Name:	Debjit Ghosal
UID:	20233000065
Experiment No.	7A

AIM:	Program on Abstraction:Implement a Program to demonstrate Abstraction using abstract class
Program 1	
PROBLEM STATEMENT :	**Program 1: 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:	/*
	Program 1: 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.
	*/
	
	/*
	Flow =
	Hard-coded Quiz. No user input. User will only give the test.
	5 question
	3 mcq
	2 true/false
	
	abstract methods:
	
	*/
	import java.util.Scanner;
	
	abstract class Questions {
	String question_text;
	
	public Questions(String question_text) {
	this.question_text = question_text;

```
    }

    abstract boolean verifyAnswer(String giveAns);

    abstract String getQuestion();

    abstract String getSolution();
}

class TrueFalse extends Questions {
    String mySolution;

    public TrueFalse(String question_text, String mySolution) {
        super(question_text);
        this.mySolution = mySolution;
    }

    @Override
    boolean verifyAnswer(String giveAns) {
        if(mySolution.equals(giveAns)) {
            return true;
        } else {
            return false;
        }
    }

    @Override
    String getQuestion() {
        return question_text;
    }

    @Override
    String getSolution() {
        return mySolution;
    }

}

class MCQ extends Questions {
    String mySolution;
```

```
    public MCQ(String question_text, String mySolution) {
        super(question_text);
        this.mySolution = mySolution;
    }

    @Override
    boolean verifyAnswer(String giveAns) {
        if(mySolution.equals(giveAns)) {
            return true;
        } else {
            return false;
        }
    }

    @Override
    String getQuestion() {
        return question_text;
    }

    @Override
    String getSolution() {
        return mySolution;
    }
}

public class Quiz {
    public static void main(String[] args) {
                    // Three MCQ Array.
        MCQ[] mcq = new MCQ[3];
        mcq[0] = new MCQ("What is the national animal of India?", "Tiger");
        mcq[1] = new MCQ("What is the national fruit of India?", "Mango");
        mcq[2] = new MCQ("What is the national bird of India?", "Peacock");
                     // Two TrueFalse Array.
         TrueFalse[] tf = new TrueFalse[3];
        tf[0] = new TrueFalse("India is a democratic country.", "true");
        tf[1] = new TrueFalse("India got independence in 1945.", "false");

        System.out.println("Let the Quiz begin!");
        Scanner in = new Scanner(System.in);
```

```
        int score = 0;
        System.out.println("MCQ Quiz:");
        for(int i = 0; i < 3; i++) {
            System.out.println(mcq[i].getQuestion());
            String ans = in.nextLine();
            if(mcq[i].verifyAnswer(ans)) {
                System.out.println("Correct!");
                score++;
            } else {
                System.out.println("Incorrect Ans.!");
            }
        }

        System.out.println("True - False Quiz: true/false input only.");
        for(int i = 0; i < 2; i++) {
            System.out.println(tf[i].getQuestion());
            String ans = in.nextLine();
            if(tf[i].verifyAnswer(ans)) {
                System.out.println("Correct!");
                score++;
            } else {
                System.out.println("Incorrect Ans.!");
            }
        }

        System.out.println("Your score is: " + score + " out of 5");
        in.close();
    }
}
```

RESULT:

```
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ javac Quiz.java
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ java Quiz
Let the Quiz begin!
MCQ Quiz:
What is the national animal of India?
Lion
Incorrect Ans.!
What is the national fruit of India?
Mango
Correct!
What is the national bird of India?
Peacock
Correct!
True - False Quiz: true/false input only.
India is a democratic country.
true
Correct!
India got independence in 1945.
false
Correct!
Your score is: 4 out of 5
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$
```

Program 2	
PROBLEM STATEMENT :	**Program 2: The task is to write a Java program in which a user will get K trials to guess a randomly generated number. Below are the rules of the game:** **If the guessed number is bigger than the actual number, the program will respond with the message that the guessed number is higher than the actual number.** **If the guessed number is smaller than the actual number, the program will respond with the message that the guessed number is lower than the actual number.** **If the guessed number is equal to the actual number or if the K trials are exhausted, the program will end with a suitable message.** **Approach: Below are the steps:** **The approach is to generate a random number using Math.random() method in Java.** **Now using a loop, take K input from the user and for each input print whether the number is smaller or larger than the actual number.** **If within K trials the user guessed the number correctly, print that the user won.**

	Else print that he was not able to guess and then print the actual number. **Define the Player class with a abstract function named getGuess().**

PROGRAM:	/*Program 2:
	The task is to write a Java program in which a user will get K trials
	to guess a randomly generated number. Below are the rules of the game:
	If the guessed number is bigger than the actual number, the program will respond with the message
	that the guessed number is higher than the actual number.
	If the guessed number is smaller than the actual number, the program will respond with the message
	that the guessed number is lower than the actual number.
	If the guessed number is equal to the actual number or if the K trials are exhausted,
	the program will end with a suitable message.
	
	Approach: Below are the steps:
	The approach is to generate a random number using Math.random() method in Java.
	Now using a loop, take K input from the user and for each input print whether the number is smaller
	or larger than the actual number.
	If within K trials the user guessed the number correctly, print that the user won.
	Else print that he was not able to guess and then print the actual number.
	Define the Player class with a abstract function named getGuess().
	*/
	import java.util.Scanner;
	
	abstract class Player{
	int k;
	abstract void getGuess();
	}
	
	class User extends Player{
	static int max = 100;
	static int min = 1;
	static int range = max - min + 1;

```
    int chances = 5;
    int current = 0;

    Scanner in = new Scanner(System.in);
    static int number = (int)(Math.random()*range) + min;
    void getGuess(){

        for ( int current = 0; current < chances; current++ ) {

            System.out.println("Guess the number between 1 and 100");
            k = in.nextInt();

            if (k == number)
            {
                System.out.println("You've Guessed the number. Congratulations!");
                break;
            }
            else if(k>number)
                System.out.println("Your number is higher. Attempts Remaining: "
+ (chances - current - 1));
             else if (k < number)
                System.out.println("Your number is lower. Attempts Remaining: " +
(chances - current - 1));
            }
        if (chances == current)
            System.out.println("No Attempts Remaining // Game Over //");
        }
    }
public class Game{
    public static void main(String[] args) {

        User u1 = new User();
        u1.getGuess();
    }
}
```

RESULT:

```
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ javac Game.java
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ java Game
Guess the number between 1 and 100
50
Your number is lower. Attempts Remaining: 4
Guess the number between 1 and 100
70
Your number is higher. Attempts Remaining: 3
Guess the number between 1 and 100
60
Your number is lower. Attempts Remaining: 2
Guess the number between 1 and 100
65
Your number is higher. Attempts Remaining: 1
Guess the number between 1 and 100
62
Your number is higher. Attempts Remaining: 0
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ java Game
Guess the number between 1 and 100
65
Your number is lower. Attempts Remaining: 4
Guess the number between 1 and 100
70
Your number is lower. Attempts Remaining: 3
Guess the number between 1 and 100
80
Your number is lower. Attempts Remaining: 2
Guess the number between 1 and 100
90
Your number is higher. Attempts Remaining: 1
Guess the number between 1 and 100
88
Your number is higher. Attempts Remaining: 0
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$
```

```
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ java Game
Guess the number between 1 and 100
50
Your number is higher. Attempts Remaining: 4
Guess the number between 1 and 100
25
Your number is lower. Attempts Remaining: 3
Guess the number between 1 and 100
13
Your number is lower. Attempts Remaining: 2
Guess the number between 1 and 100
35
Your number is lower. Attempts Remaining: 1
Guess the number between 1 and 100
45
You've Guessed the number. Congratulations!
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$
```

CONCLUSION:	I have learnt about Abstraction and more precisely the use of abstract class.