METHODS

What are methods:

- Methods are members of classes which provide functionality for classes.
- We can write our own methods in the classes.
- The functions performing on the data are known as methods.
- When a method returns a value then the method itself takes the value.
- A method will have its own copy of variable.
- Skeleton of method:

returnType methodName(parameter list) →signature/header

{

}

Example program:

```
class test

{
   static int max(int x, int y)
   {
      x++;
      if(x>y)
        return x;
      else
        return y;
   }
   public static void main(String args[])
   {
     int a=10,b=15,c;
     c=max(a,b);
     System.out.println(c);
   }
}
```

Passing object as parameters:

- To call a method from main method it is needed to be maid static.
- When the method is called the value of actual parameters are copied in formal parameters which is the only parameter passing method in java.
- The address of object in formal and actual parameter is Same.
- String cannot be modified as it is immutable.
- A method can also return an object.

  Example program:

```
class test

{
   static void update(int A[])
   {
     A[0]=25;
   }
   public static void main(String args[])
   {
      int A[]={2,3,4,5,6};
      update(A);
      System.out.println(A[0]);
   }
}
```

- A method can have its object as the return type.

Parameter passing in java.

- Whoever is calling a method is called as a caller or a method call.
- The method which is called by a caller is known as called method.
- The parameters/arguments passed in calling method are called as actual parameters.

- And the parameters of a called method are called as formal parameters.
- Formal parameters are nothing but input into a method where the return type is known as output to a method.
- The contents of actual parameters are copied in formal parameters is the only method of parameter passing in java.
- Passing of objects also follow the same method.
- Parameter passing for primitive datatypes the values are copied in formal parameters, whereas in parameter passing of objects the reference of the object id is copied in formal parameters.
- In short the primitive datatypes are passed by value and the objects are passed by reference.

Example program:

```
class test

{
   int add(int x, int y)
   {
     int z;
     z=x+y;
     return z;
   }
   public static void main(String args[])
   {
      int a=10,b=5,c;
      c=add(a,b);
      System.out.prinlt(c);
   }
}
```

Method overloading:

- Method overloading means writing more than one method having same names but different parameter list or data types.

- Compiler will call the corresponding method depending upon the parameter list.

Variable arguments:

- It is nothing but writing a single method which can run for number of parameters of same data types.
- For example – void show(int ...x)

  Where ... represents the variable arguments.
- It is similar to ellipsis in c/c++.
- The parameters passed are converted into an array.
- The parameters can be directly passed using an anonymous array.
- Variable argument should always be the last parameter.
- Printf is based on variable arguments in version 1.7 java.

Command line arguments:

- Java programs can utilize command line arguments.
- Dos file is used for command line arguments.
- There are different commands like.
  - cls: to clear the screen.
  - dir : to display the contents of the disk.
  - cd windows : to change the directory.
- C: \Windows> dir v*.*
  - the above is command line in which

  Dir is command and v*.* is an argument

RECURSION

- A recursive method is the one which will call itself.
- When the recursive function can not call itself further because of the base condition it will return back along the same path.
- Not to make the program lengthy loops are used instead of recursive functions.
- The recursions are used in problem solving.