1. Write a Simple Program in Java to Print First Fifty Prime Numbers.

```
import java.util.*;

public class Practical1{

   public static void main(String args[])
   {
      Scanner s = new Scanner(System.in);
      int x, y, flg,N;

      System.out.print("Enter the Number : ");
      N = s.nextInt();

      System.out.println("All the Prime numbers within 1 and " + N+ " are:");

      for (x = 1; x <= N; x++) {

         if (x == 1 || x == 0)
            continue;

         flg = 1;
         for (y = 2; y <= x / 2; ++y) {
            if (x % y == 0) {
               flg = 0;
               break;
            }
         }

         if (flg == 1)
            System.out.print(x + " ");
      }
   }
}
```

Output :-

```
Enter the Number : 50
All the Prime numbers within 1 and 50 are:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47
```

2. Write a Program in Java to Print Factorial of Given Number Using Recursion.

```
import java.util.*;

public class Practical2 {

   public static int fact(int n)
{
      if(n==0){
         return 1;
      }
      else if(n==1){
         return 1;
      }
      return fact(n-1)*n;
   }

   public static void main(String[] args) {

         Scanner sc = new Scanner(System.in);
         int n;

         System.out.println(“Enter the Number : ”);
         n = sc.nextInt();

      int Result;
      Result = fact(5);
      System.out.println("Factorial is : "+Result);

   }
}
```

Output :-

```
Enter the Number : 5
Factorial is : 120
```

3. Write a program in Java to Print Fibonacci series in Given Series.

```
import java.util.*;

public class Practical3 {

   public static void main(String[] args) {

      Scanner s = new Scanner(System.in);

      System.out.print("Enter the Number : ");
      int n = s.nextInt();

      int n1 = 0;
      int n2 = 1;

      System.out.print("Fibonacci Series is : "+n1 + " "+n2);

      for(int i=2;i<=n;i++){
         int n3 = n1+n2;
         System.out.print(" "+n3);
         n1=n2;
         n2=n3;
      }

   }
}
```

Output :-

```
Enter the Number : 50
Fibonacci Series is :
0,1,1,2,3,5,8,13,21,34,55,89,144,233,377,610,987,1597,2584,4181,6765,10946,1
7711,28657,46368,75025,121393,196418,317811,514229,832040,1346269,217
8309,3524578,5702887,9227465,14930352,24157817,39088169,63245986,102
334155,165580141,267914296,433494437,701408733,1134903170,18363119
03,-1323752223,512559680,-811192543,-298632863
```

4. Write a Program in Java to Demonstrate Command Line Arguments.

```
public class CommandLineArgument{
   public static void main(String args[]){

      int i;
      int count = 0;

      for(i=0 ; i<args.length ; i++){
        count = count + Integer.parseInt(args[i]);
      }

      System.out.println("Sum is : "+count);

      }
   }
```

Output :-

```
C:\Users\lalit\Desktop\TY BSC CS 2024\Java Tutorial>
javac CommandLineArgument.java

C:\Users\lalit\Desktop\TY BSC CS 2024\Java Tutorial>
java CommandLineArgument 1 2 3 4 5
Sum is : 15
```

5. Write a Program in Java to Create Student Information Using Array.

```
import java.util.*;

public class StudentArray {
   Scanner s = new Scanner(System.in);
   int id;
   String name;
   float fees;

   public void getdata(int i){
      System.out.println("Enter the Student "+(i+1) +" Detail's : ");
      System.out.print("Enter the Student Id : ");
      id = s.nextInt();
      System.out.print("Enter the Student Name : ");
      name = s.next();
      System.out.print("Enter the Student Fees : ");
      fees = s.nextFloat();

   }

   public void Display(int i){
      System.out.println("Student "+(i+1) +" Details is : ");
      System.out.println("Student Id is : "+id);
      System.out.println("Student Name is :"+name);
      System.out.println("Student Fees is : "+fees);
   }

   public static void main(String[] args) {

     StudentArray s[] = new StudentArray[5]; // Decleration...

      for(int i=0 ; i<5 ; i++){
         s[i] = new StudentArray(); // Initialization....
         s[i].getdata(i);

      }

      for(int i=0 ; i<5 ; i++){
         s[i].Display(i);
      }
   }
}
```

Output :-

```
Enter the Student Id : 101
Enter the Student Name : Yogesh
Enter the Student Fees : 1000
Enter the Student 2 Detail's :
Enter the Student Id : 102
Enter the Student Name : Chetan
Enter the Student Fees : 2000
Enter the Student 3 Detail's :
Enter the Student Id : 103
Enter the Student Name : Dipak
Enter the Student Fees : 3000
Enter the Student 4 Detail's :
Enter the Student Id : 104
Enter the Student Name : Shubham
Enter the Student Fees : 4000
Enter the Student 5 Detail's :
Enter the Student Id : 105
Enter the Student Name : Jayesh
Enter the Student Fees : 5000
Student 1 Details is :
Student Id is : 101
Student Name is :Yogesh
Student Fees is : 1000.0
Student 2 Details is :
Student Id is : 102
Student Name is :Chetan
Student Fees is : 2000.0
Student 3 Details is :
Student Id is : 103
Student Name is :Dipak
Student Fees is : 3000.0
Student 4 Details is :
Student Id is : 104
Student Name is :Shubham
Student Fees is : 4000.0
Student 5 Details is :
Student Id is : 105
Student Name is :Jayesh
Student Fees is : 5000.0
```

6. Write a Program in Java to Implement User Defined Package.

```
//Save this Program as First.java
//in mypack Folder

package mypack;
public class First{
   public void msg(){
      System.out.println("I am a First CLass Method in mypack. Package.");
   }
}


// in the another Program import the Package

import mypack.*;
public class Practical5 {
   public static void main(String[] args) {
   First obj = new First();
   obj.msg();
   }
}
```

Output :-

I am a First Class Method in mypack Package.

7. Write a Program in Java to Implement Default & Parameterized Constructor.

```
class Demo{

   Demo(){
      System.out.println("Default Constructor is invoked..!!");
   }

   Demo(int a){
      System.out.println("Value is : "+a);
   }
}

public class Constructor {
   public static void main(String[] args) {
      Demo obj = new Demo();
      Demo obj2 = new Demo(101);
   }
}
```

Output :-

```
Default Constructor is invoked..!!
Value is : 101.
```

8. Write a Program in Java to Demonstrate Various Operations on String Functions.

```
public class StringFunctions {

   public static void main(String[] args) {

      String str1 = "Hello";
      String str2 = "World";

      //String Concatination
      System.out.println("Concatinated String is : "+(str1+str2));//Using Operator
      System.out.println("Concatinated String is : "+str1.concat(str2));

      //Find Length of String
      System.out.println("Length of String 1 is : "+str1.length());
      //UpperCase and LowerCase Conversion of String

      System.out.println("Lower Case String is : "+str1.toLowerCase());
      System.out.println("Upper Case String is : "+str1.toUpperCase());

      //Substring :- Extract the Substring
      System.out.println("SubString is : "+str2.substring(1, 4));//orl

      //Replace the String to old character to new Character
      System.out.println("Replaced String is : "+str1.replace('e', 'o'));

      //Trim the String to Leading and trailing
      System.out.println("Trimed String is : "+(str1.trim()));

      //check the equality of String
      System.out.println("Equality of String1 & String2 : "+(str1.equals(str2)));

      //compare the Two String
      System.out.println("Comparison of Two Strings is "+(str1.compareTo(str2)));

      //See the Paticular character Location by indexOf
      System.out.println("Searched Element is : "+str1.indexOf('o'));

   }
}
```

Output :-

```
Concatinated String is : HelloWorld
Concatinated String is : HelloWorld
Length of String 1 is : 5
Lower Case String is : hello
Upper Case String is : HELLO
SubString is : orl
Replaced String is : Hollo
Trimed String is : Hello
Equality of String1 & String2 : false
Comparison of Two Strings is : -15
Searched Element is : 4
```

9. Write a Program in Java to Demonstrate Wrapper Classes.

```
public class WrapperClass {

   public static void main(String[] args) {

      Integer a = Integer.parseInt("501");
      Byte b = Byte.parseByte("2");
      Long l = Long.parseLong("1234567890");
      Float f = Float.parseFloat("12.11");
      Double d = Double.parseDouble("3.14");

      System.out.println("After Converted Integer Value is : "+a);
      System.out.println("After Converted Byte Value is : "+b);
      System.out.println("After Converted Long Value is : "+l);
      System.out.println("After Converted Floating Value is : "+f);
      System.out.println("After Converted Double Value is : "+d);

   }

}
```

Output :-

```
After Converted Integer Value is : 501
After Converted Byte Value is : 2
After Converted Long Value is : 1234567890
After Converted Floating Value is : 12.11
After Converted Double Value is : 3.14
```

10. Write a Program in Java to Demonstrate Abstract Class.

```
abstract class ParentClass {

   abstract void method();

}

class ChildClass extends ParentClass {

   void method() {
      System.out.println("I am a Abstract Class Method..!!");
   }
}

public class AbstractClass {

   public static void main(String[] args) {

      ChildClass obj = new ChildClass();
      obj.method();
   }
}
```

Output :-

```
I am a Abstract Class Method..!!
```

11. Write a Program in Java to Implement Inheritance.

```
// Demonstration of Inheritance in Java.....

class Parent{

   void ParentMethod(){
      System.out.println("I am Parent Class Method.");
   }
}

class Child extends Parent{

   void ChildMethod(){
      ParentMethod();
      System.out.println("I am a Child Class Method.");
   }
}

public class Single_Inheritance{

   public static void main(String[] args) {

      Child obj = new Child();
      obj.ChildMethod();
   }
}
```

Ouput :-

```
I am Parent Class Method.
I am a Child Class Method.
```

12. Write a Program in Java to Demonstrate Inner Class.

```
public class OuterClass {

   public int id = 101;

   public class Inner{
      public void print(){
         System.out.println("This is a Content of Innner Class..!!");
      }
   }

   public void OuterPrint(){
      Inner obj = new Inner();
      obj.print();
   }

}

public class InnerClassDemo{

   public static void main(String[] args) {

      OuterClass obj = new OuterClass();
      obj.OuterPrint();
   }
}
```

Ouput :-

```
This is a Content of Innner Class..!!
```

13. Write a Program in Java to Demonstrate Reflection.

```
import java.lang.Class;
import java.lang.reflect.*;

class SuperDemo{

}
class Demo extends SuperDemo{

   Demo(){
      System.out.println("Default Constructor is invoked..!!");
   }
   void Display(){
      System.out.println("I am a Display Method..!!");
   }
}

public class Reflection {
   public static void main(String[] args) {
      Demo d1 = new Demo();

      Class obj = d1.getClass();

      String Name = obj.getName();

      System.out.println("Name of Class is : "+Name);

      int modifier = obj.getModifiers();
      String Mod = Modifier.toString(modifier);
      System.out.println("Modifier is : "+Mod);

      Class superClass = obj.getSuperclass();
      System.out.println("Name of Super Class is : "+superClass.getName());

   }
}
```

Output :-

```
Default Constructor is invoked..!!
Name of Class is : Demo
Modifier is :
Name of Super Class is : SuperDemo
```

14. Write a Program in Java to Demonstrate Exception Handling.

```
import java.util.*;
import java.lang.*;
public class Exception {

   private static java.lang.Exception NullPointerException;

   public static void main(String[] args)throws java.lang.Exception {
      Scanner sc = new Scanner(System.in);

      int a,b;
      int c;

      System.out.print("Enter the Value of a : ");
      a = sc.nextInt();

      System.out.print("Enter the Value of b : ");
      b = sc.nextInt();

      try{
         if(b==0){

            throw NullPointerException;
         }
         else{
            c = a/b;
            System.out.println("Divison is : "+c);
         }
      }
      catch(NullPointerException e){
         System.out.println("Divison By Zero is Not Possible");
      }
   }
}
```

Output :-

```
Enter the Value of a : 12
Enter the Value of b : 0
Divison By Zero is Not Possible
```

15. Write a Program in Java to Demonstrate Text Stream Object that Take Input From User & Write it into Text File.

```
import java.io.*;

public class File1 {
   public static void main(String[] args) throws IOException {
      FileInputStream in = null;
      FileOutputStream out = null;

      try {
       in = new FileInputStream("D:/TY BSC CS 2024/JavaTutorial/sourcefile.txt");
       out = new FileOutputStream("D:/TY BSC CS 2024/JaTutorial/targetfile.txt");

         int c;
         while ((c = in.read()) != -1) {
            out.write(c);
         }
      }

      finally {
         if (in != null) {
            in.close();
         }
         if (out != null) {
            in.close();
         }
      }

   }
}
```

Output :-

//Content of sourcefile.txt is : HelloJavaProgramming

//Before Program Run targetfile.txt is Empty

//After Execute the Program
The Content of targetfile.txt is HelloJavaProgramming.