Adrita Ghosh

Work Log

1. **17th December 2023, 20:43 - 22:08**   
   I started working on a quiz game. I'm creating a flow chart to represent the full game. I've detailed the important checkpoints throughout the entire game in the flowchart.

First, a starting place and the player's name will be requested. There will be welcome greetings and Quiz subject categories displayed. A category will be selected; Questions will be posed; The program will take the player's answer; The program will show the player if the answer is correct or incorrect. The final score will be shown. The final score will be saved in an additional file. The final statement will be shown. The game will end.

**Display welcome message and categories**

**Start**

**Get the Player Name**

**Ask the question from Chosen Category**

**Choose the category**

**Get the answer from the Player and if wrong display the correct answer**

**Display final score**

**End**

**Display closing message**

**Save score to file**

1. **17th December 2023, 23:45 - 18th December 2023, 12:30**  
   I'm working on a Trivia Quiz Game programme. I declared a static variable, "NameOfThePlayer," under the "Final\_Game" namespace, internal class named "Programme," to record the player's name. The "Main" Procedure is the program's starting point, where I invoke the "GetPlayerName" Procedure to question the user for their name. Following that, I show a personalised welcome message that introduces the participant to the Trivia Quiz Game, as well as two categories: Science and History. The "GetPlayerName" Procedure is in charge of obtaining the player's name via the console.

I want to use File I/O and List in the program, that’s why I keep ‘System.Collections.Generic’ namespace and add ‘System.IO’ namespace along with ‘System’.

1. **27th December 2023, 17:08 - 23:45**  
   I’m planning to create a structure and to populate a list inside that structure. With the help of that List, I want to Load questions inside the program. I find a helpful piece of writing online regarding this and adapt the way of coding to populate List with a structure (Stack Overflow, 2018). I create a structure called “QuizQuestions” to hold individual quiz questions. I included a question string, a string array for multiple-choice options, and a character to define the correct answer, inside the structure. I wrote one procedure under the name of “LoadQuizQuestionsFromFile”. I create the procedure in such a way, that it reads every quiz question from a defined location and incorporate them inside the program. For storage, I give a file location and a collection of "QuizQuestions". I chose to implement the "file.ReadAllLines" function instead of the "StreamReader" to read all the lines. I adapt the idea from another helpful piece of writing online (Stack Overflow, 2018). Then I add a “for” loop to process the lines iteratively. But when I tried to assigned last line with “RightAnswer”, it’s showing error as initially I assigned value-type char with “RightAnswer”. So, to mitigate the error, I used [0] and I adapt the idea from one more piece of writing online (Stack Overflow, 2021). Now with each iteration consisting of a group of six lines matching to a quiz question. And to iterate each question, inside the “for” loop, I have added ‘.Add’ method. I learned about this method from Microsoft documentation on the List<T>.Add(T) Method (n.d.).
2. **28th December 2023, 14:08 - 20:57**

To make sure everything works right, I've set up a "try-catch block" in the "LoadQuizQuestionsFromFile" procedure to manage any possible mistakes when loading.

I've declared two global variables as static lists, named "ScienceQuiz" and "HistoryQuiz", storing questions for science and history quizzes. In the main process, I run "LoadQuizQuestionsFromFile" twice to fill these lists with questions from two different text files. These are called "SciQstn.txt" and "HistQstn.txt". This setup makes sure that each subject area has its own group of questions.

I create a function called "CategorySelectionInput" where I set the rules. These include what to write for comment, a lower limit, and an upper limit. In this function, I ask the gamer to put in a number choice from what was given according to the range. To make sure the information is right, I use a do-while loop. It takes what the user types in, tries to understand it as a number and check if it’s inside set limits. If the input is right, it leaves the loop and ends. If not, it shows a mistake message. It asks the user try again to pick their choice until they give right input. To run the program in this way, I use “if-else” loop inside the “do-while” loop. But while creating the condition for “if” I encountered an error. I wanted to convert entered value of the user to integer value with “int.TryParse(Console.ReadLine()” method but it turns out that the condition requires an “out” return and the requirement was showing by Microsoft Visual studio. I learned about this “out” keyword from Microsoft documentation on the out (C# Reference) (n.d.) and use out “CategoryInput” with “int.TryParse(Console.ReadLine()” and the error was removed. I initialized int “CategoryInput” with 0 previously.

I make another new function called "SelectedCategory" using the given input. If the number is 1, this function gives a list of science-related quiz questions. Otherwise, it provides a list of history-linked quiz questions.

I use the function "CategorySelectionInput" to ask the user which category they want. They can choose 1 or 2, and this choice is saved in a variable called "ChosenCategory". After that, I use the function "SelectedCategory", giving it "ChosenCategory" as well as the lists for "ScienceQuiz" and "HistoryQuiz". The list of quiz questions from the picked category is given to a variable called "ChosenQuestions".

1. **28th December 2023, 21:11 - 23:06**  
   Today, I have created a function named “AnswerSelectionFromChoices” with four parameters. One string value-type for comment, two char value-types for LowerLimit and UpperLimit, and one int value-type for maxAttempts to limit the attempts of the invalid entry. In this function, I aim to ask the user to input a valid character from a predefined range. I have set the range with value-type char (as I have set options with alphabetic serial) and initialized char GamerInput with the default value ‘\0’. Inside the function, I declared the value-type bool CorrectInput as "false”, and I initialize the “attempts” under integer to 0. I put entire coding lines to validate the input inside a do-while loop so that the function will run until the valid input is provided.

Next, I initialized static variable “Score” to zero.

I have created another procedure, "AskQuizQuestion” to call the entire quiz question with options and the correct answer. I’m planning to call this procedure in the main procedure. I’m planning to call the function “AnswerSelectionFromChoices” inside the procedure. First, I call the question from the List, then I create a do-while loop to prompt the user for an answer. I set the answer range based on available choices, and for that reason, I set the option range under the parameter of the function “AnswerSelectionFromChoices” among chars ‘A’ to ‘D’. Inside the loop, I increment the “Score” when the answer is correct.

In the main procedure, I have incorporated a “foreach” loop to iterate through every quiz question present in the set of “ChosenQuestions” and call the procedure “AskQuizQuestion” to present the question inside the console to the user and handle their responses.

In the last section of the main procedure, I display the final score, the goodbye message, and I have included Console.ReadLine() to view the result inside the console.

After executing the program, I identified a glitch that the user performance is not accurately reflecting. The program is not terminating until the correct answer is given.

1. **29th December 2023, 10:11 - 12:14**  
   I have resolved the previously encountered issue where the score was inaccurately recorded. While examining the “AskQuizQuestion” procedure, I identified that the “while” statement in the “do-while” loop used a boolean parameter set to true, causing the loop to persist until a correct answer was given. To address this bug and control attempts, I introduced a new parameter to limit the number of tries. I increase the “Attempts” variable, incorporating a maximum attempt limit set to 2. Subsequently, I verify if the attempts are below the maximum allowed count. If true, I notify the user of an incorrect answer and display the remaining attempts. Conversely, if the maximum attempts have been reached, I inform the user that all attempts have been used, revealing the correct answer to the question. The “while” statement now checks if the number of attempts is less than the maximum allowed attempts (Attempts < MaxAttempts).  
   A new issue has surfaced. An exception was triggered, signalling a problem while loading questions from the file named “HisQstn.txt”.  
   Another question emerges: How can I restrict the entry of invalid inputs?
2. **29th December 2023, 12:30 - 13:42**

Exception of History file not being loaded is recovered. Upon analysis, I identified that the text file had been saved in the system with the name “HistQstn.txt”, The letter “t” was mentioned in the system but not in the program. In the program it was mentioned as “HistQstn.txt”. Consequently, the program failed to read the file, as its name did not match the expected file name in the program. The bug has been rectified by ensuring the correct file name.

To limit the invalid input, in the “AnswerSelectionFromChoices” function, I incorporated the following changes:  
I introduced a constant, MaxInvalidAttempts, set to 2. Following that, I initialized a variable, InvalidAttempts, to 0. Inside the else statement, I included the line InvalidAttempts++;. Subsequently, I created an additional “If” loop with the condition (InvalidAttempts == MaxInvalidAttempts). If this condition is met, I display a message indicating that the maximum attempts for invalid input have been reached and move to the next question. However, despite these modifications, the limitation on invalid attempts is not effective.

1. **29th December 2023, 19:30 - 21:12**  
   In the “AskQuizQuestion” procedure, I included the following statement: “Moving on to Next question”. Additionally, all the previously implemented modifications intended to restrict invalid input was reverted.  
   I attempted to address the problem of not restricting invalid input by catching the code, but my efforts are unsuccessful.
2. **29th December 2023, 22:00 - 23:19**  
   I introduced a new procedure called “ScoreFile” to log the score. Within this procedure, I utilized a try-catch block to handle potential exceptions. Using a “StreamWriter”, I appended the player’s name, score, and the current date to the specified file. A success message is displayed on successful recording. In case of an exception, an error message is shown. I invoke the “ScoreFile” procedure in the main procedure of the program to record the score.  
   While coding the “ScoreFile” procedure, I employ the “using” statement to enfold the “StreamWriter”.  
   I introduced randomization for the questions by including the following lines in the “AskQuizQuestion” procedure:  
   Random random = new Random();

QuizQuestions question = EveryQuestion[random.Next(EveryQuestion.Count)];  
Additionally, when invoking the procedure in the main program, I modified the parameter of “AskQuizQuestion” with “ChosenQuestions”.

1. **30th December 2023, 22:06 - 31st December 2023, 00:00**  
   I explored websites and read several articles in an attempt to resolve the problem of not restricting invalid input. To limit the invalid input, I have added another integer parameter to limit the attempt of invalid input, “maxAttempts” in “AnswerSelectionFromChoices” function. Despite of trying to catch the code, I was unable to succeed in my efforts.
2. **31st December 2023, 13:46 - 16:41**  
   I introduced a new category, Math, and included additional questions and answers in the Science and History text files. However, upon adding the Math question text file to the program, I encountered an error when loading Science and History questions. I couldn’t pinpoint the issue initially. To identify the error, I included {ex message} in the catch statement within the “LoadQuizQuestionsFromFile” procedure. This revealed the error message: “Index was outside the bounds of the array”.

I have taken all the Maximum of the math questions from an online math article (Byjus, 2020).

1. **31st December 2023, 19:08 - 19:21**  
   Recognizing the significance of {ex message} in the catch statement of the “LoadQuizQuestionsFromFile” procedure, I incorporated the same {ex message} into the catch statement of the “ScoreFile” procedure.
2. **2nd January 2024, 13:45 - 16:41**  
   I resolved the loading error with Science and History questions. It appears that, after adding questions to both files, I inadvertently included two extra blank lines. This caused the “outside the bounds of the array” error. I rectified the issue by removing those blank lines, and now the program is running without any errors.
3. **2nd January 2024, 17:54 - 20:42**  
   In line 32, in WriteLine section I changed “Three from Two”.   
   Tried to limit both invalid and incorrect attempts. But it turns out, I can either limit the invalid attempts or I can limit the incorrect attempts.
4. **6th January 2024, 17:35 - 19:11**  
   I cleared all the changes I had made to limit incorrect attempts from the procedure “AskQuizQuestion”.

Now when I run the program, I have realised another error, that is, for randomization, questions are getting repeated. I have tried to mitigate the problem.

I need to find the solution for this purpose.

1. **6th January 2024, 20:00 - 23:15**  
   After thoroughly researched, I have learned about a method “.RemoveAt”. With this method I’m deleting all the questions which are being asked. I have changed the List name to “AskedQuestion”. I have declared int randomIndex variable and I have added the value of “random.Next(0, AskedQuestion.Count)” to it. Next, I retrieved the “QuizQuestions” element at randomIndex and assigned it to the variable “question”. Next after invoking the question, I have added “.RemoveAt” method, to delete the “AskedQuestion” means already asked question. In this way I have rectified the error or repeating question. I took the inspiration from an online article (freeCodeCamp, 2022).

Next, I have done few modifications in my code. I have added Console.Clear() after choosing the category, and after completing each question to clear my console.

After that, I have assigned one variable named “FinalScore”, where I’m calculation the percentage of the player and displaying the same. Based on the “FinalScore”, I have assigned Grades. Additionally, I have added another int named FinalScore in the “ScoreFile” procedure to add the final score in the saved file as well.

In “LoadQuizQuestionsFromFile”, I have replaced the magic number 6 with integer “numberoflines”.

Evaluation of Code Quality:  
I consider the overall quality of my code to be Good.

To carry out the main programme, I have developed a number of processes and functions using the correct modularity principles, enclosing everything inside the Main Procedure. I've connected the structure I made with List. I've used Loops to generate appropriate conditional branching. The "AnswerSelectionFromChoices" and "CategorySelectionInput" functions both employ constants for category and option limitations. I have used "try-catch" statements to manage failures. I have commented clearly in the programme. Magic number handling has been done by me in the "LoadQuizQuestionsFromFile" process. I implemented comprehensible Grading logic into the application.

However, there are also restrictions in my code.

Global variables are what I utilised for "ScienceQuiz," "HistoryQuiz," and "MathQuiz." I believe the script's variable nomenclature is unclear. The problem of not capping the wrong response is still not being fixed. I think I could have dealt with mistakes more deeply.

References:

Stack Overflow (2018). How do you populate a list with a structure at form load in C#  
Retrieved from  
<https://stackoverflow.com/questions/53252690/how-do-you-populate-a-list-with-a-structure-at-form-load-in-c-sharp>

Stack Overflow (2018). File.ReadAllLines or Stream Reader  
Retrieved from  
<https://stackoverflow.com/questions/23989677/file-readalllines-or-stream-reader>

Stack Overflow (2021). C# - How to convert string to char?  
Retrieved from  
<https://stackoverflow.com/questions/33946594/c-sharp-how-to-convert-string-to-char>  
Microsoft (n.d.). List<T>.Add(T) Method  
Retrieved from  
<https://learn.microsoft.com/en-us/dotnet/api/system.collections.generic.list-1.add?view=net-8.0>  
Microsoft (n.d.). out (C# Reference) (n.d.)  
Retrieved from  
<https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/keywords/out>

Byjus (2020). Maths Quiz Questions with Answers (MCQs)  
Retrieved from  
<https://byjus.com/maths/maths-quiz-questions/>

freeCodeCamp (2022). How to Remove an Item from a List in C#  
Retrieved from  
<https://www.freecodecamp.org/news/how-to-remove-an-item-from-a-list-in-c/#:~:text=at%20that%20index.-,Below%20is%20a%20code%20snippet%20showing%20how%20to%20use%20the,an%20item%20from%20a%20list.&text=)%3B%20%7D%20%7D%20%7D%20%7D-,In%20the%20code%20above%2C%20FirstName.,at%200%2C%20not%201>