OOP Quiz - Challenge

I had suggested that over the break, people could perhaps revisit the **previous Quiz Challenge (https://canvas.qub.ac.uk/courses/11041/pages/challenge-3-pub-quiz)** from earlier in the course and consider how they could put the various new concepts we have studied into practice to improve it.

I didn't set a formal spec at the time, and I'm not going to do so now either, but will put a few videos and coded solutions here, trying to show my own approach to the idea and in doing so, try to revisit and revise as many of the newer OOP and File Reading/Writing concepts covered on the course as I can (where they would be useful and practical, not just for the sake of it).

This will be a work in progress initially so perhaps check back later for a final version, but don't let that stop you making a start on your own approach. There is no "right" way to do it.

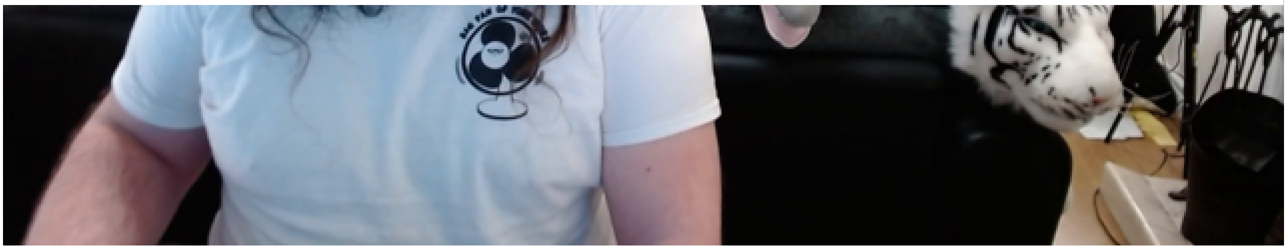
Some things I will try to explore:

- Creating an appropriately designed Question Class.
- Creating a Quiz Class with appropriate instance variables (mainly a list of Question objects) and methods to run Quiz functionality
- Creating a main method/driver class to instantiate the objects and invoke the necessary methods to create/play a quiz.
- Explore different ways of getting the question/answer data into the objects and where to put that (Quiz Constructor? Main Method of Driver etc). Also passing data from String arrays or from file reading. (file reading probably more flexible and also a good revision opportunity)
- Exploring polymorphism and inheritance etc with the Question objects by implementing some different types of questions and the considerations that that would bring. (Multiple choice will be the main alternative question type I'll work on and knowing me I'll probably overcomplicate it a bit, but have a think about it yourself if you haven't considered it already)

Initial getting started video below, with more to come. (watch this space)

OOP QuizChallenge 1 - Design and Thought Process

86 views · 1 like · 1 comment



Direct Link if required on your device: **Link (https://web.microsoftstream.com/video/196af313-57f3-4a14-8ceb-d5b3f978ce11)**

Solution (one possible):

Played around with a few ideas and produced the following sample solution. Will also do some walk through videos discussing them, but try examining/debugging the code yourself to see if you can understand how and why it works the way it does.

Add the following files to your project src to test

Question.java (https://canvas.qub.ac.uk/courses/11041/files/1503971/download?wrap=1) ↓ (https://canvas.qub.ac.uk/courses/11041/files/1503971/download?download_frd=1) , TextMatchQuestion.java (https://canvas.qub.ac.uk/courses/11041/files/1503975/download?wrap=1) ↓ (https://canvas.qub.ac.uk/courses/11041/files/1503975/download?download_frd=1) , QuestionMCQ.java (https://canvas.qub.ac.uk/courses/11041/files/1503972/download?wrap=1) ↓ (https://canvas.qub.ac.uk/courses/11041/files/1503972/download?download_frd=1)

Quiz.java (https://canvas.qub.ac.uk/courses/11041/files/1503973/download?wrap=1) ↓ (https://canvas.qub.ac.uk/courses/11041/files/1503973/download?download_frd=1)

QuizDriverClass.java (https://canvas.qub.ac.uk/courses/11041/files/1504041/download?wrap=1) ↓ (https://canvas.qub.ac.uk/courses/11041/files/1504041/download?download_frd=1)

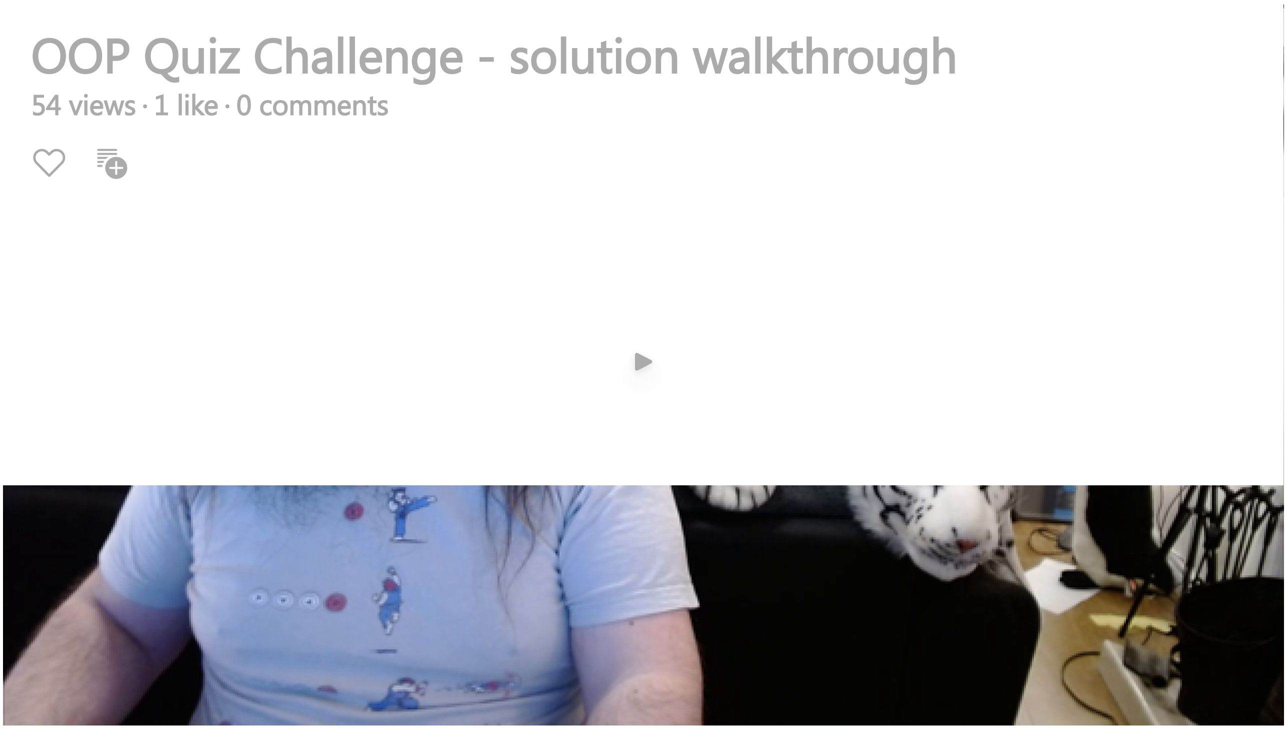
Question.java is an abstract class and the other 2 Question types inherit from it
The Quiz object has a number of different constructors so Quizes can be created in different ways

The Driver class does some testing/demoing of the main functionality though not everything has been fully tested/verified before uploading.

Driver class also uses the following sample csv file to test one of the Quiz constructors. Add to the root of your project as usual: **[QList.csv](https://canvas.qub.ac.uk/courses/11041/files/1503977/download?wrap=1)** **(https://canvas.qub.ac.uk/courses/11041/files/1503977/download?wrap=1)** **(https://canvas.qub.ac.uk/courses/11041/files/1503977/download?download_frd=1)**

Could perhaps criticise aspects of this solution - eg validation business logic in constructor rather than setters. Setters not provided for all fields etc.
Some of these were deliberate choices, but worth thinking about regardless.

Walkthrough Video



OOP Quiz Challenge - solution walkthrough

54 views · 1 like · 0 comments

Direct Link if required: **Link** **(https://web.microsoftstream.com/video/3fa6c4ac-8192-4e23-ad3f-9465f84159b5)**