COMP23111 Coursework II Report

10675461 Yi Lu

Nov 28, 2021

Contents

1 Part A: Normalisation

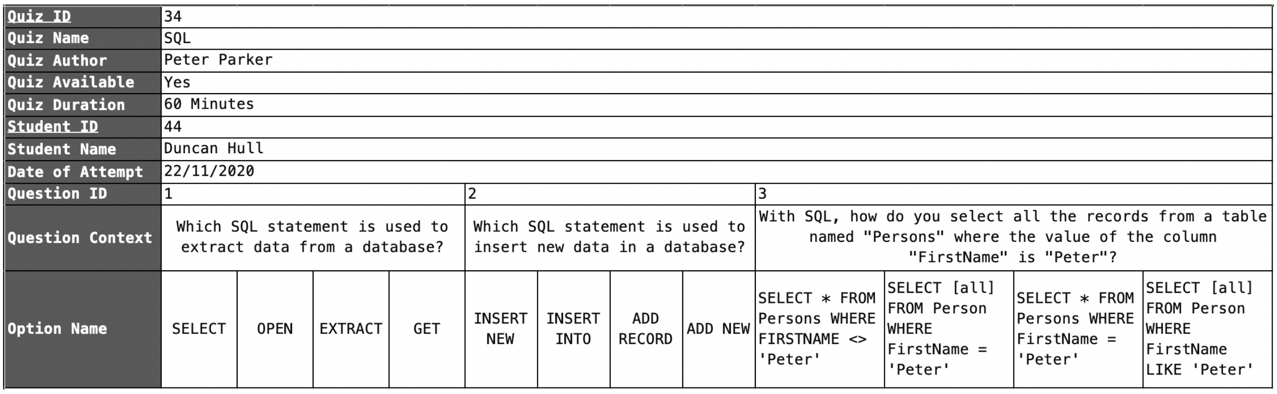
Requirement:

You are to take the information source (see next page) and normalise to 3NF to create associated relations and the attributes within them.

In this section, I transformed the given Information source into 3NF with every step presented as follows.

1.1 UNF

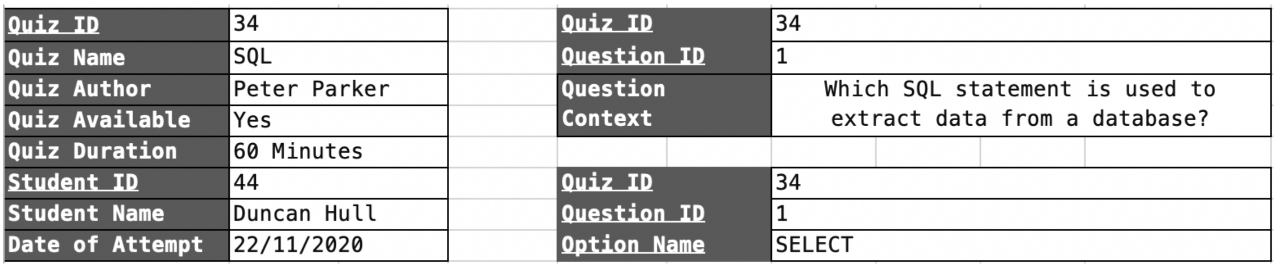
What I did for UNF: Identified all fields and sample data, identified keys and underlined them.



Quiz ID	34											
Quiz Name	SQL											
Quiz Author	Peter Parker											
Quiz Available	Yes											
Quiz Duration	60 Minutes											
Student ID	44											
Student Name	Duncan Hull											
Date of Attempt	22/11/2020											
Question ID	1				2				3			
Question Context	Which SQL statement is used to extract data from a database?				Which SQL statement is used to insert new data in a database?				With SQL, how do you select all the records from a table named "Persons" where the value of the column "FirstName" is "Peter"?			
Option Name	SELECT	OPEN	EXTRACT	GET	INSERT NEW	INSERT INTO	ADD RECORD	ADD NEW	SELECT * FROM Persons WHERE FIRSTNAME <> 'Peter'	SELECT [all] FROM Person WHERE FirstName = 'Peter'	SELECT * FROM Persons WHERE FirstName = 'Peter'	SELECT [all] FROM Person WHERE FirstName LIKE 'Peter'

1.2 1NF

What I did for 1NF: Removed repeating attributes, identified new compound keys.



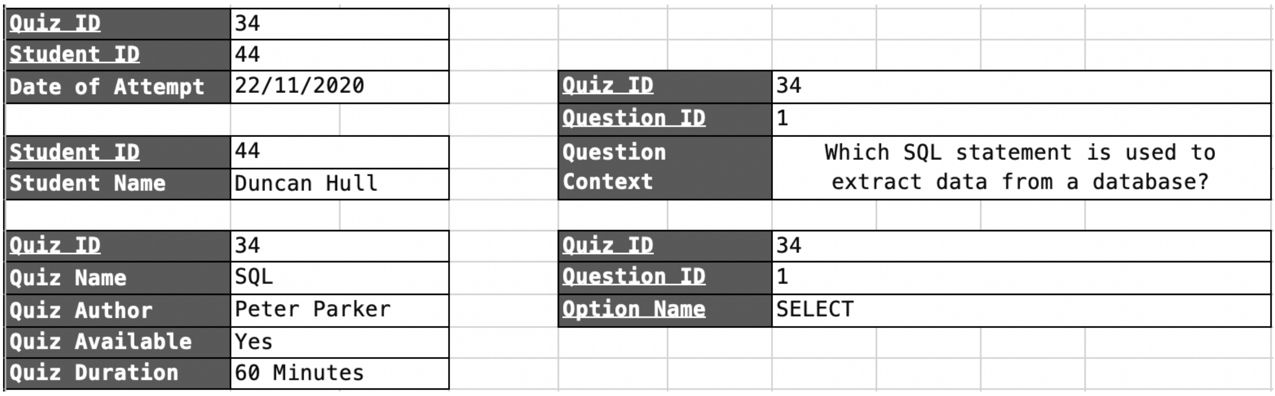
Quiz ID	34
Quiz Name	SQL
Quiz Author	Peter Parker
Quiz Available	Yes
Quiz Duration	60 Minutes
Student ID	44
Student Name	Duncan Hull
Date of Attempt	22/11/2020

Quiz ID	34
Question ID	1
Question Context	Which SQL statement is used to extract data from a database?

Quiz ID	34
Question ID	1
Option Name	SELECT

1.3 2NF

What I did for 2NF: Checked partial dependencies, removed any to new table with copy of determinant.



Quiz ID	34
Student ID	44
Date of Attempt	22/11/2020

Student ID	44
Student Name	Duncan Hull

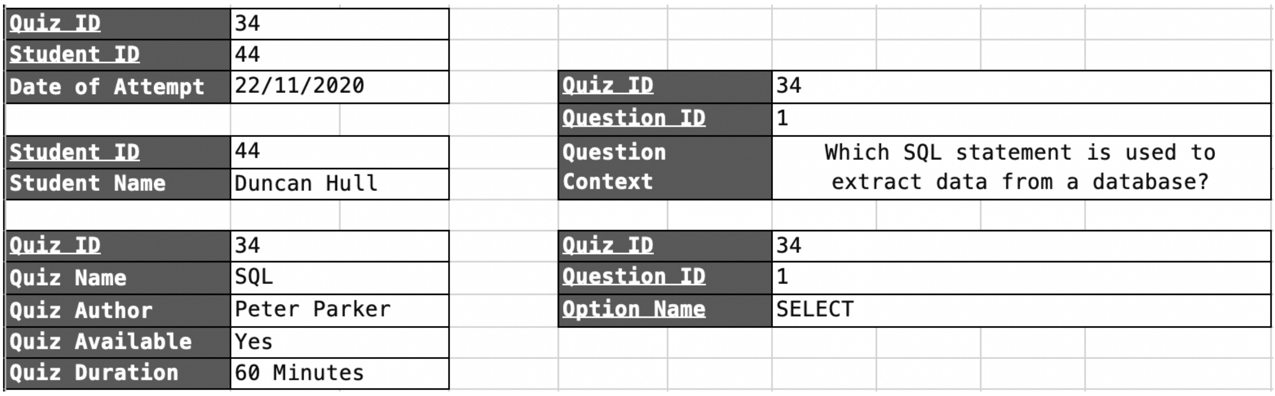
Quiz ID	34
Quiz Name	SQL
Quiz Author	Peter Parker
Quiz Available	Yes
Quiz Duration	60 Minutes

Quiz ID	34
Question ID	1
Question Context	Which SQL statement is used to extract data from a database?

Quiz ID	34
Question ID	1
Option Name	SELECT

1.4 3NF

What I did for 3NF: Checked transitive dependencies, removed any to new table with copy of determinant.



Quiz ID	34
Student ID	44
Date of Attempt	22/11/2020

Student ID	44
Student Name	Duncan Hull

Quiz ID	34
Quiz Name	SQL
Quiz Author	Peter Parker
Quiz Available	Yes
Quiz Duration	60 Minutes

Quiz ID	34
Question ID	1
Question Context	Which SQL statement is used to extract data from a database?

Quiz ID	34
Question ID	1
Option Name	SELECT

2 Part B: Relational Schema

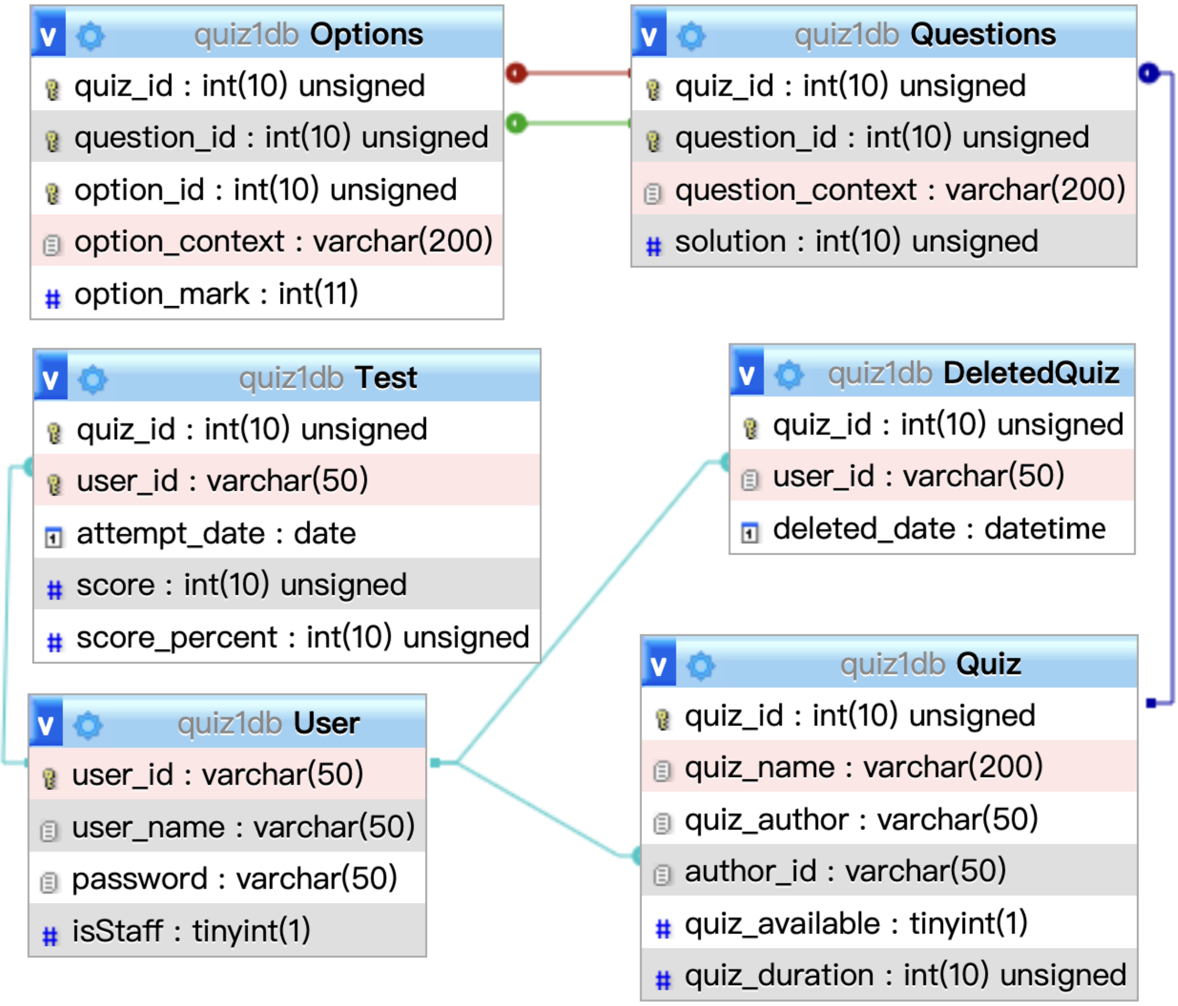
Requirement:

Create a relational Schema that clearly identifies the relations, the attributes, and any constraints.

In this section, I rendered the relational schema of my design and specified its PKs and FKs. The worked out relational schema is provided as follows:

2.1 Relational Schema

The relational schema:



quiz1db **Options**
- quiz_id : int(10) unsigned
- question_id : int(10) unsigned
- option_id : int(10) unsigned
- option_context : varchar(200)
- option_mark : int(11)

quiz1db **Questions**
- quiz_id : int(10) unsigned
- question_id : int(10) unsigned
- question_context : varchar(200)
- solution : int(10) unsigned

quiz1db **Test**
- quiz_id : int(10) unsigned
- user_id : varchar(50)
- attempt_date : date
- score : int(10) unsigned
- score_percent : int(10) unsigned

quiz1db **DeletedQuiz**
- quiz_id : int(10) unsigned
- user_id : varchar(50)
- deleted_date : datetime

quiz1db **User**
- user_id : varchar(50)
- user_name : varchar(50)
- password : varchar(50)
- isStaff : tinyint(1)

quiz1db **Quiz**
- quiz_id : int(10) unsigned
- quiz_name : varchar(200)
- quiz_author : varchar(50)
- author_id : varchar(50)
- quiz_available : tinyint(1)
- quiz_duration : int(10) unsigned

2.2 Constraints

Specification on Primary Keys

The primary keys implemented in this design is specified as follow:

1. Table ‘User”s primary key is (user_id).
2. Table ‘Quiz”s primary key is (quiz_id).

3. Table ‘DeletedQuiz”s primary key is (user_id).
4. Table ‘Questions”s primary key is unified: (quiz_id, question_id).
5. Table ‘Option”s primary key is unified: (quiz_id, question_id, option_id).
6. Table ‘Test”s primary key is unified: (quiz_id, user_id).

Specification on Foreign Keys

The foreign keys implemented in this design is specified as follow:

1. foreign key ‘autho_id’ in table ‘Quiz’ references ‘user_id’ in table ‘User’.
2. foreign key ‘user_id’ in table ‘DeletedQuiz’ references ‘user_id’ in table ‘User’.
3. foreign key ‘quiz_id’ in table ‘Questions’ references ‘quiz_id’ in table ‘Quiz’.
4. foreign key ‘quiz_id, question_id’ in table ‘Options’ references ‘quiz_id, question_id’ in table ‘Questions’.
5. foreign key ‘user_id’ in table ‘Test’ references ‘user_id’ in table ‘User’.

2.3 Identification of Changes/Assumptions

Changes on original data

1. We consider each option has one Option ID, which is the order it presents in certain questions.
2. We introduced ‘User’ entity to eliminate the fundamental difference between ‘Students’ and ‘Staff (Quiz Author)’.
3. To store quiz score, we introduced ‘Test’ entity to store the information on each quiz attempt, which includes score, score stored in percentage form, attempt date and so on.
4. To store the log generated by the Trigger, we introduced ‘DeletedQuiz’ entity to audit the deleting process.

Assumptions

1. We decided to interpretate ‘Quiz Duration’ to be the ‘Time Constraint’ on certain quizzes.
2. We assume each question can have up to infinite many options, and every question may have more than one correct choice.
3. We assume each user (student or staff) is to be determined by a unique value: Nickname. In reality, it can be any string which is unique in the database.
4. In table ‘User’, we assume that the flag to control the user’s type is ‘isStaff’.
5. We limits that every staff can only edit/delete quizzes which are created by hiis(her) own.

3 Part C: Implementation

Requirement:

Implement your design being mindful of the data types for your fields and any referential actions for foreign keys to update or delete data from related tables.

In this section, I implemented my design in Part B with the help of 'SQL'.

3.1 SQL Code

The following are the SQL statements written for implementing the table design.
Note that SQL statements for 'Stored Procedure' and 'Trigger' are attached at 'Part E'.

Listing 1: **Main Database Construction**

```
-- ------------------------------
-- TABLE 'User'
-- ------------------------------
CREATE TABLE User (
    user_id VARCHAR(50) NOT NULL,
    user_name VARCHAR(50) NOT NULL,
    password         VARCHAR(50) NOT NULL,
    isStaff BOOLEAN NOT NULL,

    CONSTRAINT User_pk
        PRIMARY KEY (user_id)
);



-- ------------------------------
-- TABLE 'QuizInfo'
-- ------------------------------
CREATE TABLE Quiz (
    quiz_id INT UNSIGNED NOT NULL AUTO_INCREMENT,
    quiz_name VARCHAR(200) NOT NULL,
    quiz_author VARCHAR(50) NOT NULL,
    author_id VARCHAR(50) NOT NULL,
    quiz_available BOOLEAN NOT NULL,
    quiz_duration INT UNSIGNED NOT NULL,

    CONSTRAINT QuizInfo_pk
        PRIMARY KEY (quiz_id),
    CONSTRAINT  author_id_User_user_id
        FOREIGN KEY (author_id) REFERENCES User (user_id)
);



-- ------------------------------
-- TABLE 'DeletedQuiz'
-- ------------------------------
CREATE TABLE DeletedQuiz (
    quiz_id INT UNSIGNED NOT NULL,
    user_id VARCHAR(50) NOT NULL,
    deleted_date DATETIME NOT NULL,

    CONSTRAINT DeletedQuiz_pk
        PRIMARY KEY (quiz_id),
    CONSTRAINT user_id_DeletedQuiz_User_user_id
        FOREIGN KEY (user_id) REFERENCES User (user_id)
);



-- ------------------------------
-- TABLE 'Questions'
-- ------------------------------
CREATE TABLE Questions (
    quiz_id INT UNSIGNED NOT NULL,
    question_id INT UNSIGNED NOT NULL ,

    question_context VARCHAR(200) NOT NULL,

    # solution == some options in this question
    solution INT UNSIGNED NOT NULL,

    CONSTRAINT Questions_pk
        PRIMARY KEY (quiz_id, question_id),
    CONSTRAINT quiz_id_Questions_Quiz_quiz_id
        FOREIGN KEY (quiz_id) REFERENCES Quiz (quiz_id) ON DELETE CASCADE
);



-- ------------------------------
-- TABLE 'Options'
-- ------------------------------
CREATE TABLE Options (
    quiz_id INT UNSIGNED NOT NULL,
    question_id INT UNSIGNED NOT NULL,
    option_id INT UNSIGNED NOT NULL,
    option_context VARCHAR(200) NOT NULL,
    option_mark int NOT NULL,

    CONSTRAINT Options_pk
        PRIMARY KEY (quiz_id, question_id, option_id),
    CONSTRAINT quiz_id_Options_Question_quiz_id
```

```
        FOREIGN KEY (quiz_id, question_id) REFERENCES Questions (quiz_id, question_id) ON DELETE CASCADE
);



-- ------------------------------
-- TABLE 'TestInfo'
-- ------------------------------
CREATE TABLE Test (
    quiz_id INT UNSIGNED NOT NULL,
    user_id VARCHAR(50) NOT NULL ,
    attempt_date DATE NOT NULL,
    score INT UNSIGNED NOT NULL,
    score_percent INT UNSIGNED NOT NULL,

    CONSTRAINT TestInfo_pk
        PRIMARY KEY (quiz_id, user_id),
    CONSTRAINT user_id_Test_User_user_id
        FOREIGN KEY (user_id) REFERENCES User (user_id)
);
```

Note that PK, FK and other constraints are clearly stated in the code.

4 Part D: The Application

Requirement:
Using PHP and MySQL create an application and front-end for your database.

I developed a simple web application as the interface of my database. It's developed using HTML5, Javascript and jQuery on the front-end, using MySQL for hosting the Database, and chose PHP as the backbone of the back-end.

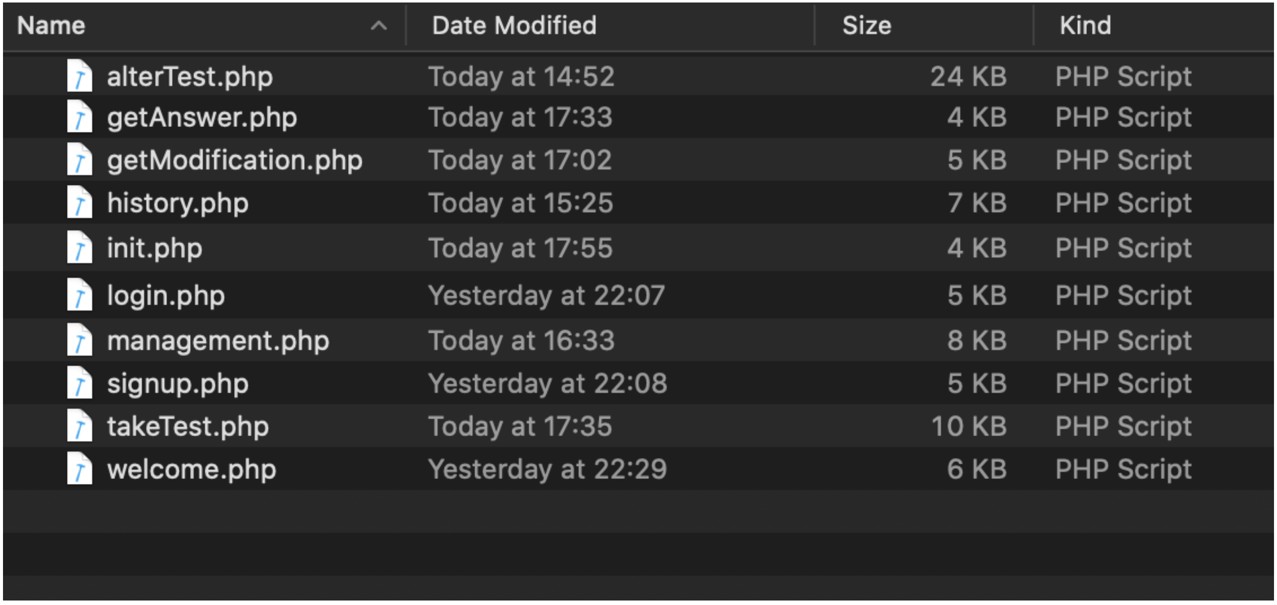
Here are the explainations and guidance of how to run and use this web application.

4.1 Structure

The application can be splitted into 3 parts:

1. An initilization helper, 'init.php', which is responsible for building up the database.
2. Several pure-php scripts, 'getAnswer.php' and 'getModification.php', responsible for taking the incoming data from the frontend, write them back into the database after processing.
3. Other front-end interfaces, which are all other '.php' files, responsible for front-end web rendering and user-interaction.

The specific files are shown as below:



Name	Date Modified	Size	Kind
alterTest.php	Today at 14:52	24 KB	PHP Script
getAnswer.php	Today at 17:33	4 KB	PHP Script
getModification.php	Today at 17:02	5 KB	PHP Script
history.php	Today at 15:25	7 KB	PHP Script
init.php	Today at 17:55	4 KB	PHP Script
login.php	Yesterday at 22:07	5 KB	PHP Script
management.php	Today at 16:33	8 KB	PHP Script
signup.php	Yesterday at 22:08	5 KB	PHP Script
takeTest.php	Today at 17:35	10 KB	PHP Script
welcome.php	Yesterday at 22:29	6 KB	PHP Script

4.2 Run the application

To be able to run the application, you need 'MySQL' and 'PHP' server running on your machine. by default, the application accesses the database using account 'root' and password 'root' at 'localhost', so make sure such account exists in your SQL server.

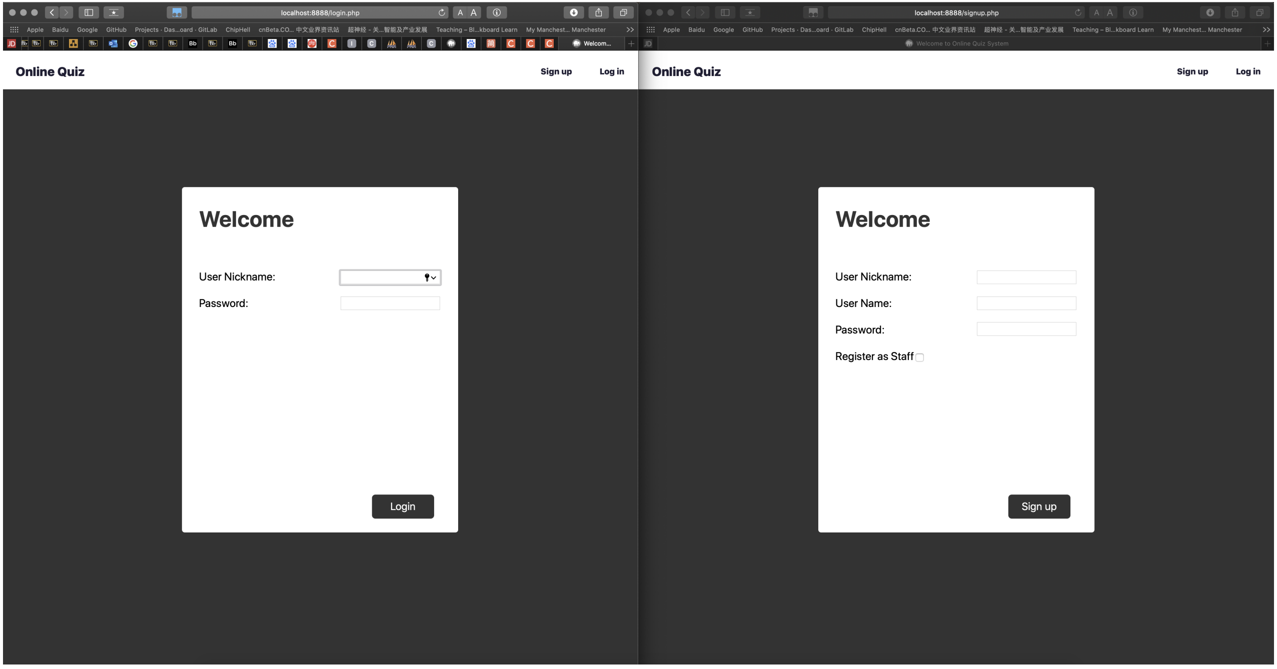
You can also choose to run the version using account 'root' and no password at directory './Online Quiz(noPassword)'. However, it's still suggested that you change your password and use the regular version.

To run the application, you need to copy all these files shown above into the home directory of your PHP server. After that, either run 'init.php' ONCE or import the backed-up database file 'quizdb.sql'. 'init.php' will communicate to the SQL server and construct the database 'quizdb' used for the application.

After that, you can access either ‘login.php’ or ‘signup.php’, which are responsible for user log in or account register respectively.

4.3 Sign up and Log in

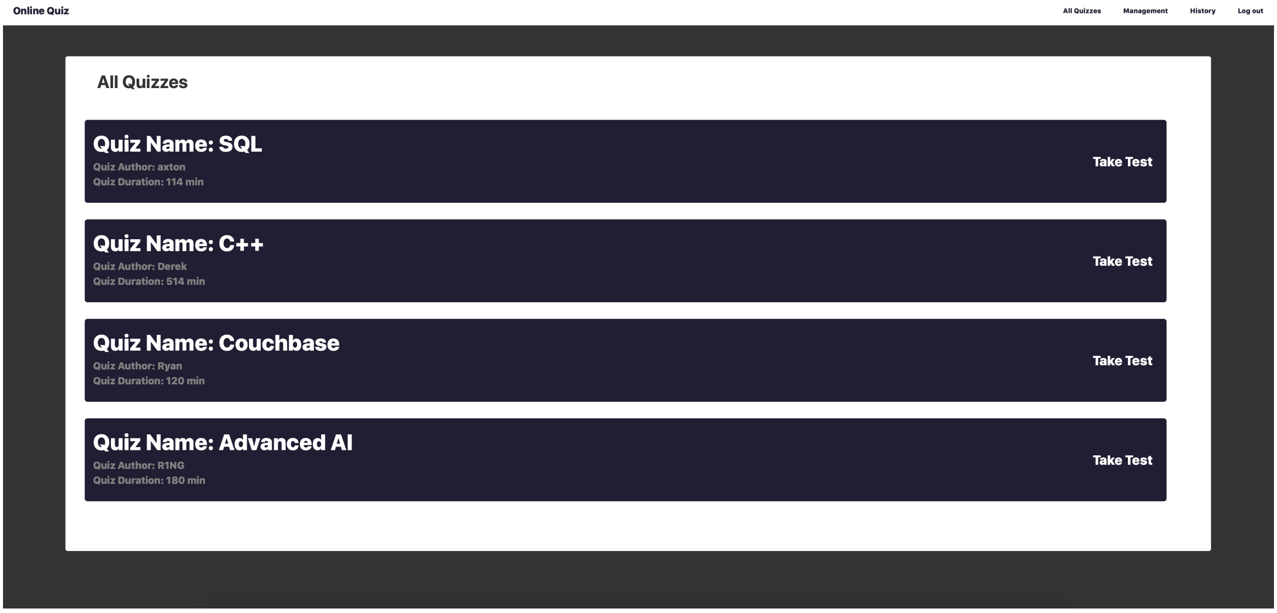
Then you will need to register. head to ‘signup.php’, you need to provide a ‘nickname’ as the unique user ID (its uniqueness will be checked by the server) , your name and the password. After the registration succeed, you’ll be redirected to the login page. Enter your credentials and click ‘Log in’, you will be Then redirected to the main page: ‘welcome.php’.



4.4 Main Page

You can check every available quizzes in the main page. Each quiz is displayed with name, author and time limit, so there’s no way that you will mistaken two tests.

After scroll down and find what you need to take, you can click on the ‘Take Test’ text to begin the quiz.



Online Quiz

All Quizzes Management History Log out

All Quizzes

Quiz Name: SQL
Quiz Author: axton
Quiz Duration: 114 min
Take Test

Quiz Name: C++
Quiz Author: Derek
Quiz Duration: 514 min
Take Test

Quiz Name: Couchbase
Quiz Author: Ryan
Quiz Duration: 120 min
Take Test

Quiz Name: Advanced AI
Quiz Author: R1NG
Quiz Duration: 180 min
Take Test

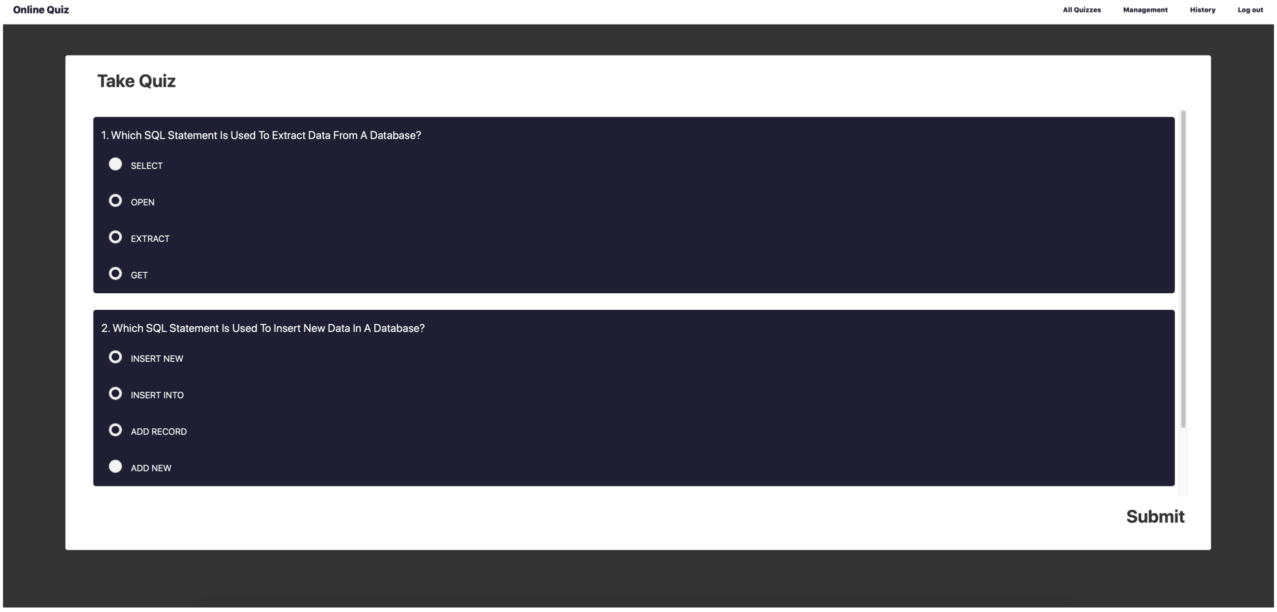
The navigation header provides you with four handy shortcuts. You can click on them to either navigate yourself around the application, or logging out in one click.

Note that 'Management' shortcut is disabled for students (users who are not registered as staff).

4.5 Take Quiz

Scroll up and down to navigate, click on the checkboxes to make your choice. After finished all questions, just click the 'Submit' button to hand in your solution.

By the time you are being redirected to the Test History page, the server calculates the score and store it into the database.



Online Quiz

All Quizzes Management History Log out

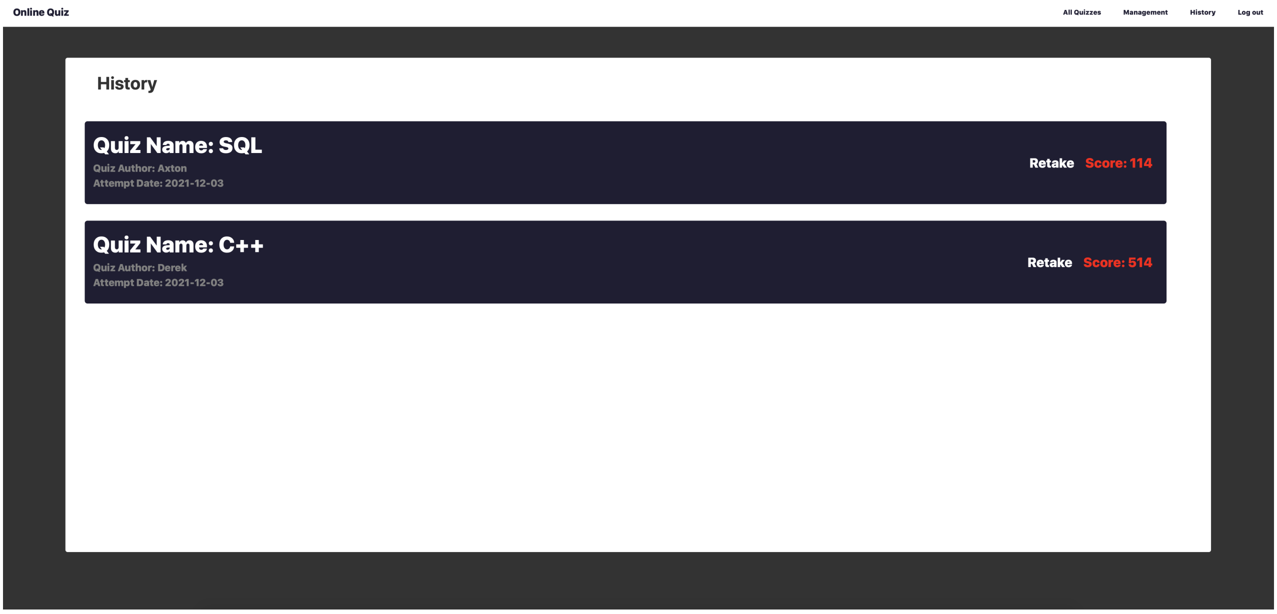
Take Quiz

1. Which SQL Statement Is Used To Extract Data From A Database?
- ● SELECT
- ○ OPEN
- ○ EXTRACT
- ○ GET

2. Which SQL Statement Is Used To Insert New Data In A Database?
- ○ INSERT NEW
- ○ INSERT INTO
- ○ ADD RECORD
- ● ADD NEW

Submit

4.6 Browse Test History



History

Quiz Name: SQL
Quiz Author: Axton
Attempt Date: 2021-12-03
Retake Score: 114

Quiz Name: C++
Quiz Author: Derek
Attempt Date: 2021-12-03
Retake Score: 514

In the History page, you can browse the information of all your previously taken tests with your score, attempt date clearly indicates.

You can also choose to retake anyone of them by clicking on the 'Retake' text.

Note that the score displayed on this page is NOT transformed into the percentage format, but directly display how many points you got from the test.

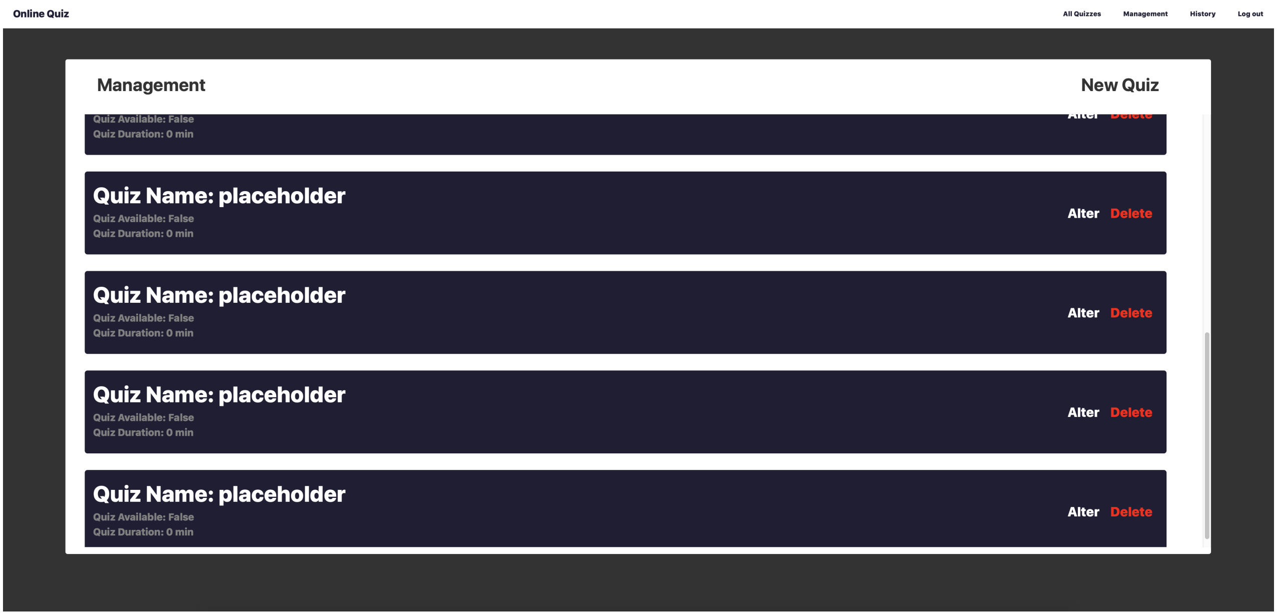
It's the quiz author who decides how many points should be considered as full mark.

4.7 Manage quizzes

This page and the following one is only opened to Staff users. In this page, you can access any quizzes created by you or create a brand new quiz by clicking on text 'New Quiz' at the top-right corner.

Any quiz which is newly generated will have a default name, time limit and availability (automatically set to false, of course).

You can edit any test by clicking on 'Alter' text, or delete any of them in a single click on 'Delete'.



Quiz Available: False
Quiz Duration: 0 min
Alter Delete

Quiz Name: placeholder
Quiz Available: False
Quiz Duration: 0 min
Alter Delete

Quiz Name: placeholder
Quiz Available: False
Quiz Duration: 0 min
Alter Delete

Quiz Name: placeholder
Quiz Available: False
Quiz Duration: 0 min
Alter Delete

Quiz Name: placeholder
Quiz Available: False
Quiz Duration: 0 min
Alter Delete

4.8 Editing Quizzes

MCQ mechanics

This interface resides within the quiz editing interface. It's probably the most complex one among all.

In this page, every question of the quiz and their corresponding options are displayed to you. You can clearly identify the ID of questions and option IDs.

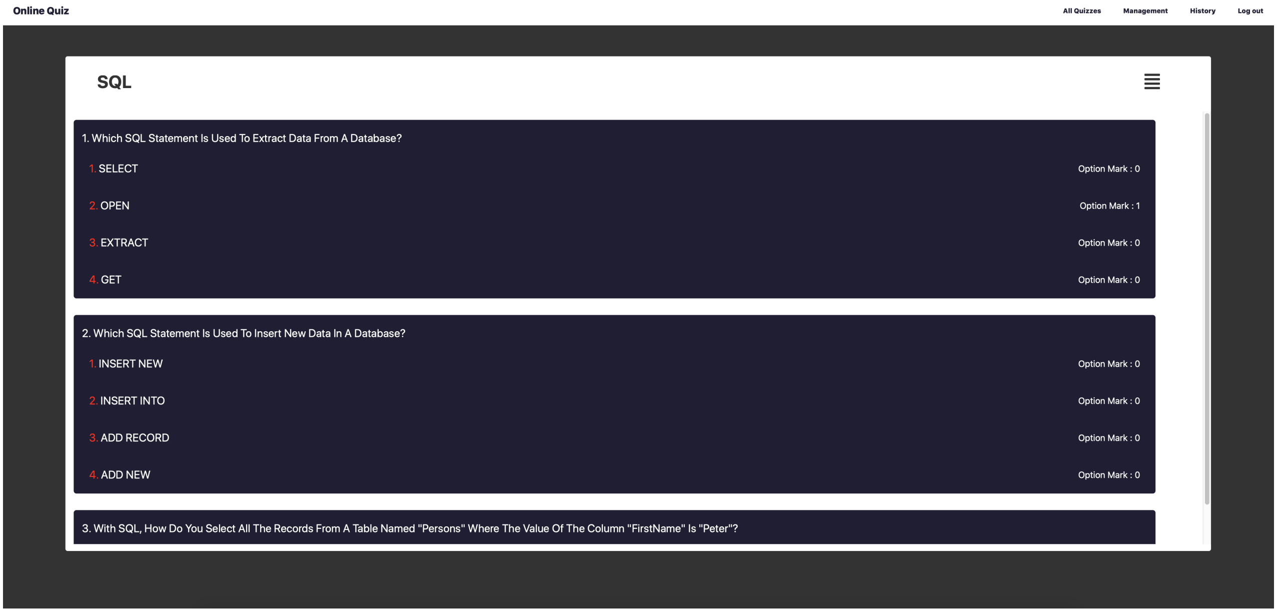
You can also clearly see the mark allocated for each options.

In this online quiz system, every question is designed to be multiple-choiced. Student can select many selections, but quiz author can modify the number of correct choices by controlling the mark given for each question.

It works like the following: options with non-zero marks are considered as (partially) correct, while zero-mark options are flagged as wrong answers.

If the student choose any 'wrong' answers, the student's score on this question will be counted as ZERO, otherwise it'll be the sum of each (partially) correct answer's allocated marks.

Only by choosing all 'correct' answers, the student can get the full mark on this question.



SQL

1. Which SQL Statement Is Used To Extract Data From A Database?

1. SELECT — Option Mark : 0
2. OPEN — Option Mark : 1
3. EXTRACT — Option Mark : 0
4. GET — Option Mark : 0

2. Which SQL Statement Is Used To Insert New Data In A Database?

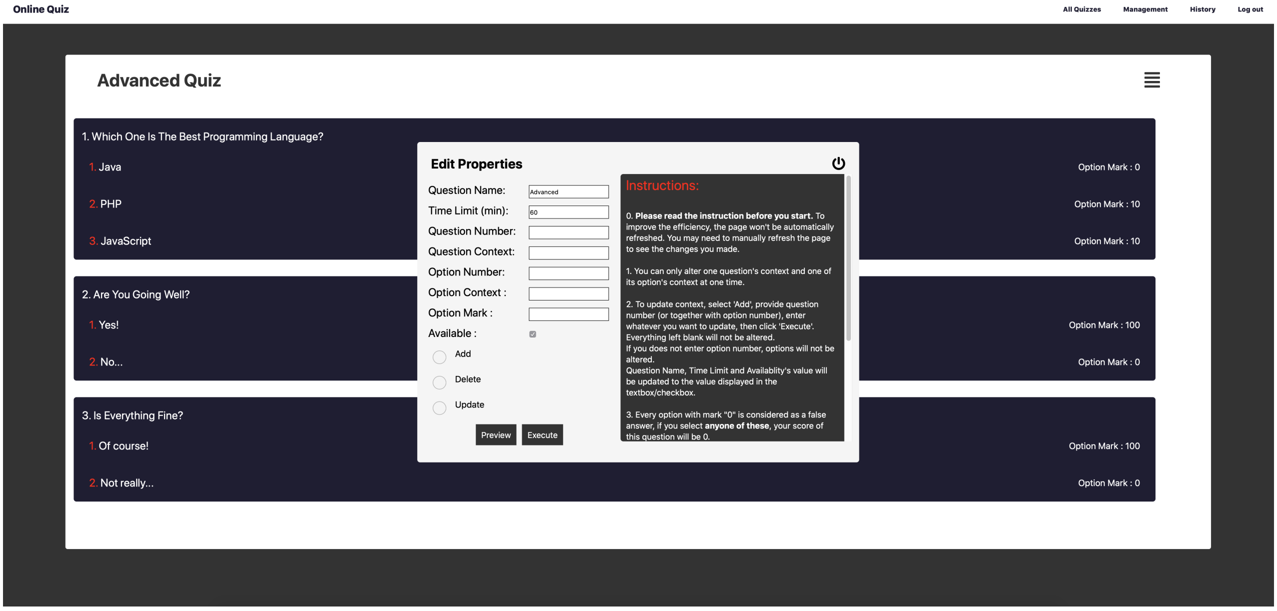
1. INSERT NEW — Option Mark : 0
2. INSERT INTO — Option Mark : 0
3. ADD RECORD — Option Mark : 0
4. ADD NEW — Option Mark : 0

3. With SQL, How Do You Select All The Records From A Table Named "Persons" Where The Value Of The Column "FirstName" Is "Peter"?

Unify Editing

You can access the Unify Editing tool by clicking on the menu icon at the top-right side.

By using this tool, you can edit/create/delete any quiz elements in one place. Select one editing mode among three, type in the information needed, click ‘Preview’ button to foreseen what’s gonna happen, hit ‘Execute’ button to proceed. The detailed instruction on how to fill in the information to be inserted is displayed at the right-hand side. After you finished, click the ‘Power-Off’ button to close it. The changes needs a page refresh to be seen.



Advanced Quiz

1. Which One Is The Best Programming Language?

1. Java — Option Mark : 0
2. PHP — Option Mark : 10
3. JavaScript — Option Mark : 10

2. Are You Going Well?

1. Yes! — Option Mark : 100
2. No... — Option Mark : 0

3. Is Everything Fine?

1. Of course! — Option Mark : 100
2. Not really... — Option Mark : 0

Edit Properties

Question Name: Advanced

Time Limit (min): 60

Question Number:

Question Context:

Option Number:

Option Context :

Option Mark :

Available : ☑

○ Add

○ Delete

○ Update

Instructions:

0. **Please read the instruction before you start.** To improve the efficiency, the page won't be automatically refreshed. You may need to manually refresh the page to see the changes you made.

1. You can only alter one question's context and one of its option's context at one time.

2. To update context, select 'Add', provide question number (or together with option number), enter whatever you want to update, then click 'Execute'.
Everything left blank will not be altered.
If you does not enter option number, options will not be altered.
Question Name, Time Limit and Availablity's value will be updated to the value displayed in the textbox/checkbox.

3. Every option with mark "0" is considered as a false answer, if you select **anyone of these**, your score of this question will be 0.

Due to technological constraints and the time limit, it’s impossible for me to implement the WYSIWYG editing tool. Therefore, this ‘Unify Editing’ tool comes with several constraints:

1. When adding new question/option, you cannot modify the question/option context. The modification must be executed through the ‘Edit’ mode.
2. Question/Options are only allowed to be added at the end of the test/option queue.

3. Quiz metadata: ‘Quiz Name’, ‘Quiz Duration’ and ‘Availability’ can only be renewed in ‘Update’ mode.
4. If the question is empty, i.e. it has no valid options to choose, the test result will be considered as invalid and won’t be stored.

5 Part E: Stored Procedures and Triggers

Requirements:

1. Create a stored procedure that displays the student names and their scores for the quizzes where they achieved less than 40 percent.
2. Create a trigger that will log the staff id, the quiz id and the current date and time, when a staff user deletes a quiz.

In this section, I created a stored procedure which is capable of filtering out students who failed on some tests. it’s also capable of displaying which test they failed by displaying the ‘quiz_id’.

I also created a trigger which automatically audits ‘Which staff deleted which Quiz at which time’ and record such data into the table ‘DeletedQuiz’.

Both the stored procedure and trigger are presentd in the database dump.

5.1 Stored Procedure

The Stored Procedure required is:

Listing 2: **StoredProcedure Implementation**

```
DELIMITER //
    CREATE PROCEDURE GetBelowForty()
    BEGIN
       SELECT user_id, quiz_id, score FROM Test WHERE score_percent < 40;
    END //
DELIMITER ;
```

Note that attribute ‘score_percent’ holds an integer ranges from 0 to 100, which is converted from the score percentage.

The score percentage is calculated along with ‘score’ when the user hands in the test solution to the server.

5.2 Trigger

The trigger required is:

Listing 3: **Trigger Implementation**

```
CREATE TRIGGER `auditQuizDelete`
AFTER DELETE
    ON `Quiz`
    FOR EACH ROW
        insert into DeletedQuiz
            set quiz_id=OLD.quiz_id,
                user_id=OLD.author_id,
                deleted_date=NOW()
```