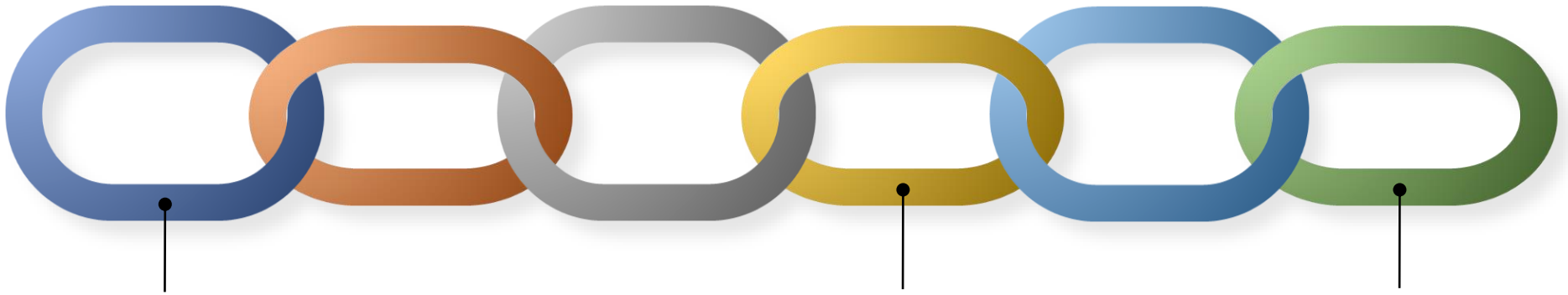
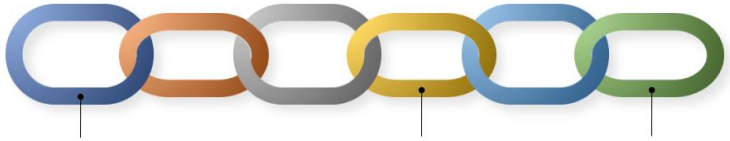


Unit 8: Arrays



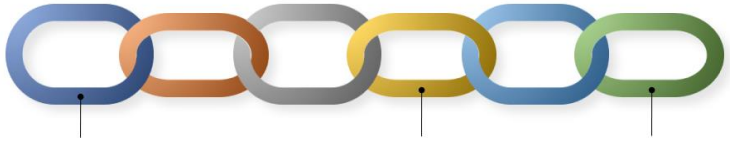
Definition: An *"array"* is a type of data capable of storing multiple values. It is used to group data that are very similar and related to each other.

Syntax1:
```
type_of_variable[] n; //N is defined as an array of type_of_variable

    n = new type_of_variable[syze] //Reserve System Memory.
```

Syntax2:
```
type_of_variable n[] = type_of_variable [syze];
```

The first element of Array is **n[0].**

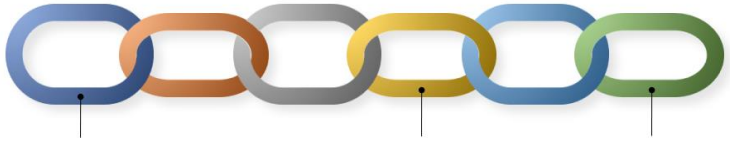


Initialize the values of an Array:

```
type_of_variable n[] = {value1, value2,…,value_n};
```

- Each element of the array can be used exactly like any other variable. It can be assigned a value or it can be used within a expression.

- All the elements of an array must be of the same type.



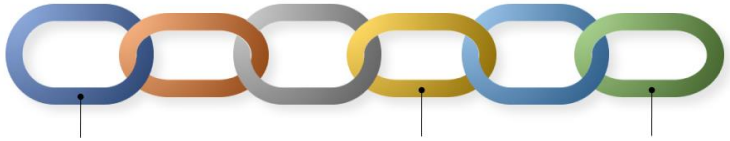
Sort Array:

- The arrays are used to sort elements.

- The most important sorting algorithms are:
  - Selection sort.
  - Bubble sort.
  - Insertion sort.
  - Merge sort.
  - Quick sort.

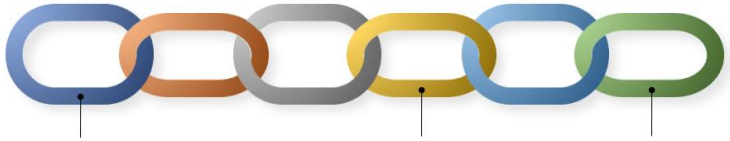
Arrays Class:

This class contains various methods for manipulating arrays (such as sorting and searching). This class also contains a static factory that allows arrays to be viewed as lists.



Arrays class methods:

- The methods are used with Arrays class:
  - **Sorting ->** sort(array) Method.
  - **Searching** -> The array should be sorted in increasing order.
    - “bynarySearch(array, value)”-> Return index of element inside the arrays if exists.
  - **Comparing**-> “equals(array1,array2)” method. Return “true” or “false” if they are equals.
  - **Filling->** fill(array, value) method. Insert value in each elements of array.



Two-Dimensional Array:

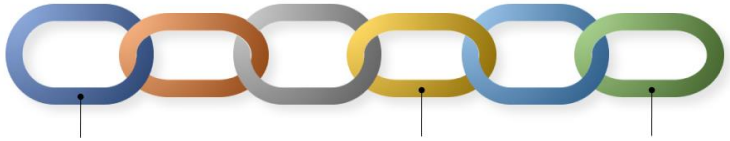
- A two-dimensional array uses two indices to locate each element. We can see this type of data as an array that, in turn, contains other arrays.
- It looks like a **grid** in which the data is distributed in rows and columns.
- Two-dimensional arrays are frequently used to place objects on a plane.

- **Syntax:**

```
type_of_variable [][] n = type_of_variable [syze1][size2];
```

- **Initialize the values of an two-dimensional Array:**

```
type_of_variable[][] n = {{val1,val2,…valn},…,{val1,val2,…,valn}}
```



For each:

- Indicate the name of the array you want to traverse and in which variable each element is going to be placed with each iteration of the loop.
- You don’t to have to specify which index the loop starts and ends with, that's taken care of by Java.
- Used to avoid errors in the size of indexes.