

Java01 – Einführung „Feld“ bzw. „array“

Ein **Feld** (engl. **array**) ist eine Folge einer endlichen Anzahl n von Variablen des gleichen Datentyps.

Die Zahl n ist die **Länge des Feldes**. Jede der n Variablen wird als **Element** oder Feld-Element bezeichnet.

Jedes Element hat einen **Index**, das erste Element den Index 0, das letzte der n Elemente den Index n – 1.

Beispiel:

Feld „feld“ aus 5 ganzen Zahlen

Index i	0	1	2	3	4
Feld-Element	feld[0]	feld[1]	feld[2]	feld[3]	feld[4]
Feld-Element-Wert (z. B.)	7	9	1	5	3

Schritt 1 – Neues BlueJ-Projekt erstellen

Öffne zunächst BlueJ und erstelle mit „Projekt“ → „Neues Projekt“ ein neues BlueJ-Projekt mit dem Namen „**Java_02_Feld**“ und speichere es in Deinem Home-Laufwerk („Location“ → „Choose“).

Schritt 2 – Klasse „Feld“ erstellen

Erstelle in Deinem neuen Projekt eine neue Klasse „Feld“ und öffne den Quelltext mit einem Doppelklick auf das Klassen-Symbol. Lösche alles zwischen „public class Feld{“ und der letzten geschweiften Klammer ganz unten.

Schritt 3 – Variable „feld“ deklarieren

In Java wird ein Feld als Datentyp deklariert durch eckige Klammern direkt nach der Angabe des Datentyps:

```
public class Feld {  
    private int[] feld;           // Deklaration von „feld“ als Feld von „int's“  
}
```

Schritt 4 – Konstruktor mit Festlegung der Feld-Länge und Initialisierung der Elemente

Erst im Konstruktor wird die Feld-Länge festgelegt. Die Feld-Elemente werden beim Aufruf initialisiert.

```
public class Feld {  
    private int[] feld;           // Deklaration von „feld“ als Feld von „int's“  
    public Feld(int a1, int a2, int a3, int a4, int a5){      // Konstruktor  
        feld = new int[5];          // Die Festlegung der Feld-Länge erfolgt hier.  
        this.feld[0] = a1;          // Zuordnung der beim Aufruf des Konstruktor  
        this.feld[1] = a2;          // einzugebenden ganzen Zahlen („int's“)  
        this.feld[2] = a3;          // zu den einzelnen Feld-Elementen.  
        this.feld[3] = a4;          // Achtung: Zählung beginnt bei 0 (Ende: 4)  
        this.feld[4] = a5;  
    }  
}
```

„Übersetze“ den Quellcode und erzeuge eine neuen Objekt der Klasse „Feld“. Gib dabei 5 ganze Zahlen ein. Inspiziere das neue erzeugte Objekt der Klasse „Feld“ (Rechtsklick auf das Objekt → Inspizieren).

Schritt 5 – „feldElementIAusgeben(int i)“ zum Ausgeben des i-ten Element-Wertes

Die hier erscheinenden Zeilenumbrüche bei ****** musst Du löschen!

```
public void feldElementIAusgeben(int i) {  
    System.out.println("Das Element mit dem Index " + i + " hat den  
    ** Wert " + feld[i] + ".");  
}
```

Schritt 6 – „feldElementISetzen(int i, int neuerWert)“ zum Setzen des i-ten Element-Wertes

Die hier erscheinenden Zeilenumbrüche bei ****** musst Du löschen!

```
public void feldElementISetzen(int i, int neuerWert) {  
    feld[i] = neuerWert;  
    System.out.println("Das Element mit dem Index " + i + " hat nun den  
    ** Wert " + feld[i] + ".");  
}
```