CS-602: Server-Side Web Development

Assignment 5

This document should not be disseminated outside the purview of its intended purpose.

General Rules for Homework Assignments

- You are strongly encouraged to add comments throughout the program. Doing so will help your instructor to understand your programming logic and grade you more accurately.
- You must work on your assignments individually. You are not allowed to copy the answers from others.
- Each assignment has a strict deadline. Assignments submitted after the deadline will carry a penalty.
- When the term **`lastName`** is referenced in an assignment, please replace it with your last name.

Download and extract the starter template zip file, **`CS602_HW5_lastName`**. Rename the folder with your last name. Complete the corresponding assignment files in this folder.

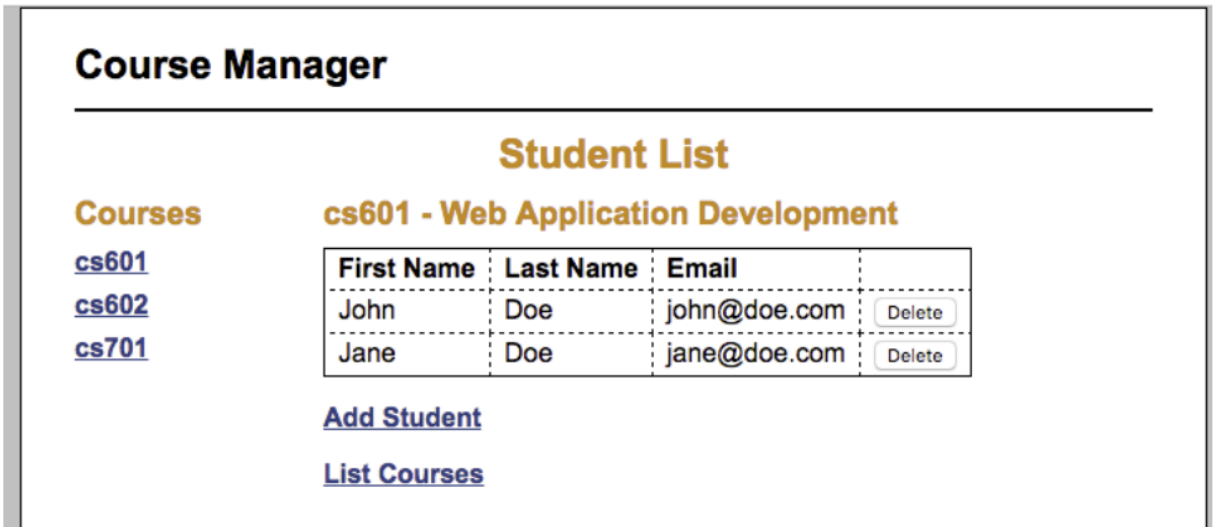
PDO – Course/Student Manager (100 Points)

Create the necessary PHP files using PDO in order to provide the following functionality. The application maintains a list of courses. Each course has a list of students enrolled in that course. The database structure and initial data used for the application is shown at the end of this document.

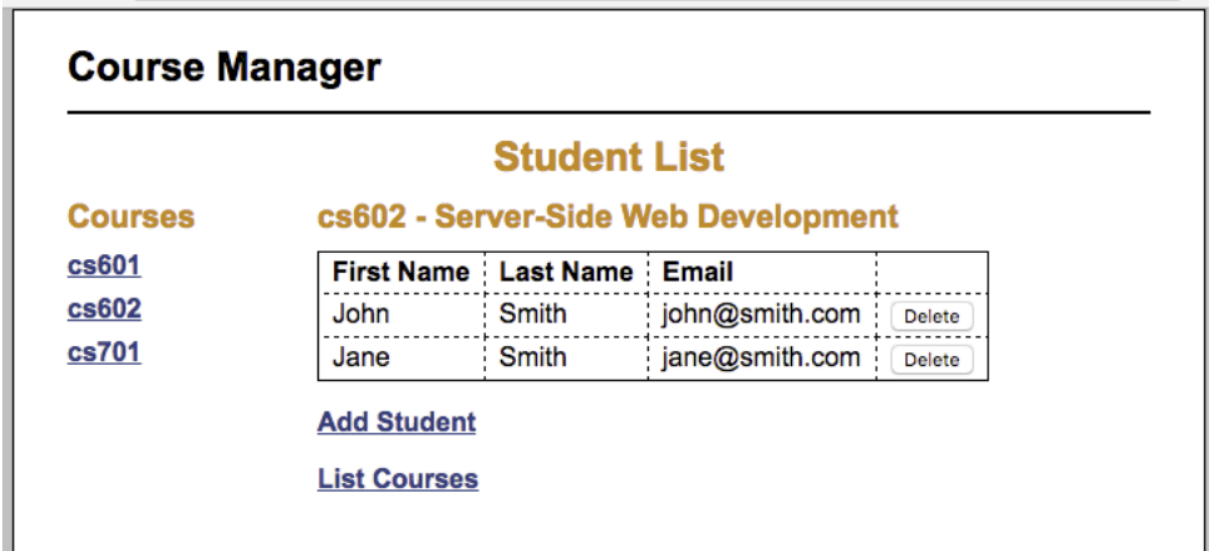
The home page, **`index.php`**, shows a list of courses. The students enrolled in the first course are shown as well. The home page has links for deleting a student, adding a student, and a link to manage the list of courses, as shown below.

Boston University Metropolitan College

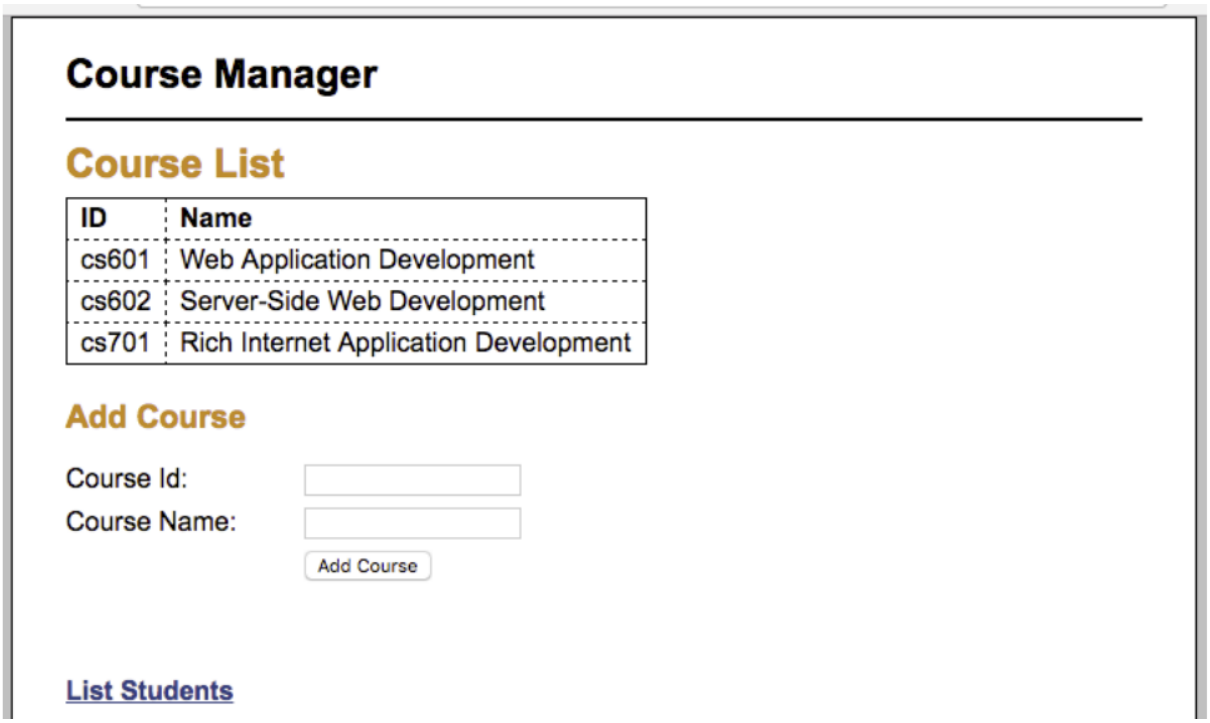
Ed Orsini | edorsini@bu.edu



Selecting a different course shows the same home page along with the request parameter corresponding to the selected course_id as shown below.

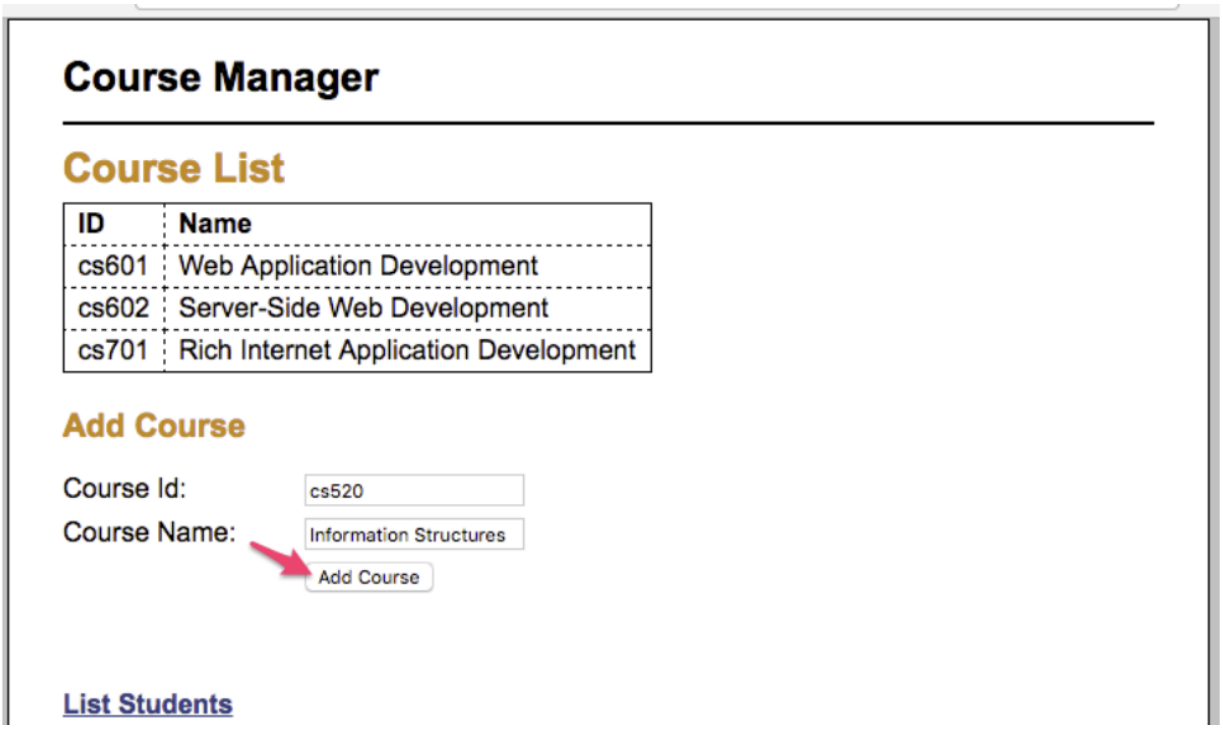


The List Courses link shows the course_list.php page as shown below. The current set of courses is shown. The page also includes a form for adding a new course. The user can also go to the home page by clicking the List Students link.

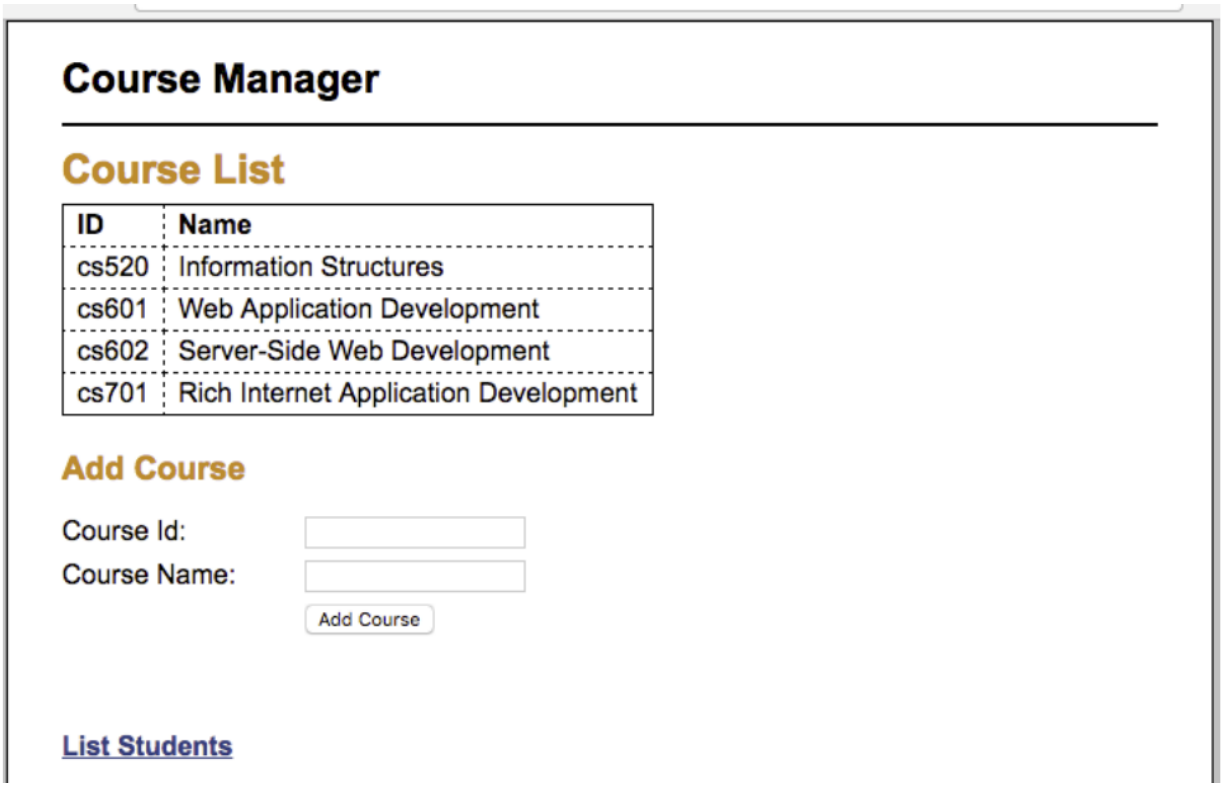


The user can enter new course information and submit the form (use POST) by clicking the Add Course button, as shown below.

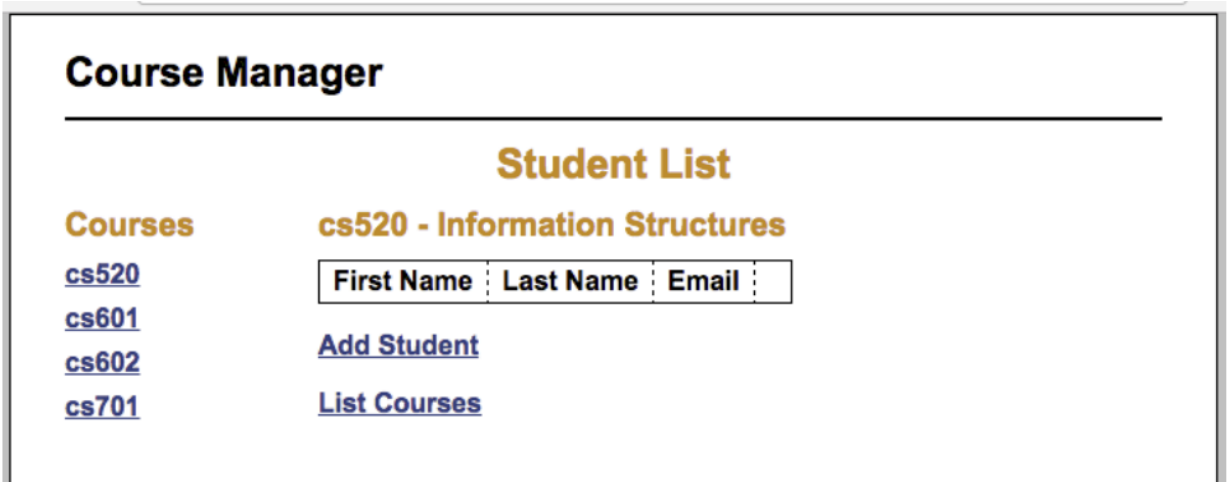
Ed Orsini | edorsini@bu.edu



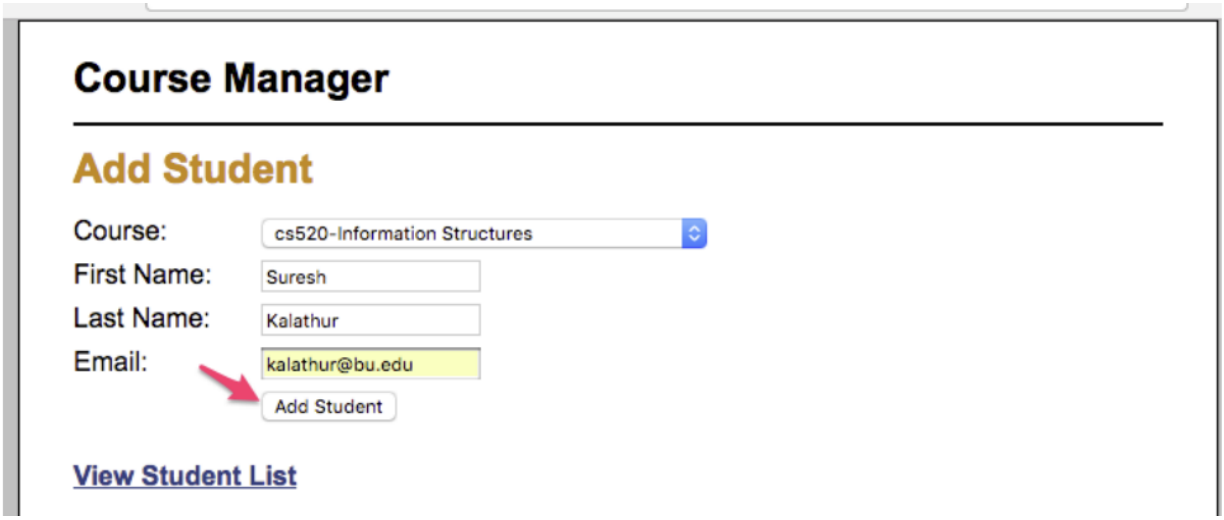
The add_course.php script handles the above form submission. The result includes the contents of course_list.php, as shown below. Note that the courses are always displayed in the ascending order of their course id.



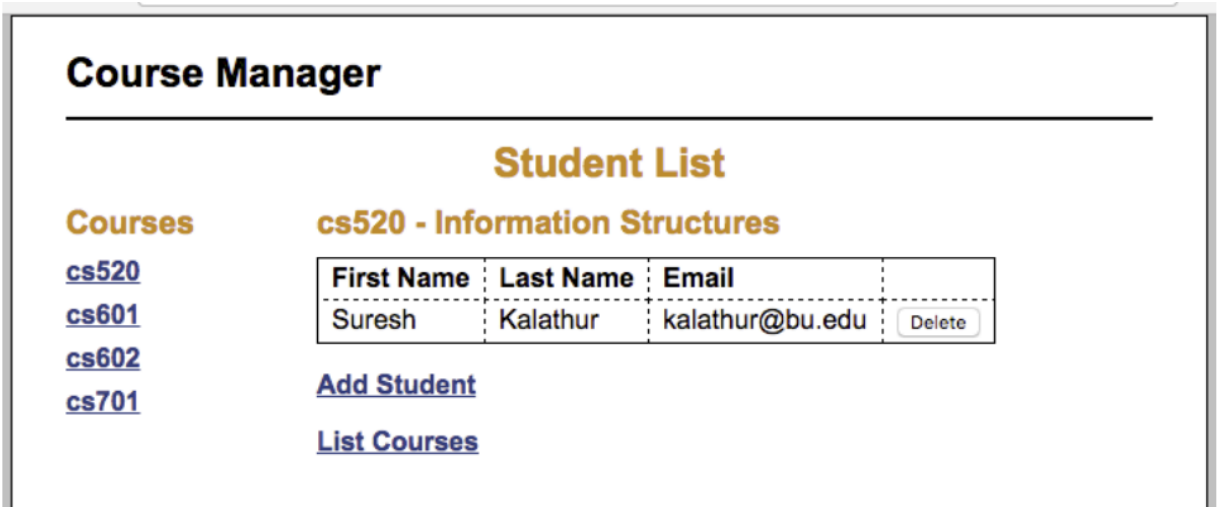
The List Students link shows the home page, index.php. The newly added course has no students as shown below.



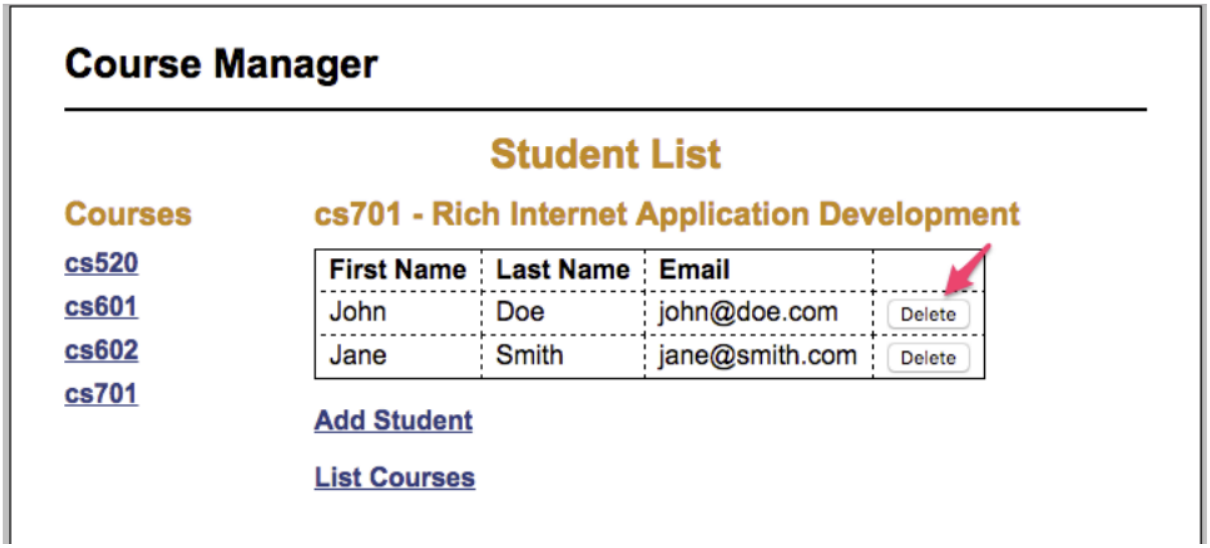
The Add Student link shows the **`add_student_form.php`** page. The form shows a drop-down list of courses, and the fields for entering the first name, last name, and email of the student. The form is submitted with POST.



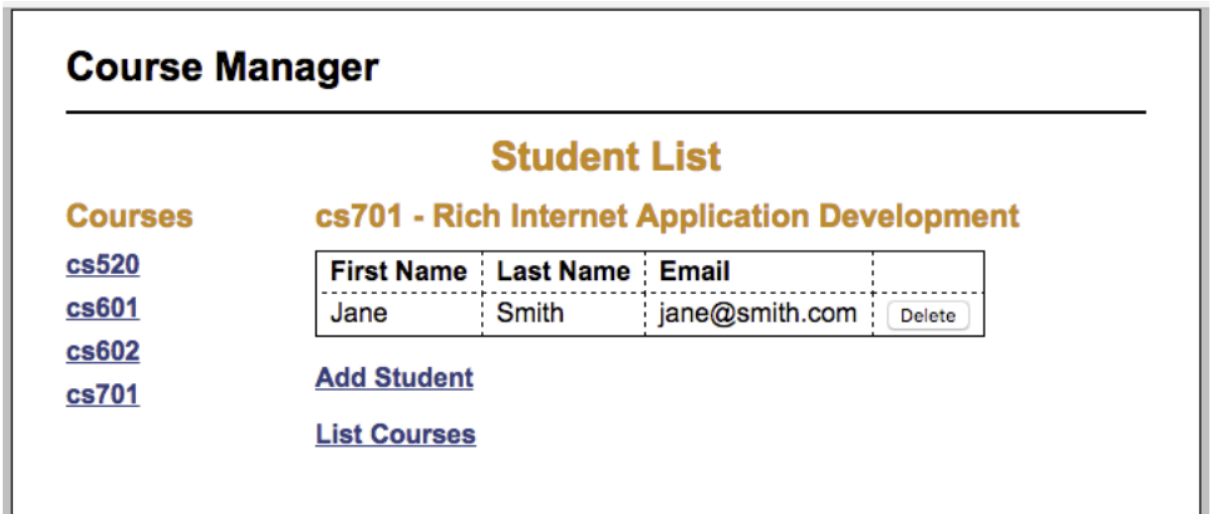
The above form submission is handled by **`add_student.php`**. The result includes the home page, showing the course selection and the resulting students in that selected course.



The Delete button deletes the corresponding student from the database. The following figure shows the current list of students in the cs701 course.



The delete action is handled by **`delete_student.php`**. After the student is deleted, the result includes the home page with the current course.



The following should be used for connecting to the database, **`database.php`**.

database.php

```php
<?php
    $dsn = 'mysql:host=localhost;dbname=cs602';
    $username = 'cs602_user';
    $password = 'cs602_secret';

    try {
        $db = new PDO($dsn, $username, $password);
    } catch (PDOException $e) {
        $error_message = $e->getMessage();
        include('database_error.php');
        exit();
    }
?>
```

Use the following database schema and the sample data shown below. The tables and the columns within the tables should match the schema shown.

Ed Orsini | edorsini@bu.edu

```sql
-- create the tables
CREATE TABLE sk_courses (
  courseID        VARCHAR(12)     NOT NULL,
  courseName      VARCHAR(255)    NOT NULL,
  PRIMARY KEY (courseID)
);

CREATE TABLE sk_students (
  studentID     INT(11)           NOT NULL AUTO_INCREMENT,
  courseID      VARCHAR(12)       NOT NULL,
  firstName     VARCHAR(255)      NOT NULL,
  lastName      VARCHAR(255)      NOT NULL,
  email         VARCHAR(255)      NOT NULL,
  PRIMARY KEY (studentID)
);


-- insert data into the database
INSERT INTO sk_courses VALUES
('cs601', 'Web Application Development'),
('cs602', 'Server-Side Web Development'),
('cs701', 'Rich Internet Application Development');

INSERT INTO sk_students VALUES
(1, 'cs601', 'John', 'Doe', 'john@doe.com'),
(2, 'cs601', 'Jane', 'Doe', 'jane@doe.com'),
(3, 'cs602', 'John', 'Smith', 'john@smith.com'),
(4, 'cs602', 'Jane', 'Smith', 'jane@smith.com'),
(5, 'cs701', 'John', 'Doe', 'john@doe.com'),
(6, 'cs701', 'Jane', 'Smith', 'jane@smith.com');
```


Ed Orsini | edorsini@bu.edu

Submission

Export your **`CS602_HW5_lastName`** folder containing all the relevant files as a zip file, and upload the zip file to the Assignment section.

Boston University Metropolitan College

Ed Orsini | edorsini@bu.edu