

NOTES: Introduction to One-Dimensional(1D) Arrays

Name of Array: *scores* Number of Elements (Array Length): *4*

Index #	<i>0</i>	<i>1</i>	<i>2</i>	<i>3</i>
Data/Element	<i>95</i>	<i>76</i>	<i>81</i>	<i>40</i>

Name of Array: *percents* Array Length: *3*

Index #	<i>0</i>	<i>1</i>	<i>2</i>
Data/Element	<i>92.1</i>	<i>56.4</i>	<i>83.2</i>

Name of Array: *friends* Array Length: *2*

Index #	<i>0</i>	<i>1</i>
Data/Element	<i>"Jack"</i>	<i>"Jill"</i>

Declaring and Initializing Values in a 1D-Array Using One Statement:

data type stored

```
int[] scores = {095, 176, 281, 340};  
double[] percents = {92.1, 56.4, 83.2};  
String[] friends = {"Jack", "Jill"};
```

Accessing/Mutating Array Data and Finding Array Lengths

```
System.out.println(scores[3])      // 40      ← value we are accessing  
System.out.println(percents[0])    // 92.1  
System.out.println(friends[1])     // Jill  
System.out.println(scores[6])      // Out of bounds error  
scores[3] = 50;                    // mutates the data  
System.out.println(scores[3])      // 50  
friends[0] = "Fred";               // mutates the data  
System.out.println(friends[0])     // Fred  
System.out.println(friends.length) // 2  
System.out.println(scores.length)  // 4
```

↑
NO PARENTHESES!

Declaring and Initializing Values in a 1D-Array Using Multiple Statements:

Name of Array:

Number of Elements (Array Length):

Index #	0	1	2	3
Data/Element	95	76	81	40

data type ↓
name of array ↓
new 1D Array ←
length ↑

```
int[] scores = new int[4];  
scores[0] = 95;  
scores[1] = 76;  
scores[2] = 81;  
scores[3] = 40;
```

Using a For Loop to Initialize Data into an 1D-Array

Let's suppose you want to create a 1D-array named `numbers`, containing the numbers 1, 2, 3, 4, 5.

Name of Array: `numbers` Array Length: `5`

Index #	0	1	2	3	4
Data/Element	1	2	3	4	5

```
// declare your 1D array similar to the example above
```

```
int[] numbers = new int[5];
```

```
// Use a "for" loop to store the desired data
```

```
for(int i = 0; i < numbers.length; i++)
```

```
{  
    numbers[i] = i+1;
```

```
}
```

PRACTICE: Declaring and Initializing 1D Arrays

Array #1: An Inventory of Prices

Name of Array: prices Data Type: double Array Length: 3

Index #	0	1	2
Data/Element	55.99	4.99	6.00

Using One Initialization Statement:

`double [] prices = { 55.99, 4.99, 6.00 }`

Using Multiple Statements:

`double [] prices = new double [3]`
`prices [0] = 55.99;`
`prices [1] = 4.99;`
`prices [2] = 6.00;`

Array #2: Teachers in Applied Tech

Name of Array: techTeam Data Type: String Array Length: 4

Index #	0	1	2	3
Data/Element	"Kennedy"	"Monley"	"Goldstien"	"Miller"

Using One Initialization Statement:

`String [] techTeam = { "Kennedy", "Monley", "Goldstein", "Miller" };`

Using Multiple Statements:

`String [] techTeam = new String [4]`
`techTeam [0] = "Kennedy"`
`techTeam [1] = "Monley"`
`techTeam [2] = "Goldstein"`
`techTeam [3] = "Miller"`