

# Selenium WebDriver Training

Arrays

# The Golden Circle

What

## What are Arrays?

Collection of items of same data type at contiguous memory locations

Why

## Why are arrays needed?

Used to store multiple values in a single variable

How

## How to implement Arrays?

Multiple ways - Refer next slides

# What is Array?

Collection of items of

- same data type
- stored at consecutive memory locations

# How to Create Array ? [1 Out of 2]

## Using Array Literal

- **Syntax 1: type name[] = {values};**

# How to Create Array ? [1 Out of 2]

## Using Array Literal

- Syntax 1: `type name[] = {values};`
- Syntax 2: `type[] name = {values};`

# How to Create Array ? [1 Out of 2]

## Using Array Literal

- Syntax 1: `type name[] = {values};`
- Syntax 2: `type[] name = {values};`
- Example: `int[] scores = {98, 87, 91, 99, 87};`

# How to Create Array ? [1 Out of 2]

## Using Array Literal

- Syntax 1: `type name[] = {values};`
- Syntax 2: `type[] name = {values};`
- Example: `int[] scores = {98, 87, 91, 99, 87};`



# How to Create Array ? [1 Out of 2]

## Using Array Literal

- Syntax 1: `type name[] = {values};`
- Syntax 2: `type[] name = {values};`
- Example: `int[] scores = {98, 87, 91, 99, 87};`



0

1

2

3

4



# How to Create Array ? [1 Out of 2]

## Using Array Literal

- **Syntax 1:** `type name[] = {values};`
- **Syntax 2:** `type[] name = {values};`
- **Example:** `int[] scores = {98, 87, 91, 99, 87};`

98	87	91	99	87
0	1	2	3	4

# How to Create Array ? [1 Out of 2]

## Using Array Literal

- Syntax 1: `type name[] = {values};`
- Syntax 2: `type[] name = {values};`
- Example: `int[] scores = {98, 87, 91, 99, 87};`
- Example: `String[] coaches = {"Babu", "Hari", "SSP"};`

# How to Create Array ? [2 Out of 2]

## Using Instantiation

- **Syntax 1: `type name[] = new type[size];`**

# How to Create Array ? [2 Out of 2]

## Using Instantiation

- **Syntax 1: `type name[] = new type[size];`**
- **Example: `int[] scores = new int[5];`**

# How to Create Array ? [2 Out of 2]

## Using Instantiation

- **Syntax 1:** `type name[] = new type[size];`
- **Example:** `int[] scores = new int[5];`



# How to Create Array ? [2 Out of 2]

## Using Instantiation

- **Syntax 1:** `type name[] = new type[size];`
- **Example:** `int[] scores = new int[5];`



0

1

2

3

4

# How to Create Array ? [2 Out of 2]

## Using Instantiation

- **Syntax 1:** `type name[] = new type[size];`
- **Example:** `int[] scores = new int[5];`  
`scores[0] = 97;`



0

1

2

3

4

# How to Create Array ? [2 Out of 2]

## Using Instantiation

- Syntax 1: `type name[] = new type[size];`
- Example: `int[] scores = new int[5];`  
`scores[0] = 97;`



0

1

2

3

4



# How to Create Array ? [2 Out of 2]

## Using Instantiation

- **Syntax 1:** `type name[] = new type[size];`
- **Example:** `int[] scores = new int[5];`  
`scores[1] = 84;`

97	84			
0	1	2	3	4

# Summary

- Arrays – Fixed Length & Same Data Type at Consecutive memories
- **Literal** (based on data) & **Instantiation** (based on Size)

# Classroom Exercise (Breakout)

- Write a program to find the duplicate numbers in the given array

```
int[] nums = {2, 5, 7, 7, 5, 9, 2, 3};
```

- Before writing the code – follow the 3 step process:
  - Understand the problem
  - Solve the problem (Using Pseudocode)
  - Write the code