Algorithm 

|  |  |
| --- | --- |
| **Product Name** | Applied Degree in Software Engineering (BDSE) |
| **Qualification Name** | Applied Degree in Software Engineering |
| **Product Name** | Programming Foundations |
| **Module Name (BDSE)** | **Programming Foundations** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Learner name | | | Assessor name | |
| Agung Yuda Pratama | | |  | |
| **Date issued** | **Completion date** | | | **Submitted on** |
|  |  | | |  |
|  | | | | |
| **Project title** | | **Algorithm** | | |

### Plagiarism

Plagiarism covers both copying or paraphrasing with the source without proper understanding and citation. The source includes but is not limited to

* Books, Journal Articles, Online resources
* Others Assignment or project
* Software program
* Images

The learners must submit their work without plagiarism while producing the evidence against the assessment criteria. They must apply proper citation and Harvard referencing. Their work must go through the Turnitin software to check for the similarity index. Please consult the relevant module instructor or mentor if you need any further advice.

|  |
| --- |
| **Learner declaration** |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  Student signature: Date:23-11-2022 |
|  |

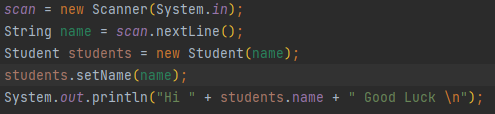
Lithan | EDU Class © 2022

I**MPLEMENTATION**

Step 1 - Greeting the user.



Step 2 - Take the name as input from the user.



Step 3- Give the options for the MCQ exam to the user.

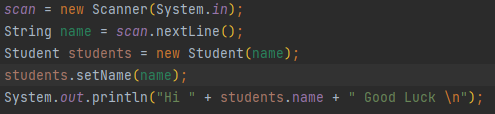
private void showFiles(File[] *files*) {  
 for (int *i* = 0; *i* < *files*.length; *i*++){  
 System.*out*.println("Here The MCQ Set: " + *files*[*i*].getName().replace(".csv", " ") + "(choose " + (*i* + 1) + ")");  
 }  
}  
public void chooseTheTest() {  
 showFiles(Objects.*requireNonNull*(directoryPath.listFiles()));  
 choice = new Scanner(System.*in*);  
 String[] *f* = directoryPath.list();  
 int *userChoice* = 0;  
 boolean *trueChoice* = true;  
 while (*trueChoice*){  
 System.*out*.print("\nInput The Number: ");  
 *userChoice* = choice.nextInt();  
 System.*out*.println();  
 if (*userChoice* == 0 || *userChoice* < 0 || *userChoice* > Objects.*requireNonNull*(*f*).length){  
 System.*out*.println("Please Input The Correct Number!");  
 } else {  
 *trueChoice* = false;  
 }  
 }  
 String *path* = "MCQ/" + *f*[*userChoice* - 1];  
 read.startTest(*path*);

**ALGORITHM**

**Step 1- Take the name input from the user.**

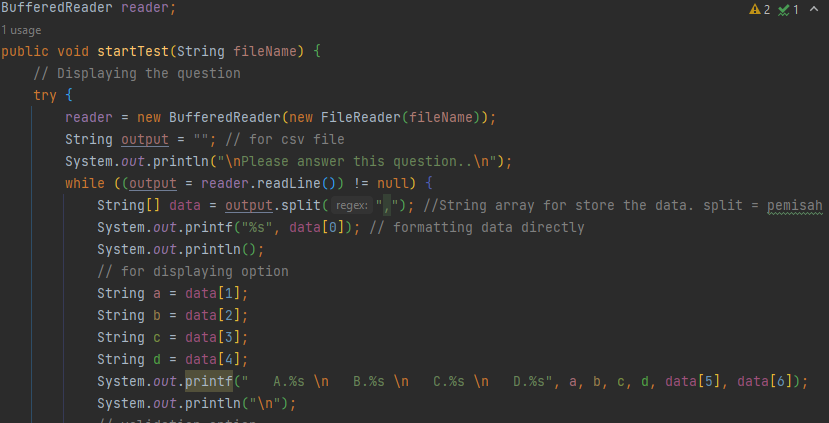
Creating a Scanner object.

Using scan.nextLine(), we are taking the name input from the user and storing it in the name variable.



**Step 2- Giving options to the user.**

Getting the list of all the files in the MCQ folder and storing it in the *files* array.



Using a for loop to iterate through each file in the *files* array and displaying the file name with .csv removed.

private void showFiles(File[] *files*) {  
 for (int *i* = 0; *i* < *files*.length; *i*++){  
 System.*out*.println("Here The MCQ Set: " + *files*[*i*].getName().replace(".csv", " ") + "(choose " + (*i* + 1) + ")");  
 }  
}

**Step 3- Take the choice input from the user.**

We use a scanner object to take input from the user and store it in the *choice* variable.

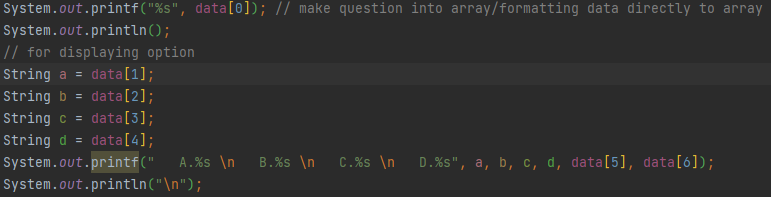


**Step 4- Depending upon the choice, read the corresponding file line by line.**



**Step 5- Separate the questions, options, and correct answers from the line. So it will display questions and options.**

We use an index to show and separate the question, option and correct answer



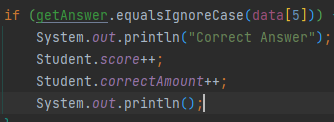
**Step 7- Take the answer input from the user.**

Use a while and scanner for getting input from the user. Use if-else for validation input from user

boolean *optionTest* = true;  
String *getAnswer* = null;  
while (*optionTest*){  
 Scanner *answer* = new Scanner(System.*in*);  
 // for displaying answer  
 System.*out*.print("Your answer: ");  
 *getAnswer* = *answer*.nextLine();  
 if (Objects.*equals*(*getAnswer*, "a") || Objects.*equals*(*getAnswer*, "b") || Objects.*equals*(*getAnswer*, "c") || Objects.*equals*(*getAnswer*, "d")){  
 *optionTest* = false;  
 } else {  
 System.*out*.println("\nYou can only input A,B,C,D\n");  
 }  
}

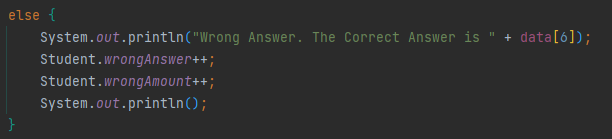
**Step 8- If the user answers correctly, update the score.**

Use if for condition correct answer



**Step 9- else the user answers wrong, update the wrong answer.**

Use else for condition wrong answer and show the correct answer to user



**Step 10- Display the score when the entire file is read.**

Make finalScore, name and in the print section you add name+correctAmount+wrongAmount+finalScore to display the Final Score from User

private static void showResult() {  
 int *finalScore* = Student.*getScore*();  
 String *name* = *s*.name;  
 System.*out*.println("Name: " + *name* + "\nYou answered "+ Student.*correctAmount* + " Questions Right, " + Student.*wrongAmount* + " Question Wrong for a total of 10 Questions " + "\nScore: " + *finalScore* +"0%");  
 System.*out*.println("\nThank you for attending The Test \nHave a nice day ");  
}

