**Objective:** To write a program with getter and setter methods.

package java2\_Project13;

public class Student {

  private String lastName;

  private String firstName;

  private int studentId;

  private double[] projects;

  private double[] quizzes;

  public Student(String lastName, String firstName, int studentId){

     this.lastName=lastName;

     this.firstName=firstName;

     this.studentId=studentId;

     projects = new double[] {-1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0,

         -1.0,-1.0, -1.0, -1.0, -1.0,};

     quizzes = new double [] {-1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0};

  }

public boolean setProjectScore(double newProjects, int projectNumber) {

        projects[projectNumber] = newProjects;

        if (projectNumber >= 0 || projectNumber <= 15)

            return true; else

            return false;

      }

  public boolean setQuizScore(double newQuizzes, int quizNumber) {

         quizzes[quizNumber] = newQuizzes;

         if (quizNumber >= 0 || quizNumber <= 15)

            return true; else

            return false;

      }

  public double getProjectScore(int projectNumber){

      if (projects[projectNumber] < 0 || projects[projectNumber] > 15)

            return -1.0; else

              return projects[projectNumber];

  }

  public double getQuizScore(int quizNumber){

          if (quizzes[quizNumber] < 0 || quizzes[quizNumber] > 15)

                return -1.0; else

                  return quizzes[quizNumber];

  }

  public int getNextProjectIndex(){

      int nextProjectIndex=0;

      while (nextProjectIndex <= 10)

          if (projects[nextProjectIndex] == -1.0)

              return nextProjectIndex;

          nextProjectIndex++;

      return -1;

  }

  public int getNextQuizIndex(){

      int nextQuizIndex=0;

      while (nextQuizIndex <= 10)

          if (projects[nextQuizIndex] == -1.0)

              return nextQuizIndex;

          nextQuizIndex++;

      return -1;

  }

  public String getLastName() {

    return lastName;

  }

  public void setLastName(String lastName) {

    this.lastName = lastName;

  }

  public String getFirstName() {

    return firstName;

  }

  public void setFirstName(String firstName) {

    this.firstName = firstName;

  }

  public int getStudentId() {

    return studentId;

  }

  public void setStudentId(int studentId) {

    this.studentId = studentId;

  }

}

Write a program in Java which generates a random number between 0 and 1. If the number generated is less than 0.5 then the program must print “The Value is less than 0.5” and if the number generated is greater than or equal to 0.5, then the program must print “The value is greater than 0.5”. Write the program using a) If – else b) using the Ternary operator

public class RandomUtil {

public static void main(String a[]){

// Math.random() number return a random double value between

// 0 and 1, where 0 is inclusive and 1 is exclusive.

System.out.println("Random number between 0 and 1 : " + Math.random());

System.out.println("Random number between 0 and 1 : " + Math.random());

// Now suppose you need random integer between 0 to 10

// you can do following

System.out.println("Random integer between 0 and 10 : "

+ getRandom(10));

System.out.println("Random integer between 0 and 10 : "

+ getRandom(10));

// Now let's get random number between 1 and 10

System.out.println("Random value between 1 and 10 : "

+ getRandomInteger(10, 1));

System.out.println("Random value between 1 and 10 : "

+ getRandomInteger(10, 1));

// Now let's find random number between 1 and 100

System.out.println("Random number between 1 and 100 : "

+ getRandomInteger(100, 1));

System.out.println("Random number between 1 and 100 : "

+ getRandomInteger(100, 1));

// generate random number between 1000 and 9999

System.out.println("Random value between 1000 and 9999 : "

+ getRandomInteger(1000, 10000));

System.out.println("Random value between 1000 and 9999 : "

+ getRandomInteger(100, 10000));

}

/\*

\* Java method to return random integer between 0 and

\* given number. Pay attention to brackets while casting

\* (int) Math.random\*max will return incorrect result.

\*/

public static int getRandom(int max){

return (int) (Math.random()\*max);

}

/\*

\* returns random integer between minimum and maximum range

\*/

public static int getRandomInteger(int maximum, int minimum){

return ((int) (Math.random()\*(maximum - minimum))) + minimum;

}

}

Output

Random number between 0 and 1 : 0.04465203527687556

Random number between 0 and 1 : 0.914215990673822

Random integer between 0 and 10 : 9

Random integer between 0 and 10 : 8

Random value between 1 and 10 : 5

Random value between 1 and 10 : 2

Random number between 1 and 100 : 29

Random number between 1 and 100 : 37

Random value between 1000 and 9999 : 3517

Random value between 1000 and 9999 : 3581

**Objective:** To sort a given set of numbers

public class ExArraySortElement

{

public static void main(String[] args)

{

int n, temp;

//scanner class object creation

Scanner s = new Scanner(System.in);

//input total number of elements to be read

System.out.print("Enter the elements you want : ");

n = s.nextInt();

//integer array object

int a[] = new int[n];

//read elements

System.out.println("Enter all the elements:");

for (int i = 0; i < n; i++)

{

a[i] = s.nextInt();

}

//sorting elements

for (int i = 0; i < n; i++)

{

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

{

if (a[i] > a[j])

{

temp = a[i];

a[i] = a[j];

a[j] = temp;

}

}

}

//print sorted elements

System.out.println("Ascending Order:");

for (int i = 0; i < n ; i++)

{

System.out.println(a[i]);

}

}

}

Assignment 4: Interfaces and reference objects

public class Polymorphism

{

   public static void main(String[] args)

   {

          A ref1 = new C();

          B ref2 = (B) ref1;

          System.out.println(ref2.f());

   }

}

class A

{

   int f() { return 0; }

}

class B extends A

{

   int f() { return 1; }

}

class C extends B

{

   int f() { return 2; }

}

class MyClass

{

   public static void main(String[] args)

   {

          MyClass a;

          MySubclass b;

          a = new MyClass();           //(1)

          b = new MySubclass();  //(2)

          a = b;                           //(3)

          b = a;                           //(4)

          a = new MySubclass();  //(5)

   }

}

class MySubclass extends MyClass

{

}