Array:-

* It is the collection of similar type of data which is stored in continuous memory location.
* We can access the elements of an array by index number. So, in Java array is index based.
* Index position or index number of an array always starts from 0 (zero). So, an array allocates the memory for elements upto size -1.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 23 | 29 | 72 | 64 | 18 | 56 | 34 |

a[5] = 56 a[1] = 29 a[3] = 64

Here, ‘a’ is the subscript value & “[5], [1], [3]” are the index numbers.

Types of Array:-

* Single Dimensional Array (S.D.A.)
* Multi/Double Dimensional Array (D.D.A.)

Single Dimensional Array (S.D.A.):-

* Single dimensional array (S.D.A.) represents a row or a column of elements.

Syntax for declaration of a S.D.A.:-

<Data type> <name of the variable> [size] = new <data type> [size];

Example:-

int a[] = new int[10];  
int a[] = {1, 2, 3, 4, 5};  
char a[] = {‘a’, ‘b’, ‘c’};  
String a[] = {“Ram”, “Shyam”};

Double Dimensional Array (D.D.A.):-

* A double dimensional array represents the elements in both row and column.

Syntax for declaration of a D.D.A.:-

<Data type> <name of the variable> [size][size]= new <data type> [size][size];

row column  
Example:-

int a[3][3] = new int [3][3];

Strings:-

* It is the collection of group of characters enclosed with double quotes (“ “).

Declaration of a String:-

String <variable name>

Example:-

String s = “computer”;

We can create an object to string class by allocating memory using new operator.

String s = new String (“Hello”);

String Methods:-

(String) concat()

* It is used to join two strings.

Example:-

String S1 = “Hello”;  
String S2 = “World”;

S1.concat(S2)

object parameter

* We can use ‘+’ for space in between the two strings while joining them.
* We can also concatenation or join two strings using ‘+’ operator. This is called as string concatenation operator.

Example:-

String S1 = “Hello”;  
String S2 = “World”;

String S3 = S1+S2;

String S3 = S1 + “ “ + S2;

(int) length()

* It returns the length of the string i.e. the number of characters present in a string.

Example:-

String S1 = “Hello”;  
int l = S1.length()  
 = 5

(char) charAt()

* It returns the characters at the specified location or position.

Example:-

String S1 = “Hello”;  
char ch = S1.charAt(2)

= l

(int) compareTo

* This method is used to compare two strings i.e. string is bigger, smaller or equal.

Example:-

String S1 = “Hello”;  
String S2 = “World”;  
S1.compareTo(S2)

* If S1 & S2 strings are equal then, this method will return 0(zero).
* If S1 is greater than S2 then, this method will return a positive number or positive value.
* If S1 is less than S2 then, this method will return a negative number or negative value.

(int)compareToIgnore

* This is same as compareTo but this doesn’t take the case of strings. (Upper & Lower)

boolean equals()

* This method returns true of two strings are same, otherwise false. This is case sensitive.

Example:-

String S1 = “box”;  
String S2 = “box”;

S1.equals(S2)  
  
True

boolean equalsIgnore()

* This method returns true if two strings are equal/same otherwise false. It performs case sensitive.

Example:-

String S1 = “Box”;  
String S2 = “box”;

S1.equalsIgnore(S2)

True

boolean startsWith()

* This method returns true if a string is beginning with sub-string. This method is case sensitive.

Example:-

String S1 = “box”;  
String S2 = “bo”;  
S1.startsWith(S2)  
  
True

boolean endsWith()

* This method returns true if a string ends with the given sub-string. This method is case sensitive.

Example:-

String S1 = “Hello”;  
String S2 = “lo”;  
S1.endsWith(S2)  
  
True

(int) indexOf()

* It returns an integer value i.e. the position of a character or sub-string.

Example:-

String S1 = “hello”;  
S1. indexOf(l)  
  
2

String S1 = “hello”;  
S1. indexOf(l, 2)  
  
3

(int) lastindexOf()

* This method returns the last occurance of the sub-string in the main string. If it is not found then it returns a negative number or negative value.

Example:-

String S1 = “World”;  
S1. lastindexOf(d)

4

(string) replace()

* This method replaces all the occurance of characters by a new character.

Example:-

String S1 = “Hello”;  
 String S2 = S1.replace(‘l’, ‘x’)  
 = “Hexxo”

(string) substring()

* This method is used to extract sub-string from a main string.

(string) substring(int i)

* It returns the characters starting from the ith position till the end of the string.

Example:-

String S1 = “Computer”;  
 String S2 = S1.substring(5)  
 = “ter”

(string) substring(int i, int j)

* This method returns a new string starting from the ith  position to the jth position characters included (jth – 1).

Example:-

String S1 = “Computer”;  
 String S2 = S1.substring(3,6)  
 = “put”