**Unterrichtsstunde Arrays**

Übersicht:

In dieser Stunde wird das Konzept von Arrays erklärt. Dabei wird besonders auf die Erstellung, Befüllung und Modifikation von Arrays eingegangen. Im weiteren Schritt wird die Verwendung von Arrays mit Schleifen erläutert. Diese Schritte sollen die Schüler anhand von einfachen Aufgaben und Beispielen lernen und verstehen.

Lernziel:

Nach dieser Stunde haben die Schüler das Konzept eines Arrays verstanden, können selbst Arrays erstellen, befüllen und Modifizieren/Ändern. Die Schüler das Prinzip des Indexes verstanden und verstehen, dass dieser in einem Array bei 0 beginnt. Zusätzlich haben sie verstanden, dass man mit Schleifen arbeiten kann, um die Werte in einem Array besser auszugeben oder besser mit ihnen arbeiten zu können.

Zeitplan:

1. Einführung in die Materie 5min
2. Erklären des Grundprinzips eines Arrays 10min
3. Grundsyntax zum Erstellen von Arrays 10min
4. Übung - Erstellen von ersten Arrays 15min
5. Erklärung von Methoden zum Hinzufügen/Ändern von Arrays 10min
6. Übung - Ändern von Arrays 10min
7. Erklärung von Schleifen in Verbindung mit Arrays 15min
8. Übung - Arrays und Schleifen 15min

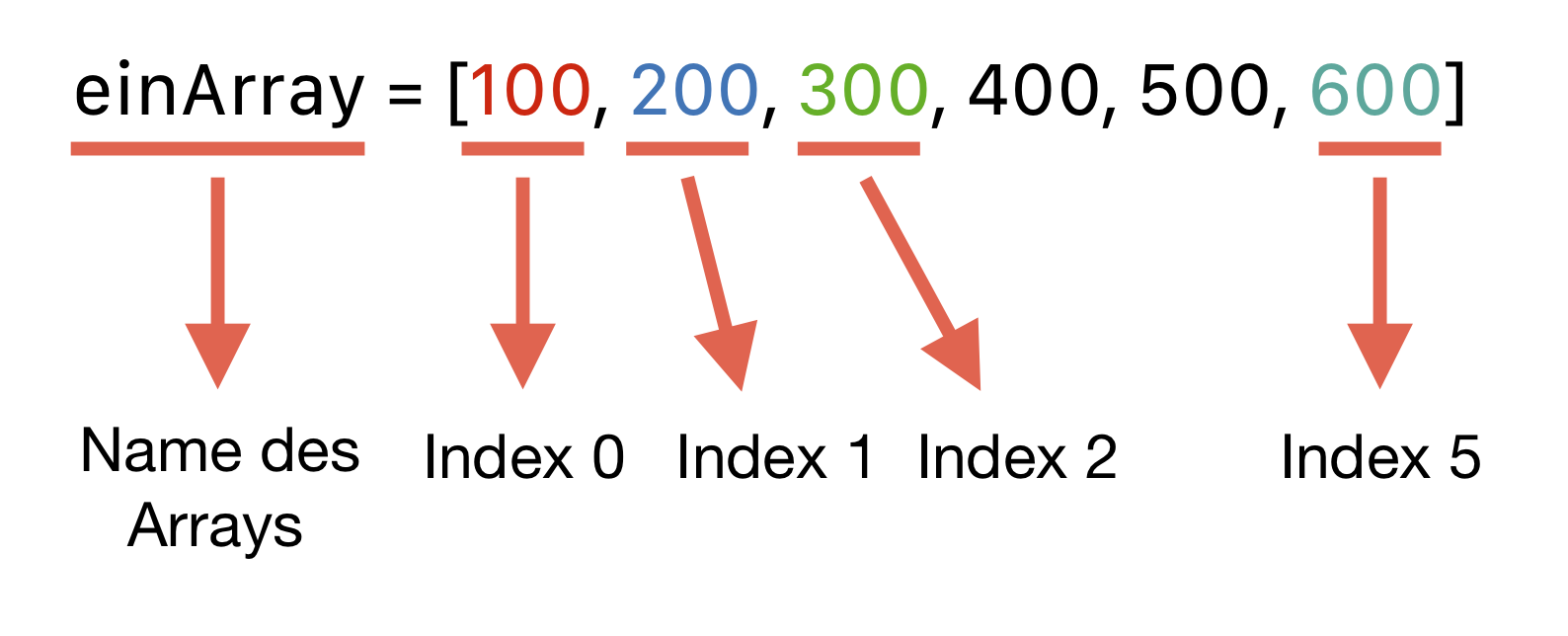
Theorie:

1. **Einführung in die Materie:**

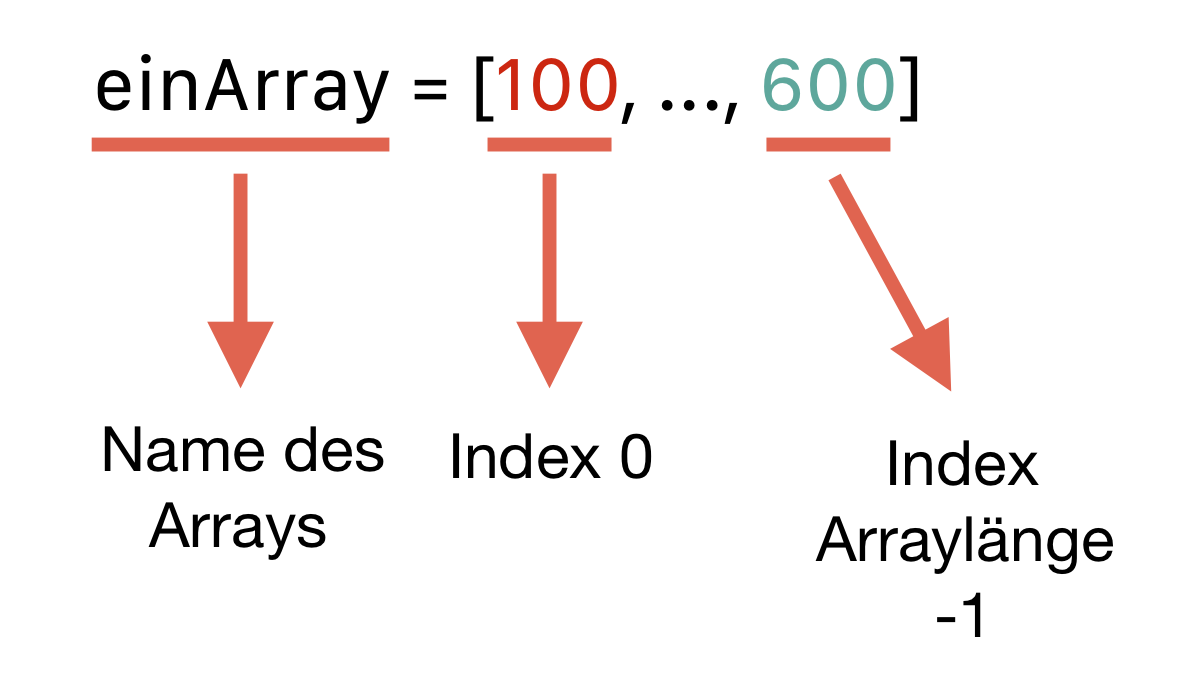
* Frage an die Schüler: *“Was könnt ihr euch unter einer “Sammlung” vorstellen?”* (Im Bezug auf Softwareentwicklung)
* **Begriffe an der Tafel sammeln**

1. **Erklären des Grundprinzips eines Arrays:**

* Ein Array ist eine Sammlung an Elementen eines bestimmten Datentyps
* Man kann es sich bildlich als Container oder Box vorstellen, in der verschiedene Werte geordnet liegen
* Diese Sammlung beinhaltet verschiedene Werte und kann einer Variable zugewiesen werden
* An der Tafel (alternativ Beamer) visuell zeigen, wie ein Array aussieht und was ein Array ist:



* Anhand der Grafik das Prinzip der Indexe erklären (startet bei 0 und endet bei Länge - 1):



1. **Grundsyntax für Erstellen von Arrays:**

* In der IDE zeigen, wie man einen Array in Java erstellt (**2 Methoden**):

1. int[] neuesArray = new int[10]; Definiert einen Array der Länge 10.

* neuesArray[0] = 100; Setzt ein Int = 100 an der ersten Stelle (Index 0) des Arrays ein.  
  neuesArray[1] = 200; etc.
* println(neuesArray[0]); Zeigen, dass der Wert an Index 0 ausgegeben wird.

1. int[] neuesArray = {100, 200, 300}; Erstellt einen neuen Array und setzt die Länge entsprechend der Menge der in der geschweiften Klammer definierten Inhalte. Fügt die Inhalte innerhalb einer Zeile auch direkt hinzu, ist eine **schnellere** Methode.

* println(neuesArray[1]); Zeigen, dass der Wert an Index 1 ausgegeben wird.
* Erklären, dass man bei Arrays verschiedene Datentypen verwenden kann:
  + String[] namen = {"Hannes", "Elias", "Paul"}; etc.

1. **Übung - Erstellen von ersten Arrays**

* Übung am Beamer zeigen und die Schüler bearbeiten lassen (ca. 10min)
* Lösungen in der IDE zeigen und erklären (ca. 5min)

1. **Erklärung von Methoden zum Hinzufügen/Ändern von Arrays**

* In der IDE zeigen wie man einen Wert im Array verändert:
  + int[] neuesArray = {100, 200, 300}; Erstellt einen neuen Array.
  + neuesArray[0] = 200; Setzt den Wert des Arrays am Index 0 auf 200 (davor definiert auf 100).
  + **Wichtig**: Ist ein Array als Int definiert so kann der Wert keinen String erhalten! (neuesArray[0] = "Hallo"; *// Error!*)
* Es ist **nicht** möglich einen Array im nachhinein zu vergrößern! Es können keine neuen Werte hinzugefügt werden, nur bestehende geändert werden. **Achtung**: In anderen Sprachen ist das aber teilweise möglich (z.B. Swift).

1. **Übung - Ändern von Arrays**

* Übung am Beamer zeigen und die Schüler bearbeiten lassen (ca. 5min)
* Lösungen in der IDE zeigen und erklären (ca. 5min)

1. **Erklärung von Schleifen in Verbindung mit Arrays**

* Arrays können auch fortgeschritten benutzt werden
* Um mit den Werten in einem Array zu arbeiten, benutzt man oft Schleifen
* Man Iteriert über den Array und führt dabei eine Operation aus (d.h. man nimmt jeden Wert des Arrays und macht etwas mit ihm)
* **Beispiele**:
  + Man hat einen Array mit Zahlen und möchte alle Zahlen addieren und die Summe ausgeben
  + Man hat einen Array mit Namen und möchte alle Namen ausgeben
  + Man hat einen Namen und möchte Wissen, ob dieser sich in einem Array befindet (sinnvoll bei Formularen oder Eingaben zur Prüfung)
* Alle diese Operationen sind mit **Schleifen** lösbar
* In der IDE zeigen und erklären, wie man eine Schleife auf einem Array verwendet

|  |
| --- |
| String[] staedte = {"Berlin", "Frankfurt", "Hamburg", "Stuttgart", "Leipzig"}; for(int i = 0; i < staedte.length; i++){  println(staedte[i]); }; |

* + Im ersten Schritt wird ein Array namens staedte erstellt und befüllt.
  + Danach wird mit einer for-Schleife über den Array iteriert und jeder Wert auf die Konsole ausgegeben.
  + **Wichtig**: Innerhalb der Schleife wird ein int i erstellt, welches die Bedingung i < staedte.length erfüllen muss. Damit ist i immer kleiner als die Länge des Arrays und hat deshalb auch immer einen gültigen Index! Nur dann wird i auch um 1 inkrementiert.
  + **Erklärung**: .length gibt die Anzahl der in einem Array erhaltenen Elemente aus. Hat man nun einen Array mit 5 Elementen, so befindet sich der letzte Wert des Arrays an Index 4 (siehe Erklärung am Anfang der Stunde). Deshalb muss die Variable i in der Schleife immer kleiner sein, als die Anzahl der Elemente in einem Array, da sie sich sonst nicht mehr innerhalb eines gültigen Indexes befindet.
  + Die Schleife beginnt also mit dem Wert 0 und wiederholt sich so oft, bis sie den Wert 4 erreicht (Länge des Arrays - 1). Danach erfüllt i die Bedingungen nicht mehr und die Schleife wird nicht mehr ausgeführt. Der Code innerhalb der Schleife wird also 5 mal ausgeführt und bei jedem neuen ausführen hat i einen anderen Wert.
  + **Wichtig**: Man muss sich merken, wie so eine Schleife aufgebaut ist.
  + **Schüler fragen, ob sie das Prinzip verstanden haben und eventuelle Unklarheiten klären.**
* Weiteres Beispiel in der IDE zeigen und erklären:

|  |
| --- |
| int result = 0; int [] numbers = {2,4,6,8}; for(int e = 0; e < numbers.length; e++){  result += numbers[e]; }; println(result); |

* + Hier wird ein int result erstellt, welches den Wert 0 besitzt.
  + Danach wird ein Array mit Zahlen erstellt.
  + Die Schleife läuft wieder wie beim Beispiel davor über den Array.
  + Innerhalb der Schleife wird nun aber der Wert des Arrays am Index e dem vorher erstellten result addiert.
  + Sobald die Schleife fertig ist, wird result auf der Konsole ausgegeben.
  + **Schüler** **fragen**, was hier möglicherweise das Ergebnis (auf der Konsole) ist.
* Beispiel wenn man prüfen möchte ob ein Name in einem Array vorhanden ist

1. **Übung - Arrays und Schleifen**

* Übung am Beamer zeigen und die Schüler bearbeiten lassen (ca. 10min)
* Lösungen in der IDE zeigen und erklären (ca. 5min)

**Übungen:**

**1. Übung - Erstellen von ersten Arrays**

1. **Array mit definierter Länge**
   1. Erstelle einen Array namens zahlen, welcher eine vordefinierte Länge von 4 Elementen hat und Integer beinhaltet.
   2. Befülle den Array mit den Zahlen von 1-4.
   3. Lasse dir die Zahl 3 über den Array an der Konsole ausgeben.
2. **Array mit Städten**
   1. Erstelle einen Array namens staedte, welcher eine vordefinierte Länge von 5 Elementen hat und Strings beinhaltet.
   2. Befülle den Array mit den Städten Berlin, Frankfurt, Hamburg, Stuttgart und Leipzig.
   3. Lasse dir alle Städte die sich im Array befinden an der Konsole ausgeben.
   4. Versuche den oben erstellten Array mit nur einem Code-Statement (innerhalb einer Code-Zeile) zu generieren.

**2. Übung - Ändern von Arrays**

1. **Wert in einem Array verändern**
   1. Erstelle einen Array names integers, welcher 5 verschiedene Integer Werte beinhaltet.
   2. Ändere den zweiten und vierten Wert auf eine andere Zahl.
   3. Ändere alle Werte im Array auf andere Zahlen.

**3. Übung - Arrays und Schleifen**

1. **Jeden Wert in einem Array ausgeben und addieren**
   1. Erstelle einen Array und befülle ihn mit den Namen deiner Freunde.
   2. Lasse dir nun mithilfe einer Schleife alle Namen die sich im Array befinden auf die Konsole ausgeben.
   3. Ändere den Array, sodass er die Alterszahlen deiner Freunde beinhaltet und lasse dir mithilfe einer Schleife die Summe auf die Konsole ausgeben.
2. **BONUS - Knobelaufgaben**
   1. Erstelle und befülle einen Array mit Namen und überprüfe mithilfe einer Schleife, ob sich dein eigener Name in diesem Array befindet. Gib dementsprechend eine Nachricht auf der Konsole aus (z.B.: “Mein Name ist leider nicht enthalten”, “Mein Name ist enthalten!”, etc.).

**TIPP**: Erstelle für deinen Namen eine eigene String-Variable. Versuche die Aufgabe aufzuteilen, kümmere dich zuerst um die Schleife und dann um die entsprechende Ausgabe auf der Konsole. Ein Boolean-Wert könnte hilfreich sein.

**Lösungen**

**Immer in der IDE nachvollziehbar und langsam vorführen! Ergebnisse am besten in der Konsole ausgeben, damit die Schüler direkt sehen was passiert.**

**1. Erstellen von ersten Arrays**

1. **Array mit definierter Länge**
   1. int[] zahlen = new int[4];

|  |
| --- |
| zahlen[0] = 1;  zahlen[1] = 2; zahlen[2] = 3; zahlen[3] = 4; |

* 1. println(zahlen[2]);

1. **Array mit Städten**
   1. String[] staedte = new String[5];

|  |
| --- |
| staedte[0] = "Berlin"; staedte[1] = "Frankfurt"; staedte[2] = "Hamburg"; staedte[3] = "Stuttgart"; staedte[4] = "Leipzig"; |

* 1. println(staedte);

|  |
| --- |
| String[] staedte = {"Berlin", "Frankfurt", "Hamburg", "Stuttgart", "Leipzig"}; |

**2. Ändern von Array** s

1. **Wert in einem Array verändern**
   1. int[] integers = {20, 40, 50, 60, 70};

|  |
| --- |
| integers[1] = 1; integers[3] = 12; |



|  |
| --- |
| integers[0] = 1; integers[1] = 2; integers[2] = 3; integers[3] = 4; integers[4] = 5; |

**3. Arrays und Schleifen**

1. **Jeden Wert in einem Array ausgeben und addieren**
   1. String[] names = {"Hans", "Peter", "Marc"};

|  |
| --- |
| for(int i = 0; i < names.length; i++){  println(names[i]); }; |

|  |
| --- |
| int result = 0; int [] numbers = {2,4,6,8}; for(int i = 0; i < numbers.length; i++){  result += numbers[i]; }; println(result); |

1. **Knobelaufgabe**

|  |
| --- |
| String meinName = "Marc"; Boolean response = false; String[] names = {"Hans", "Peter", "Marc", "Ben", "Johannes", "Paul"};  for(int i = 0; i < names.length; i++){  if(names[i] == meinName){  response = true;  }; };  if(response){  println("Name ist erhalten"); } else {  println("Name ist nicht erhalten"); }; |