

# **APIP #3**

## **Arrays**



# Agenda

- Basis
- Syntax
- Spelletje spelen
- Complexere dingen doen
- Oefenen
- Huiswerk

# The basics

- Soms is het handig om meerdere variabelen op te slaan in een container.
- Een array is een container waarin je meerdere variabelen van hetzelfde type kan opslaan.
- Een array is een gestructureerde dataopslag.
- Arrays werken met indexen.



## VB: Sla de naam van elke student op

```
public class Main {  
    public static void main(String[] args) {  
  
        Scanner scanner = new Scanner(System.in);  
        String naam = scanner.nextLine();  
  
        while (!naam.equals("quit")) {  
  
            naam = scanner.nextLine();  
            // Wat is er met de eerste naam gebeurd?  
  
        }  
    }  
}
```

## Oplossing?

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        String naam1 = scanner.nextLine();
        String naam2 = scanner.nextLine();

        //...
        String naam55 = scanner.nextLine();
    }
}
```



## Oplossing!

```
public class Main {  
    public static void main(String[] args)  
    {  
        //... aanmaken van scanner  
  
        String[] array = new String[55];  
  
        //... Inlezen van namen  
    }  
}
```

array
[0] = "Ab"
[1] = "Bas"
[2] = "Charles"
[...] = "..."
[54] = "Zoë"

# Werken met...

- **Aanmaken:**

```
String[] namen = new String[10]; // 10 plekken
```

```
int[] cijfers = new int[10];
```

```
boolean[] uitkomsten = new boolean[10];
```

- De grootte van een array kan na declaratie niet meer aangepast worden: 10 = 10
- Wat wel kan is een nieuwe array aanmaken

```
uitkomsten = new boolean[100];
```

# Werken met...

- **Indexen:**  
Arrays werken met indexen; superlastig!  
Index is pointer naar plek in de array.  
Arrays starten bij 0 (althans in Java)

Waarde	1	1	2	3	5	8	13
Index	0	1	2	3	4	5	6





## Werken met...

- Waardes plaatsen:

```
String[] docenten = new String[1]; //een lege array
docenten[0] = "Fatih"; //let op 0 is eerste plek
docenten[1] = "Maltie";
docenten[2] = "Jelle";
docenten[3] = "Dinesh";
```

Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 1

# Werken met...

- **Aanmaken II**

```
String[] docenten = new String[4];  
docenten[0] = "Fatih";  
docenten[1] = "Maltie";  
docenten[2] = "Jelle";  
docenten[3] = "Dinesh";
```

**Dit is hetzelfde als:**

```
String[] docenten = {"Fatih", "Maltie", "Jelle", "Dinesh"};
```

# Werken met...

- Uitlezen

```
int[] cijfers = {1,2,3,4,5,6,7,8,9,10};
```

```
System.out.println(cijfers[0]);
```

Vraag: stop het laatste getal in een variabele

```
int resultaatVoorJaap = cijfers[?];
```

# Array kunnen slechts één datatype bevatten

```
int[] cijfers = new int[10];  
cijfers[0] = 5;  
cijfers[1] = "test"; //dit gaat niet  
cijfers[2] = 5.5; //dit ook niet
```

**In Python:**

```
array = [5, 'test', 5.5] //Geldige code
```

# Array zijn gestructureerd

```
double[] kommaGetallen = new double[5];  
kommaGetallen[0] = 10.1;  
kommaGetallen[1] = 7.3;  
kommaGetallen[2] = 42.0;  
kommaGetallen[3] = 3.7;  
kommaGetallen[4] = 0.1;  
System.out.println(kommaGetallen[0]);
```

kommaGetallen[0] zal altijd 10.1 teruggeven (totdat je de waarde wijzigt).



# Kahoot!

## Wat staat er bij default in een array


- It depends
- Bij primatieve typen zoals `int`, `double`, `float`, `boolean`, etc
  - 0 of een variant daarvan (0.0, false)
- Bij objecten altijd `null`
  - `null` betekent 'geen waarde'
  - Niet te verwarren met 0

# Opdracht

Nu jullie:

- Maakt een int array aan met 5 plekjes.
  - Vul die met de waarden: 5, 10, 150, 20, 25.
  - Print het eerste getal uit.
  - Verander de waarde van het 3<sup>de</sup> plekje naar 15. (Let op de index)
- 
- Haal ondertussen een kopje koffie (of iets anders)



**Korte pauze**  
**Terug over ? minuten** 

## Opdracht: antwoorden

Nu jullie:

- Maakt een int array aan met 5 plekjes.
- Vul die met de waarden: 5, 10, 150, 20, 25.
- Print het eerste getal uit.
- Verander de waarde van het 3<sup>de</sup> plekje naar 15. (Let op de index)

```
int[] array = {5, 10, 150, 20, 25};  
System.out.println(array[0]);  
array[2] = 15;
```

## VB: Alle namen printen

### Array

[0] = "Ab"

[1] = "Bas"

[2] = "Charles"

[3] = "Zoë"

```
String[] namen = {"Ab", "Bas", "Charles", "Zoë"};
```

```
System.out.println(namen[0]); //"Ab"  
System.out.println(namen[1]); //"Bas"  
System.out.println(namen[2]); //"Charles"  
System.out.println(namen[3]); //"Zoë"
```

```
Ab  
Bas  
Charles  
Zoë  
  
Process finished with exit code 0
```

**Dit is hetzelfde als:**

```
for (int i = 0; i < 4; i++) { // lengte van de array is 4  
    System.out.println(namen[i]); //We gebruiken de teller "i" als  
                                   index-waarde  
}
```

# Array.length

```
String[] namen = {"Ab", "Bas", "Charles", "Zoë"}; //lengte (4) zichtbaar
```

- Soms is de lengte onbekend.

```
String[] namen = {"Ab", "Bas", "Charles", "Dirk", "Ella", "Faruk", ...}
```

- Dat kan lastig zijn.

```
for (int i = 0; i < 4; i++) {  
    //4 werkt dan hoogstwaarschijnlijk niet meer  
}
```

- Oplossing: Array.length geeft de lengte terug van de array.

```
for (int i = 0; i < namen.length; i++) {  
    // werkt ongeacht de lengte  
}
```

## VB: Alle namen uitprinten

```
String[] namen = {"Ab", "Bas", "Charles", "Zoë"};  
  
for (int i = 0; i < namen.length; i++) {  
    System.out.println(namen[i]);  
}
```

```
Ab  
Bas  
Charles  
Zoë  
  
Process finished with exit code 0
```

## Werken met... een waarde zoeken

- Soms wil je weten of een bepaald waarde in de array zit

```
String[] namen = {"Ab", "Bas", "Charles", "Zoë"};  
boolean aanwezig = false; //Hierin sla je op of de waarde aanwezig is  
String zoekterm = "Dirk";  
  
for (int i = 0; i < namen.length; i++) {  
    if (namen[i].equals(zoekterm)) {  
        aanwezig = true;  
    }  
}  
  
System.out.println(aanwezig); //false
```

## Werken met... een waarde toevoegen

- Soms wil je een extra waarde toevoegen
- "Dirk" moet toegevoegd worden
- Weet je nog: arrays kunnen niet groter worden

array
[0] = "Ab"
[1] = "Bas"
[2] = "Charles"
[3] = "Zoë"
"Dirk"

### Pseudocode

```
// Maak een nieuwe array aan die 1 groter is dan orginele array
```

```
// Itereer (loop) over de orginele array heen en kopieer waardes  
naar de grotere array
```

```
// Voeg Dirk toe als laatste waarde aan de grotere array
```

```
// Overschrijf de orginele array door de grotere array
```

## Werken met... een waarde toevoegen

- Soms wil je een extra waarde toevoegen

```
String[] namen = {"Ab", "Bas", "Charles", "Zoë"};
```

```
// Maak een nieuwe array aan die 1 groter is dan orginele array
```

```
String[] grotereArray = new String[namen.length + 1];
```

```
// Itereer (loop) over de orginele array heen en kopieer waardes naar de  
grotere array
```

```
for (int i = 0; i < namen.length; i++) {  
    grotereArray[i] = namen[i];  
}
```

```
// Voeg Dirk toe als laatste waarde aan de grotere array
```

```
grotereArray[grotereArray.length - 1] = "Dirk"; //laatste plek
```

```
// Overschrijf de orginele array door de grotere array
```

```
namen = grotereArray;
```

```
Ab  
Bas  
Charles  
Zoë  
Dirk  
  
Process finished with exit code 0
```



## Werken met... een waarde verwijderen

- Soms wil je een waarde verwijderen
- "Dirk" moet verwijderd worden
- Weet je nog: arrays kunnen ook niet kleiner worden

### Pseudocode

```
// Maak een nieuwe array aan die 1 kleiner is dan orginele array
```

```
// Itereer (loop) over de kleinere array heen en kopieer waardes  
vanuit de orginele array
```

```
// Overschrijf de orginele array door de kleinere array
```

array
[0] = "Ab"
[1] = "Bas"
[2] = "Charles"
[3] = "Zoë"
[4] = "Dirk"

## Werken met... een waarde verwijderen

```
String[] namen = {"Ab", "Bas", "Charles", "Zoë", "Dirk"};
```

```
// Maak een nieuwe array aan die 1 kleiner is dan orginele array
```

```
String[] kleinereArray = new String[namen.length - 1];
```

```
// Itereer (loop) over de kleinere array heen en kopieer waardes vanuit de  
orginele array
```

```
for (int i = 0; i < kleinereArray.length; i++) {  
    kleinereArray[i] = namen[i];  
}
```

```
Ab  
Bas  
Charles  
Zoë  
  
Process finished with exit code 0
```

```
// Overschrijf de orginele array door de kleinere array
```


```
namen = kleinereArray;
```

## Werken met... waardes omwisselen

- Soms wil je waardes wisselen

```
String[] namen = {"Ab", "Bas", "Charles", "Zoë", "Dirk"};  
//Dirk moet op "zijn plek" komen
```

```
String temp = namen[3]; //We slaan Zoë tijdelijk op  
namen[3] = namen[4]; //Dirk staat nu op twee plekken  
namen[4] = temp;
```

**Korte pauze**  
**Terug over ? minuten** 

# Reference I

```
int orgineel = 1;  
int kopie = orgineel;
```

```
System.out.println(orgineel); // 1  
System.out.println(kopie); // 1
```

```
orgineel = 50;
```

```
System.out.println(orgineel); // 50  
System.out.println(kopie); // 1
```

## Reference II

```
int[] orgineel = {1}; //initialeer een array met 1 plek  
int[] kopie = orgineel;
```

```
System.out.println(orgineel[0]); // 1  
System.out.println(kopie[0]); // 1
```


```
orgineel[0] = 50;
```


```
System.out.println(orgineel[0]); // 50  
System.out.println(kopie[0]); // 50
```

## Reference III: visueel

`int orgineel = 1;` → 

`int kopie = orgineel;` → 

`int[] orgineel = {1};` → 

`int[] kopie = orgineel;` → 

## Reference IV: een oplossing

```
int[] origineel = {1};

int[] kopie = new int[origineel.length];
// we gebruiken dezelfde lengte

for (int i = 0; i < origineel.length; i++) {
    kopie[i] = origineel[i];
}

origineel[0] = 50;

System.out.println(origineel[0]); // 50
System.out.println(kopie[0]); // 1
```



# Oefening

```
int[] ongesorteerd = {1,5,2,6,5,3};  
int[] gesorteerd = {1,2,3,4,5,6};
```

- Schrijf code om te controleren of de array gesorteerd is en print dat
- Dikke tip: Veel van wat zojuist besproken is, heb je nodig
- Uitdaging: Sorteer de array

## Oefening: antwoord

```
int[] ongesorteerd = {1, 5, 2, 6, 5, 3};  
int[] gesorteerd = {1, 2, 3, 4, 5, 6};  
int[] array = ongesorteerd;
```

```
boolean resultaat = true;  
int laatsteGetal = array[0];
```

```
for (int i = 1; i < array.length; i++) {  
    if (array[i] < laatsteGetal) {  
        resultaat = false;  
    }  
    laatsteGetal = array[i];  
}
```

```
System.out.println("Array gesorteerd?: " + resultaat);
```

# Huiswerk

- **Schrijf code om "Dirk" op de juist plek toe te voegen (tussen Charles en Zoë)**

Je hoeft nog niet de woorden met elkaar te vergelijken

```
String[] namen = {"Ab", "Bas", "Charles", "Zoë"};
```

- **Schrijf vervolgens code om "Dirk" te verwijderen**

```
String[] namen = {"Ab", "Bas", "Charles", "Dirk", "Zoë"};
```

- **Schrijf code om te controleren of de array gelijk is aan een andere array**

```
int[] basis = {1,5,2,6,5,3};
```

```
int[] gelijk = {1,5,2,6,5,3};
```

```
int[] ongelijk = {9,4,6,3};
```

- **Neem je vragen mee naar dinsdag**



*“That’s all Folks!”*