

Who wants to be a millionaire

Generated by Doxygen 1.14.0

1 Class Index	1
1.1 Class List	1
2 File Index	3
2.1 File List	3
3 Class Documentation	5
3.1 CategoryManager Class Reference	5
3.1.1 Detailed Description	5
3.1.2 Constructor & Destructor Documentation	5
3.1.2.1 CategoryManager()	5
3.1.3 Member Function Documentation	6
3.1.3.1 CategoryCount()	6
3.1.3.2 ChooseCategory()	6
3.1.3.3 RemoveCategory()	6
3.1.3.4 ShowCategories()	7
3.1.3.5 ShowCategory()	7
3.1.4 Property Documentation	7
3.1.4.1 HasCategories	7
3.2 Logo Class Reference	7
3.2.1 Detailed Description	7
3.2.2 Member Function Documentation	8
3.2.2.1 FinishLogo()	8
3.2.2.2 StartLogo()	8
3.3 Program Class Reference	8
3.3.1 Detailed Description	8
3.4 Question Class Reference	8
3.4.1 Detailed Description	9
3.4.2 Constructor & Destructor Documentation	9
3.4.2.1 Question()	9
3.4.3 Member Function Documentation	9
3.4.3.1 LoadQuestions()	9
3.4.4 Property Documentation	10
3.4.4.1 Answers	10
3.4.4.2 CorrectAnswer	10
3.4.4.3 Point	10
3.4.4.4 Text	10
4 File Documentation	11
4.1 CategoryManager.cs File Reference	11
4.1.1 Detailed Description	11
4.2 HintSystem.cs File Reference	11
4.2.1 Detailed Description	11

4.3 Logo.cs File Reference	12
4.3.1 Detailed Description	12
4.4 PlayerManager.cs File Reference	12
4.5 Program.cs File Reference	12
4.5.1 Detailed Description	12
4.6 Question.cs File Reference	12
4.6.1 Detailed Description	13
4.7 RecordManager.cs File Reference	13
4.7.1 Detailed Description	13
4.8 Utils.cs File Reference	13
4.8.1 Detailed Description	13
Index	15

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

CategoryManager

Manages the list of question categories for the quiz game. Responsible for loading categories from files, displaying them, allowing selection, and updating the list after each selection 5

Logo

Static class responsible for showing stylized console logos at the start and end of the game session 7

Program

Main class containing the game entry point 8

Question

Represents a single quiz question, including the question text, multiple choice answers, the correct answer index, and the point value 8

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

CategoryManager.cs	
Contains the CategoryManager (p. 5) class for handling question categories	11
HintSystem.cs	
Provides hint functionalities for the quiz game	11
Logo.cs	
Displays start and finish logos for the game	12
PlayerManager.cs	12
Program.cs	
Entry point for the quiz game	12
Question.cs	
Represents a quiz question with answers and related data	12
RecordManager.cs	
Provides functionality to save and display player records	13
Utils.cs	
Utility methods for user input validation and game flow control	13

Chapter 3

Class Documentation

3.1 CategoryManager Class Reference

Manages the list of question categories for the quiz game. Responsible for loading categories from files, displaying them, allowing selection, and updating the list after each selection.

Public Member Functions

- **CategoryManager** (string folderPath="Questions")
*Initializes a new instance of the **CategoryManager** (p. 5) class. Loads all category files from the specified folder path.*
- void **ShowCategories** ()
Displays the list of available categories to the user.
- string **ChooseCategory** (int choose)
Returns the file path for the chosen category based on user input.
- void **ShowCategory** (string filePath)
Displays the name of the selected category.
- void **RemoveCategory** (string filePath)
Removes the specified category from the list of available categories.
- int **CategoryCount** ()
Gets the number of available categories.

Properties

- bool **HasCategories** [get]
Gets a value indicating whether there are any categories remaining.

3.1.1 Detailed Description

Manages the list of question categories for the quiz game. Responsible for loading categories from files, displaying them, allowing selection, and updating the list after each selection.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 CategoryManager()

```
CategoryManager.CategoryManager (  
    string folderPath = "Questions")
```

Initializes a new instance of the **CategoryManager** (p. 5) class. Loads all category files from the specified folder path.

Parameters

<i>folderPath</i>	Path to the folder containing category files (default: "Questions").
-------------------	--

3.1.3 Member Function Documentation

3.1.3.1 CategoryCount()

```
int CategoryManager.CategoryCount ()
```

Gets the number of available categories.

Returns

The count of categories.

3.1.3.2 ChooseCategory()

```
string CategoryManager.ChooseCategory (  
    int choice)
```

Returns the file path for the chosen category based on user input.

Parameters

<i>choice</i>	User's numeric choice (1-based index).
---------------	--

Returns

File path of the selected category.

Exceptions

<i>Exception</i>	Thrown if the choice is invalid.
------------------	----------------------------------

3.1.3.3 RemoveCategory()

```
void CategoryManager.RemoveCategory (  
    string filePath)
```

Removes the specified category from the list of available categories.

Parameters

<i>filePath</i>	Path to the category file to remove.
-----------------	--------------------------------------

3.1.3.4 ShowCategories()

```
void CategoryManager.ShowCategories ()
```

Displays the list of available categories to the user.

3.1.3.5 ShowCategory()

```
void CategoryManager.ShowCategory (
    string filePath)
```

Displays the name of the selected category.

Parameters

<i>filePath</i>	Path to the selected category file.
-----------------	-------------------------------------

3.1.4 Property Documentation

3.1.4.1 HasCategories

```
bool CategoryManager.HasCategories [get]
```

Gets a value indicating whether there are any categories remaining.

The documentation for this class was generated from the following file:

- **CategoryManager.cs**

3.2 Logo Class Reference

Static class responsible for showing stylized console logos at the start and end of the game session.

Static Public Member Functions

- static void **StartLogo** ()
Displays the start logo with a welcome message.
- static void **FinishLogo** ()
Displays the finish logo with a thank you message and pauses for 10 seconds before closing.

3.2.1 Detailed Description

Static class responsible for showing stylized console logos at the start and end of the game session.

3.2.2 Member Function Documentation

3.2.2.1 FinishLogo()

```
void Logo.FinishLogo () [static]
```

Displays the finish logo with a thank you message and pauses for 10 seconds before closing.

3.2.2.2 StartLogo()

```
void Logo.StartLogo () [static]
```

Displays the start logo with a welcome message.

The documentation for this class was generated from the following file:

- **Logo.cs**

3.3 Program Class Reference

Main class containing the game entry point.

3.3.1 Detailed Description

Main class containing the game entry point.

The documentation for this class was generated from the following file:

- **Program.cs**

3.4 Question Class Reference

Represents a single quiz question, including the question text, multiple choice answers, the correct answer index, and the point value.

Public Member Functions

- **Question** (string text, string[] answers, int correctAnswer, int point)
*Initializes a new instance of the **Question** (p. 8) class.*

Static Public Member Functions

- static List< **Question** > **LoadQuestions** (string filePath)
Loads a list of questions from a file. Each line in the file should be formatted as: QuestionText|Answer1|Answer2|↔ Answer3|Answer4|CorrectAnswerIndex|Points.

Properties

- string **Text** [get]
Gets the question text.
- string[] **Answers** [get]
Gets the array of answer options.
- int **CorrectAnswer** [get]
Gets the index (1-based) of the correct answer.
- int **Point** [get]
Gets the number of points awarded for a correct answer.

3.4.1 Detailed Description

Represents a single quiz question, including the question text, multiple choice answers, the correct answer index, and the point value.

3.4.2 Constructor & Destructor Documentation

3.4.2.1 Question()

```
Question.Question (  
    string text,  
    string[] answers,  
    int correctAnswer,  
    int point)
```

Initializes a new instance of the **Question** (p. 8) class.

Parameters

<i>text</i>	The question text.
<i>answers</i>	Array of 4 possible answers.
<i>correctAnswer</i>	The correct answer index (1-based).
<i>point</i>	Points awarded for correct answer.

3.4.3 Member Function Documentation

3.4.3.1 LoadQuestions()

```
List< Question > Question.LoadQuestions (  
    string filePath) [static]
```

Loads a list of questions from a file. Each line in the file should be formatted as: QuestionText|Answer1|Answer2|↵ Answer3|Answer4|CorrectAnswerIndex|Points.

Parameters

<i>filePath</i>	Path to the questions file.
-----------------	-----------------------------

Returns

List of loaded questions.

3.4.4 Property Documentation

3.4.4.1 Answers

```
string [] Question.Answers [get]
```

Gets the array of answer options.

3.4.4.2 CorrectAnswer

```
int Question.CorrectAnswer [get]
```

Gets the index (1-based) of the correct answer.

3.4.4.3 Point

```
int Question.Point [get]
```

Gets the number of points awarded for a correct answer.

3.4.4.4 Text

```
string Question.Text [get]
```

Gets the question text.

The documentation for this class was generated from the following file:

- **Question.cs**

Chapter 4

File Documentation

4.1 CategoryManager.cs File Reference

Contains the **CategoryManager** (p. 5) class for handling question categories.

Classes

- class **CategoryManager**

Manages the list of question categories for the quiz game. Responsible for loading categories from files, displaying them, allowing selection, and updating the list after each selection.

4.1.1 Detailed Description

Contains the **CategoryManager** (p. 5) class for handling question categories.

4.2 HintSystem.cs File Reference

Provides hint functionalities for the quiz game.

Classes

- class **HintSystem**

A static class that handles the hint system during a question. Offers different types of hints to assist the player.

4.2.1 Detailed Description

Provides hint functionalities for the quiz game.

4.3 Logo.cs File Reference

Displays start and finish logos for the game.

Classes

- class **Logo**

Static class responsible for showing stylized console logos at the start and end of the game session.

4.3.1 Detailed Description

Displays start and finish logos for the game.

4.4 PlayerManager.cs File Reference

Classes

- class **PlayManager**

Static class that handles the display and interaction logic for a quiz question, including processing user input, checking correctness, and updating the score.

4.5 Program.cs File Reference

Entry point for the quiz game.

Classes

- class **Program**

Main class containing the game entry point.

4.5.1 Detailed Description

Entry point for the quiz game.

This file contains the main game loop logic, including player interaction, category and question selection, scoring, and record saving.

4.6 Question.cs File Reference

Represents a quiz question with answers and related data.

Classes

- class **Question**

Represents a single quiz question, including the question text, multiple choice answers, the correct answer index, and the point value.

4.6.1 Detailed Description

Represents a quiz question with answers and related data.

4.7 RecordManager.cs File Reference

Provides functionality to save and display player records.

Classes

- class **RecordManager**

Static class responsible for managing game records, including saving player scores to a file and displaying the records table.

4.7.1 Detailed Description

Provides functionality to save and display player records.

4.8 Utils.cs File Reference

Utility methods for user input validation and game flow control.

Classes

- class **Utils**

Static helper class providing methods for validating user input, asking for user decisions during the game, and retrieving random wrong answers.

4.8.1 Detailed Description

Utility methods for user input validation and game flow control.

Index

- Answers
 - Question, 10
- CategoryCount
 - CategoryManager, 6
- CategoryManager, 5
 - CategoryCount, 6
 - CategoryManager, 5
 - ChooseCategory, 6
 - HasCategories, 7
 - RemoveCategory, 6
 - ShowCategories, 6
 - ShowCategory, 7
- CategoryManager.cs, 11
- ChooseCategory
 - CategoryManager, 6
- CorrectAnswer
 - Question, 10
- FinishLogo
 - Logo, 8
- HasCategories
 - CategoryManager, 7
- HintSystem.cs, 11
- LoadQuestions
 - Question, 9
- Logo, 7
 - FinishLogo, 8
 - StartLogo, 8
- Logo.cs, 12
- PlayerManager.cs, 12
- Point
 - Question, 10
- Program, 8
- Program.cs, 12
- Question, 8
 - Answers, 10
 - CorrectAnswer, 10
 - LoadQuestions, 9
 - Point, 10
 - Question, 9
 - Text, 10
- Question.cs, 12
- RecordManager.cs, 13
- RemoveCategory
 - CategoryManager, 6
- ShowCategories
 - CategoryManager, 6
- ShowCategory
 - CategoryManager, 7
- StartLogo
 - Logo, 8
- Text
 - Question, 10
- Utils.cs, 13