Zahl] ist gerade.“ oder „Die Zahl [Zahl] ist ungerade.“

- Erstelle dazu zwei Methoden: eine, die den ternären Operator verwendet, und eine, die eine if-Anweisung verwendet.
- Vergleiche die beiden Ansätze und erkläre, wann der ternäre Operator sinnvoll eingesetzt wird. (In einem Kommentar)



Wann sollten ternäre Operatoren verwendet werden?

FRAGE/DISKUSSION

Vergleich der Schleifen

Schleife	Wann verwendet?
for	Für bekannte Iterationen oder Zählerbasierte Schleifen.
foreach	Wenn über Sammlungen oder Arrays iteriert werden soll, ohne Indizes.
while	Wenn die Anzahl der Iterationen nicht bekannt ist, aber eine Bedingung gegeben ist.
do-while	Wenn der Code mindestens einmal ausgeführt werden muss.



Info: Kopfgesteuerte Schleifen prüfen die Bedingung vor der Ausführung, fußgesteuerte nach der ersten Ausführung (z.B. do-while).



Aufgabe

Schreibe zwei Methoden, die die Zahlen von 1 bis 10 sowohl mit einer kopfgesteuerten als auch mit einer fußgesteuerten Schleife ausgibt.

- Verwende eine for-Schleife für die kopfgesteuerte Schleife
- Verwende eine do-while-Schleife für eine fußgesteuerte Schleife
- Vergleiche die beiden Ansätze und erkläre, wann welche Schleife sinnvoll eingesetzt wird. (In einem Kommentar)

Methoden der Wrapper-Klassen

- **parseInt(String s):** Wandelt eine String-Repräsentation eines primitiven Datentyps (z. B. int) in den entsprechenden Wert um.

```
int num = Integer.parseInt("123");
```

- **valueOf(String s):** Wandelt eine String-Repräsentation in das entsprechende Wrapper-Objekt um.

```
Integer num = Integer.valueOf("123");
```

- **toString():** Gibt den Wert des Wrapper-Objekts als String zurück.

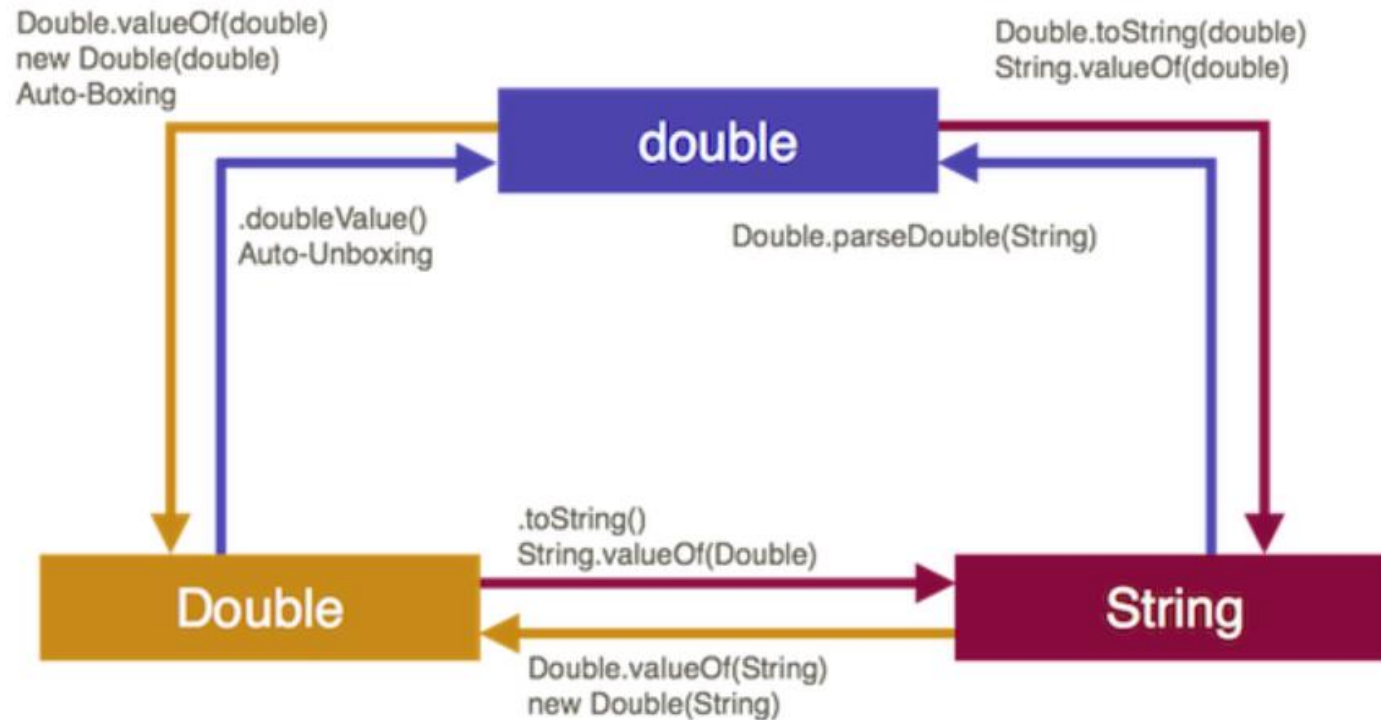
```
Integer num = 10;  
System.out.println(num.toString());
```

- **compareTo():** Vergleicht zwei Wrapper-Objekte.

```
Integer num1 = 10;  
Integer num2 = 20;  
System.out.println(num1.compareTo(num2));
```

Methoden der Wrapper-Klassen

Beispiel:





Quiz

Was ist der Unterschied zwischen einem primitiven Datentyp und einer Wrapper-Klasse?

- a) Wrapper-Klassen sind keine Objekte und bieten keine zusätzlichen Funktionen.
- b) Primitive Datentypen sind Objekte und Wrapper-Klassen bieten zusätzliche Methoden.
- c) Primitive Datentypen sind einfache Werte, während Wrapper-Klassen Objekte sind, die diese Werte umhüllen und zusätzliche Methoden bieten.
- d) Es gibt keinen Unterschied, sie sind dasselbe.

Implizites Casten (widening casting)

- Daten eines kleineren Datentyps werden durch den Compiler **automatisch** dem größeren angepasst

byte -> short -> char -> int -> long -> float -> double

```
int num = 10;  
double doubleNum = num;
```

Explizites Casten (narrowing casting)

- Wechsel von einem größeren in einen kleineren Datentyp
- Es ist eine explizite Anweisung erforderlich
- Es können dabei Daten verloren gehen!

double -> float -> long -> int -> char -> short -> byte

```
double doubleNum = 9.99;  
int intNum = (int) doubleNum;
```

Casten von Objekten

- Umwandlung eines Objekts von einem Typ in einen anderen, der eine Oberklasse oder Unterklasse ist

Upcasting:

Ein Objekt einer Unterklasse wird in ein Objekt der Oberklasse umgewandelt. Dies geschieht automatisch.

Downcasting:

Ein Objekt einer Oberklasse wird in ein Objekt einer Unterklasse umgewandelt.



Aufgabe

Stell dir vor, du hast ein System zur Verwaltung von Produktpreisen in einem Online-Shop. Der Preis eines Produkts wird als `double` gespeichert, aber du musst auch mit verschiedenen Wrapper-Klassen arbeiten, um mit der Datenbank zu interagieren (die Werte als `Double` speichert) und später mit den Produktpreisen Rechnungen zu erstellen, die auf `int` basieren.

1. Erstelle eine Methode, die einen Produktpreis als `double` empfängt, diesen in ein `Double` (Wrapper-Klasse) umwandelt und die Mehrwertsteuer (z.B. 19%) darauf berechnet.
2. Erstelle eine Methode, die den Mehrwertsteuerbetrag als `Double` empfängt, diesen in den primitiven Datentyp `double` umwandelt und den Endpreis des Produkts berechnet (Preis inkl. MwSt.).
3. Erstelle eine weitere Methode, die einen `Double` Preis empfängt, diesen in `int` umwandelt und den Endpreis in Cent zurückgibt.

Quellen

<https://pixabay.com/de/vectors/pfeil-kreislauf-recyceln-159146/>