Ruotong Ding

23111-Cwk2-S-Advanced Databases, COMP23111,27/11/2021

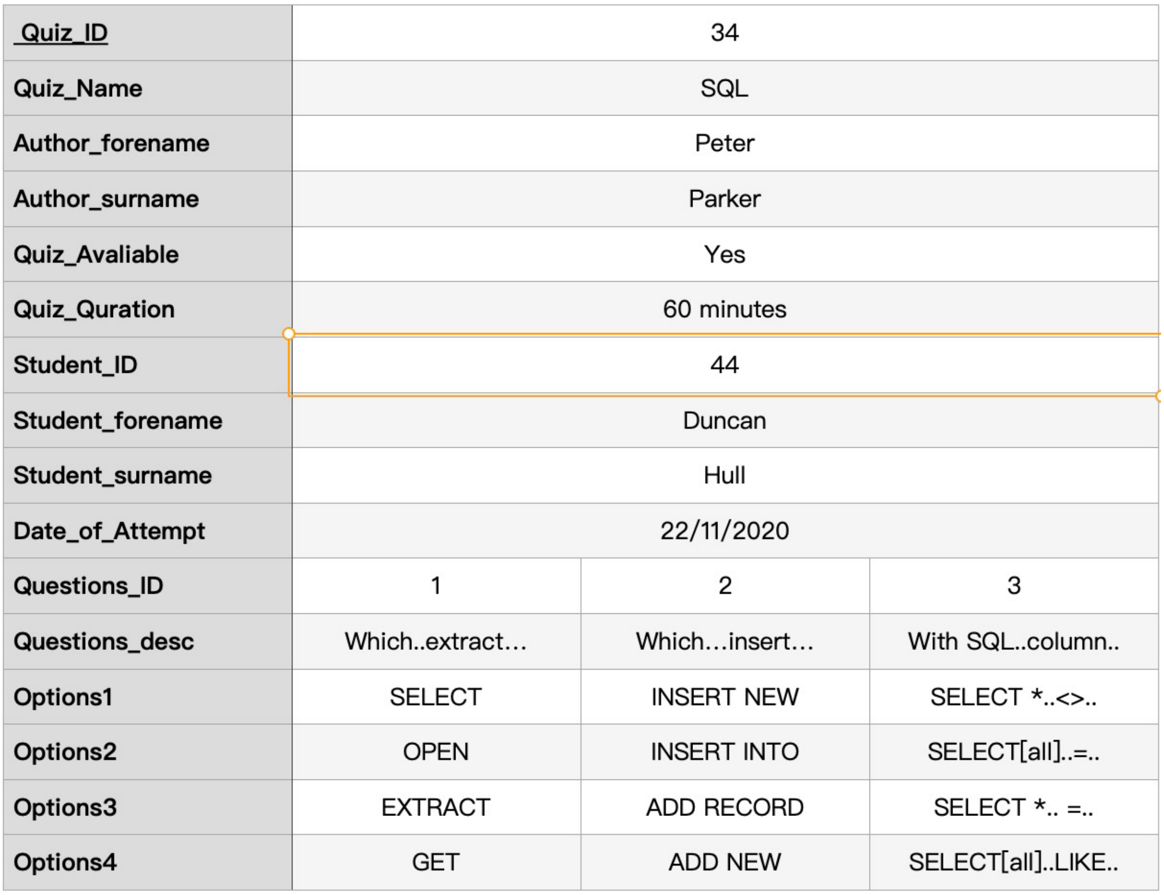
Table of contents

Part A: Normalization

Takes the information source and normalize to 3NF to create associated relations and the attributes within them

UNF:

I Identified all the field of the information source, also identified the repeating values and represent then together, below is my UNF table:



Quiz_ID	34		
Quiz_Name	SQL		
Author_forename	Peter		
Author_surname	Parker		
Quiz_Avaliable	Yes		
Quiz_Quration	60 minutes		
Student_ID	44		
Student_forename	Duncan		
Student_surname	Hull		
Date_of_Attempt	22/11/2020		
Questions_ID	1	2	3
Questions_desc	Which..extract…	Which…insert…	With SQL..column..
Options1	SELECT	INSERT NEW	SELECT *..<>..
Options2	OPEN	INSERT INTO	SELECT[all]..=..
Options3	EXTRACT	ADD RECORD	SELECT *.. =..
Options4	GET	ADD NEW	SELECT[all]..LIKE..

1NF:

I removed the repeating attributes by inserting them into a new table, and identified the new compound key. Below are my 1NF relations:

Quizinfo(Quiz_ID, Quiz_name, Author_forename, Author_surname, Quiz_available, Quiz_duration, Student_ID, Student_forename, Student_surname, Date_of_Attempt)

QuestionInfo(Quiz_ID, Question_ID, Question_desc, Options1, Options2, Options3, Options4)

2NF:
I checked the partial dependencies of my 1NF and remove them to new table. Below are my 2NF relations

QuizInfo(Quiz_ID, Quiz_Name, Author_forename, Author_surname, Quiz_available, Quiz_Duration)

TakenInfo(Quiz_ID, Student_ID, Attempt_date)

StudentInfo(Student_ID, Student_forename, Student_surname)

QuestionInfo(Quiz_ID, Question_ID, Question_desc, Options1, Options2, Options3, Options4)

3NF:
I checked the transitive dependencies and remove them to new table with copy of determinant. I found that Quiz_name has partial dependencies, but while I remove it to a new table it is a 1 to 1 relation so I merged back. Question_desc also has transitive dependencies so I remove it to a new table.

QuizInfo(Quiz_ID, Quiz_Name, Author_forename, Author_surname, Quiz_available, Quiz_Duration)

TakenInfo(Quiz_ID, Student_ID, Attempt_date)

StudentInfo(Student_ID, Student_forename, Student_surname

OptionInfo(Quiz_ID, Question_ID, Question_desc, Options1, Options2, Options3, Options4)

QuestionInfo(Quiz_ID, Question_ID, Question_desc)

Part B: Relational Schema

In this section, I designed a relational schema base on 3NF in part A, but made some changes and added the constraints

Staff(staff_id, username, password, forename, surname)

Not null constraint(username, password, forename, surname)

Auto increment(staff_id)

Unique constraint(username)

Quiz(quiz_id, quiz_name, quiz_available, quiz_duration, fullmark, author_id)
FK(author_id) --> Staff(staff_id)

Not null constraint(quiz_name, quiz_available, quiz_duration, author_id)

Auto increment constraint(quiz_id)

Questions(question_id, quiz_id, queston_desc, question_mark)

FK quiz_id --> Quiz(quiz_id)

Not null constraint(quiz_id, question_id, desc)

Options(option_id, quiz_id, question_id, option_desc, correct_ans)

FK quiz_id --> Quiz(quiz_id)

FK question_id --> Questions(question_id)

Not null constraint(option_id, quiz_id, question_id, option_desc,correct_ans)

Student(student_id, username, forename, surname, password)

Not null constraint(username, forename, surname, password)

Auto increment constraint(student_id)

Unique constraint(username)

Taken(quiz_id, student_username, attempt_date, fullmark)

FK quiz_id --> quiz(Quiz_id)

FK student_username --> Student(student_username)

Not null constraint(quiz_id, student_username, attempt_date, mark)

Part C: Implementation

In this section I implemented my design with specifying datatypes and any constraints

```
create database quizsystem;
```

```
use quizsystem;
```

```
create table if not exists Staff(
  staff_id int primary key not null auto_increment,
  username varchar(30) not null unique,
  forename varchar(30) not null,
  surname varchar(30) not null,
  password varchar(30) not null
);

create table if not exists Quiz(
  quiz_id int primary key auto_increment,
  quiz_name varchar(50) not null unique,
  quiz_available varchar(10) not null,
  quiz_duration varchar(20) not null,
  fullmark int,
  author_id int not null,
  foreign key (author_id) references Staff(staff_id) ON DELETE RESTRICT
);

create table if not exists Questions(
  question_id int not null,
  quiz_id int not null,
  question_desc varchar(70) not null,
  question_mark int,
  primary key(question_id,quiz_id),
  foreign key (quiz_id) references Quiz(quiz_id) ON DELETE RESTRICT
);

create table if not exists Options(
  option_id int not null,
  quiz_id int not null,
  question_id int not null,
  option_desc varchar(70) not null,
  correct_ans varchar(10) not null,
  primary key(option_id, quiz_id,question_id),
  foreign key (quiz_id) references Quiz(quiz_id) ON DELETE RESTRICT,
  foreign key (question_id) references Questions(question_id) ON DELETE
RESTRICT
);

create table if not exists Student(
  student_id int primary key auto_increment,
  username varchar(30) not null unique,
  forename varchar(30) not null,
  surname varchar(30) not null,
  password varchar(30) not null
);

create table if not exists Taken(
  quiz_id int not null,
  student_username varchar(30) not null,
  attempt_date date not null,
  fullmark int,
  primary key (quiz_id, student_username),
  foreign key (quiz_id) references Quiz(quiz_id) ON DELETE RESTRICT,
  foreign key (student_username) references Student(username) ON DELETE
RESTRICT
);
```

Part D: The Application

NOTE: I set my database confidentials as “$pdo = new pdo('mysql:host=localhost;dbname=quizsystem', 'root', 'drt789mju');”
Please update my confidentials with yours in each PHP file, or you can change your database password same as mine temporarily.
I also used some header like “header('location:http://127.0.0.1:99/cw2/quizstaff.php');”
Please change the “127.0.0.1.99” corresponded to your configuration.
I am sorry for the inconveinience.

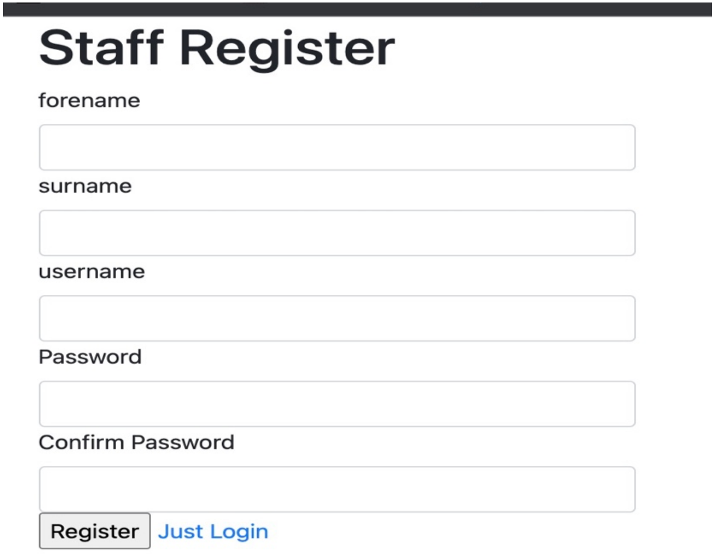
In this section I used PHP and MySQL created an application and a brief front-end for my database.

Here is the guide on how to use the database

.

First you should decide you want to use staff or student to use the application, I suggest use staff first as the staff should create quiz then student can do it.

To register as a staff, go to staffregister.php the(I did not include interface and localhost in this address as I think your details of interface and localhost may differ than mine). As the figure below shows:



Staff Register

forename

surname

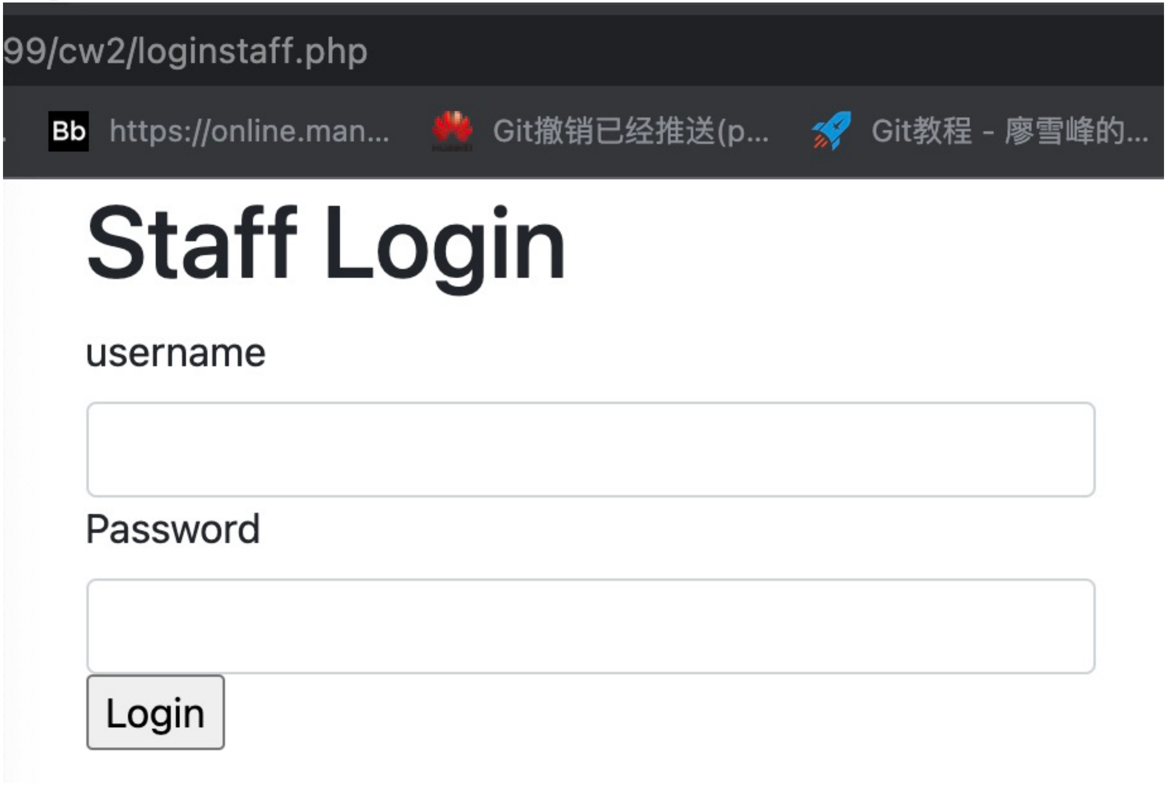
username

Password

Confirm Password

If you do not have an account, then fill in the form of your details. If you already have an account, click the 'Just Login' on the right of the register button. After you fill in the form your details will be added to the database. In this php I create the database and a table to store the staff's details, have a check after you register.

After you register, the page will direct you to the staff log in page(loginstaff.php), as the picture below shows:



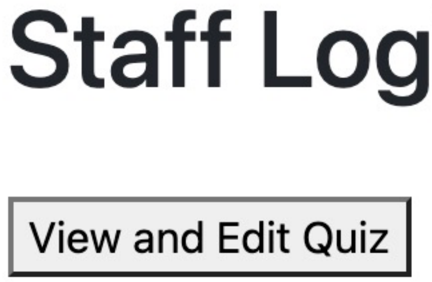
Staff Login

username

Password

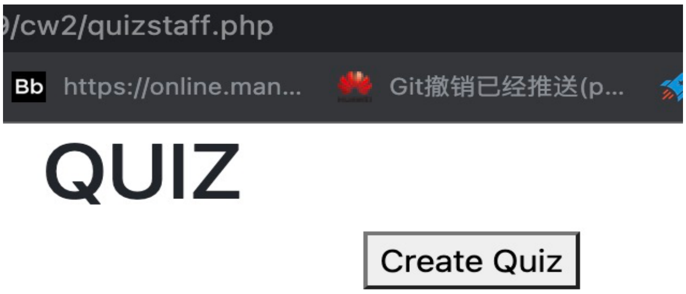
Give the username and password you created before to login, if you entered the wrong message which does not match the record in database you will get a alert to warn you.

Then we are ready to go to the quiz pages for staff, click the Login button,



Staff Login

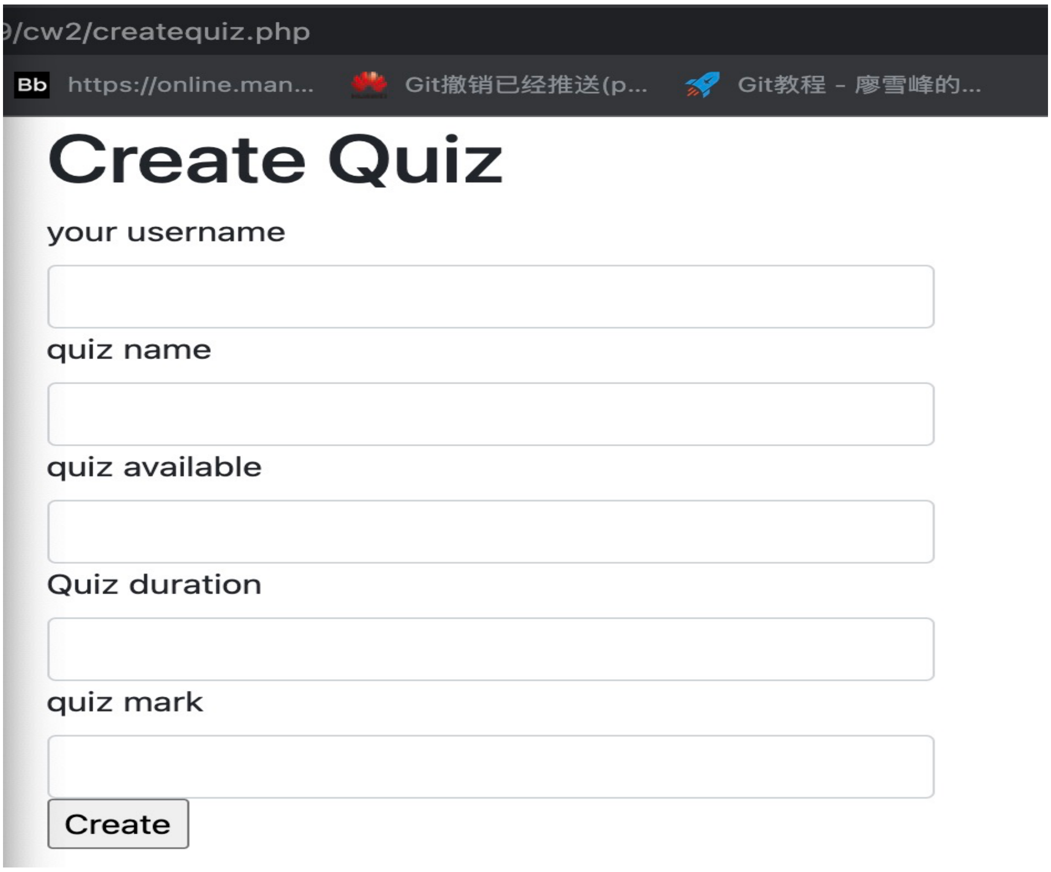
When you see this ‘view and edit’ button it means your details is correct, click the ‘view and edit’ button, which will direct you to the quiz pages for staff(quizstaff.php) which be like below:



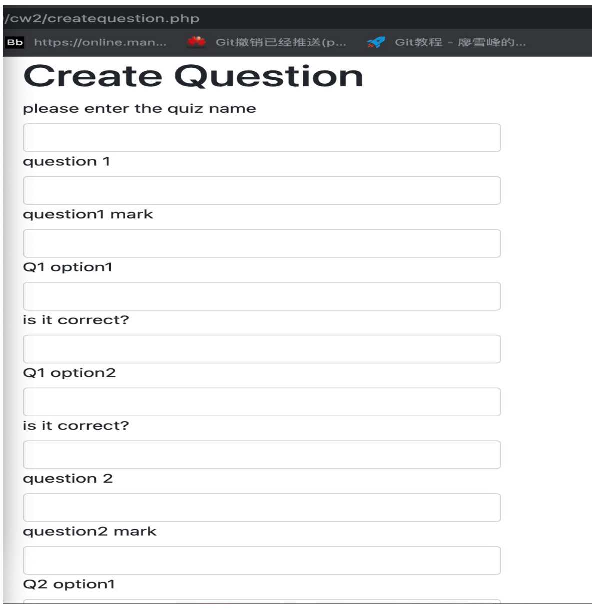
)/cw2/quizstaff.php

QUIZ

As we have not created a quiz yet, this page just shows a create button, click the button, then you will see the create page(createquiz.php) as below picture:

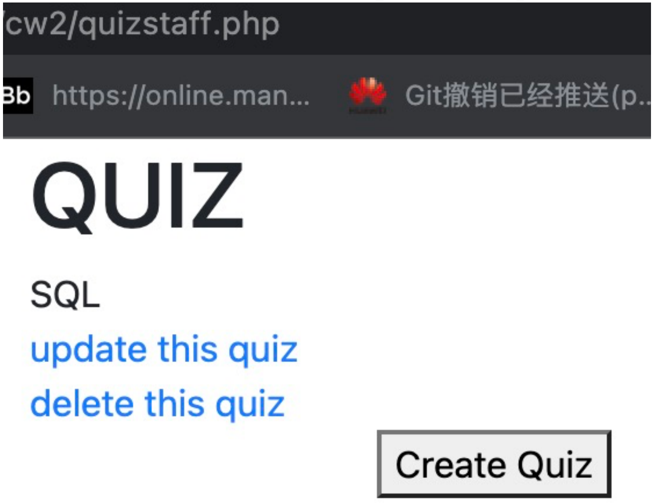


Please enter your username and quiz information here and the click the create button. Then you can check the database and will find a table quiz is created and your quiz Information has already stored in it.(if your enterd is not a staff username in database, It will alert “you are not allowed to created a quiz)
Then you will be prompt to create questions and options page(createquestion.php), which is shown as below:



First you should enter the quiz name which is exactly same you entered before, then you can create your questions and options, here I gave you three questions to create

and which question has two options, below the 'is it correct' of each option, enter "yes" if it is and "no" if it is not. Then enter the question mark for each question. One thing denote the total mark of each question mark should equal to the mark of quiz you entered before. Then you check the database will see a table questions and another table options have been created and your records will be in there.
Then you will be direct back to the main quiz page (quizstaff.php), and now you can see your quiz just created and two button "update this quiz" and "delete this quiz".



/cw2/quizstaff.php

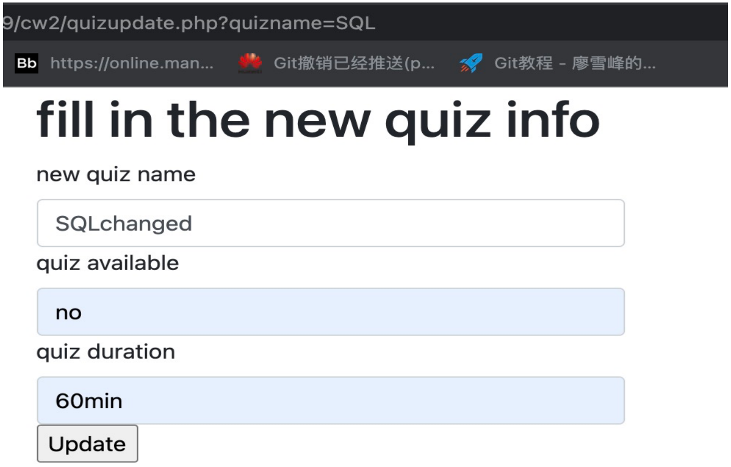
QUIZ

SQL

update this quiz

delete this quiz

If you click to update this quiz
It will direct you to the update page(quizupdate.php) which is shown as below:



9/cw2/quizupdate.php?quizname=SQL

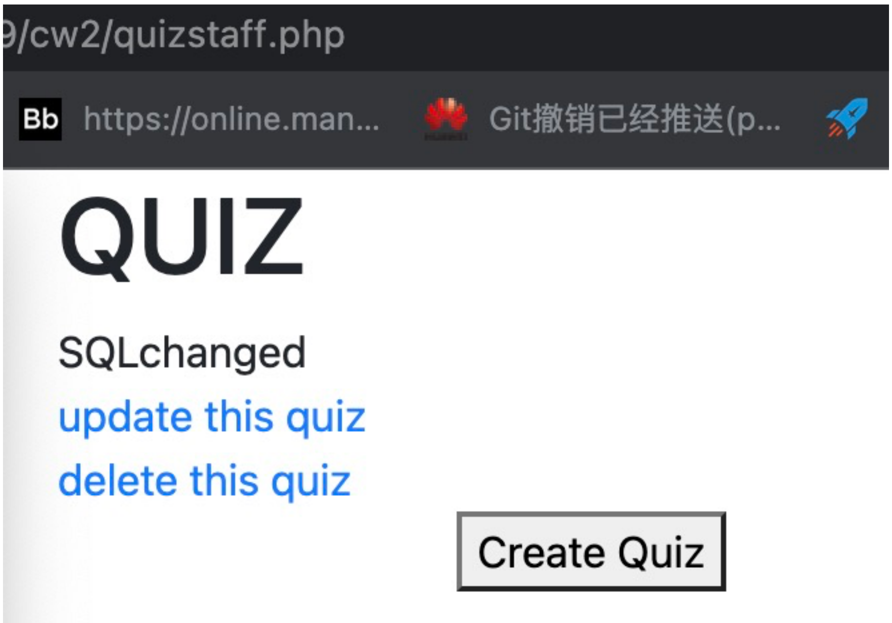
fill in the new quiz info

new quiz name: SQLchanged

quiz available: no

quiz duration: 60min

Please fill in the information, same as before, please enter yes or no in the quiz available part, here I changed the quiz name from "SQL" to "SQLchanged", then we should click update and will be back to the main quiz page (quizstaff.php) again.
As you can see in the below picture, the quiz name "SQL" has been changed to "SQLchanged"



9/cw2/quizstaff.php

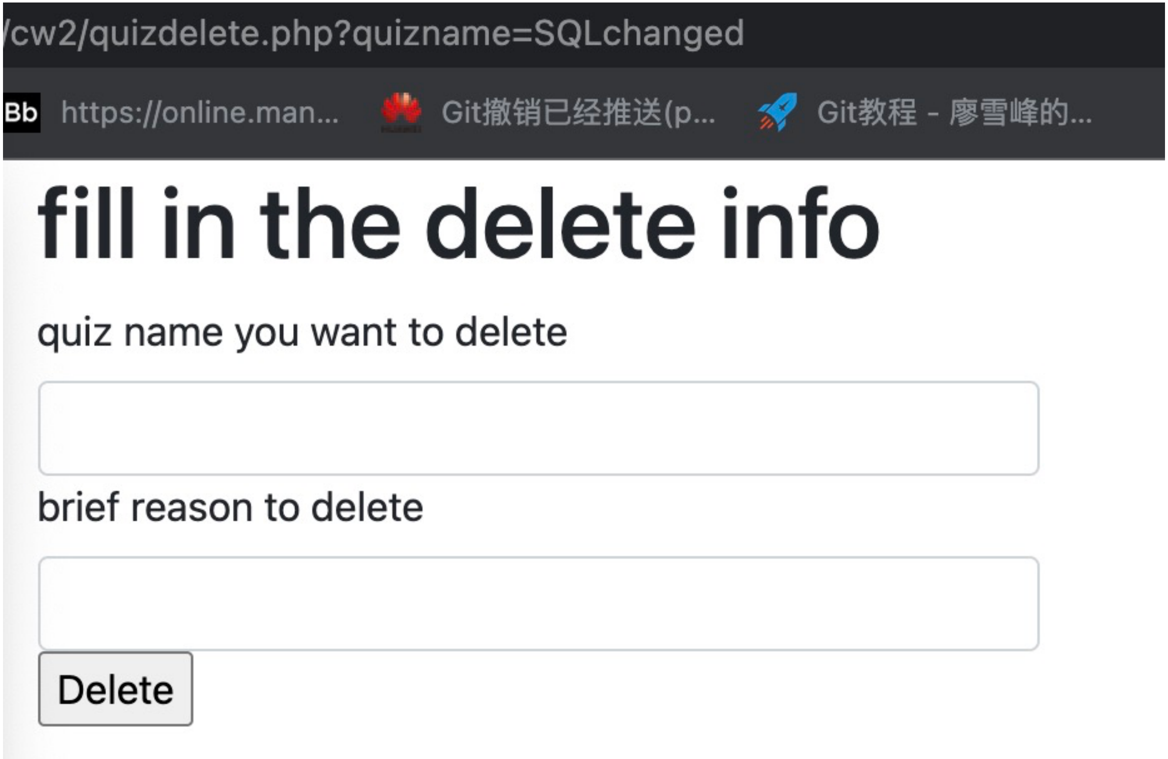
QUIZ

SQLchanged

update this quiz

delete this quiz

To test the delete function, click “delete this quiz”, then you will be directed to delete quiz page(quizdelete.php) which is shown as below:



/cw2/quizdelete.php?quizname=SQLchanged

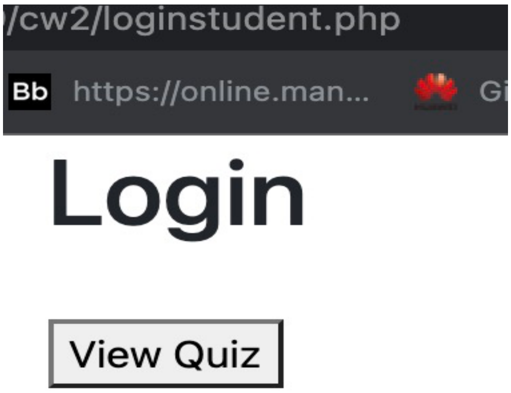
fill in the delete info

quiz name you want to delete

brief reason to delete

Please fill the form, the first should exactly the same of the quiz name you want to delete, If you are not sure have a check on the url the quiz name is shown just after the .php and the reason you want to delete you can write whatever you want. After you click the “Delete” button you will be directed to the main quiz page again(quizstaff.php). this time you will find the quiz you chose does not show anymore and have a check of database you you will find the records related to that quiz are also be removed. Please have a run of my trigger and then go back to another quiz then check the database.

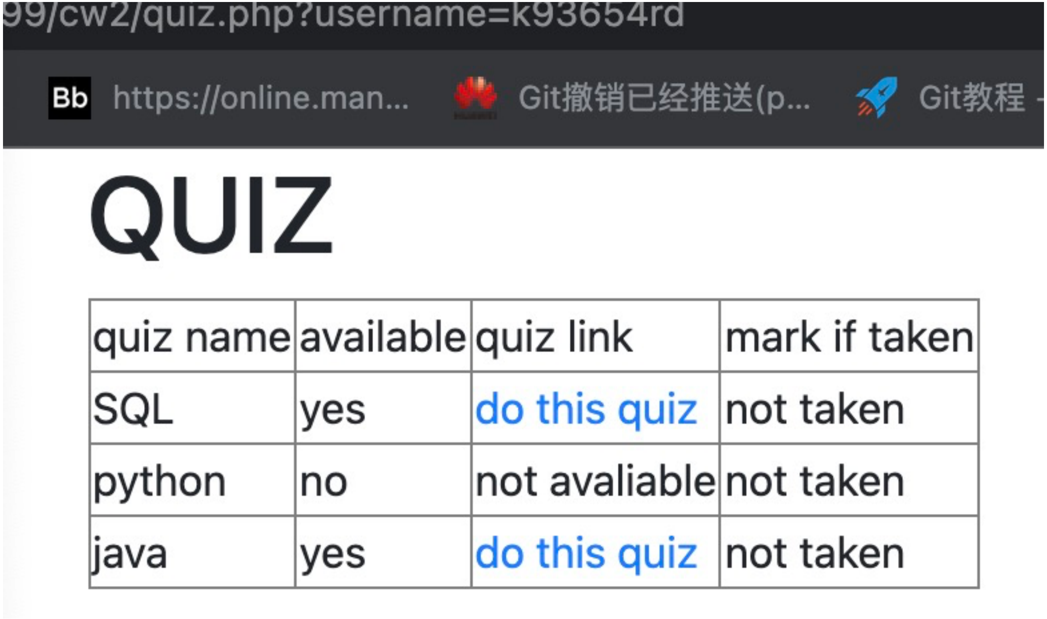
Then we have finished the tour of staff part, let’s begin the student part. Much similar as staff part, go to studentregister.php to register as a student and go to loginstudent.php to log in as a student. Once your username and password matches where database records, the below picture will appear:


/cw2/loginstudent.php

Login

Click the button “View quiz” to go to the quiz main page for student(quiz.php).

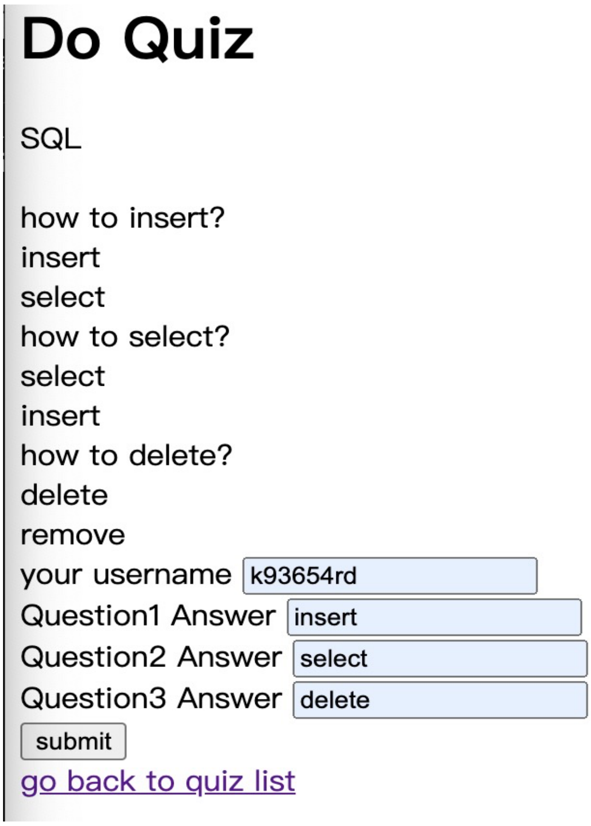
The main quiz page of student user is shown as below:


99/cw2/quiz.php?username=k93654rd

QUIZ

quiz name	available	quiz link	mark if taken
SQL	yes	do this quiz	not taken
python	no	not avaliable	not taken
java	yes	do this quiz	not taken

Here I have already created three quizzes using a staff user account. The table here shows Quiz name, quiz available, if the quiz is available then a quiz link will bring you to do the If the student has already taken this quiz the mark of the quiz will also be shown. Here we Click “do this quiz” in SQL row, this will bring us to the quiz take page(quiztake.php), which is shown as below:

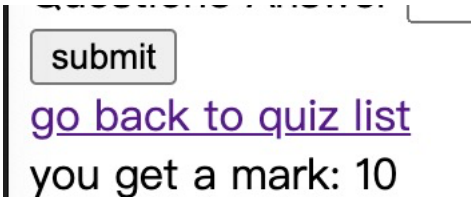


Do Quiz

SQL

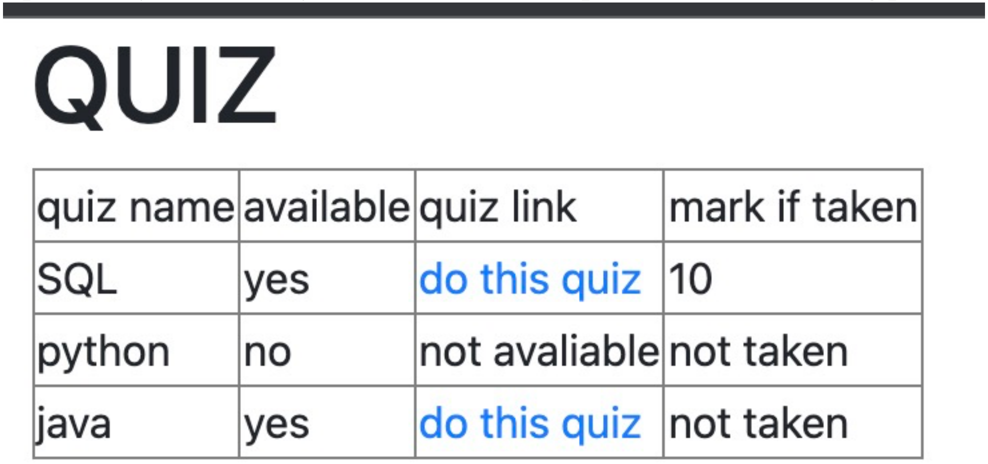
how to insert?
insert
select
how to select?
select
insert
how to delete?
delete
remove
your username k93654rd
Question1 Answer insert
Question2 Answer select
Question3 Answer delete

The quiz name comes at top of the body, then the three questions followed by the options Will be shown. And below, first you need to enter your username, then your answers of the 3 questions. After you finished, just click “submit”, then your mark will be shown below as the following picture shows:



you get a mark: 10

and now your taken records will be stored in the database in the table taken. Then click “go back to quiz list” you will see your status has been updated as the following picture shows:



QUIZ

quiz name	available	quiz link	mark if taken
SQL	yes	do this quiz	10
python	no	not avaliable	not taken
java	yes	do this quiz	not taken

As you can see here, The SQL row the mark has been updated. If you log out and change Student account to login, when you log in you will find the table is different that is any users can only view the taken information of themselves.

Part E: Stored Procedures and Triggers

In this section I wrote the MySQL to create a stored procedure that display the student names for the quizzes where they achieved less than 40%

I also wrote the MySQL to create a trigger that will log the staff id, quiz id and current date and time. Before the trigger is created, a new table delete_record records the deleted data is created

Stored procedure:

```
DELIMITER ^
    CREATE PROCEDURE getLessForty()
        BEGIN
        select forename, surname from student where username in
        (select username from taken where fullmark<40)
        union
        select fullmark from taken where fullmark<40;
   END ^
DELIMITER ;
```

Trigger:

```
create table if not exists delete_record(
    staff_id int primary key not null,
    quiz_id int not null,
        deletedate date not null,
        deletetime time not null,
);

DELIMITER ^
CREATE TRIGGER record_quiz_deletion
    BEFORE DELETE ON quiz FOR EACH ROW
    BEGIN
      INSERT INTO delete_record
        SET staff_id = OLD.author_id,
            quiz_id = OLD.quiz_id,
            deletedate = curdate(),
             deletetime =curtime();
    END ^
DELIMITER ;
```