2 Tier Client-Server PHP MySQL Connector
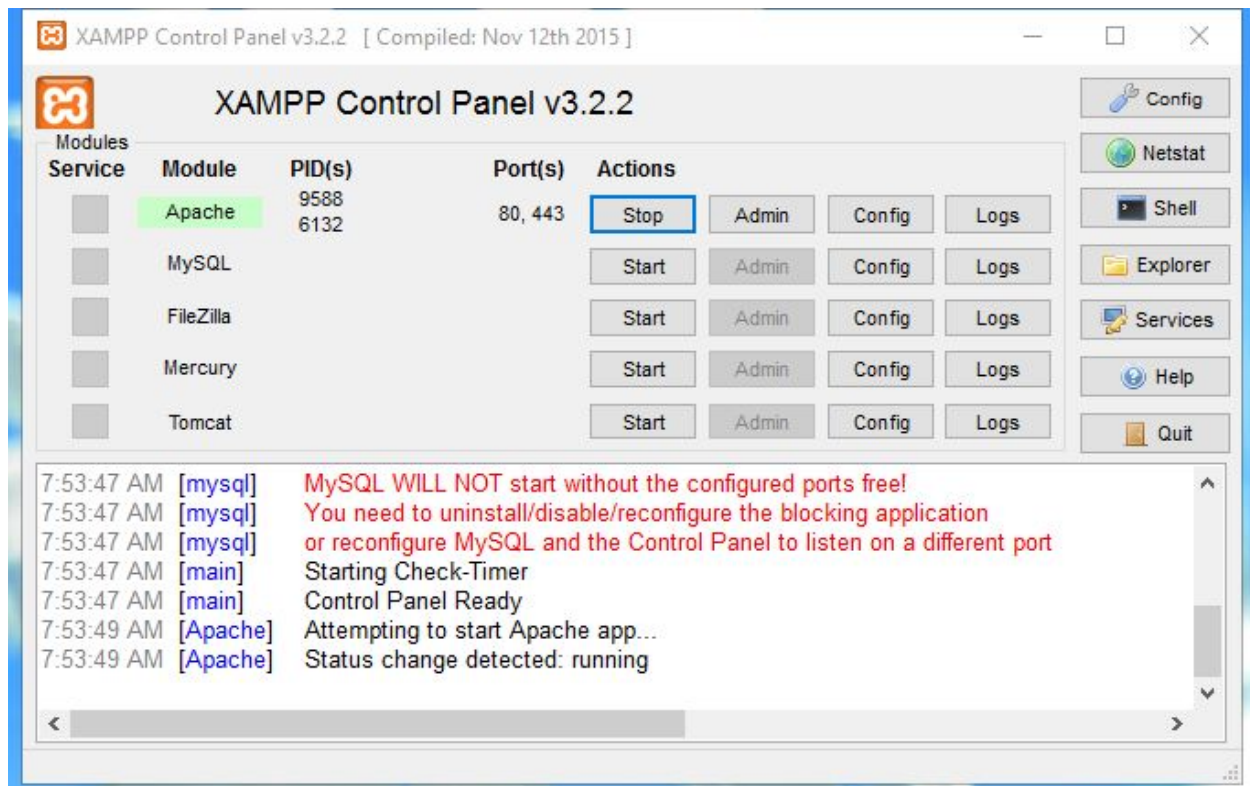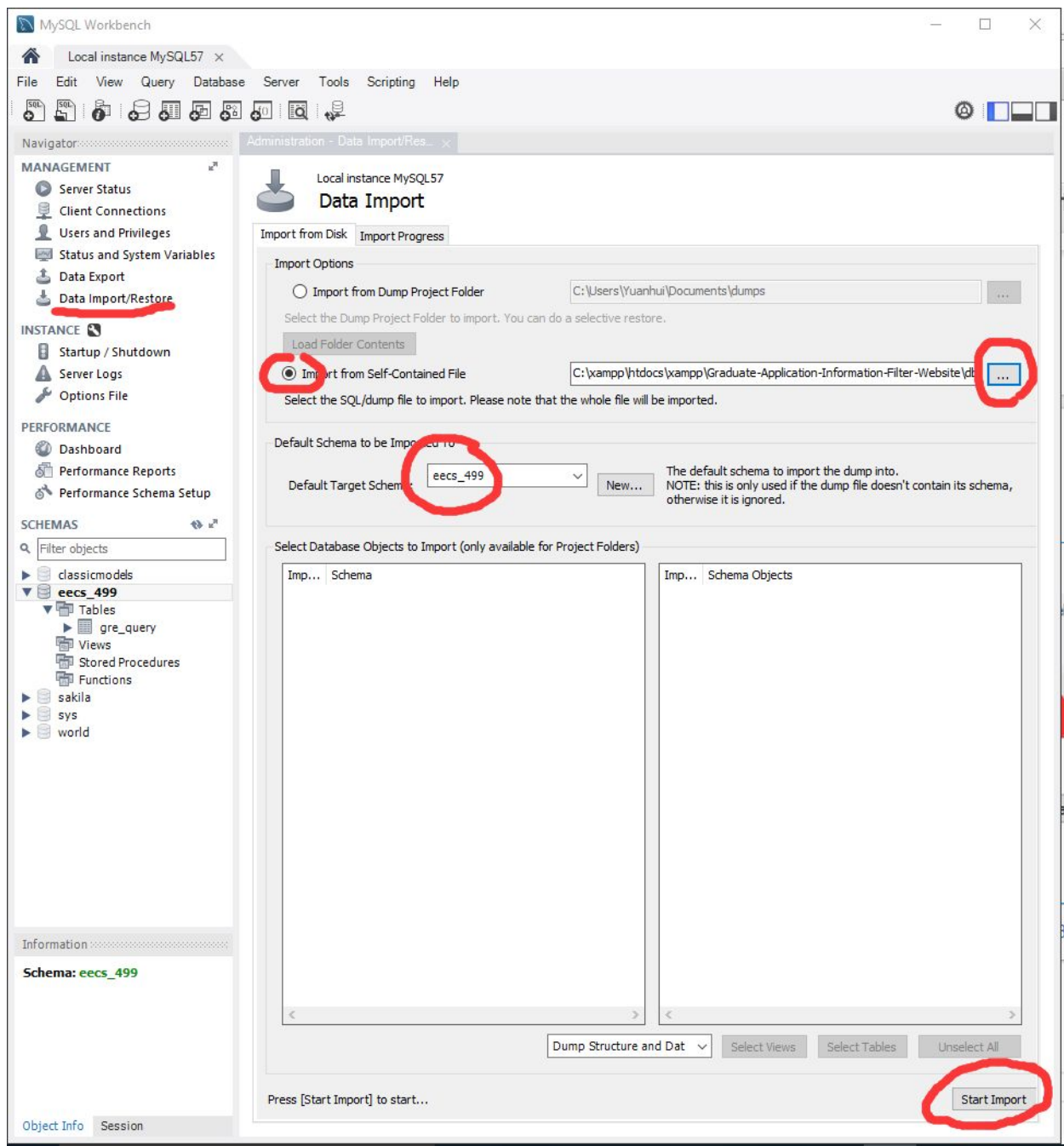## Yuanhui Yang

1 Build Environment

* From here to download XAMPP PHP development environment. Install it in Windows 10 OS, then move the entire dictionary, 'Graduate-Application-Information-Filter-Website/', in 'C:\xampp\htdocs\xampp'. Run it and launch its Apache Module.

* Create 'eecs_499' schema in MySQL Workbench and import eecs_499.sql into eecs_499 schema.

* Keep MySQL and Apache alive.

2 Dictionary Structure:

C:\xampp\htdocs\xampp

C:\xampp\htdocs\xampp\Graduate-Application-Information-Filter-Website

C:\xampp\htdocs\xampp\Graduate-Application-Information-Filter-Website\connector

C:\xampp\htdocs\xampp\Graduate-Application-Information-Filter-Website\css

C:\xampp\htdocs\xampp\Graduate-Application-Information-Filter-Website\db

C:\xampp\htdocs\xampp\Graduate-Application-Information-Filter-Website\js

3 JavaScript AJAX call PHP Connector by [XMLHttpRequest API](#)

`xmlhttp.open("GET", "connector/mysql-connector-to-schema-eecs_499.php?ping=" + ping + "&method=" + method, true)`; is the key line that call asynchronously connector/mysql-connector-to-schema-eecs_499.php with parameters of 'ping' and 'method'.

/js/myJs.js

```
    ping = "SELECT" + attrRes + " FROM " + table + " WHERE " + res;
    console.log(res);
    $("#Search").html(ping);



    tableHead = "";
    for ($idx = 0; $idx < checkAttr.length; $idx++) {
      tableHead = tableHead + "<th>";
      tableHead = tableHead + checkAttr[$idx];
      tableHead = tableHead + "</th>";
    }
    $("#title").html(tableHead);



    if (ping == "") {
      document.getElementById("body").innerHTML = "";
```

```
        return;
    } else {
      if (window.XMLHttpRequest) {
          // code for IE7+, Firefox, Chrome, Opera, Safari
          xmlhttp = new XMLHttpRequest();
      } else {
          // code for IE6, IE5
          xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
      }
      xmlhttp.onreadystatechange = function() {
        if (xmlhttp.readyState == 4 && xmlhttp.status == 200) {
            document.getElementById("body").innerHTML = xmlhttp.responseText;
        }
      };
      xmlhttp.open("GET", "connector/mysql-connector-to-schema-eecs_499.php?ping=" + ping +
"&method=" + method, true);
      xmlhttp.send();
    }
```

4 PHP Connector by [MySQL Improved Extension](#)

Received the 2 parameters from /js/myJs.js, PHP Connector connect MySQL's 'eecs_499' database where user is 'root' and password='root' then print the selected table to webpage by echo function.

connector/mysql-connector-to-schema-eecs_499.php

```php
<?php
$ping = $_GET["ping"];
$method = $_GET["method"];
// Hostname: localhost, username: root, password: root, db: eecs_499
$mysqli = new mysqli('localhost', 'root', 'root', 'eecs_499');
      if ($mysqli->connect_errno) {
      echo "Sorry, this website is experiencing problems.";
      echo "Error: Failed to make a MySQL connection, here is why: \n";
      echo "Errno: " . $mysqli->connect_errno . "\n";
      echo "Error: " . $mysqli->connect_error . "\n";
      exit;
}
if (!$pong = $mysqli->query($ping)) {
      echo "Sorry, the website is experiencing problems.";
      echo "Error: Our query failed to execute and here is why: \n";
      echo "Query: " . $ping . "\n";
```

```php
        echo "Errno: " . $mysqli->errno . "\n";
        echo "Error: " . $mysqli->error . "\n";
        exit;
}
if ($pong->num_rows === 0) {
        echo "We could not find a match, sorry about that. Please try again.";
        exit;
}
if ($method === "Search") {
        while ($row = $pong->fetch_array(MYSQLI_NUM)) {
                echo "<tr>";
                for ($column = 0; $column < $pong->field_count; $column++) {
                        echo "<td>";
                        echo $row[$column];
                        echo "</td>";
                }
                echo "</tr>";
        }
}
$pong->free();
$mysqli->close();
?>
```

5 Test

Front-end webpage generates MySQL query words,

SELECT EmployID, Program, Gender, Ethnicity, Country FROM GRE_Query WHERE (
(Gre_verbal > 150) OR (Gre_quan < (SELECT avg(Gre_quan) FROM GRE_Query)))
AND ((Gre_verbal > (SELECT min(Gre_verbal) FROM GRE_Query)) OR (Gre_quan
< (SELECT max(Gre_quan) FROM GRE_Query))) AND ((Gre_verbal > (SELECT
avg(Gre_verbal) FROM GRE_Query)) OR (Gre_quan < (SELECT avg(Gre_quan)
FROM GRE_Query)))

PHP connector forwards it to Back-end server and print the server's reply on webpage.

Select

☑ Employee ID ☑ Program ☑ Gender ☑ Ethnicity ☑ Country of Citizenship

Filter

	Gre_verbal ▼	> ▼	value ▼	150	**+** OR
or	Gre_quan ▼	< ▼	avg ▼	165	

AND

	Gre_verbal ▼	> ▼	min ▼		**+** OR
or	Gre_quan ▼	< ▼	max ▼		

AND

	Gre_verbal ▼	> ▼	avg ▼			**+** OR
or	Gre_quan ▼	< ▼	avg ▼		

+ AND

⬏ Clear

🔍 Search

	⤒ Clear
	🔍 Search

SELECT EmployID, Program, Gender, Ethnicity, Country FROM GRE_Query WHERE ((Gre_verbal > 150) OR (Gre_quan < (SELECT avg(Gre_quan) FROM GRE_Query))) AND ((Gre_verbal > (SELECT min(Gre_verbal) FROM GRE_Query)) OR (Gre_quan < (SELECT max(Gre_quan) FROM GRE_Query))) AND ((Gre_verbal > (SELECT avg(Gre_verbal) FROM GRE_Query)) OR (Gre_quan < (SELECT avg(Gre_quan) FROM GRE_Query)))

Result

EmployID	Program	Gender	Ethnicity	Country
2417008	Computer Science: PhD (C45PH)	Female	Asian	UNITED STATES
2496110	Computer Science: MS (C45MS)	Male	White	UNITED STATES
2512001	Computer Science: MS (C45MS)	Male	Asian	CHINA
2545141	Electrical Engineering: PhD (E16PH)	Male	Asian	MYANMAR
2545885	Computer Science: MS (C45MS)	Male	Asian	UNITED STATES
2553805	Electrical Engineering: PhD (E16PH)	Male	Asian	UNITED STATES
2562529	Electrical Engineering: MS (E16MS)	Male	Asian	KOREA (REPUBLIC OF)
2562807	Computer Science: PhD (C45PH)	Male	Asian	UNITED STATES
2563203	Computer Science: MS (C45MS)	Male	Asian	CHINA
2579184	Computer Science: PhD (C45PH)	Male	White	UNITED STATES
2601445	Computer Science: MS (C45MS)	Male		CHINA
2612373	Electrical Engineering: PhD (E16PH)	Female	Asian	CHINA
2622290	Computer Science: PhD (C45PH)	Male		CHINA
2656535	Electrical Engineering: PhD (E16PH)	Male	Asian	CHINA
2678961	Computer Science: PhD (C45PH)	Male	White	UNITED STATES