**Why Array:**

Int x = 20; // we need to have multiple values for x need to store

Int x=20,30,30 -- fail it will throw error because it contains single container

**Arrays:**

* **Collection of same element**
* Arrays are given with notation or Square bracket **[]**

Int [] values = new int[] // Int [] values declaring the variable data type as integer arrays

// = new int[] Assigning the integer array with new keyword for object creating

Int [] values = new int[100] // we are creating 100 slots of array



Int [] values = new int[100]

values [0] = 1000;

values [99] = 93432;

Chart

Description automatically generated

System.out.println(values[55]);

//if u put 55 index it will give 0 because no values assigned to 55th index

//so if we give value it should give 1000

System.out.println(values[0]);

**Key point**

* If u 100 then index will be 0 to 99

So System.out.println(values[100]); // does not exist bcz 100 index is not there

* If u put string in integer array it will error mismatch

**String Array:**

String[] words = new String[] {“My” , “Name”, “is” };

System.out.println(words[2]);

**Another way to do it**

String[] words = new String[3]

words [0] =”My”;

words[1]=”name”;

words[30=” is”;

System.out.println(words[2]);

**Keypoint:**

* Once we run above string array we cannot increase or decrease ,example increase

String[] words = new String[4]

Why we cant bcz we are creating a new object array when we instantiate

So if needed to add 4 arryas we need to create another array

Graphical user interface, text

Description automatically generated with medium confidence

String[] words = new String[4]

If we try to take assigned array value in the different string array it will give NULL

Here

Words = new String [10]

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

System.out.println(words[2]);

Out put is NULL,bcz previous array string becomes Null after we created new string array

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

OK, the old data is gone at this point.

The old data is gone from this variable.

And now the 08 index position no longer it no longer has my it just has this this keyword called null.

**Types of Variables**

There are three types of variables in Java

1. local variable
2. instance variable
3. static variable

**1) Local Variable**

* A variable declared inside the body of the method is called local variable.
* You can use this variable only within that method and the other methods in the class aren't even aware that the variable exists.

**2) Instance Variable**

* A variable declared inside the class but outside the body of the method, is called an instance variable.
* It is not declared as static

**3) Static variable**

* A variable that is declared as **static** is called a static variable. It cannot be local.
* You can create a single copy of the static variable and share it among all the instances of the class. Memory allocation for static variables happens only once when the class is loaded in the memory

**Example to understand the types of variables in java**

public class A

{

static int m=100;//static variable

void method()

{

int n=90;//local variable

}

public static void main(String args[])

{

int data=50;//instance variable

}

}//end of class