**1.Create a class that keeps track of the number of instances created. Implement a static variable and method to accomplish this.**

**Ans:**

```
public class assignmnet {
    public static void main(String[] args) {
        StaticAssignment st = new StaticAssignment();
        StaticAssignment st1 = new StaticAssignment();
        StaticAssignment st2 = new StaticAssignment();
         StaticAssignment st3 = new StaticAssignment();
        System.out.println("Number of instances created: " +
StaticAssignment.getInstanceCount());
    }
}

class StaticAssignment {
    private static int instanceCount = 0;

    public StaticAssignment() {
        instanceCount++;
    }

    public static int getInstanceCount() {
        return instanceCount;
    }
}
OUTPUT:
Number of instances created: 4
```

**2. Write a program and create a constructor with parameters and initialise the variable using a constructor.**

**Ans:**

```
public class assignmnet {
    public static void main(String[] args) {
        A ref = new A(10, "Avijit");
        ref.disp();
    }
}

class A {
   private int a;
    private String str;
```

```
  A(int a, String str) {
    this.a = a;
    this.str = str;
  }
  public void disp(){
     System.out.println(a + " " + str);
     System.out.println();
  }
}
OUTPUT:
10 Avijit
```

**3. Use a private keyword for a variable and use setter and getter methods to initialise and print the values.**

**Ans:**

```
class Student {
  private int age;
  private String name;
  private int rollNo;

  public int getAge() {
     return age;
  }

  public void setAge(int age) {
     this.age = age;
  }

  public String getName() {
     return name;
  }

  public void setName(String name) {
     this.name = name;
  }

  public int getRollNo() {
     return rollNo;
  }

  public void setRollNo(int rollNo) {
     this.rollNo = rollNo;
  }
```

```
    public void studentDetails() {
        System.out.println(name + " and " + age + " " + rollNo);
    }
}
public class launch2 {
    public static void main(String[] args) {
        Student std = new Student();
        std.setAge(24);
        std.setName("Avijit");
        std.setRollNo(1);

        // std.studentDetails();
        int stdent1 = std.getAge();
        String studentName = std.getName();
        int studentRoll = std.getRollNo();
        System.out.print(stdent1 + "");
        System.out.print(studentName + " ");
        System.out.print(studentRoll);
        System.out.println();

        Student std1 = new Student();
        std1.setAge(18);
        std1.setName("Demon");
        std1.setRollNo(2);
        int stdent2 = std1.getAge();
        String student2Name = std1.getName();
        int student2Roll = std1.getRollNo();
        System.out.print(stdent2 + " ");
        System.out.print(student2Name + " ");
        System.out.print(student2Roll);
        System.out.println();
    }
}
OUTPUT:
24 Avijit 1
18 Demon 2
```

**4. Write a program to call a method without creating an object of a class.**

**Ans:**

```
public class assignmnet {
    public static void main(String[] args) {
            assignmnet.disp();
    }
    public static void disp() {
```

```
      System.out.println("Static Method");
   }
}
OUTPUT: Static Method
```

**5. Write a program which has static block and constructor overloading, initialise variables using constructors and print it.**

**Ans:**

```
public class assignmnet {
   public static void main(String[] args) {
         assignmnet.disp();
   }
   public static void disp() {
      System.out.println("Static Method");
   }
}
OUTPUT: Static Method
```