Arrays:

1. Arrays are **Object** and **fixed in Size**
2. Arrays can have **homogenious data type**
3. Arrays are **inheriting** from **Object Class**, and **implementing Serializable** and **Clonable Interfaces**.
4. Arrays are present in **java.util package.**
5. When ever we create an **Array** then we need to specify the **Array Size.**
6. We are able to add the elements up to the **fixed size** only.
7. If we are trying to add the elements over the **size** then JVM will rise an Exception. **ArrayIndexOutOfBoundException**
8. Arrays can hold both **primitives and object types.**

Types of Arrays:

Single Dimensional Array
int [] a; int dataType and a is arrayName

Multidimensional Array
int[][] a = new int[3][3]; int dataType and a is arrayName
int[][] a = { { 10, 20, 30 }, { 40, 50, 60 }, { 70, 80, 90 } };

Ways to declare an array

```
int[]  a;
```
Declare an array first and provide elements to it

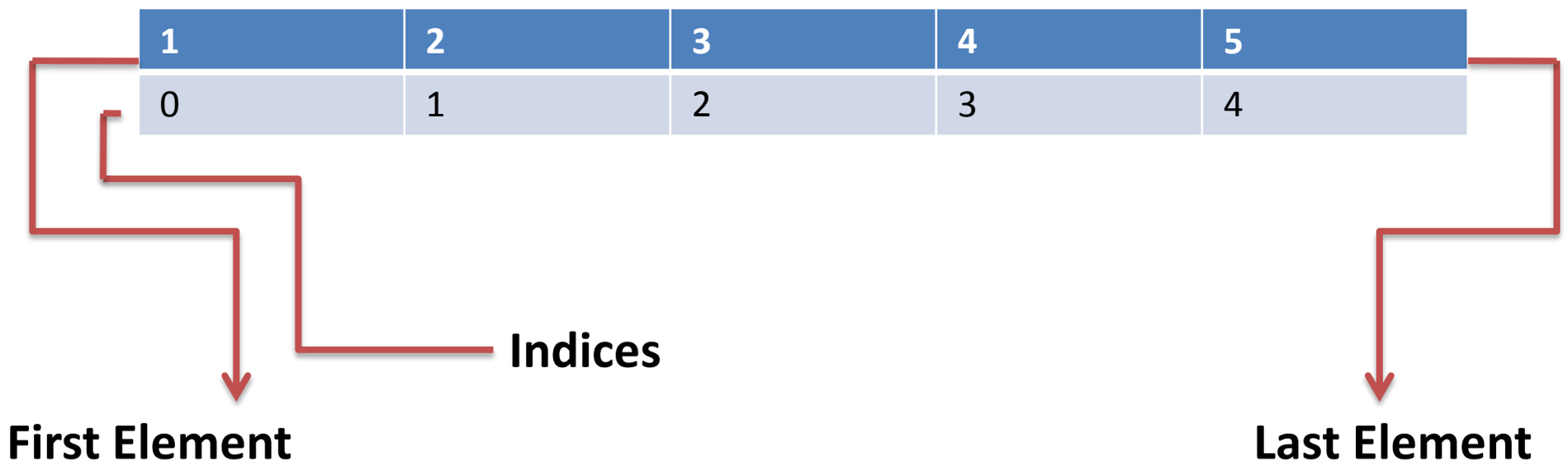
```
int i[] = { 10, 20, 30, 40, 50 };
```
Declare an array first and size, provide elements later

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

Array Index:

You can access Array elements on basis of Indices.

Int I = new int[5]



The default initial value of elements of an array is 0 for numeric types and false for boolean.

Length Varibale vs Length() Method

Length Variable:

1. **length** is a **final variable** applicable for **arrays**.
2. With the help of **length variable**, we can obtain the **size of the array**.

Length Method()

1. **length() method** is a **final method** which is applicable for **string objects.**
2. **length() method** returns the number of **characters presents** in the **string.**

```
package com.dl.one.array;

public class Eg1 {

int[] a;
int b[];

public static void main(String[] args) {

System.out.println(new Eg1().a); // null
System.out.println(new Eg1().b);// null

}
}
```

```java
package com.dl.one.array;

public class Eg2 {

public static void main(String[] args) {

// Declare an array first and provide elements to it
int i[] = { 10, 20, 30, 40, 50 };
System.out.println(i[0]); // 10
System.out.println(i[1]); // 20
System.out.println(i[2]); // 30
System.out.println(i[3]); // 40
System.out.println(i[4]); // 50
// System.out.println(i[5]); // java.lang.ArrayIndexOutOfBoundsException

}
}
```

```java
package com.dl.one.array;

public class Eg3 {

public static void main(String[] args) {

// Declare an array and size first, provide elements later
int j[] = new int[5];
j[0] = 100;
j[1] = 200;
j[2] = 300;
j[3] = 400;
System.out.println(j[0]); // 100
System.out.println(j[1]); // 200
System.out.println(j[2]); // 300
System.out.println(j[3]); // 400
System.out.println(j[4]); // 0 // default value
}
}
```

```java
package com.dl.one.array;
//modifiers
public class Eg4 {

public static int[] a;
private static int[] b;
protected static int[] c;
int[] d = { 10, 20, 30, 40, 50 };

public static void main(String[] args) {
System.out.println(a); // null
System.out.println(b); // null
System.out.println(c); // null

Eg4 eg = new Eg4();
System.out.println(eg.d[0]); // 10
System.out.println(eg.d[1]); // 20
System.out.println(eg.d[2]); // 30
System.out.println(eg.d[3]); // 40
System.out.println(eg.d[4]); // 50
}
}
```

```java
package com.dl.one.array;

//length variable
public class Eg5 {

public static void main(String[] args) {

int a[] = { 10, 20, 30, 40, 50 };
System.out.println(a[0]); // 10
System.out.println(a.length); // 5
System.out.println(a.length - 1); // 4
System.out.println(a.length + 2); // 7
System.out.println(a.length); // 5
System.out.println(a.length * 4);// 20
}
}
```

```java
package com.dl.one.array;

public class Eg6 {

public static void main(String[] args) {

// Variable must provide either dimension expressions or an array initializer
// int a[] = new int[];

int a[] = new int[0];
System.out.println(a); // [I@515f550a
// a[0] = 10; //java.lang.ArrayIndexOutOfBoundsException

int b[] = new int[-5];
// System.out.println(b); // java.lang.NegativeArraySizeException
// b[-4] = 10; //java.lang.NegativeArraySizeException: -5

}
}
```

```java
package com.dl.one.array;

//array using for loop
public class Eg7 {

public static void main(String[] args) {

int a[] = { 10, 20, 30, 40, 50 };
for (int i = 0; i < a.length; i++) {
System.out.println(a[i]);
}

System.out.println(" ");

for (int i = a.length -1; i >= 0; i--) {
        System.out.println(a[i]);
    }
}
}
```

```
10
20
30
40
50

50
40
30
20
10
```

```java
package com.dl.one.array;

//array using while loop
public class Eg8 {
public static void main(String[] args) {

int a[] = { 10, 20, 30, 40, 50 };

int i = 0;
while (i < 5) {
System.out.println(a[i]);
i++;
}

System.out.println("");

int j = 4;
while (j >= 0) {
System.out.println(a[j]);
j--;
}
}
}
```

```
10
20
30
40
50

50
40
30
20
10
```

```java
package com.dl.one.array;

//array using do while loop
public class Eg9 {

public static void main(String[] args) {

int a[] = { 10, 20, 30, 40, 50 };
int i = 0;
do {
System.out.println(a[i]);
i++;
} while (i < 5);

System.out.println(" ");

int j = 4;
do {
System.out.println(a[j]);
j--;
}while(j>=0);
}
}
```

```
10
20
30
40
50

50
40
30
20
10
```

Foreach loop is an **enhanced loop**, it is convenient way to **retrieve elements of an array or collection**
For loops execute a **block of code until an expression becomes false**
for(datatype item : collection){

}

```
package com.dl.two.array;
public class Eg1 {

public static void main(String[] args) {
int a[] = { 10, 20, 30, 40, 50, 60, 70 };
System.out.println(a.length); // 7

for (int i = 0; i < 5; i++) {
System.out.println(a[i]);
}
}
}
```

```
10
20
30
40
50
```

Note:we use for loop **if we know number of iterations in advance**, here **length is 7, but the expected condition is i<5; explicitly we can change the condition here,** if we **add new elements also as per the condition only it will execute**

```
package com.dl.two.array;

public class Eg2 {
public static void main(String[] args) {

int a[] = { 10, 20, 30, 40, 50, 60, 70 };

for (int i = 0; i < a.length; i++) {
System.out.println(a[i]);
}
}
}
```

```
10
20
30
40
50
60
70
```

Note: Here we are using **length variable in condition part**, if we add new elements **it will execute new elements even**

```
package com.dl.two.array;

public class Eg3 {

public static void main(String[] args) {

int a[] = { 10, 20, 30, 40, 50, 60, 70 };

for (int i = 0; i <= a.length; i++) {
System.out.println(a[i]);
}
}
}
```

```
10
20
30
40
50
60
70
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException:
Index 7 out of bounds for length 7
at com.dl.two.array.Eg3.main(Eg3.java:10)
```

Note: if we **use condition i<=a.length; we got an error java.lang.ArrayIndexOutOfBoundsException**
So, in this scenario **we cant add new elements, better to use foreach loop**

```java
package com.dl.two.array;
//foreach loop
public class Eg4 {

public static void main(String[] args) {
int a[] = { 10, 20, 30, 40, 50, 60, 70 };

for (int i : a) {
System.out.println(i);
}
}
}
```

```
10
20
30
40
50
60
70
```

```java
package com.dl.two.array;

//length and length()
public class Eg5 {
public static void main(String[] args) {
int[] a = new int[3];
System.out.println("Length of Int Array: " + a.length);
a[0] = 10;
a[1] = 20;
a[2] = 30;
for (int i = 0; i < a.length; i++) {
System.out.println("Elements: " + a[i]);
}

System.out.println("-----");

String[] s = new String[3];
System.out.println("Length of String Array: " + s.length);
s[0] = "Apple";
s[1] = "Mango";
s[2] = "Goa";
for (int i = 0; i < s.length; i++) {
System.out.println("Length of Strings: " + s[i].length());
}
}
}
```

```
Length of Int Array: 3
Elements: 10
Elements: 20
Elements: 30
-----
Length of String Array: 3
Length of Strings: 5
Length of Strings: 5
Length of Strings: 3
```

```java
package com.dl.two.array;
//MultiDimensional Array
public class Eg6 {

public static void main(String[] args) {

int[][] a = { { 10, 20, 30 }, { 40, 50, 60 }, { 70, 80, 90 } };

System.out.println(a[0][0]);// 10 //a[row-index] [column-index]
System.out.println(a[0][1]);// 20
System.out.println(a[0][2]);// 30

System.out.println(a[1][0]);// 40
System.out.println(a[1][1]);// 50
System.out.println(a[1][2]);// 60

System.out.println(a[2][0]);// 70
System.out.println(a[2][1]);// 80
System.out.println(a[2][2]);// 90

System.out.println(a[0]); //[I@515f550a
System.out.println(a[1].length); //3
}
}
```

```
10
20
30
40
50
60
70
80
90
[I@515f550a
3
```