1. Creating and calling method

public class meth {

    static void myMethod() {

        System.out.println("I just got executed!");

    }

  public static void main(String[] args) {

    myMethod();

  }

}

1. Method Overloading
2. public class over {
3. public  int add(int a,int b)
4. {
5. int c = a+b;                                // this method is not static there fore we create obj to use it .
6. return c;
7. }
8. public static float add(float a,float c)
9. {
10. return a;
11. }
12. public static float add(float g)                    // static : always our main is static therefore  whenever we call method it should be static or create a object of class to use method, like we use it in fist call of method
13. {
14. return g\*g;
15. }
16. public static void main(String[] args) {
17. over obj = new over();                          // to call method we create an obj
18. int i = obj.add(5,9);                       //now we link it to pass arguments
19. float j = add(5.5f);
20. float k = add(5.5f,5.0f);                   //5.0f here f is very important to pass float value in method, else it will generate error
22. System.out.println(i);
23. System.out.println(j);
24. System.out.println(k);
26. }
27. }
28. Class method
    1. Static and public
29. public class staticpublic {
30. public  int add(int a,int b)
31. {
32. int c = a+b;                                // this method is not static there fore we create obj to use it .
33. return c;
34. }
35. public static float add(float a,float c)
36. {
37. return a;
38. }
39. public static float add(float g)                    // static : always our main is static therefore  whenever we call method it should be static or create a object of class to use method, like we use it in fist call of method
40. {
41. return g\*g;
42. }
43. public static void main(String[] args) {
44. over obj = new over();                          // to call method we create an obj
45. int i = obj.add(5,9);                       //now we link it to pass arguments
46. float j = add(5.5f);
47. float k = add(5.5f,5.0f);                   //5.0f here f is very important to pass float value in method, else it will generate error
49. System.out.println(i);
50. System.out.println(j);
51. System.out.println(k);
53. }
54. }
    1. Accessing method with and without object
55. Constructors
    1. Default
    2. Parameterized
    3. Copy contructor
56. Modifiers :
    1. Access modifiers for classs
       1. Public
       2. Default
    2. Access modifiers for attributes and methods
       1. Public
       2. Private
       3. Defualt
       4. Procteded