CASE STUDY DOCUMENTATION ON
MCQ BASED ONLINE EXAM SYSTEM

Overview

This project implements an online Multiple-Choice Question (MCQ) based exam system using Python. It consists of two main parts:

1. **Question management** (handled by `question_master.py`)

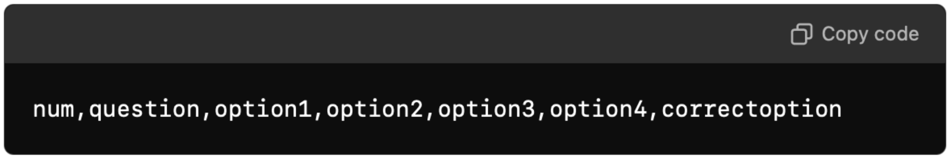
2. **Exam client** (handled by `exam_client.py`)

The system allows the admin to manage exam questions (CRUD operations) and participants to take the exam, with their scores recorded in a leaderboard.

1. File Description

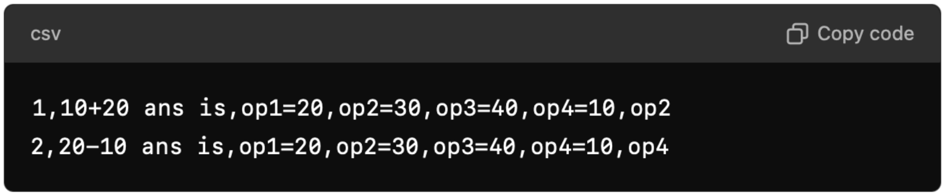
1.1 questions.csv

This CSV file stores the exam questions and their multiple-choice answers. It follows this structure:



```
num,question,option1,option2,option3,option4,correctoption
```

Example:



```csv
1,10+20 ans is,op1=20,op2=30,op3=40,op4=10,op2
2,20-10 ans is,op1=20,op2=30,op3=40,op4=10,op4
```

1.2 question_master.py

This file is responsible for managing the questions that appear in the exam. It allows the admin to:

- Add new questions (auto-numbered)

- Search for a question by number

- Delete a question by number

- Modify an existing question

- Display all questions

Functions in question_master.py:

- `load_questions()`:

  - Reads the `questions.csv` file and loads it into a Python data structure (list of dictionaries).

- `add_question()`:

  - Adds a new question to the data structure and updates the CSV file.

- `search_question(num)`:

  - Searches for and displays a specific question based on the question number.

- `save_questions()`:

  - Writes the updated question list to `questions.csv`.

1.3 exam_client.py

This file is the main exam interface for participants. It performs the following tasks:

- Displays the current date and time.

- Prompts the participant for their name and university.

- Presents the questions one by one.

- Records the participant's answers and calculates their score.

- Saves the participant's score to the leaderboard.

- Displays the top 5 participants on the leaderboard.

Functions in exam_client.py:

- `load_questions()`:
  - Loads the questions from the `questions.csv` file.
- `take_exam(student_name, university)`:
  - Runs the exam by presenting each question and capturing the participant's answer.
  - Scores the exam based on the correct answers.
  - Displays the final score.
- `save_score(name, score)`:
  - Appends the participant's name and score to the `leaderboard.csv` file.
- `display_leaderboard()`:
  - Reads and sorts the leaderboard and displays the top 5 participants.

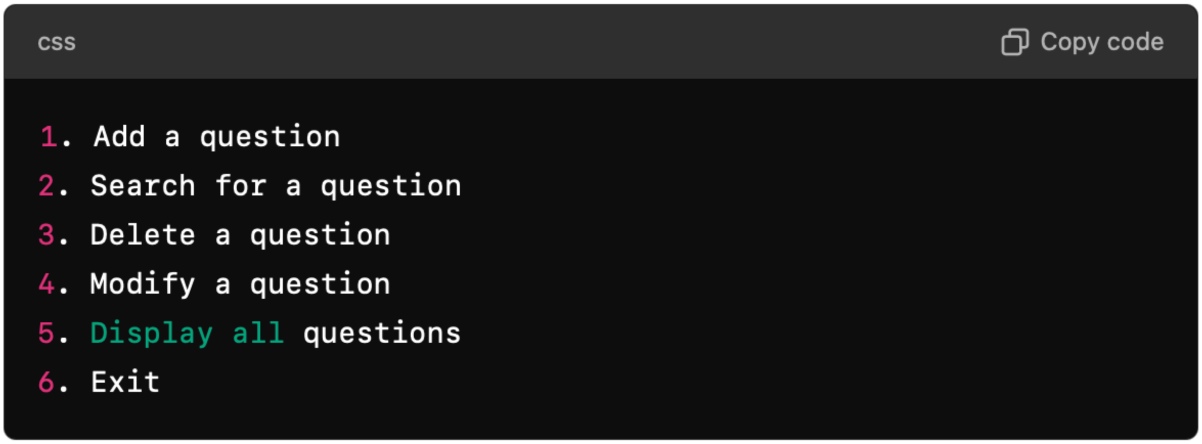
2. Program Execution

2.1 Running the `question_master.py` (Admin)

The admin can run `question_master.py` to manage exam questions. Upon execution, the program provides a menu to: Add a question

- Search for a question by number
- Delete a question by number
- Modify an existing question
- Display all questions
- Exit

Example usage:



```
1. Add a question
2. Search for a question
3. Delete a question
4. Modify a question
5. Display all questions
6. Exit
```

The admin selects an option by inputting the corresponding number. For instance, selecting "1" will prompt the admin to add a new question.

2.2 Running the `exam_client.py` (Participant)

Participants can run `exam_client.py` to take the exam. Upon starting, they are asked to enter:

- Their name

- Their university

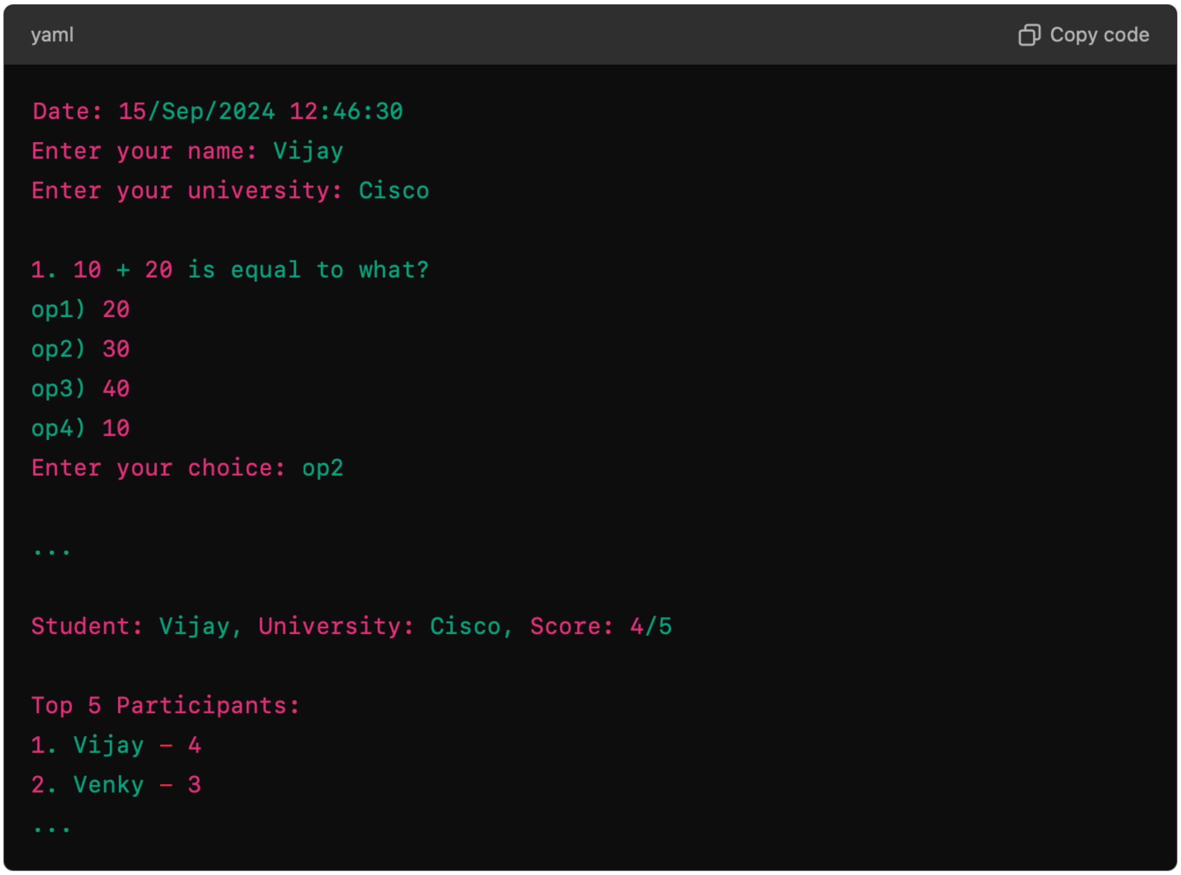
The system then:

- Displays the current date and time.

- Presents each question in turn, along with the available answer options.

- Asks the participant to enter their choice (e.g., `op1`).

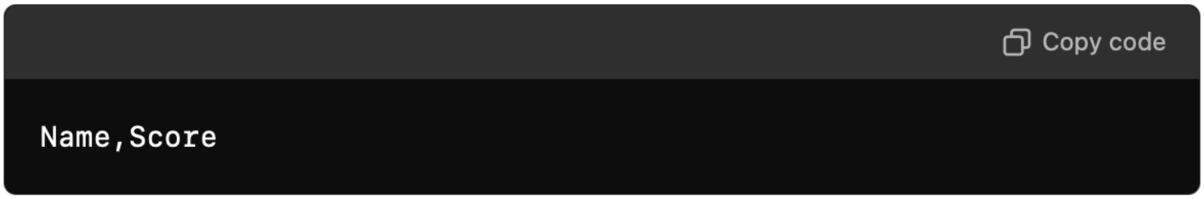
At the end of the exam, the participant's score is displayed, and their result is saved in `leaderboard.csv`. The system will then display the leaderboard with the top 5 participants.



```
Date: 15/Sep/2024 12:46:30
Enter your name: Vijay
Enter your university: Cisco

1. 10 + 20 is equal to what?
op1) 20
op2) 30
op3) 40
op4) 10
Enter your choice: op2

...

Student: Vijay, University: Cisco, Score: 4/5

Top 5 Participants:
1. Vijay - 4
2. Venky - 3
...
```

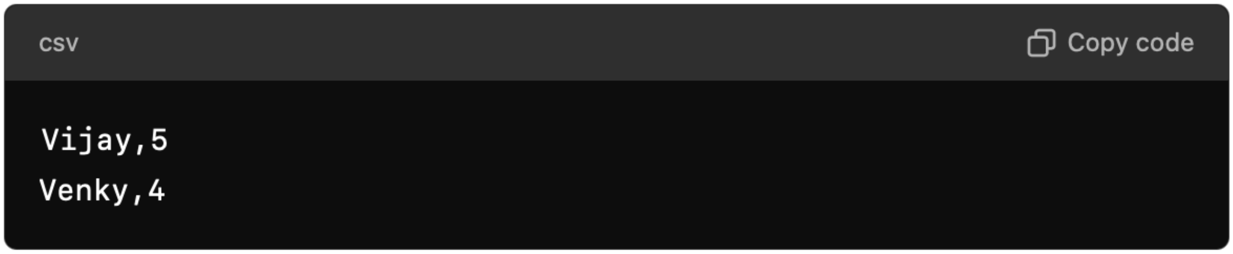
3. Leaderboard

The leaderboard.csv file stores the participants' scores in the following format:



```
Name,Score
```

Each time a participant completes the exam, their score is saved in this file. The leaderboard displays the top 5 participants ranked by their score.



```csv
Vijay,5
Venky,4
```

4. Additional Features

- **Modularity:** The code is modular, with clear separation between the question management and the exam client system.

- **Error Handling:** The program uses try-except blocks for robust error handling.

- **Logging:** The system can be extended to include logging mechanisms for better debugging and tracing.

- **Customization:** The system can be customized to include additional features such as time limits, user authentication, and more.

P; The leaderboard.csv file stores the participants' scores in the following format:
The leaderboard.csv file stores the participants' scores in the following format:

This documentation covers the major functionalities and file structure of the MCQ-based online exam system.