UE19CS353 – OOADJ

Lab Assignment – 2

Date – 31 - Jan - 2022

|  |  |  |
| --- | --- | --- |
| SRN – PES1UG19CS090 | Name – Arvind Krishna | Sem - 6  Sec - B |

Question:

Text, letter

Description automatically generated

Code:

/\*

    - Arvind Krishna

    - 31/01/2022 04:13 PM

    - path - /mnt/d/pesu/Sem 6/java/OOAD-Labs/week-2

\*/

// define an class TestQuestion that has a String variable called question, and a readQuestion method

import java.util.Scanner;

class TestQuestion {

    protected String question;

    public *void* readQuestion(Scanner *scan*){}

}

// define 3 subclasses,

// shortAnswer - numLines - by default 1

// LongAnswer - numLines

// MCQ - must have numChoices, and array of string for choices

class ShortAnswer extends TestQuestion {

    private *int* numLines = 1;

    public *void* readQuestion(Scanner *scan*) {

        System.out.print("Enter the question: ");

        question = *scan*.nextLine();

        System.out.print("Enter the number of lines: ");

        String nextIntString = *scan*.nextLine();

        numLines = Integer.parseInt(nextIntString);

    }

    //toString method

    public String toString() {

        return "Question: " + question + "\n\tNumber of Lines: " + numLines;

    }

}

class LongAnswer extends TestQuestion {

    private *int* numLines = -1;

    public *void* readQuestion(Scanner *scan*) {

        System.out.print("Enter the question: ");

        question = *scan*.nextLine();

        System.out.print("Enter the number of lines: ");

        String nextIntString = *scan*.nextLine();

        numLines = Integer.parseInt(nextIntString);

    }

    //toString method

    public String toString() {

        return "Question: " + question + "\n\tNumber of Lines: " + numLines;

    }

}

class MCQ extends TestQuestion {

    private *int* numChoices = -1;

    private String[] choices;// = new String[numChoices];

    public *void* readQuestion(Scanner *scan*) {

        System.out.print("Enter the question: ");

        question = *scan*.nextLine();

        System.out.print("Enter the number of choices: ");

        String nextIntString = *scan*.nextLine(); //get the number as a single line

        numChoices = Integer.parseInt(nextIntString);

        choices = new String[numChoices];

        for (*int* i = 0; i < numChoices; i++) {

            System.out.print("Enter choice " + (i + 1) + ": ");

            choices[i] = *scan*.nextLine();

        }

    }

    //toString method

    public String toString() {

        String str = question + "\nThe options are: ";

        for (*int* i = 0; i < numChoices; i++) {

            str += "\n\t" + (i + 1) +". "+ choices[i];

        }

        return str;

    }

}

//the main method in TQManager class should contain an array of TestQuestion references of any of the 3 subclasses.

// In the main function, the user should be able to add a question of any of the 3 subclasses, and accordingly add it to the array and readQuestion method is invoked

// Thereafter, display all the questions by implicitly invoking the toString method.

public class TQManager {

    public static *void* main(String[] *args*) {

        TestQuestion[] questions = new TestQuestion[10];

        Scanner scan = new Scanner(System.in);

*int* choice = -1;

*int* numQuestions = 0;

        do {

            System.out.println("1. Short Answer");

            System.out.println("2. Long Answer");

            System.out.println("3. Multiple Choice");

            System.out.println("4. Exit");

            System.out.print("Enter your choice: ");

            String nextIntString = scan.nextLine(); //get the number as a single line

            choice = Integer.parseInt(nextIntString);

            switch (choice) {

            case 1:

                questions[numQuestions] = new ShortAnswer();

                questions[numQuestions].readQuestion(scan);

                numQuestions++;

                break;

            case 2:

                questions[numQuestions] = new LongAnswer();

                questions[numQuestions].readQuestion(scan);

                numQuestions++;

                break;

            case 3:

                questions[numQuestions] = new MCQ();

                questions[numQuestions].readQuestion(scan);

                numQuestions++;

                break;

            case 4:

                break;

            default:

                System.out.println("Invalid choice!");

                break;

            }

        } while (choice != 4);

        scan.close();

        System.out.println("\nThe questions are: ");

        for (*int* j = 0; j < numQuestions; j++) {

            System.out.println(j + 1 + ") " + questions[j].toString() + "\n");

        }

    }

}

Output Screenshots:

Text

Description automatically generated

Text

Description automatically generated

Printing all of the saved questions:

Text

Description automatically generated