**Array is collection of similar data type.**

Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value.

**Advantages of Arrays in Java**

* Java arrays enable you to access any element randomly with the help of indexes
* It is easy to store and manipulate large data sets

**Disadvantages of Arrays in Java**

* The size of the array cannot be increased or decreased once it is declared—arrays have a fixed size
* Java cannot store heterogeneous data. It can only store a single type of primitives

**How to declare and initialise array observe [] is next to type.**

String**[]** cars = {"Volvo", "BMW", "Ford", "Mazda"};

|  |  |  |  |
| --- | --- | --- | --- |
| **0** | **1** | **2** | **3** |
| **10** | **20** | **30** | **40** |
| **2000** | **2004** | **2008** | **2012** |

System.out.println(cars.length);//4

int**[]** myNum = {10, 20, 30, 40};

myNum

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

**How to iterate**

**Int len=** myNum.length;//4

for (int i = 0; i <len; i++) {

System.out.println(myNum[i]);//Value at index

}

**For each loop**

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | 1 | 2 | 3 |
| 10 | 20 | 30 | 40 |

for (int i **:** myNum) {

System.out.println(i);//Value🡺10 20 30 40

}

Declare array and take input from user.

Stack

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |
| 5 | 3 | 2 |

Heap

2000

arr

**import** **static** java.lang.System.***out***;

**import** java.util.\*;

**public** **class** Myclass {

**public** **static** **void** main(String[] args) {

**int**[] arr;

Scanner sc=**new** Scanner(System.***in***);

avg

3.3333

sum

0 5 8 10

r=sc.nextInt();

arr=**new** **int**[r];

**int** i,sum=0;

int len= arr.length;

**for**(i=0;i<len;i++)

{

arr[i]=sc.nextInt();//5 3 2

sum=sum+arr[i];

} float avg= (float)sum/r; out.print(avg);

**for**(**int** n:arr)

***out***.println(n);

}

}

new key word is same as malloc of C which will create memory block at runtime on heap.

What is difference between new and malloc ?

new is operator and malloc is function and both will create a memory block at **runtime** on heap.

Can we take size of array from user?

Yes

**import** **static** java.lang.System.***out***;

**import** java.util.\*;

**public** **class** Myclass {

**public** **static** **void** main(String[] args) {

Scanner sc=**new** Scanner(System.***in***);

***out***.print("enter size of array");

**int** size=sc.nextInt();//8

**int**[] arr=**new** **int**[**size**];//8

**int** i;

int s= arr.length;

**for**(i=0;i<s;i++)

{

arr[i]=sc.nextInt();

}

for (int i = 0; i < arr.length; i++) {

System.out.println(arr[i]);

}

}

}