

Washington State University
School of Electrical Engineering and Computer Science
Spring 2019

CptS 479 Mobile Application Development

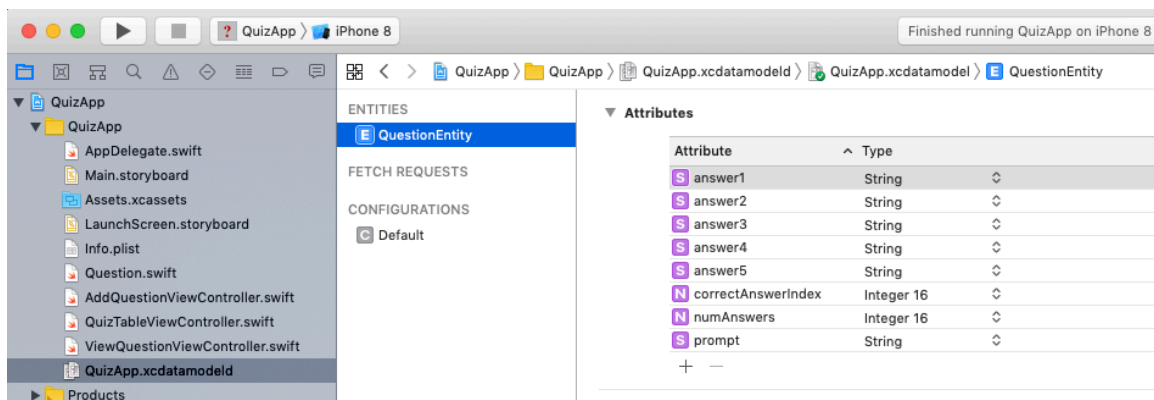
Homework 11

Due: April 2, 2019 (11:59pm)

General Instructions: Put the entire app directory into one zip file and submit as an attachment under Content → Homework 11 for this course on the Blackboard Learn system by the above deadline. Note that you may submit multiple times, but only the most recent entry submitted before the above deadline will be graded.

This homework is essentially the same as Homework 5, except that the quiz questions will be persistently stored using Core Data. My solution to Homework 5 is available on Blackboard Learn. You may use my solution or your own as a starting point. Specifically,

1. Add the CoreData boilerplate code into your app. You can get this code by creating a new project in Xcode and checking the “Use CoreData” box on the “Choose options for your new project” screen. The code will be at the end of the AppDelegate.swift file. Don’t forget to also include the saveContext line in the applicationWillTerminate method.
2. Create a Data Model for a Question. You can add a Data Model to your project by choosing File→New→File and choosing the “Data Model” file under “Core Data”. The data model should consist of a single entity with attributes corresponding to the Question properties: prompt, answer1, answer2, answer3, answer4, answer5, numAnswers, and correctAnswerIndex. Hint: Don’t give the entity the same name as your Question class, or Xcode will get confused. Below is a screenshot of my QuestionEntity.



3. In the Quiz Table View Controller, add a method for fetching questions from CoreData. When the app starts, all the questions stored in CoreData should be loaded and displayed

in the table view. *Do not initialize the app with any sample questions; the initial table of questions should be empty when the app is run for the first time.*

4. In the Quiz Table View Controller, add a method for adding a new question, which should result in adding the question's data to CoreData. This method should be called upon returning from the Add Question view after tapping Save.
5. In the Quiz Table View Controller, add a method for deleting a question. This method should be called when the user swipes-to-delete a question in the table. You will need some way to uniquely identify the CoreData object corresponding to the deleted question, so you can delete that object from CoreData. Below are three ways to do this.
 - a. Maintain a separate array of QuestionEntity objects corresponding to your quiz question array. Modifications to the latter result in modifications to the former along with the appropriate CoreData object inserts and deletes.
 - b. Generate and store a unique identifier with each question, both in the Question class and the QuestionEntity. Then fetch the object having this unique identifier and delete it.
 - c. Each CoreData object has a unique object ID of type NSManagedObjectID. You can store this ID in the Question class (e.g., `question.objectID = managedObject.objectID`). When you want to delete this question, you can use this ID to access the object directly using `managedObjectContext.object(with: question.objectID)` and then delete it. Note that the objectID for a new object is assigned only after the context is saved.
6. Make sure your app can add and delete questions, and that these changes persist even if the app is terminated and restarted.
7. Your app should still support all the functionality of Homework 5, including viewing questions that are selected from the table.
8. As usual, make sure the auto layout constraints are set so that the view elements are appropriately displayed with no overlap regardless of device orientation.