**Java Methods**  
   
I) What is Method?  
   
Method is a set of statements to perform an operation, methods are also known as procedures or functions.  
  
Generally, we use Functions (Built in and user defined) in computer programming or scripting.  
  
In Java we use methods for code reusability.  
  
II) Use of Methods:  
   
Whenever we want perform same operations multiple times then we use methods, using methods we can reduce the code size.  
  
III) Types of Methods:  
   
Basically we have two types of methods in java.  
  
i) Built in methods  
  
ii) User defined methods.  
  
IV) Built in methods  
   
> Java has a library of (pre-defined) classes and methods, organized in packages.  
  
> In order to use Built in methods, we import the packages (or classes individually). (Group of classes)  
  
> java.lang package is automatically imported, in any Java program.  
  
> Using import keyword we can import java pre-defined libraries.(We can import entire packages or particular class)  
  
Categories of Built in methods:  
   
a) String methods  
  
b) Array methods  
  
c) Character methods  
  
d) Number methods  
  
e) Date & Time methods  
  
etc...  
  
**Example for Built in method:**  
System.out.println("Hello Selenium");  
-------------------------------------  
**V) User defined methods**   
  
> Methods in Objects oriented Programming equivalent of Functions in Non object oriented programming.  
  
**Types of User defined methods**  
    i) Method without return any value  
    ii) Method with return values.  
  
Writing method:  
  
**Syntax:**  
modifier returnType methodName(Parameters) {  
// Method body  
}

public

modifier -It is optional, it defines access type of the method  
  
returnType - Method may retrun a value  
  
methodName - Name of the method.  
  
parameters - Parameters are optional, we can use mutiple parameters by separating with ,  
  
method body - set of statements define that what the method does.  
------------------------------------------------  
**Example 1: Method with returning a value.**  
public static int add(int a, int b){  
Statements  
----------  
-----------  
}  
  
**Method calling:**  
   
dataType variableName = methodName(Values)  
--------------  
int abc = add(5, 3);  
    System.out.println(abc);  
}  
    public static int add(int a, int b){  
        int result;  
        result = a + b;  
        return result;  
    }  
}  
----------------------------------  
**Example 2: Method without returning any value**  
   
modifier methodName(Parameters) {  
//Method body  
}  
  
public static void add(int a, int b) {  
Statements  
-----------  
--------  
}  
  
Examples:  
  
Internal method:  
----------------  
public static void main(String [] args){  
studentRank(499);  
}  
       public static void studentRank(int marks) {  
              if (marks >= 600) {  
                 System.out.println("Rank:A1");  
              }  
              else if (marks >= 500) {  
                 System.out.println("Rank:A2");  
              }  
              else {  
                 System.out.println("Rank:A3");  
              }  
       }  
------------------------------------  
**Example 3: External Method (Calling from external or another class).**  
   
import xyza.CopyArray;  
public class Apple extends CopyArray {  
  
    public static void main (String [] args){  
        studentRank(678);  
    }  
}  
------------------------------------------  
**VI) Method Overloading in Java**  
If a class have multiple methods with same name, but different parameters, It is known as Method overloading.  
  
There are two ways to overlad the Method in java  
  
**i) By changing number of Arguments.**  
Example: we have Two methods in our class with the name of add.  
  
a) int add (int a, int b)  
  
b) int add (int a, int b, int c)  
  
**ii) By changing data types.**  
a) int add (int a, int b)  
  
b) double add(double a, double b)  
----------------------------  
**Example Java Program for Method Overloading:**  
package javaExamples;  
  
public class MethodOverLoading {  
      
public static void main (String []args){  
    int x = add(5, 7);  
    int y = add(5, 7, 9);  
    double z = add(5.234, 7.23);  
    System.out.println(x);  
    System.out.println(y);  
    System.out.println(z);  
}  
public static int add(int a, int b){  
    int result;  
    result = a + b;  
    return result;  
}  
public static int add(int a, int b, int c){  
    int result;  
    result = a + b + c;  
    return result;  
}  
public static double add(double a, double b){  
    double result;  
    result = a + b;  
    return result;  
}  
}  
  
**Advantages of Method Overloading:**  
> It increases the readability of the Program.