JAVA LABORATORY WEEK-2

PROGRAM ON LOOPS

1.Write a Java program to calculate the power of a number without using the Math.pow() method, using a for loop.

PROGRAM:

```
import java.util.Scanner;

public class d_power {
   public static void main(String[] args) {
      Scanner s = new Scanner(System.in);
      int num = s.nextInt();
      int power = s.nextInt();
      long ans= 1;
      for (int i = 0; i < power; i++) {
         ans*= num;
      }
      System.out.print(“The power of number is : ”+ans);

   }
}
```

OUTPUT

```
2 3

The power of the number is : 8
```

2.Write a Java program to simulate the Collatz Conjecture for a given number using a while loop.

Collatz Conjecture Logic:

- **keep calculating the next number in the sequence until it reaches 1.**
- **If the current number is even, it is divided by 2.**
- **If the current number is odd, it is replaced by 3 * number + 1.**

Sample Input and Output

- **Input: 6 Output: 6 3 10 5 16 8 4 2 1**
- **Input: 19 Output: 19 58 29 88 44 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1**

PROGRAM:

```
import java.util.*;

public class CollatzConjecture {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);

      System.out.print("Enter a number to apply Collatz Conjecture: ");
      int number = scanner.nextInt();

      System.out.print(number + " ");

      while (number != 1) {
         if (number % 2 == 0) {

            number = number / 2;
         } else {

            number = 3 * number + 1;
         }
         System.out.print(number + " ");
      }
   }
}
```

OUTPUT:

```
Enter a number to apply Collatz Conjecture: 6
     6    3 10 5 16 8 4 2 1
```

3.Write a Java program to find the reverse of a number using a while loop.

PROGRAM:

```
import java.util.*;

public class ReverseNumber {
   public static void main(String[] args) {
      Scanner s = new Scanner(System.in);

      System.out.print("Enter a number: ");
      int number = s.nextInt();

      int rev= 0;

      while (number != 0) {
        rev = rev* 10 +number % 10;
         number = number / 10;
      }
```

```
        System.out.println("Reversed number: " + rev);
    }
}
```

OUTPUT:

```
Enter a number: 478
Reversed number: 874
```

4.Write a Java program to print all prime numbers between 1 and 100 using nested for loops.

PROGRAM:

```
import java.util.*;
class D_Prime{
    public static void main(String[] args) {
        System.out.println("Prime numbers between 1 and 100:");

        for (int i = 2; i <= 100; i++) {
            boolean isPrime = true;

            for (int j = 2; j *j<=i; j++) {
                if (i % j == 0) {
                    isPrime = false;
                    break;
                }
            }
            if (isPrime) {
                System.out.print(i + " ");
            }
        }
    }
}
```

OUTPUT:

```
Prime numbers between 1 and 100:

            3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97
```

5.Write a Java program to find the LCM (Least Common Multiple) and greatest common divisor (GCD) of two numbers using a while loop

PROGRAM:

```
import java.util.Scanner;

public class D_lcm_gcd {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);

      System.out.print("Enter first number: ");
      int a = scanner.nextInt();
      System.out.print("Enter second number: ");
      int b= scanner.nextInt();

      int gcd = findGCD(a, b);

      int lcm = (a * b) / gcd;

      System.out.println("GCD  is: " + gcd);
      System.out.println("LCM is: " + lcm);
   }
   public static int findGCD(int x, int y) {
      while (y != 0) {
         int temp = y;
         x = x % y;
         x = temp;
      }
      return x;
   }
}
```

OUTPUT:

```
Enter first number: 6
Enter second number: 12
GCD is: 6
LCM is: 12
```

6.Write a Java program to print the first n terms of the Fibonacci series using a for loop.

PROGRAM:

```
import java.util.Scanner;

public class FibonacciSeries {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
```

```
        System.out.print("Enter the number of terms in Fibonacci series: ");
        int n = scanner.nextInt();

        int firstTerm = 0, secondTerm = 1, nextTerm;

        System.out.print("Fibonacci Series up to " + n + " terms:");

        for (int i = 1; i <= n; ++i) {

            System.out.print(" " + firstTerm);

            nextTerm = firstTerm + secondTerm;

            firstTerm = secondTerm;
            secondTerm = nextTerm;
        }
    }
}
```

OUTPUT:

```
Enter the number of terms in Fibonacci series: 8
Fibonacci Series up to 10 terms: 0 1 1 2 3 5 8 13
```

PROBLEMS ON NESTED LOOPS

1.Write a Java program to print the multiplication table for numbers 1 to 10 using nested loops.

PROGRAM:

```
import java.util.*;
class MultiplicationTable {
    public static void main(String[] args) {
        for (int i = 1; i <= 10; i++) {
            System.out.println("Multiplication table for " + i + ":");

            for (int j = 1; j <= 10; j++) {
                int product = i * j;
                System.out.println(i + " * " + j + " = " + product);
            }

            System.out.println();
        }
    }}
```

OUTPUT:

```
Multiplication table for 1:
1 * 1 = 1
1 * 2 = 2
1 * 3 = 3
1 * 4 = 4
1 * 5 = 5
1 * 6 = 6
1 * 7 = 7
1 * 8 = 8
1 * 9 = 9
1 * 10 = 10
```

2.Write a Java program to print a inverted right-angled triangle of stars where the number of rows is provided by the user.

```
* * * *

* * *

* *

*
```

PROGRAM:

```
import java.util.Scanner;

public class D_right {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);

        System.out.print("Enter the number of rows: ");
        int r = s.nextInt();

        for (int i = r; i >= 1; i--) {
            for (int j = 1; j <= i; j++) {
                System.out.print("* ");
            }
            System.out.println();
        }
    }
}
```

OUTPUT:

```
Enter the number of rows: 5
```

```
* * * * *
* * * *
* * *
* *
*
```

3.Write a Java program to print Floyd's Triangle with a given number of rows.

PROGRAM:

```
import java.util.Scanner;

public class FloydsTriangle {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of rows for Floyd's Triangle: ");
        int numRows = scanner.nextInt();

        int number = 1;

        for (int i = 1; i <= numRows; i++) {
            for (int j = 1; j <= i; j++) {
                System.out.print(number + " ");
                number++;
            }
            System.out.println();
        }
    }
}
```

OUTPUT:

```
Enter the number of rows for Floyd's Triangle: 5
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
```

4.Write a Java program to print a diamond pattern using stars where the maximum width of the diamond is provided by the user.

PROGRAM:

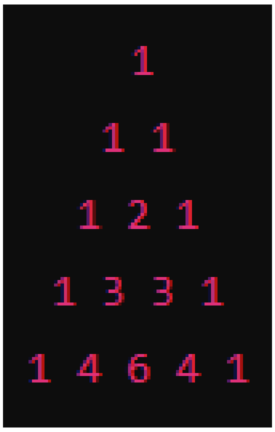
```
import java.util.*;

public class D_DiamondPattern {
    public static void main(String[] args) {
        Scanner s= new Scanner(System.in);
        int maxWidth = s.nextInt();
        int height = (maxWidth / 2) + 1;
        for (int i = 1; i <= height; i++) {
            for (int j = 1; j <= height - i; j++) {
                System.out.print(" ");
            }
            for (int k = 1; k <= 2 * i - 1; k++) {
                System.out.print("*");
            }
            System.out.println();
        }
        for (int i = height - 1; i >= 1; i--) {
            for (int j = 1; j <= height - i; j++) {
                System.out.print(" ");
            }
            for (int k = 1; k <= 2 * i - 1; k++) {
                System.out.print("*");
            }
            System.out.println();
        }
    }
}
```

OUTPUT:

5

```
  *
 ***
*****
 ***
  *
```

5.Write a Java program to print Pascal's Triangle up to a given number of rows.



```
    1
   1 1
  1 2 1
 1 3 3 1
1 4 6 4 1
```

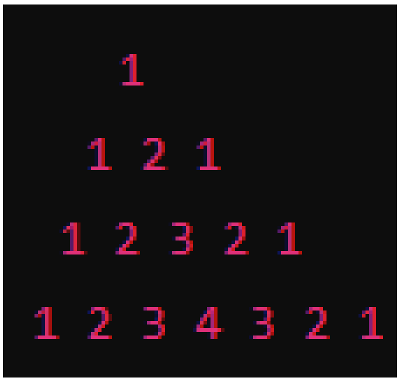
PROGRAM:

```
import java.util.Scanner;

public class D_Pascal {
   public static void main(String[] args) {
      Scanner s = new Scanner(System.in);
      System.out.print("Enter the number of rows : ");
      int r = s.nextInt();

      for (int i = 0; i < r; i++) {
         int n= 1;
         for (int j = 0; j < r - i; j++) {
            System.out.print("   ");
         }
         for (int j = 0; j <= i; j++) {
            System.out.printf("%6d", n);
            n = n * (i - j) / (j + 1);
         }
         System.out.println();
      }
   }
}
```

OUTPUT:

```
Enter the number of rows : 5

                  1
               1     1
            1     2     1
         1     3     3     1
      1        4     6     4     1
```

6.Write a Java program to print a pyramid of numbers.



```
    1

   1 2 1

  1 2 3 2 1

 1 2 3 4 3 2 1
```

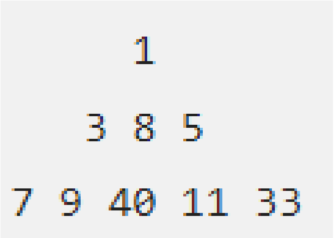
PROGRAM:

```
import java.util.*;

public class D_pyramid{
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the number of rows : ");
        int r = s.nextInt();

        for (int i = 1; i <= r; i++) {

            for (int j = 1; j <= r - i; j++) {
                System.out.print("  ");
            }
            for (int j = 1; j <= i; j++) {
                System.out.printf("%2d", j);
            }
            for (int j = i - 1; j >= 1; j--) {
                System.out.printf("%2d", j);
            }
            System.out.println();
        }
    }
}
```

OUTPUT:

```
Enter the number of rows : 5
        1
      1 2 1
    1 2 3 2 1
  1 2 3 4 3 2 1
1 2 3 4 5 4 3 2 1
```

9. Print the following pattern using Java



```
        1

     3  8  5

7  9  40  11  33
```

PROGRAM:

```
import java.util.Scanner;
class D_pattern {
   public static void main(String[] args) {
      Scanner s=new Scanner(System.in);
      System.out.print("Enter the number of rows: ");
      int n=s.nextInt();
      int i,j,a,z=1;
      for(i=0;i<n;i++){
         for(j=0;j<n-i-1;j++)
            System.out.print("  ");
         a=i*(z+z+(2*i-1)*2);
         for(j=0;j<2*i+1;j++){
            if(i!=0 && j==((2*i+1)/2))
               System.out.print(a+" ");
            else{
               System.out.print(z+" ");
               z+=2;
            }
         }
         System.out.println();
      }
      System.out.println();
   }
}
```

OUTPUT:

```
Enter the number of rows: 3

    1

  3 8 5

7 9 40 11 13
```