

Strings :- (Sequence of characters)

- String name = new String ("Ajay");
- String name = "Ajay"

Methods -

- ① name.length()
- ② name.toLowerCase()
- ③ name.toUpperCase()
- ④ name.trim()
- ⑤ name.substring (start)
- ⑥ name.substring (start, end)
 └ end index is exclusive
- ⑦ name.replace ("ay", "a")
- ⑧ name.startsWith ("A")
- ⑨ name.endsWith ("y");

⑩ name.charAt(2)

⑪ name.indexOf("a")

⑫ name.indexOf("a", 2) → returns -1 if not found

⑬ name.lastIndexOf("a", 2)

⑭ name.equals("Ajay")

• Escape sequence in Strings -

Ex - \n, \t, \', \"

Decision making instruction in JAVA :-

① If-else statement

```
if (condition_to_be_checked){  
    // code
```

```
    }  
    else { // code  
        }
```

② switch Case Control Instruction

Switch (var) {

Case C1:
 // code

 break;

Case C2:

 // code
 break;

 !

default:
 // code

}

Loop Control Instruction -

- Types -
- ① for loop
 - ② while loop
 - ③ do-while loop

① for loop -

for (initialize; check_bool_exp ; update){

 // code

}

② while loop

```
while ( boolean condition ) {  
    // code  
}
```

③ do - while loop

```
do {  
    // code  
} while ( condition );
```

Arrays

→ Array is a collection of similar types of data.

* Declaration :-

- ① `int[] marks;` → Declaration
`marks = new int[5];` → Memory Allocation
- ② `int marks = new int[5];` → Declaration + Memory Allocation
- ③ `int marks = {100, 70, 80, 71, 78};` → Declaration + Initialise

- length - marks.length
- Display -
for (int i=0; i<marks.length; i+1){
 sout(marks[i]);
}
- for-each loop in Java -
for (int a: marks){
 sout(a);
}
- Multi dimensional Array -
int [][] data = new int [2][3];
String [][] arr = new String [2][4];

Methods in JAVA -

→ Method is a function written inside a class.

eg - int mysum(int a, int b){
 return a+b;
}

// calling
calc obj = new calc();
. obj.mysum(a,b);

* Static Keyword -

- The static keyword is used to associate a method of a given class with the class rather than the object.
- We can call a static method without creating an instance of the class.

- In JAVA, the main() method is static, so that JVM can tell the main() method directly without allocating any extra memory for object creation.
- All objects share the static method in a class.

```
eg - static int sum( int a, int b){  
    return a+b;  
}  
  
System.out.println( sum( a, b )); // calling.
```

Method Overloading in JAVA -

Two or more methods with the same name but with different parameters.

4 Ways of Method Overloading

- ① By changing the return type of the different methods

```
eg - int multiply (int a, int b){  
    return a×b;  
}  
double multiply (int a, int b){  
    return a×b;  
}
```

- ② By changing the number of arguments passed :-

```
eg - int multiply (int a, int b){  
    return a×b;  
}  
int multiply (int a, int b, int c){  
    return a×b×c;  
}.
```

Variable Arguments (VarArgs) -

Syntax -

```
public static void foo (int ...arr) {
```

// arr is available here as int []arr.

}

To solve the problem of function overloading

Variable Arguments was introduced.

```
eg - static int add (int ...arr){
```

```
    int result = 0;
```

```
    for (int a : arr) {
```

```
        result += a;
```

```
    } } return result;
```

// Calling -
cout (add(1,2)) ; cout (add(1,2,3,4));