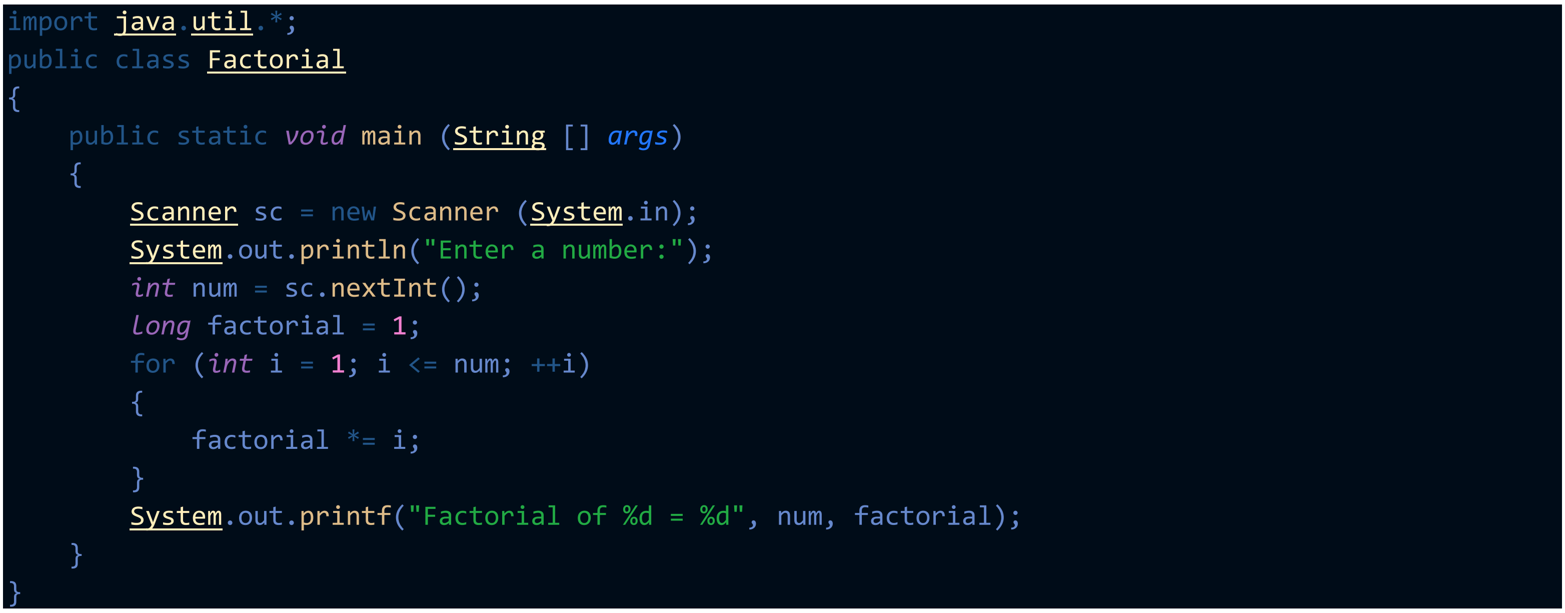
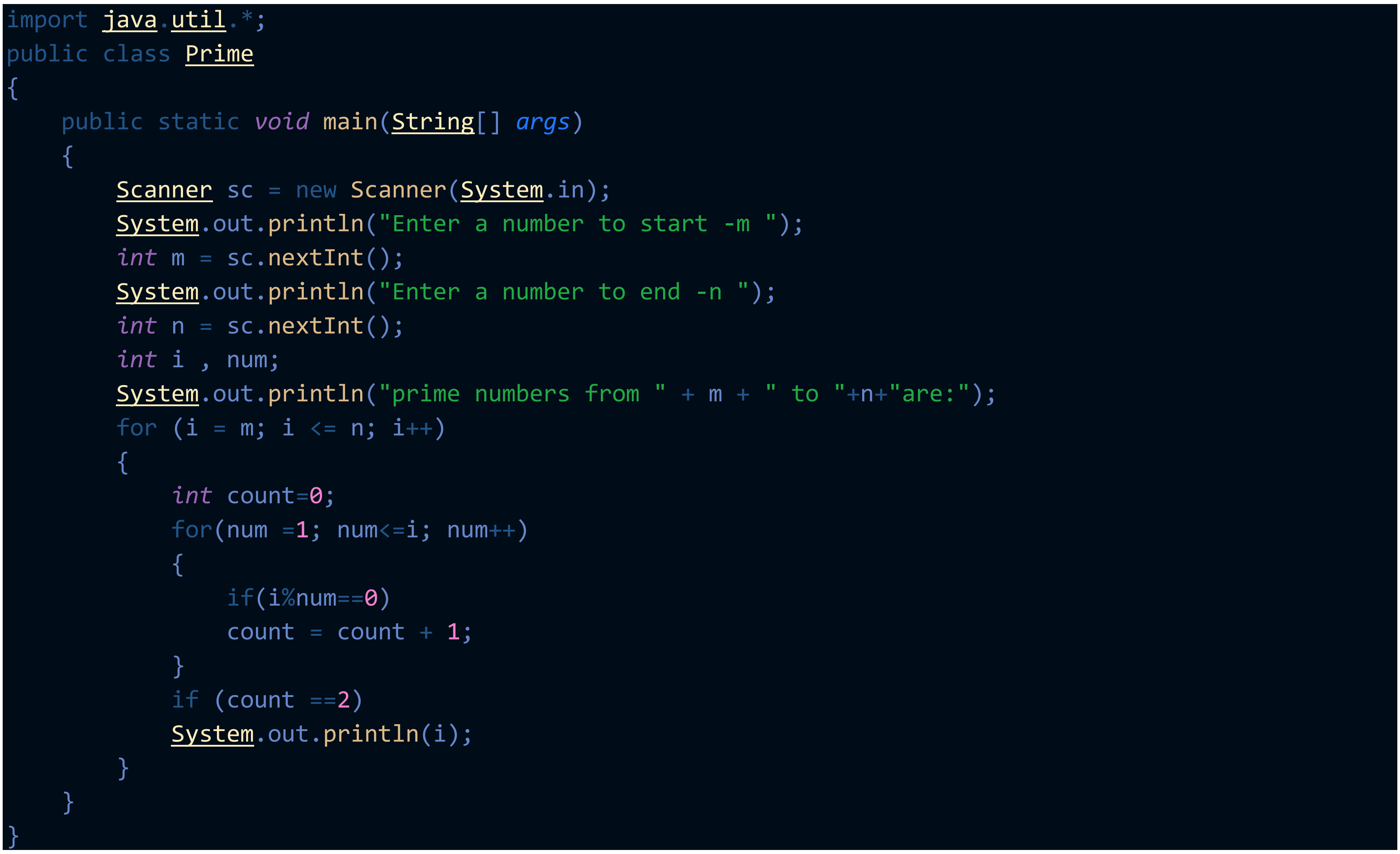
JAVA PROGRAMS

1) Program to find factorial of a number



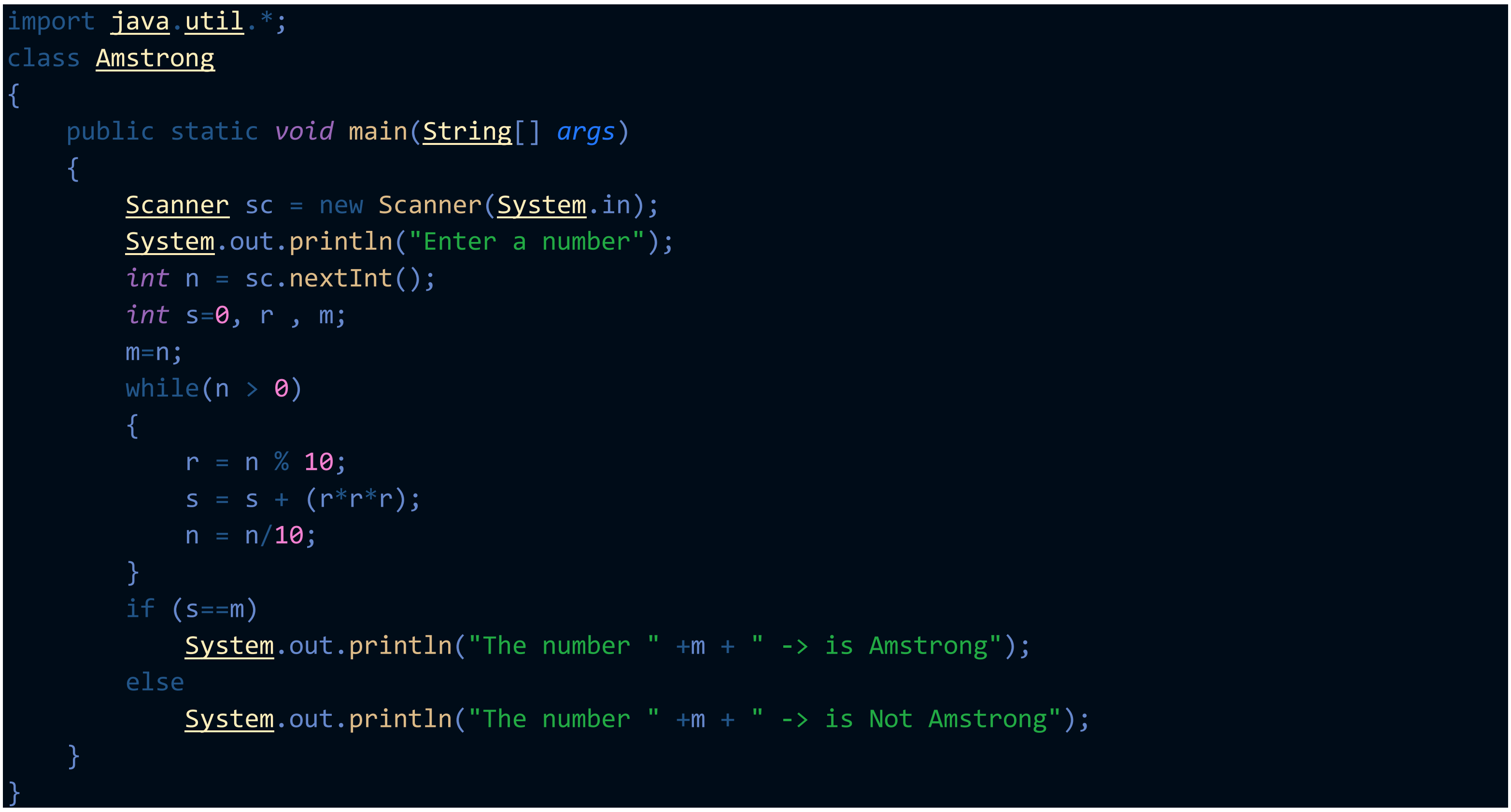
```java
import java.util.*;
public class Factorial
{
    public static void main (String [] args)
    {
        Scanner sc = new Scanner (System.in);
        System.out.println("Enter a number:");
        int num = sc.nextInt();
        long factorial = 1;
        for (int i = 1; i <= num; ++i)
        {
            factorial *= i;
        }
        System.out.printf("Factorial of %d = %d", num, factorial);
    }
}
```

2)Program to find prime numbers from m to n



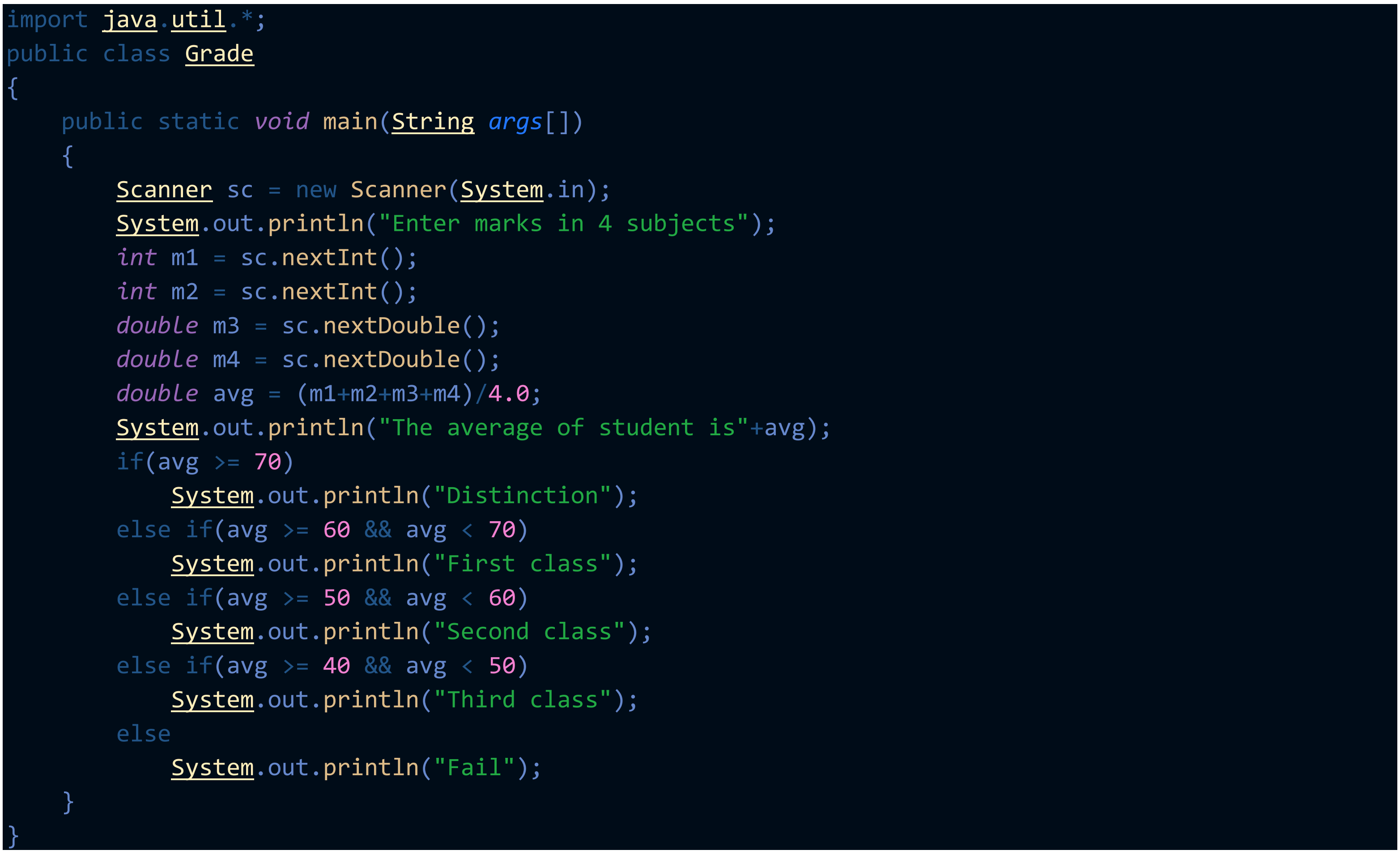
```java
import java.util.*;
public class Prime
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a number to start -m ");
        int m = sc.nextInt();
        System.out.println("Enter a number to end -n ");
        int n = sc.nextInt();
        int i , num;
        System.out.println("prime numbers from " + m + " to "+n+"are:");
        for (i = m; i <= n; i++)
        {
            int count=0;
            for(num =1; num<=i; num++)
            {
                if(i%num==0)
                count = count + 1;
            }
            if (count ==2)
            System.out.println(i);
        }
    }
}
```

3)Number is Armstrong number or not?



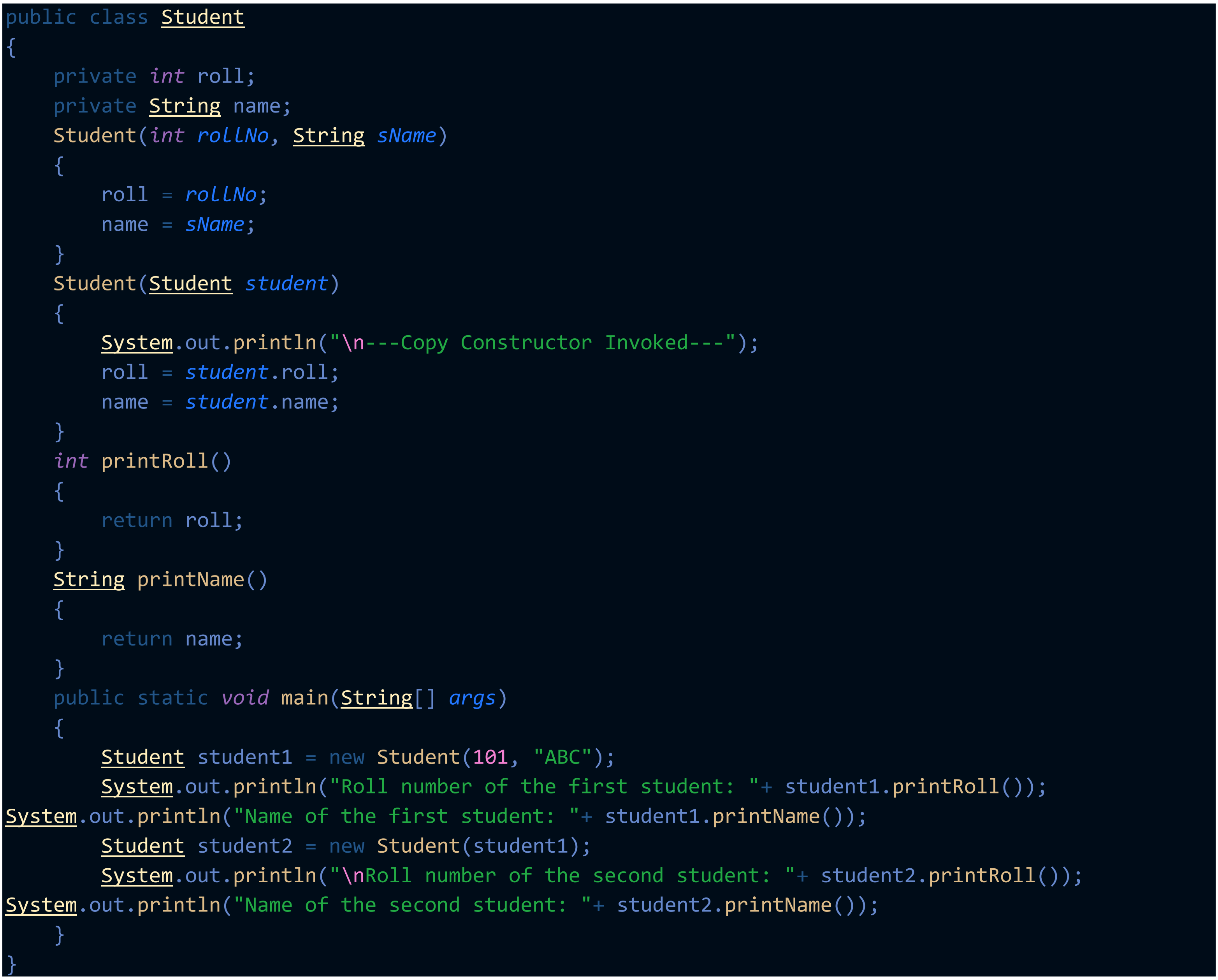
```
import java.util.*;
class Amstrong
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a number");
        int n = sc.nextInt();
        int s=0, r , m;
        m=n;
        while(n > 0)
        {
            r = n % 10;
            s = s + (r*r*r);
            n = n/10;
        }
        if (s==m)
            System.out.println("The number " +m + " -> is Amstrong");
        else
            System.out.println("The number " +m + " -> is Not Amstrong");
    }
}
```

4)To find Grade of a student



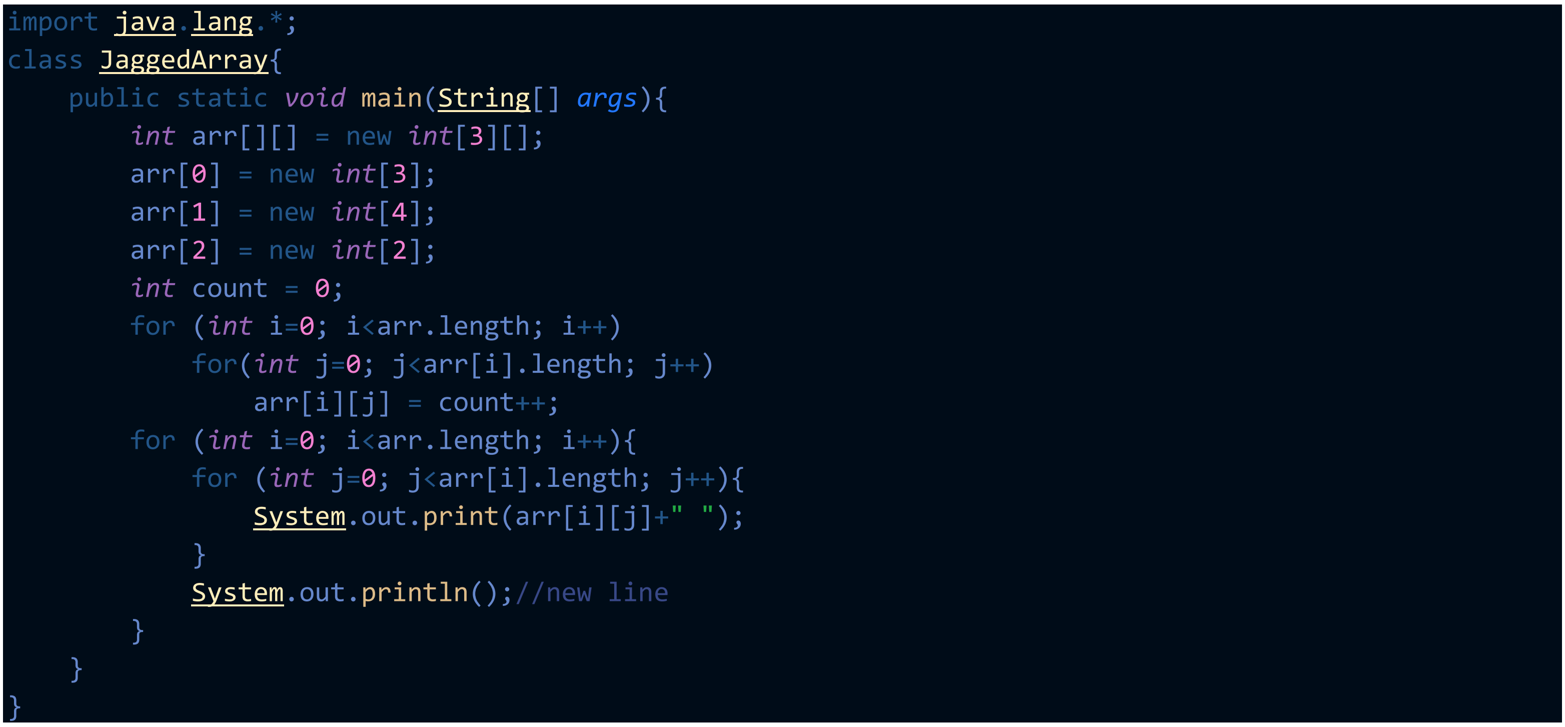
```
import java.util.*;
public class Grade
{
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter marks in 4 subjects");
        int m1 = sc.nextInt();
        int m2 = sc.nextInt();
        double m3 = sc.nextDouble();
        double m4 = sc.nextDouble();
        double avg = (m1+m2+m3+m4)/4.0;
        System.out.println("The average of student is"+avg);
        if(avg >= 70)
            System.out.println("Distinction");
        else if(avg >= 60 && avg < 70)
            System.out.println("First class");
        else if(avg >= 50 && avg < 60)
            System.out.println("Second class");
        else if(avg >= 40 && avg < 50)
            System.out.println("Third class");
        else
            System.out.println("Fail");
    }
}
```

5)Program with Student class (constructor)



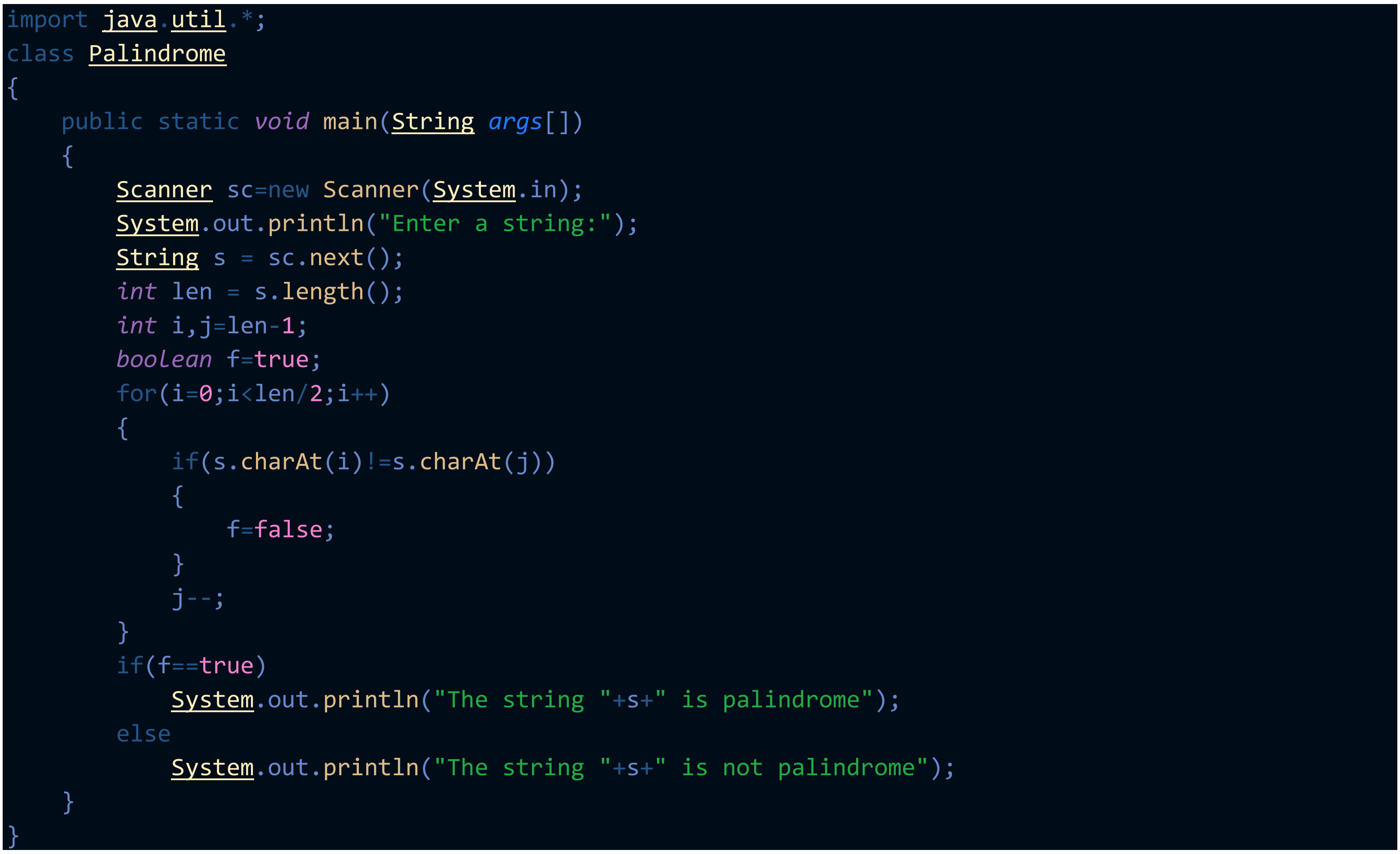
```
public class Student
{
    private int roll;
    private String name;
    Student(int rollNo, String sName)
    {
        roll = rollNo;
        name = sName;
    }
    Student(Student student)
    {
        System.out.println("\n---Copy Constructor Invoked---");
        roll = student.roll;
        name = student.name;
    }
    int printRoll()
    {
        return roll;
    }
    String printName()
    {
        return name;
    }
    public static void main(String[] args)
    {
        Student student1 = new Student(101, "ABC");
        System.out.println("Roll number of the first student: "+ student1.printRoll());
System.out.println("Name of the first student: "+ student1.printName());
        Student student2 = new Student(student1);
        System.out.println("\nRoll number of the second student: "+ student2.printRoll());
System.out.println("Name of the second student: "+ student2.printName());
    }
}
```

6)Jagged Array



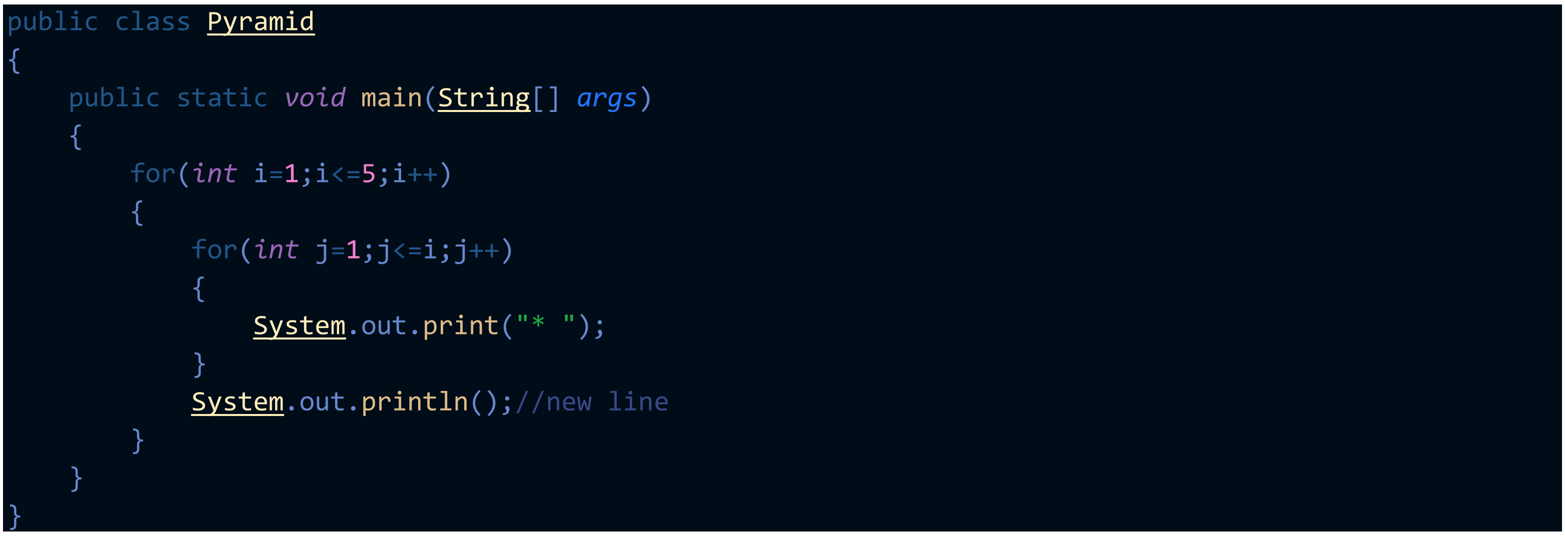
```
import java.lang.*;
class JaggedArray{
    public static void main(String[] args){
        int arr[][] = new int[3][];
        arr[0] = new int[3];
        arr[1] = new int[4];
        arr[2] = new int[2];
        int count = 0;
        for (int i=0; i<arr.length; i++)
            for(int j=0; j<arr[i].length; j++)
                arr[i][j] = count++;
        for (int i=0; i<arr.length; i++){
            for (int j=0; j<arr[i].length; j++){
                System.out.print(arr[i][j]+" ");
            }
            System.out.println();//new line
        }
    }
}
```

7)String Palindrome



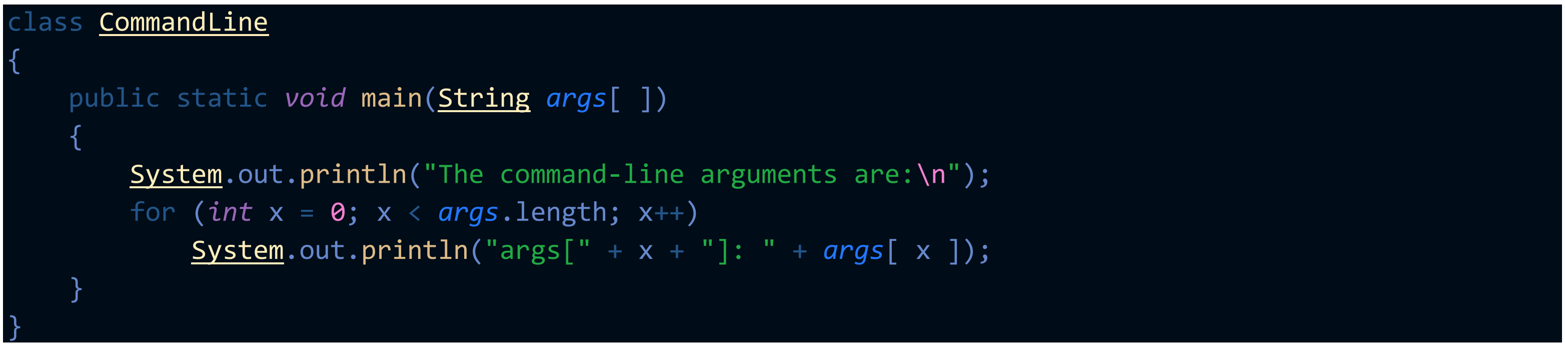
```
import java.util.*;
class Palindrome
{
    public static void main(String args[])
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter a string:");
        String s = sc.next();
        int len = s.length();
        int i,j=len-1;
        boolean f=true;
        for(i=0;i<len/2;i++)
        {
            if(s.charAt(i)!=s.charAt(j))
            {
                f=false;
            }
            j--;
        }
        if(f==true)
            System.out.println("The string "+s+" is palindrome");
        else
            System.out.println("The string "+s+" is not palindrome");
    }
}
```

8)Pyramid pattern



```
public class Pyramid
{
    public static void main(String[] args)
    {
        for(int i=1;i<=5;i++)
        {
            for(int j=1;j<=i;j++)
            {
                System.out.print("* ");
            }
            System.out.println();//new line
        }
    }
}
```

9)Command line arguments



```
class CommandLine
{
    public static void main(String args[ ])
    {
        System.out.println("The command-line arguments are:\n");
        for (int x = 0; x < args.length; x++)
            System.out.println("args[" + x + "]: " + args[ x ]);
    }
}
```

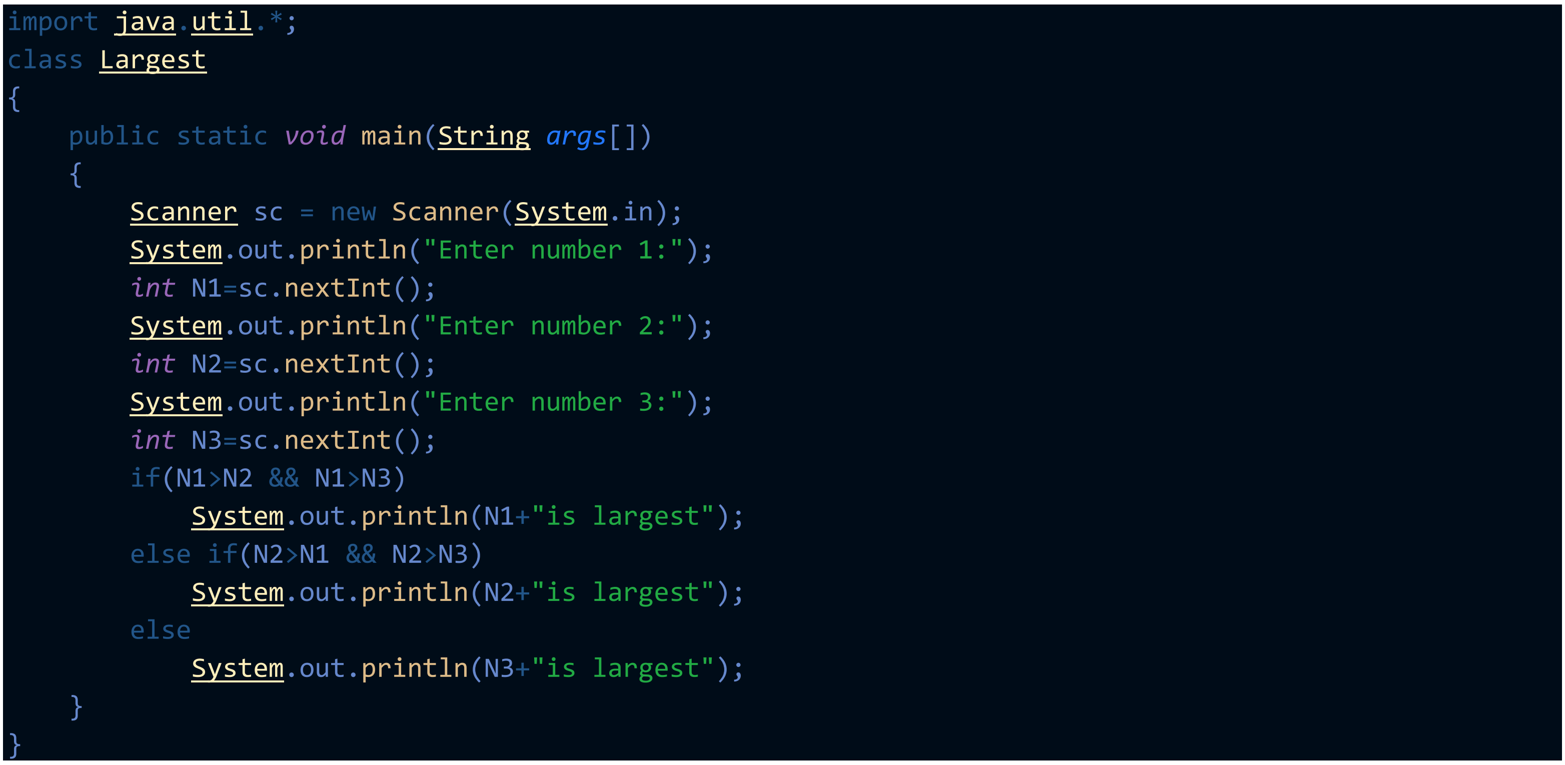
10)Leap year or not

```
import java.util.*;
public class Leap
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a year : ");
        int year = sc.nextInt();
        if (year % 4 == 0)
        {
            if (year % 100 == 0)
            {
                if (year % 400 == 0)
                    System.out.println(year + " is a leap year.");
                else
                    System.out.println(year + " is not a leap year.");
            }
            else
                System.out.println(year + " is a leap year.");
        }
        else
            System.out.println(year + " is not a leap year.");
    }
}
```

11)To check the number is Even or odd

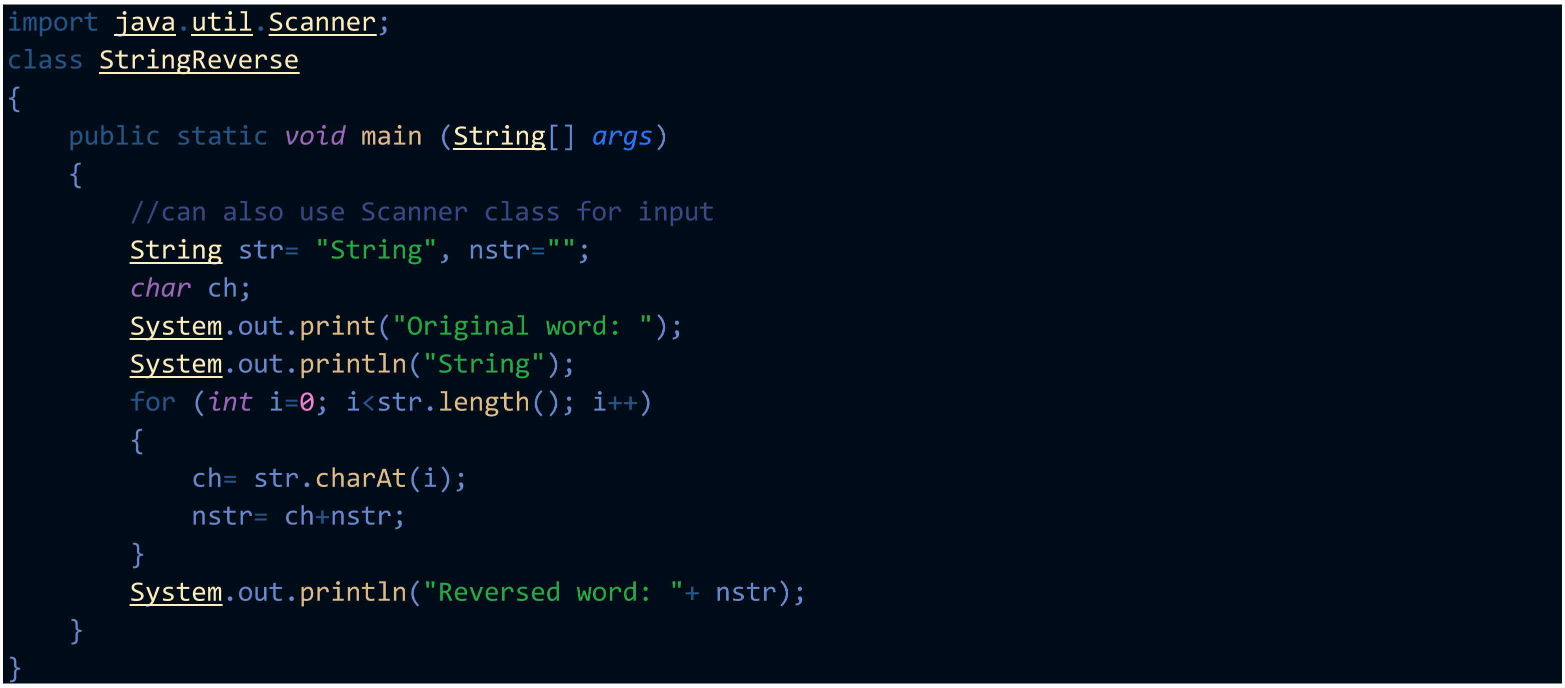
```
import java.util.*;
class EvenOrOdd
{
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a number");
        int N = sc.nextInt();
        if(N%2==0)
            System.out.println("Even");
        else
            System.out.println("Odd");
    }
}
```

12)Largest among 3 numbers



```
import java.util.*;
class Largest
{
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter number 1:");
        int N1=sc.nextInt();
        System.out.println("Enter number 2:");
        int N2=sc.nextInt();
        System.out.println("Enter number 3:");
        int N3=sc.nextInt();
        if(N1>N2 && N1>N3)
            System.out.println(N1+"is largest");
        else if(N2>N1 && N2>N3)
            System.out.println(N2+"is largest");
        else
            System.out.println(N3+"is largest");
    }
}
```

13)String Reverse



```
import java.util.Scanner;
class StringReverse
{
    public static void main (String[] args)
    {
        //can also use Scanner class for input
        String str= "String", nstr="";
        char ch;
        System.out.print("Original word: ");
        System.out.println("String");
        for (int i=0; i<str.length(); i++)
        {
            ch= str.charAt(i);
            nstr= ch+nstr;
        }
        System.out.println("Reversed word: "+ nstr);
    }
}
```

Program 1: CONSTRUCTOR OVERLOADING (ConsOL.java)

```
class ConsOL
{
    String language;
    ConsOL() {
        language = "Java";
    }
    ConsOL(String language) {
        this.language = language;
    }

    public void getName() {
        System.out.println("Programming Langauage: " + language);
    }

    public static void main(String[] args) {
        ConsOL obj1 = new ConsOL();
        ConsOL obj2 = new ConsOL("Python");
        obj1.getName();
        obj2.getName();
    }
}
```

Output:

```
Programming Langauage: Java
Programming Langauage: Python
```

Program 2: Floyd Triangle (FloydTriangleExample.java)

```
import java.util.*;
class FloydTriangle
{
    public static void main(String args[])
    {
       int rows, number = 1, count, j;
       Scanner input = new Scanner(System.in);
       System.out.print("Enter the number of rows for Floyd's triangle: ");
       rows = input.nextInt();
       System.out.println("Floyd's triangle");
       System.out.println("****************");
        for ( count = 1 ; count <= rows ; count++ )
        {
            for ( j = 1 ; j <= count ; j++ )
            {
                System.out.print(number+" ");
                number++;
            }
           System.out.println();
       }
   }
}
```

Output:

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

Program 3: Jagged Arrays (JaggedArray.java)

```
class JaggedArray {
    public static void main(String[] args)
    {
        int r = 5;
        int arr[][] = new int[r][];
        for (int i = 0; i < arr.length; i++)
            arr[i] = new int[i + 1];
        int count = 0;
        for (int i = 0; i < arr.length; i++)
            for (int j = 0; j < arr[i].length; j++)
                arr[i][j] = count++;
        System.out.println("Contents of 2D Jagged Array");
        for (int i = 0; i < arr.length; i++) {
            for (int j = 0; j < arr[i].length; j++)
                System.out.print(arr[i][j] + " ");
            System.out.println();
        }
    }
}
```

Output:

```
Contents of 2D Jagged Array
0
1 2
3 4 5
6 7 8 9
10 11 12 13 14
```

Program 4: METHOD OVERLOADING (MethodOL.java)

```
class MethodOL
{
    int add(int a, int b) {
            return (a + b);
    }
    int add(int a , int b , int c) {
        return (a + b + c) ;
    }
    double add(double a , double b) {
        return (a + b);
    }
    double add(int a , double b) {
        return (a + b);
    }
    public static void main( String args[])
    {
        MethodOL ob = new MethodOL();
        System.out.println("Calling add method with two int parameters: " +ob.add(17, 25));
        System.out.println("Calling add method with three int parameters: " +ob.add(55, 27, 35));
        System.out.println("Calling add method with two double parameters: " +ob.add(36.5, 42.8));
        System.out.println("Calling add method with one int and one double parameter: " +ob.add(11, 24.5));
    }
}
```

Output:

```
Calling add method with two int parameters: 42
Calling add method with three int parameters: 117
Calling add method with two double parameters: 79.3
Calling add method with one int and one double parameter: 35.5
```

Program 5: Demonstration of Nested Switch (NestedSwitch.java)

```
class NestedSwitch
{
    public static void main(String args[]){
        int tech = 2;
        int course = 2;
        switch(tech){
            case 1:
                System.out.println("Python");
                break;
            case 2:
                switch(course){
                    case 1:
                        System.out.println("J2EE");
                        break;
                    case 2:
                        System.out.println("Advanced Java");
                }
        }
    }
}
```

Output:

```
Advanced Java
```

Program 6: Pyramid Pattern (PyramidExample.java)

```
public class Pyramid
{
    public static void main(String[] args) {
        for(int i=1;i<=5;i++) {
            for(int j=1;j<=i;j++)
                System.out.print("* ");
            System.out.println();
        }
    }
}
```

Output:

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

Program 7: String Palindrome (StringPalindrome.java)

```
import java.util.*;
class StringPalindrome
{
    public static boolean isPal(String s)
    {
        if(s.length() == 0 || s.length() == 1)
            return true;
        if(s.charAt(0) == s.charAt(s.length()-1))
            return isPal(s.substring(1, s.length()-1));
        return false;
    }
    public static void main(String[]args)
    {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter the String for check: ");
        String string = scan.nextLine(); scan.close();
        if(isPal(string))
            System.out.println(string + " is a palindrome");
        else
            System.out.println(string + " is not a palindrome");
    }
}
```

Output:

```
Enter the String for check: Jai_Balayya
Jai_Balayya is not a palindrome

Enter the String for check: rotator
rotator is a palindrome
```

Program 8: Encapsulation Example (Encapsulation.java)

```
class Student
{
    private String name;
    private int rollno;
    public void setName(String name) {
        this.name = name;
    }
    public void setRollno(int rollno) {
        this.rollno = rollno;
    }
    public String getName() {
        return name;
    }
    public int getRollno() {
        return rollno;
    }
}
class Encapsulation {
    public static void main(String args[]) {
        Student s=new Student();
        s.setRollno(1200);
        s.setName("Jai Balayya");
        System.out.println(s.getName());
        System.out.println(s.getRollno());
    }
}
```

Output:

```
Jai Balayya
1200
```

Program 9: METHOD OVERLOADING EXAMPLE 2 (Sum.java)

```java
public class Sum {
    public int sum(int x, int y)
    {
        return (x + y);
    }
    public int sum(int x, int y, int z)
    {
        return (x + y + z);
    }
    public double sum(double x, double y)
    {
        return (x + y);
    }
    public static void main(String args[])
    {
        Sum s = new Sum();
        System.out.println(s.sum(10, 20));
        System.out.println(s.sum(10, 20, 30));
        System.out.println(s.sum(10.5, 20.5));
    }
}
```

Output:

```
30
60
31.0
```

Program 10: Switch statement (SwitchCase.java)

```java
import java.util.*;
public class SwitchCase{
      public static void main(String args[]){
            System.out.print("Enter the value for Month : ");
            Scanner scan = new Scanner(System.in);
            int month = scan.nextInt(); scan.close();
            switch(month){
                  case 1: System.out.println("january"); break;
                  case 2: System.out.println("february"); break;
                  case 3: System.out.println("march"); break;
                  case 4: System.out.println("april"); break;
                  case 5: System.out.println("may"); break;
                  case 6: System.out.println("june"); break;
                  case 7: System.out.println("july"); break;
                  case 8: System.out.println("august"); break;
                  case 9: System.out.println("september"); break;
                  case 10: System.out.println("October"); //break;
                  case 11: System.out.println("november"); //break;
                  case 12: System.out.println("december"); break;
                  default: System.out.println("enter numbers from 1 to 12");
            }
      }
}
```

Output:

```
Enter the value for Month : 10
October
november
December
```

Program 11: ENCAPSULATION Example 2 (Encapsulation.java)

```
class Encapsulate
{
    private String name;
    private int roll;
    private int age;
    public int getAge() { return age; }
    public String getName() { return name; }
    public int getRoll() { return roll; }
    public void setAge(int newAge) { age = newAge; }
    public void setName(String newName) { name = newName; }
    public void setRoll(int newRoll) { roll = newRoll; }
}
public class Encapsulation {
    public static void main(String[] args)
    {
        Encapsulate obj = new Encapsulate();
        obj.setName("Harsh");
        obj.setAge(19);
        obj.setRoll(51);
        System.out.println("Geek's name: " + obj.getName());
        System.out.println("Geek's age: " + obj.getAge());
        System.out.println("Geek's roll: " + obj.getRoll());
    }
}
```

Output:

```
Geek's name: Harsh
Geek's age: 19
Geek's roll: 51
```

Program 12: While Loop Example (WhileExample.java)

```
class WhileExample {
    public static void main(String args[]){
        int i=10;
        while(i>1){
              System.out.println(i);
              i--;
        }
    }
}
```

Program 13: While Loop Example 2 (WhileLoop.java)

```
import java.util.*;
public class WhileLoop {
    private static Scanner sc;
    public static void main(String[] args) {
        int number, sum = 0;
        sc = new Scanner(System.in);
        System.out.print("Please Enter any integer Value below 10: ");
        number = sc.nextInt();
        while (number <= 10)  {
            sum = sum + number;
            number++;
        }
        System.out.format("Sum of the Numbers from the While Loop is: %d ", sum);
    }
}
```

Output:

```
Please Enter any integer Value below 10: 6
Sum of the Numbers from the While Loop is: 40
```

Program 14: Command line arguments example (CommandLine.java)

```
class CommandLine
{
    public static void main(String args[ ])
    {
        System.out.println("The command-line arguments are:\n");
        for (int x = 0; x < args.length; x++)
            System.out.println("args[" + x + "]: " + args[ x ]);
    }
}
```

Execution:

```
java CommandLine 1 hello
```

Output:

```
The command-line arguments are:

args[0]: 1
args[1]: hello
```

Program 15: Check leap year or not (Leapyear.java)

```
import java.util.*;
class Leapyear
{
    public static void main(String arg[])
    {
        long year;
        Scanner sc=new Scanner(System.in);
        System.out.print("Enter any calendar year : ");
        year=sc.nextLong(); sc.close();
        leapOrNot(year);
    }
    static void leapOrNot(long year)
    {
        if(year > 0 && year % 100 == 0)
        {
            if(year % 100 == 0)
            {
                if(year % 400 == 0)
                    System.out.println(year+" is a leap year");
                else
                    System.out.println(year+" is not a leap year");
            }
            else
                System.out.println(year+" is not a leap year");
        }
        else
            System.out.println(year+" is not a leap year");
    }
}
```

Output:

```
Enter any calendar year : 2100
2100 is not a leap year
```

Program 16: Reversing the array (ReverseArray.java)

```
import java.util.*;
class ReverseArray
{
    public static void main(String[] args)
    {
        int n, i, j=0;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter number of elements in the array: ");
        n = s.nextInt();
        int array[] = new int[n];
        int rev[] = new int[n];
        System.out.println("Enter "+n+" elements ");
        for( i=0; i < n; i++)
            array[i] = s.nextInt();
        System.out.println("Reverse of an array is :");
        for( i=n;i>0 ; i--,j++)
        {
            rev[j] = array[i-1];
            System.out.print(rev[j] + " ");
        }
    }
}
```

Output:

```
Enter number of elements in the array: 6
Enter 6 elements
54 98 21 74 36 1
Reverse of an array is :
1 36 74 21 98 54
```

Program 17: Demonstrating Static keyword (StaticExample.java)

```
public class StaticExample {
    static int number1;
    static String str1;
    static void display(){
         System.out.println("From static method");
    }
    public static void main(String[] args) {
      display();
      number1 = 101;
      str1 = "IT";
      System.out.println("Static Number: " + number1);
      System.out.println("Static String: " + str1);
    }
}
```

Output:

```
From static method
Static Number: 101
Static String: IT
```

Program 18: String Method demonstration (StringMethodsDemo.java)

```java
public class StringMethodsDemo {
    public static void main(String[] args) {
        String targetString = "Java is fun to learn";
        String s1= "JAVA";
        String s2= "Java";
        String s3 = "  Hello Java  ";
        System.out.println("Char at index 2(third position): " + targetString.charAt(2));
        System.out.println("After Concat: "+ targetString.concat("-Enjoy-"));
        System.out.println("Checking equals ignoring case: " +s2.equalsIgnoreCase(s1));
        System.out.println("Checking equals with case: " +s2.equals(s1));
        System.out.println("Checking Length: "+ targetString.length());
        System.out.println("Replace function: "+ targetString.replace("fun", "easy"));
        System.out.println("SubString of targetString: "+ targetString.substring(8));
        System.out.println("SubString of targetString: "+ targetString.substring(8, 12));
        System.out.println("Converting to lower case: "+ targetString.toLowerCase());
        System.out.println("Converting to upper case: "+ targetString.toUpperCase());
        System.out.println("Trimming string: " + s3.trim());
        System.out.println("searching s1 in targetString: " + targetString.contains(s1));
        System.out.println("searching s2 in targetString: " + targetString.contains(s2));
        char [] charArray = s2.toCharArray();
        System.out.println("Size of char array: " + charArray.length);
        System.out.println("Printing last element of array: " + charArray[3]);
    }
}
```

Output:

```
Char at index 2(third position): v
After Concat: Java is fun to learn-Enjoy-
Checking equals ignoring case: true
Checking equals with case: false
Checking Length: 20
Replace function: Java is easy to learn
SubString of targetString: fun to learn
SubString of targetString: fun
Converting to lower case: java is fun to learn
Converting to upper case: JAVA IS FUN TO LEARN
Trimming string: Hello Java
searching s1 in targetString: false
searching s2 in targetString: true
Size of char array: 4
Printing last element of array: a
```

Week - 6

6.a - Implement Package

Converter.java

```java
package Measure;

public class Converter
{
    public double mmtocm(double mm)
    {
        return mm/10;
    }
    public double cmtom(double cm)
    {
        return cm/100;
    }
    public double mtokm(double m)
    {
        return m/1000;
    }
}
```

NeedConverter.java

```java
import java.util.*;
import Measure.Converter;

class NeedConverter
{
    public static void main(String args[])
    {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter length : ");
        double l = scan.nextDouble();

        Converter obj = new Converter();
        System.out.println("Method mm-cm returned : "+obj.mmtocm(l));
        System.out.println("Method cm-m returned : "+obj.cmtom(l));
        System.out.println("Method m-km returned : "+obj.mtokm(l));
        scan.close();
    }
}
```

Execution:

```
$javac -d . Converter.java
$javac NeedConverter.java
$java NeedConverter
```

Output:

```
Enter length : 10
Method mm-cm returned : 1.0
Method cm-m returned : 0.1
Method m-km returned : 0.01
```

6.b - Implement Interface

StdDemo.java

```java
interface Student
{
    void displayGrade();
    void attendence();
}
class UGStudent
{
    String grade;
    double attendence;
    UGStudent(String g, double a){
        grade = g; attendence = a;
    }
    void displayGrade()
    {
        System.out.println("Grade of UG Student is " + grade);
    }
    void attendence()
    {
        System.out.println("Attendence of UG Student is " + attendance);
    }
}
public class StdDemo
{
    public static void main(String[] args)
    {
        UGStudent u = new UGStudent("A+",98.2);
        u.displayGrade();
        u.attendance();
    }
}
```

Output:

```
Grade of UG Student is A+
Attendance of UG Student is 98.2
```

Week - 7

7.a -Implement Exception Handling - Number Format & Arithmetic Exceptions

ExceptionDemo.java

```
import java.util.*;
public class ExceptionDemo
{
    public static void main (String ar[])
    {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter 1st number : ");
        String a = scan.next();
        System.out.print("Enter 2nd number : ");
        String b = scan.next();
        scan.close();
        try
        {
            int n1 = Integer.parseInt(a);
            int n2 = Integer.parseInt(b);
            System.out.println("After conversion, n1 = " + n1 + " and n2 = " + n2);
            if(n2 == 0)
                throw new ArithmeticException ("Division by zero Error");
            int x = n1/n2;
            System.out.println("The result of the division " + n1 + "/" + n2 + " is " + x);
        }
        catch(NumberFormatException e)
        {
            System.out.println("! Exception : " + e + "\nPlease Try Again, Enter data in numeric 
format.");
        }
        catch(ArithmeticException e)
        {
            System.out.println("! Exception : " + e + "\nPlease Try Again, Divisor must not be 
0.");
        }
        finally
        {
            System.out.println("\nStatement in Finally block.");
        }
        System.out.println("End of program.\n");
    }
}
```

Output 1:

```
Enter 1st number : 4
Enter 2nd number : 0
After conversion, n1 = 4 and n2 = 0
! Exception : java.lang.ArithmeticException: Division by zero Error
Please Try Again, Divisor must not be 0.

Statement in Finally block.
End of program.
```

Output 2:

```
Enter 1st number : k
Enter 2nd number : 1
! Exception : java.lang.NumberFormatException: For input string: "k"
Please Try Again, Enter data in numeric format.

Statement in Finally block.
End of program.
```

7.b - Implement Exception Handling - Custom Exception

```java
class NegativeAgeException extends Exception
{
    NegativeAgeException(String str)
    {
        super(str);
    }
}

class CustomExceptionDemo
{
    static void checkAge(int age) throws NegativeAgeException
    {
        if(age < 0)
            throw new NegativeAgeException("Not Vaild Age");
        else
            System.out.println("Age = " + age + " is valid.");
    }
    public static void main(String a[])
    {
        try
        {
            checkAge(5); checkAge(23); checkAge(100); checkAge(-3); checkAge(57);
        }
        catch(Exception e)
        {
            System.out.println("! Exception : " + e + "\nPlease Try Again, Negative age is
incorrect.");
        }
        finally
        {
            System.out.println("\nStatement in Finally block.");
        }
        System.out.println("End of program.\n");
    }
}
```

Output:

```
Age = 5 is valid.
Age = 23 is valid.
Age = 100 is valid.
! Exception : NegativeAgeException: Not Vaild Age
Please Try Again, Negative age is incorrect.

Statement in Finally block.
End of program.
```

Week - 8

8.a - Implement Threads – Two Threads printing “Java is object oriented.”

```java
class Threads extends Thread
{
    Threads(String name)
    {
        super(name);
    }
    public void run()
    {
        try
        {
            for(int i = 0; i <= 3; i++)
            {
                System.out.println(getName() + " printed : Java is object oriented.");
                sleep(100);
            }
        }
        catch(Exception e)
        {
            System.out.println("Exception Caught : " + e);
        }
    }
}

class MultiThreadDemo
{
    public static void main(String a[])
    {
        Threads t1 = new Threads("Thread-1");
        Threads t2 = new Threads("Thread-2");
        t1.start();
        t2.start();
        System.out.println("End of main method.");
    }
}
```

Output:

```
End of main method.
Thread-1 printed : Java is object oriented.
Thread-2 printed : Java is object oriented.
Thread-2 printed : Java is object oriented.
Thread-1 printed : Java is object oriented.
Thread-1 printed : Java is object oriented.
Thread-2 printed : Java is object oriented.
Thread-1 printed : Java is object oriented.
Thread-2 printed : Java is object oriented.
```

8.b - Demonstrate Producer & Consumer problem.

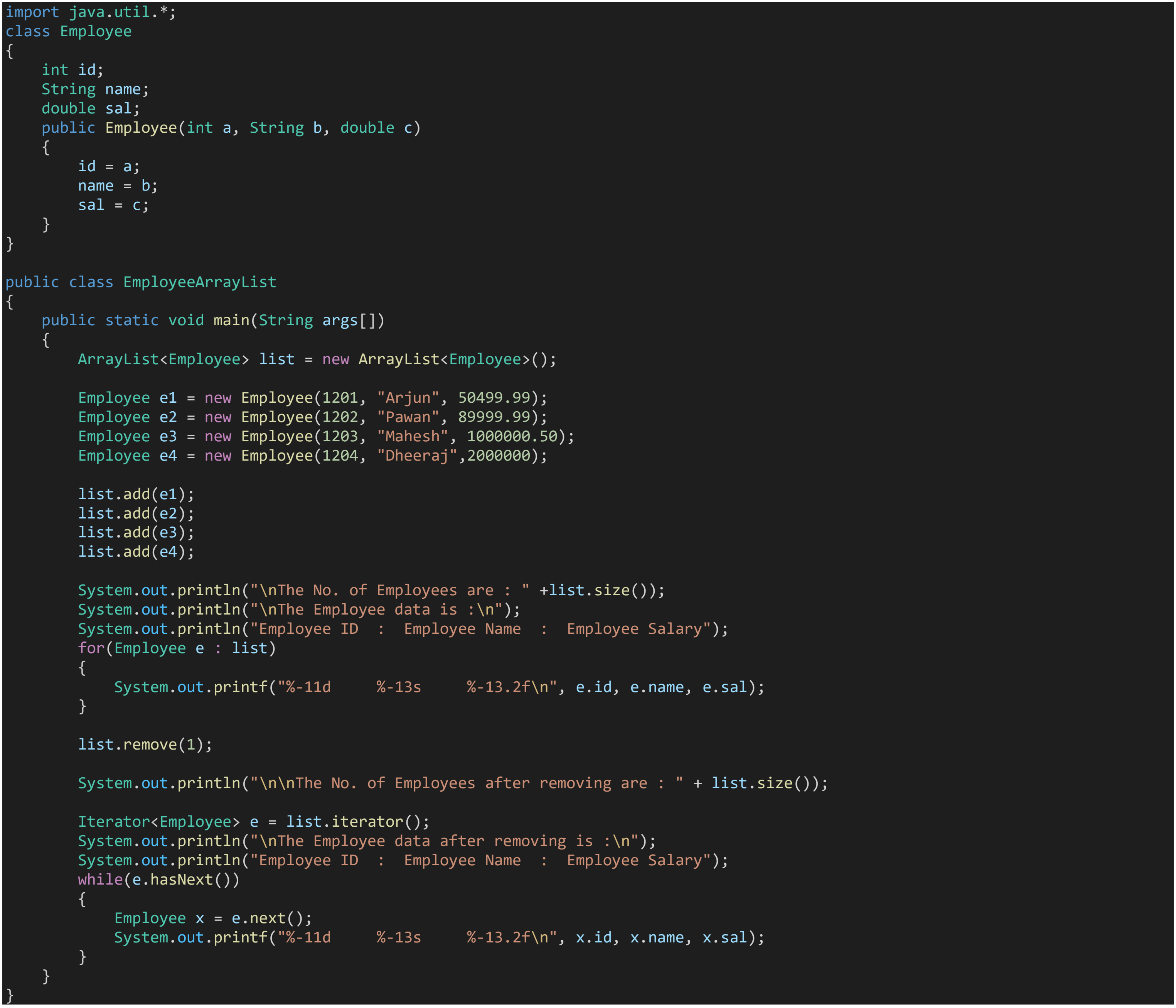
```java
class Buffer {
    int item; boolean available = false;
    synchronized void produce(int x)
    {
        if(available) {
            try { wait(); }
            catch(Exception e) {
                System.out.println("Exception Caught : " + e);
            }
        }
        item = x;
        System.out.println("\nProducer : produced --> " + item);
        available = true;
        notify();
    }
    synchronized int consume()
    {
        if(!available) {
            try { wait(); }
            catch(Exception e) {
                System.out.println("Exception Caught : " + e);
            }
        }
        System.out.println("Consumer : consumed <-- " + item);
        available = false;
        notify();
        return item;
    }
}
class Producer extends Thread {
    Buffer b;
    Producer(Buffer b) {
        this.b = b;
        start();
    }
    public void run() {
        b.produce(1); b.produce(2); b.produce(3); b.produce(4); b.produce(5);
    }
}
class Consumer extends Thread {
    Buffer b;
    Consumer(Buffer b) {
        this.b = b;
        start();
    }
    public void run() {
        b.consume(); b.consume(); b.consume(); b.consume();
    }
}
public class ProducerConsumer {
    public static void main(String a[]) {
        Buffer b = new Buffer();
        new Producer(b);
        new Consumer(b);
    }
}
```

Output:

```
Producer : produced --> 1
Consumer : consumed <-- 1

Producer : produced --> 2
Consumer : consumed <-- 2

Producer : produced --> 3
Consumer : consumed <-- 3

Producer : produced --> 4
Consumer : consumed <-- 4

Producer : produced --> 5
```

Week – 9

9.a - Implement ArrayList



```java
import java.util.*;
class Employee
{
    int id;
    String name;
    double sal;
    public Employee(int a, String b, double c)
    {
        id = a;
        name = b;
        sal = c;
    }
}

public class EmployeeArrayList
{
    public static void main(String args[])
    {
        ArrayList<Employee> list = new ArrayList<Employee>();

        Employee e1 = new Employee(1201, "Arjun", 50499.99);
        Employee e2 = new Employee(1202, "Pawan", 89999.99);
        Employee e3 = new Employee(1203, "Mahesh", 1000000.50);
        Employee e4 = new Employee(1204, "Dheeraj",2000000);

        list.add(e1);
        list.add(e2);
        list.add(e3);
        list.add(e4);

        System.out.println("\nThe No. of Employees are : " +list.size());
        System.out.println("\nThe Employee data is :\n");
        System.out.println("Employee ID  :  Employee Name  :  Employee Salary");
        for(Employee e : list)
        {
            System.out.printf("%-11d     %-13s     %-13.2f\n", e.id, e.name, e.sal);
        }

        list.remove(1);

        System.out.println("\n\nThe No. of Employees after removing are : " + list.size());

        Iterator<Employee> e = list.iterator();
        System.out.println("\nThe Employee data after removing is :\n");
        System.out.println("Employee ID  :  Employee Name  :  Employee Salary");
        while(e.hasNext())
        {
            Employee x = e.next();
            System.out.printf("%-11d     %-13s     %-13.2f\n", x.id, x.name, x.sal);
        }
    }
}
```

Output:

```
The No. of Employees are : 4

The Employee data is :

Employee ID  :  Employee Name  :  Employee Salary
1201          Arjun          50499.99
1202          Pawan          89999.99
1203          Mahesh         1000000.50
1204          Dheeraj        2000000.00


The No. of Employees after removing are : 3

The Employee data after removing is :

Employee ID  :  Employee Name  :  Employee Salary
1201          Arjun          50499.99
1203          Mahesh         1000000.50
1204          Dheeraj        2000000.00
```

9.b – Implement Hashset

```java
import java.util.*;
class HashSetDemo
{
    public static void main(String args[])
    {
        HashSet<String> set = new HashSet<String>();
        set.add("Sriharsh");
        set.add("Dheeraj");
        set.add("Sandeep");
        set.add("Ganesh");

        System.out.print("\nSet elements are : ");
        Iterator<String> i = set.iterator();
        while(i.hasNext())
        {
            System.out.print(i.next() + ", ");
        }
        System.out.println(" \b\b\b ");

        set.remove("Dheeraj");

        System.out.println("\nSet elements after removing \"Dheeraj\" : " +set);

        HashSet<String> set1 = new HashSet<String>();
        set1.add("Varshith");
        set1.add("Boom");

        set.addAll(set1);
        System.out.println("\nSet elements after adding set1 are : "+set);

        set.removeAll(set1);
        System.out.println("\nSet elements after removing set1 are : "+set);

        set.clear();

        System.out.println("\nSet elements after clearing are : "+set);
    }
}
```

Output:

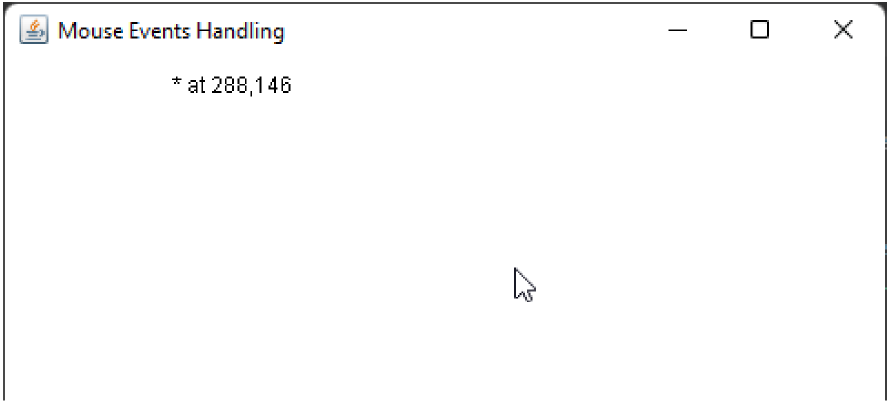
```
Set elements are : Sandeep, Ganesh, Sriharsh, Dheeraj

Set elements after removing "Dheeraj" : [Sandeep, Ganesh, Sriharsh]

Set elements after adding set1 are : [Sandeep, Ganesh, Boom, Sriharsh, Varshith]

Set elements after removing set1 are : [Sandeep, Ganesh, Sriharsh]

Set elements after clearing are : []
```

Week - 10

10.a – Implement Mouse Event Listener

```
import java.awt.*;
import java.awt.event.*;
public class MouseDemo extends Frame implements
MouseListener,MouseMotionListener
{
    int x=0, y=0;
    String msg= "";
    MouseDemo(String title) {
        super(title);
        addMouseListener(this);
        addMouseMotionListener(this);
        setSize(500,500);
        setVisible(true);
        addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent e) {
                dispose(); }
        } );
    }
    public void mouseClicked(MouseEvent e) {
        msg= "MouseClicked";
        x = e.getX();
        y = e.getY();
        repaint(); }
    public void mousePressed(MouseEvent e) {
        msg= "MousePressed";
        x = e.getX();
        y = e.getY();
        repaint(); }
    public void mouseReleased(MouseEvent e) {
        msg = "MouseReleased";
        x = e.getX();
        y = e.getY();
        repaint(); }
    public void mouseEntered(MouseEvent e) {
        msg= "MouseEntered";
        x = e.getX();
        y = e.getY();
        repaint(); }
    public void mouseExited(MouseEvent e) {
        msg= "MouseExited";
        x = e.getX();
        y = e.getY();
        repaint(); }
    public void mouseMoved(MouseEvent e) {
        msg= "*";
        x = e.getX();
        y = e.getY();
        repaint(); }
    public void mouseDragged(MouseEvent e) {
        msg= "#";
        x = e.getX();
        y = e.getY();
        repaint(); }
    pubic void paint(Graphics g) {
        g.drawString(msg + " at " + x + "," + y, 100,50); }
    public static void main(String[] args) {
        MouseDemo f = new MouseDemo("Mouse Events Handling"); }
}
```

Output:

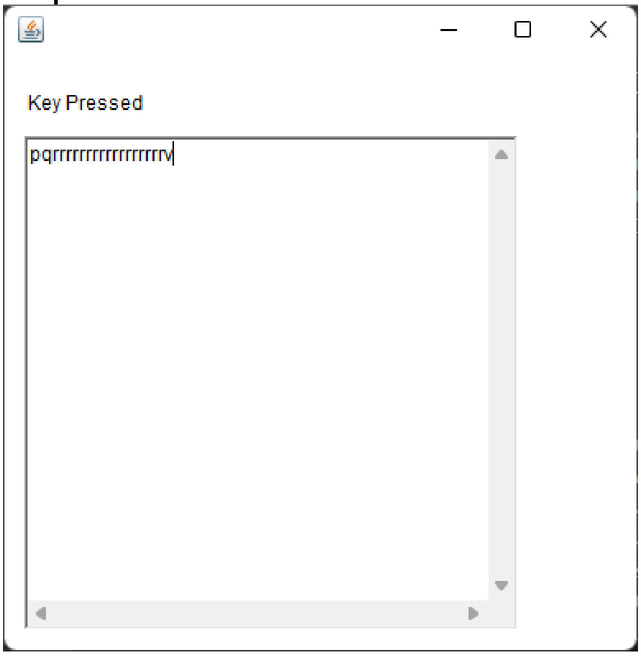


```
* at 288,146
```

10.b - Implement Key Listener

```
import java.awt.*;
import java.awt.event.*;
public class KeyDemo extends Frame implements KeyListener
{
Label l;
TextArea area;
KeyDemo()
{
l=new Label();
l.setBounds(20,50,100,20);
area=new TextArea();
area.setBounds(20,80,300, 300);
area.addKeyListener(this);
add(l);add(area);
setSize(400,400);
setLayout(null);
setVisible(true);
addWindowListener(new WindowAdapter()
{
public void windowClosing(WindowEvent e)
{
dispose();
}
} );
}
public void keyPressed(KeyEvent e)
{
l.setText("Key Pressed");
}
public void keyReleased(KeyEvent e)
{
l.setText("Key Released");
}
public void keyTyped(KeyEvent e)
{
l.setText("Key Typed");
}
public static void main(String[] args)
{
new KeyDemo();
}
}
```

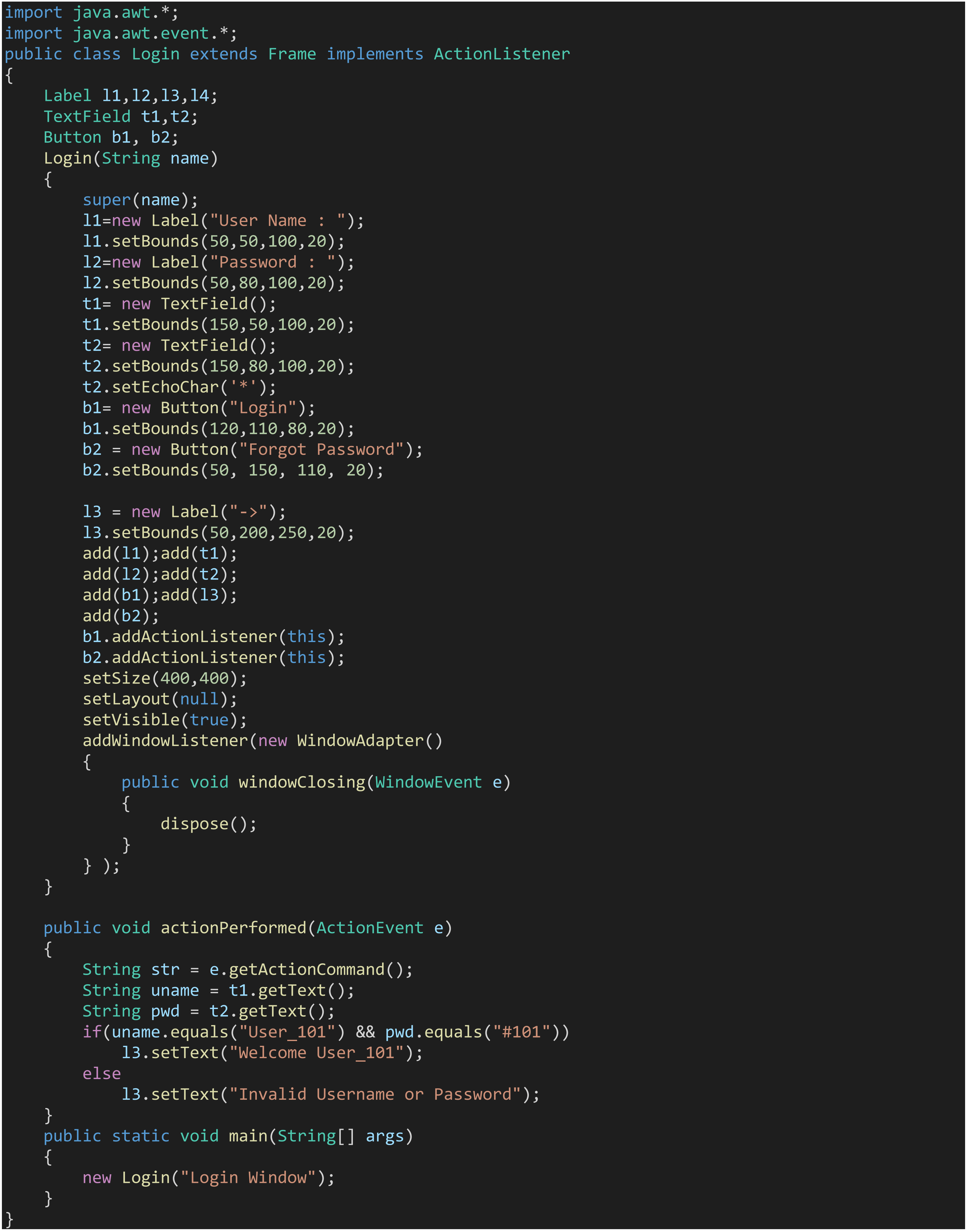
Output:



```
Key Pressed

pqrrrrrrrrrrrrrrrrrv
```

Week – 11

11.a – Login Page using AWT



```java
import java.awt.*;
import java.awt.event.*;
public class Login extends Frame implements ActionListener
{
    Label l1,l2,l3,l4;
    TextField t1,t2;
    Button b1, b2;
    Login(String name)
    {
        super(name);
        l1=new Label("User Name : ");
        l1.setBounds(50,50,100,20);
        l2=new Label("Password : ");
        l2.setBounds(50,80,100,20);
        t1= new TextField();
        t1.setBounds(150,50,100,20);
        t2= new TextField();
        t2.setBounds(150,80,100,20);
        t2.setEchoChar('*');
        b1= new Button("Login");
        b1.setBounds(120,110,80,20);
        b2 = new Button("Forgot Password");
        b2.setBounds(50, 150, 110, 20);

        l3 = new Label("->");
        l3.setBounds(50,200,250,20);
        add(l1);add(t1);
        add(l2);add(t2);
        add(b1);add(l3);
        add(b2);
        b1.addActionListener(this);
        b2.addActionListener(this);
        setSize(400,400);
        setLayout(null);
        setVisible(true);
        addWindowListener(new WindowAdapter()
        {
            public void windowClosing(WindowEvent e)
            {
                dispose();
            }
        } );
    }

    public void actionPerformed(ActionEvent e)
    {
        String str = e.getActionCommand();
        String uname = t1.getText();
        String pwd = t2.getText();
        if(uname.equals("User_101") && pwd.equals("#101"))
            l3.setText("Welcome User_101");
        else
            l3.setText("Invalid Username or Password");
    }
    public static void main(String[] args)
    {
        new Login("Login Window");
    }
}
```

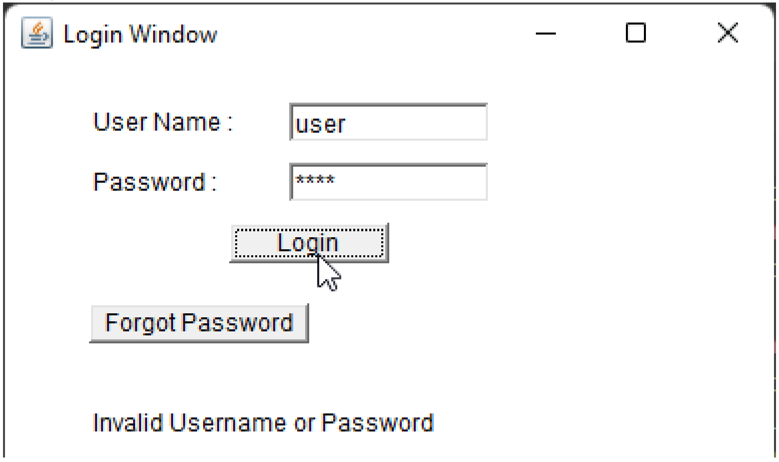
.

.

.

.

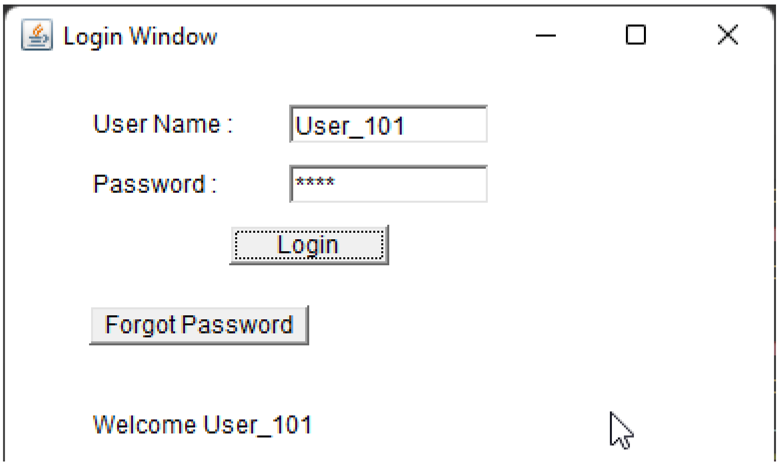
Output:



User Name : user

Password : ****

Invalid Username or Password



User Name : User_101

Password : ****

Welcome User_101