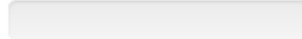
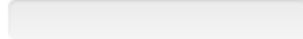


Quizzies

Aktuelle Punktzahl: 0 / 5



Beantwortet: 0 / 6



Ein Array ist eine Sammlung fixer Grösse.

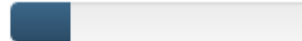
☒ Ja

☐ Nein, in Java nicht

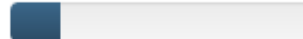
➔ 1 Punkt

Quizzies

Aktuelle Punktzahl: 1 / 5



Beantwortet: 1 / 6



Folgender Beispielcode erzeugt ein int-Array der Grösse 3.

```
int[] zahlen = new int[3];  
zahlen[2] = 2;
```

Der Inhalt des Arrays ist nach Ausführung des Codes wie folgt:

- Erstes Element: 0
- Zweites Element: 0
- Letzes Element: 2

☐ Nein

☒ Ja

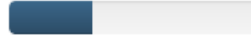
➔ 1 Punkt

Quizzies

Aktuelle Punktzahl: 2 / 5



Beantwortet: 2 / 6



Welche Aussagen treffen auf den folgenden Code zu?

```
int[] messreihe = wetterstation.getTemperaturmessungen();
int resultat = 0;
for(int i = messreihe.length-1; i >= 0; i--){
    resultat += messreihe[i];
}
resultat = resultat/messreihe.length;
```

- ☒ Dieses Codestück berechnet den Mittelwert der Temperaturmessreihe auf Ganzzahl genau.
- ☒ Es wird eine Sammlung mit variabler Grösse verwendet.
- ☐ Eine for-Schleife (oder auch eine for-each-Schleife) ist für diese Aufgabe besser geeignet als eine while-Schleife.
- ☐ Die bei der Deklaration der for-Schleife deklarierte Variable i ist nur innerhalb der for-Schleife verwendbar.
- ☐ Hier wäre besser eine while-Schleife verwendet worden.

➔ 0 Punkte

Quizzies

Test beenden

Test unterbrechen

Aktuelle Punktzahl: 2 / 5



Beantwortet: 3 / 6



Der Zugriff auf Elemente in einem Array gilt als sehr effizient.

```
Auto[] autos = new Auto[3];
autos[0] = new Auto();
autos[1] = new Auto();
autos[2] = new Auto();
```

Dies ist so, weil die Speicherposition der zu einem Index *i* gehörenden Auto Objekt im Falle von Arrays berechnet werden kann. Somit ist bei bekanntem Index ein direkter Zugriff auf die auf das Objekt zeigende Referenz möglich. Dies funktioniert wie folgt: Die Variable *autos* ist eine Referenz auf den Anfang des Array. Bekanntlich speichert ein Array von Objekten nicht die Objekte sondern lediglich Referenzen auf diese Objekte. Der Speicherbedarf einer Referenz ist bekannt. Da diese Referenzen im Speicher hintereinander liegen, kann die Speicherposition der Referenz für das Objekt mit Index *i* gemäss folgender Formel berechnet werden: $\text{Speicherradresse}(\text{autos}) + (i * \text{Speicherbedarf einer Referenz})$.

- ☐ Nein
- ☒ Ja

➔ 1 Punkt

Quizzies

Test beenden

Test unterbrechen

Aktuelle Punktzahl: 3 / 5



Beantwortet: 4 / 6



Sie haben eine ArrayList list vom Typ `ArrayList<Integer>` vor sich, aus der Sie alle Elemente entfernen wollen, deren Wert grösser ist als 100. Welche Schleifenköpfe eignen sich für diese Aufgabe?

- ☒ `Iterator<Integer> it = list.iterator();`
`while(it.hasNext())`
- ☐ `for(int i=0; i < list.size(); i++)`
- ☐ `for(Integer i : list)`
- ☒ `for(Iterator<Integer> it = list.iterator(); it.hasNext();)`
- ☐ `for(Iterator<Integer> it = list.iterator(); ; it.hasNext())`

➔ 1 Punkt