Java static keyword

The static keyword in Java is used for memory management mainly. We can apply static keyword with variables, methods, blocks and nested classes. The static keyword belongs to the class than an instance of the class.

The static can be:

1. Variable (also known as a class variable)
2. Method (also known as a class method)
3. Block
4. Nested class

1) Java static variable

If you declare any variable as static, it is known as a static variable.

- The static variable can be used to refer to the common property of all objects (which is not unique for each object), for example, the company name of employees, college name of students, etc.
- The static variable gets memory only once in the class area at the time of class loading.

Advantages of static variable

It makes your program **memory efficient** (i.e., it saves memory).

Example of static variable

//Java Program to demonstrate the use of static variable

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

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

```
  void display ()
  {
        System.out.println(rollno+" "+name+" "+college);
  }
}
public class TestStaticVariable1
{
   public static void main(String args[])
 {
 Student s1 = new Student(111,"Karan");
 Student s2 = new Student(222,"Aryan");
 s1.display();
 s2.display();
 }
}
```

2) Java static method

If you apply static keyword with any method, it is known as static method.

- A static method belongs to the class rather than the object of a class.
- A static method can be invoked without the need for creating an instance of a class.
- A static method can access static data member and can change the value of it.

Example of static method

```
//Java Program to demonstrate the use of a static method.

class Student

{

    int rollno;

    String name;

    static String college = "ITS";

        static void change()
```

```
{
    college = "BBDIT";
    }
        Student(int r, String n)
{
    rollno = r;
    name = n;
    }
        void display()
{
      System.out.println(rollno+" "+name+" "+college);
}
}
public class TestStaticMethod
{
   public static void main(String args[])
{
   Student.change();
   Student s1 = new Student(111,"Karan");
   Student s2 = new Student(222,"Aryan");
   Student s3 = new Student(333,"Sonoo");
   s1.display();
   s2.display();
   s3.display();
   }
}
```

Another example of a static method that performs a normal calculation

//Java Program to get the cube of a given number using the static method

```
class Calculate
{
 static int cube(int x)
{
 return x*x*x;
 }

 public static void main(String args[])
{
 int result=Calculate.cube(5);
 System.out.println(result);
 }
}
```

Q) Why is the Java main method static?

Ans) It is because the object is not required to call a static method. If it were a non-static method, JVM creates an object first then call main() method that will lead the problem of extra memory allocation.

3) Java static block

Is used to initialize the static data member.

It is executed before the main method at the time of classloading.

Example of static block

```
class A2
{
```

```
  Static
{
System.out.println("static block is invoked");
}
  public static void main(String args[])
{
   System.out.println("Hello main");
  }
}
```

// Java program to demonstrate use of static blocks

```
class Test
{
        static int a = 10;
        static int b;
        static
         {
                System.out.println("Static block initialized.");
                b = a * 4;
         }
        public static void main(String[] args)
        {
        System.out.println("from main");
        System.out.println("Value of a : "+a);
        System.out.println("Value of b : "+b);
        }
}
```