Using "static" Methods

In Java just like we can have static data, similarly we also can have static methods in our class and Java recommends **two situations** when we should make our method static

1. When we have a method in the class which is **only accessing static data members of the class** .

For example: In the previous program the method **showNextId()** must be declared as static because it is only accessing static data **nextId** in its body. The most important benefit of doing this will be that now we can call the method **showNextId()** without using any object, by simply using class name.

Code

=====

```
public static void showNextId()
{
  System.out.println("Next emp id will be"+nextId);
}
```

call

===

```
Emp.showNextId();
```

2. Whenever we have a class without any instance variable then all the method of that class must be declared as static.

For example:

```
public class MyMath {
    public static int add(int a,int b)
    {
        int c;
        c=a+b;
        return c;
    }
    public static int max(int a,int b)
    {
        if(a>b)
            return a;
        return b;
    }
}
```

Special Note:

========

Based on the above rule, the developers of Java langauge themselves have declared some very commonly use classes with every method as static and they are:

java.lang.Math, java.util.Arrays, java.util.Collections

```
class Account
{
    private int accNo;
    private String name;
    private double bal;
    private static double roi;

    static
    {

    }
```