| **Name:** | Mahadev Balla |
| --- | --- |
| **UID:** | 2023300010 |
| **Experiment No.** | 7B |

| **AIM:** | Implement a program to demonstrate abstract class. |
| --- | --- |
| **Program 1** | |
| **PROBLEM STATEMENT :** | Write a program to calculate the percentage of marks obtained in three subjects (each out of 100) by student A and in four subjects (each out of 100) by student B.  Create an abstract class 'Marks' with an abstract method 'getPercentage'.  It is inherited by two other classes 'A' and 'B' each having a method with the same name which returns the percentage of the students.  The constructor of student A takes the marks in three subjects as its parameters and the marks in four subjects as its parameters for student B.  Input to be taken from the user. Display the marks in highest order of student A and B. |
| **PROGRAM:** | import java.util.\*;  abstract class Marks{  private ArrayList<Double> marks = new ArrayList<>();  ArrayList<Double> getMarks(){  return marks;  }  abstract double getPercentage();  }  class A extends Marks{  private ArrayList<Double> marks = new ArrayList<>();  A(ArrayList<Double> marks){  this.marks = marks;  }  @Override  double getPercentage(){  double total=0;  for (int i=0; i<marks.size() ;i++){  total += marks.get(i);  }  return total/marks.size();  }  ArrayList<Double> getMarks(){  return marks;  }  }  class B extends Marks {  private ArrayList<Double> marks = new ArrayList<>();  B(ArrayList<Double> marks){  this.marks = marks;  }  @Override  double getPercentage(){  double total=0;  for (int i=0; i<marks.size() ;i++) {  total += marks.get(i);  }  return total/marks.size();  }  ArrayList<Double> getMarks(){  return marks;  }  }  public class stud{  public static void main(String[] args){  Scanner sc = new Scanner(System.in);  double temp;  System.out.println("Enter marks for student A -");  ArrayList<Double> marksA = new ArrayList<>();  for (int i=0; i<3; i++){  do{  System.out.print("Subject " + (i+1) + " : ");  temp = sc.nextDouble();  if(temp<0 || temp>100){  System.out.println("Enter a valid input!!");  }  else{  marksA.add(temp);  }  }  while(temp<0 || temp>100);  }  System.out.println("Enter marks for student B -");  ArrayList<Double> marksB = new ArrayList<>();  for (int i=0; i<4; i++){  do{  System.out.print("Subject " + (i+1) + " : ");  temp = sc.nextDouble();  if(temp<0 || temp>100){  System.out.println("Enter a valid input!!");  }  else{  marksB.add(temp);  }  }  while(temp<0 || temp>100);  }  A a = new A(marksA);  B b = new B(marksB);  if(a.getPercentage() > b.getPercentage()){  System.out.println("Percentage of student A : " + a.getPercentage());  System.out.println("Percentage of student B : " + b.getPercentage());  System.out.println("Student A has higher percentage!! ");  System.out.println("Marks of Student A -");  displayMarks(a);  System.out.println("Marks of Student B -");  displayMarks(b);  }  else if(a.getPercentage() < b.getPercentage()){  System.out.println("Percentage of student A : " + a.getPercentage());  System.out.println("Percentage of student B : " + b.getPercentage());  System.out.println("Student B has higher percentage: " + b.getPercentage());  System.out.println("Marks of Student B -");  displayMarks(b);  System.out.println("Marks of Student A -");  displayMarks(a);  }  else{  System.out.println("Both students have the same percentage: " + b.getPercentage());  System.out.println("Marks of Student A -");  displayMarks(a);  System.out.println("Marks of Student B -");  displayMarks(b);  }  }  private static void displayMarks(Marks student) {  ArrayList<Double> marks = student.getMarks();  ArrayList<Double> sortedMarks = new ArrayList<>(marks);  Collections.sort(sortedMarks, Collections.reverseOrder());  for (double mark : sortedMarks) {  int index = marks.indexOf(mark) + 1;  System.out.println("Subject " + index + " : " + mark);  }  }  } |
| **RESULT:** | |
| **Program 2** | |
| **PROBLEM STATEMENT :** | Design a Quiz System as following  There is a abstract class Questions. It includes a variable question\_text- This contains the text of the question. You can set question text using constructors. It includes a method verifyAnswer() to verify. It includes abstract method specifications for getQuestion(), getSolution()  There are 2 other classes True or False Questions, Multiple Choice Question. They all inherit the Question class and provide implementation for its methods. They all have a variable called mySolution( Note: different class will have different data type for this variable)  Child classes will call parents getType method to get the question type  Multiple Choice Question also has variables for different options  Note: Make a Main class where you can make and give the quiz  Write an interactive program in Java  Shuffle all the questions randomly  Each question must specify which type of question it is.  Total score and solution must be displayed in the end. |
| **PROGRAM:** | import java.util.\*;  abstract class Questions{  abstract String getQuestion();  abstract int getSolution();  }  class TrueOrFalseQuestion extends Questions{  String ques; int mySolution;  TrueOrFalseQuestion(String q, int mySolution){  this.ques = q; this.mySolution = mySolution;  }  String getQuestion(){  return ques;  };  int getSolution(){  return mySolution;  };  }  class MultipleChoiceQuestion extends Questions{  String ques; int mySolution;  MultipleChoiceQuestion(String q, int mySolution){  this.ques = q; this.mySolution = mySolution;  }  String getQuestion(){  return ques;  };  int getSolution(){  return mySolution;  };  }  class quiz {  public static void main(String[] args) {  Scanner sc = new Scanner(System.in);  int ans1, ans2, ans3, ans4, ans5, ans6; int total1=0, total2=0;  System.out.println("----- True or False Round -----\n");  TrueOrFalseQuestion q1 = new TrueOrFalseQuestion("Penguins are capable of flight.",2);  System.out.println(q1.getQuestion());  do{  System.out.print("Enter your answer(1=T or 2=F) : ");  ans1 = sc.nextInt();  if(ans1 != 1 && ans1 != 2){  System.out.println("Enter a valid input!!");  }  }  while(ans1 != 1 && ans1 != 2);  if(ans1 == q1.getSolution()){  System.out.println("Correct answer!!");  total1 += 20;  }  else{  System.out.println("Incorrect answer!!");  }  TrueOrFalseQuestion q2 = new TrueOrFalseQuestion("The Statue of Liberty was a gift from France to the United States.",1);  System.out.println(q2.getQuestion());  do{  System.out.print("Enter your answer(1=T or 2=F) : ");  ans2 = sc.nextInt();  if(ans2 != 1 && ans2 != 2){  System.out.println("Enter a valid input!!");  }  }  while(ans2 != 1 && ans2 != 2);  if(ans2 == q2.getSolution()){  System.out.println("Correct answer!!");  total1 += 20;  }  else{  System.out.println("Incorrect answer!!");  }    TrueOrFalseQuestion q3 = new TrueOrFalseQuestion("Venus is the hottest planet in our solar system.",1);  System.out.println(q3.getQuestion());  do{  System.out.print("Enter your answer(1=T or 2=F) : ");  ans3 = sc.nextInt();  if(ans3 != 1 && ans3 != 2){  System.out.println("Enter a valid input!!");  }  }  while(ans3 != 1 && ans3 != 2);  if(ans3 == q3.getSolution()){  System.out.println("Correct answer!!");  total1 += 20;  }  else{  System.out.println("Incorrect answer!!");  }  System.out.println("Total points scored in True or False Round : " + total1 + "/60");  System.out.println("\n----- MCQ Round -----\n");  MultipleChoiceQuestion q4 = new MultipleChoiceQuestion("Which of the following is the largest planet in our solar system?", 2);  System.out.println(q4.getQuestion() + "\n1. Mars\n2. Jupiter\n3. Earth\n4. Saturn");  do{  System.out.print("Enter your answer : ");  ans4 = sc.nextInt();  if(ans4 != 1 && ans4 != 2 && ans4 != 3 && ans4 != 4){  System.out.println("Enter a valid input!!");  }  }  while(ans4 != 1 && ans4 != 2 && ans4 != 3 && ans4 != 4);  if(ans4 == q4.getSolution()){  System.out.println("Correct answer!!");  total2 += 20;  }  else{  System.out.println("Incorrect answer!!");  }  MultipleChoiceQuestion q5 = new MultipleChoiceQuestion("Who is known as the father of modern physics?", 2);  System.out.println(q5.getQuestion() + "\n1. Isaac Newton\n2. Albert Einstein\n3. Galileo Galilei\n4. Nikola Tesla");  do{  System.out.print("Enter your answer : ");  ans5 = sc.nextInt();  if(ans5 != 1 && ans5 != 2 && ans5 != 3 && ans5 != 4){  System.out.println("Enter a valid input!!");  }  }  while(ans5 != 1 && ans5 != 2 && ans5 != 3 && ans5 != 4);  if(ans5 == q5.getSolution()){  System.out.println("Correct answer!!");  total2 += 20;  }  else{  System.out.println("Incorrect answer!!");  }  MultipleChoiceQuestion q6 = new MultipleChoiceQuestion("Which planet is known as the 'Red Planet'?", 1);  System.out.println(q6.getQuestion() + "\n1. Mars\n2. Neptune\n3. Venus\n4. Pluto");  do{  System.out.print("Enter your answer : ");  ans6 = sc.nextInt();  if(ans6 != 1 && ans6 != 2 && ans6 != 3 && ans6 != 4){  System.out.println("Enter a valid input!!");  }  }  while(ans6 != 1 && ans6 != 2 && ans6 != 3 && ans6 != 4);  if(ans6 == q6.getSolution()){  System.out.println("Correct answer!!");  total2 += 20;  }  else{  System.out.println("Incorrect answer!!");  }  System.out.println("Total points scored in MCQ Round : " + total2 + "/60");  System.out.println("Total points scored : " + (total1+total2) + "/120");  }  } |
| **RESULT:** | |
| **CONCLUSION:** | Studied the implementation of abstract class to solve the given problems. |