2.9 Static Members

- ***Static*** is a Non Access Modifier.
- **The Static** modifier can be applied to a variable or Method or block or inner Class.
- ***Static members*** belong to Class only not an instance.
- A Static Method **cannot** access an *instance variable*.
- Static Methods **cannot** be *overridden a*s they are Class specific and don't belong to an Instance.
- Static Methods can be ***redefined***.
- If a Class contains any static blocks then that block will be executed only when the Class is loaded in JVM.
- Creating multiple instances does not execute the static block multiple time.
- Only the constructor will be executed multiple times.
- If Class.forName("*class_name*") is called then the static block of the Class will get executed.

❑ Purpose of Static Keyword in Java

The static word can be used to attach a Variable or Method to a Class. The variable or Method that are marked static belong to the Class rather than to any particular instance.

2.9.1 Static Keyword Rules

- ***Variable or Methods*** marked static belong to the **Class** rather then to any particular Instance.
- **Static Method or variable** can be used without creating or referencing any instance of the Class.
- If there are instances, a static variable of a Class will be shared by all instances of that class, This will result in **only one copy**.
- A static Method can't access a non static variable nor can directly invoke non static Method (It can invoke or access Method or variable via *instances*).

The Static keyword is applicable to

Method

Variable

Class nested within another Class

Initialization Block

The Static keyword Not Applicable to

Class (Not Nested)

Constructor

Interfaces

Method Local Inner Classes (Difference then nested class)

Inner Class Methods

Instance Variables

Local Variables

2.9.2 Class Variables – Static Fields

- If you declare any variable as static, it is known static variable.
- The static variable can be used to refer the common property of all objects (that is not unique for each object) e.g. company name of employees, college name of students etc.
- The static variable gets memory only once in class area at the time of class loading.
- Advantage of static variable is, it makes your program **memory efficient** (i.e. it saves memory).

Example of static variable

```
class Student8{
   int rollno;
   String name;
   static String college ="ITS";

   Student8(int r,String n){
   rollno = r;
   name = n;
   }
 void display (){System.out.println(rollno+" "+name+" "+college);
}
 public static void main(String args[]){
 Student8 s1 = new Student8(111,"Karan");
 Student8 s2 = new Student8(222,"Aryan");
 s1.display();
 s2.display();
}
}
```

```
Output:
111 Karan ITS
222 Aryan ITS
```

2.9.3 Class Methods – Static Methods

- If you apply static keyword with any method, it is known as static method.
- A static method belongs to the class rather than object of a class.
- A static method can be invoked without the need for creating an instance of a class.
- static method can access static data member and can change the value of it.

Example of static method

```
class Student9{
    int rollno;
    String name;
    static String college = "ITS";

    static void change(){
    college = "NMPI";
    }

    Student9(int r, String n){
    rollno = r;
    name = n;
    }

    void display ()
    {
      System.out.println(rollno+" "+name+" "+college);
}
   public static void main(String args[]){
   Student9.change();

   Student9 s1 = new Student9 (111,"Karan");
   Student9 s2 = new Student9 (222,"Aryan");
   Student9 s3 = new Student9 (333,"Sonoo");

   s1.display();
   s2.display();
   s3.display();
   }
}
```

Output:

```
     111 Karan NMPI
     222 Aryan NMPI
     333 Sonoo NMPI
```