MSCI 346 – Spring 2019

LAB #5 - PHP/SQL

Introduction to Web Development

- We need to perform activities such as:
  - Connecting to the database server
  - Querying the database
  - Displaying results on a web form
  - Querying the database using user-input as a parameter

1. HTML to create web forms
2. Server Side Scripting (PHP)
3. SQL

Recap of Lab4

- PhpMyAdmin Tutorial
  - Create Tables
  - Bulk-load data
  - Running SQL queries
- PHP-MySQLi Connection

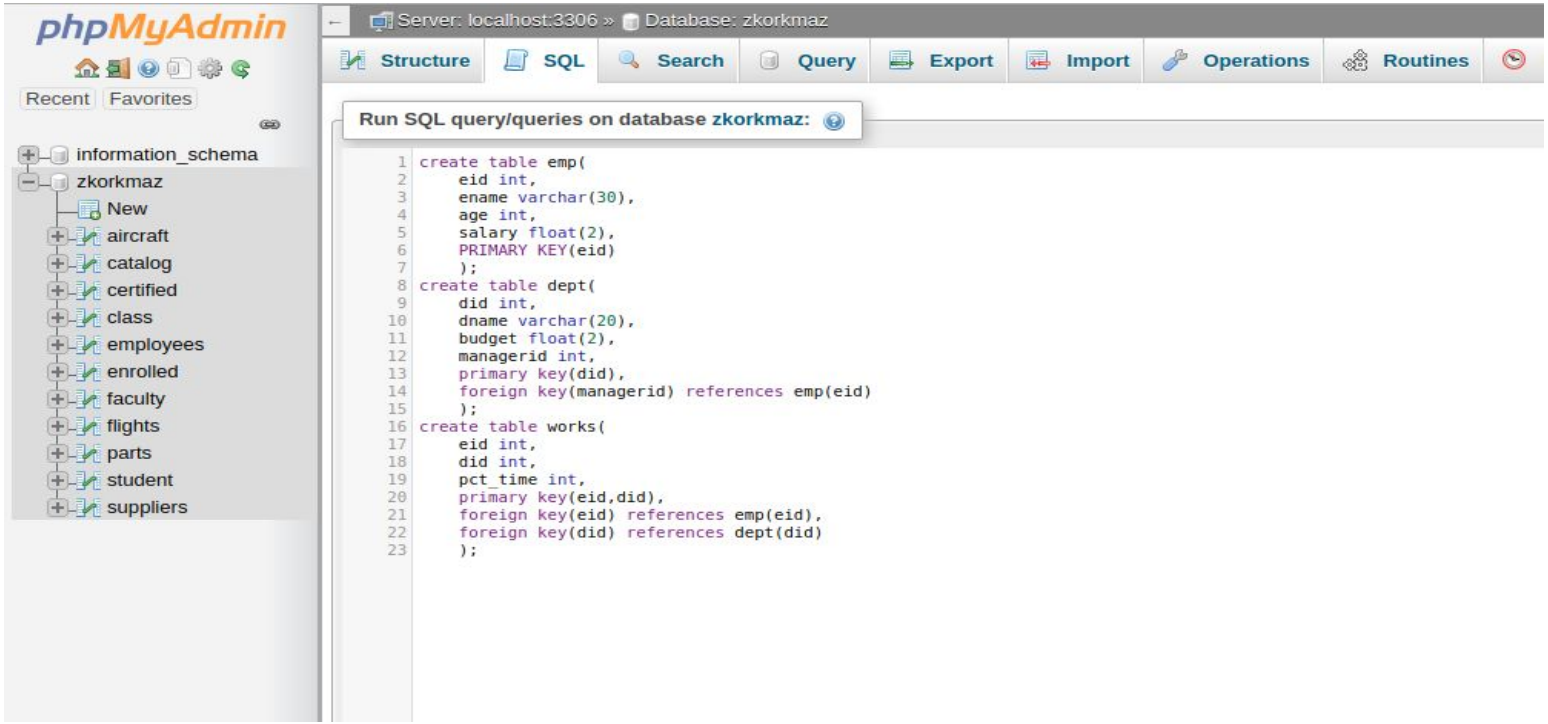
LAB 4: Exercises

- The schema for Exercise 5.2 & 5.4 is given below:

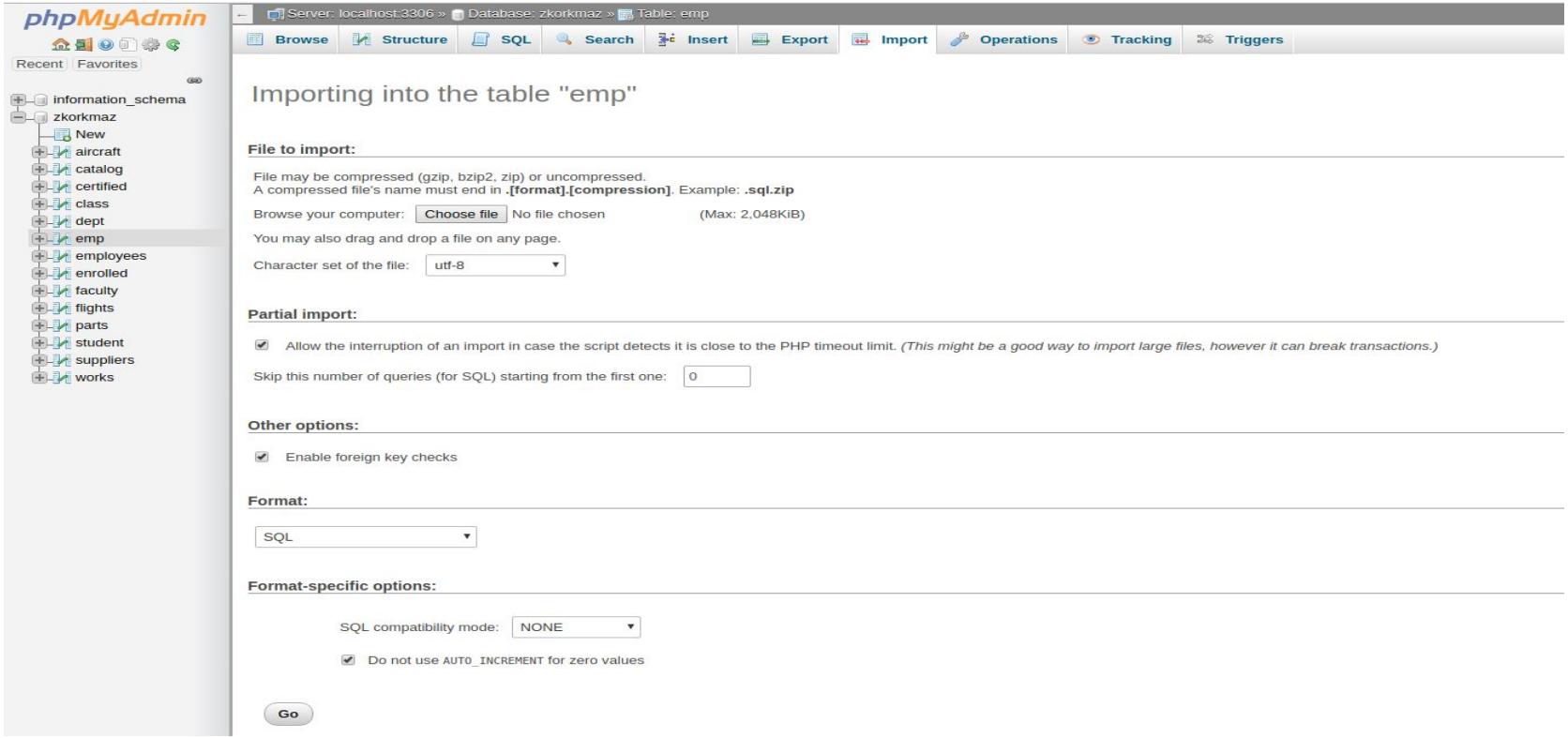
Suppliers(*sid:* `integer`, *sname:* `string`, *address:* `string`)
Parts(*pid:* `integer`, *pname:* `string`, *color:* `string`)
Catalog(*sid:* `integer`, *pid:* `integer`, *cost:* `real`)

Emp(*eid:* `integer`, *ename:* `string`, *age:* `integer`, *salary:* `real`)
Works(*eid:* `integer`, *did:* `integer`, *pct_time:* `integer`)
Dept(*did:* `integer`, *dname:* `string`, *budget:* `real`, *managerid:* `integer`)

Create company db tables using SQL



Bulk-load data - import



Today

- PhpMyAdmin
  - Advanced SQL queries
- PHP-MySQLi Connection
  - Running queries with user-supplied parameters and print query output in php

Exercise - 1

Note that the Employees relation describes pilots and other kinds of employees as well; every pilot is certified for some aircraft, and **only pilots are certified to fly.**

origin destination

Flights(*flno:* `integer`, *from:* `string`, *to:* `string`, *distance:* `integer`, *departs:* `time`, *arrives:* `time`, *price:* `real`)
Aircraft(*aid:* `integer`, *aname:* `string`, *cruisingrange:* `integer`)
Certified(*eid:* `integer`, *aid:* `integer`)
Employees(*eid:* `integer`, *ename:* `string`, *salary:* `integer`)

Write each of the following queries in SQL.

Exercise-1

A. Find the names of aircraft such that all pilots certified to operate them have salaries more than $80,000.
B. For each pilot who is certified for more than three aircraft, find the eid and the maximum cruisingrange of the aircraft for which she or he is certified.
C. Identify the routes that can be piloted by every pilot who makes more than $100,000.
D. Print the enames of pilots who can operate planes with cruisingrange greater than 3000 miles but are not certified on any Boeing aircraft.
E. Compute the difference between the average salary of a pilot and the average salary of all employees (including pilots).

Solutions

1-A) Find the names of aircraft such that all pilots certified to operate them have salaries more than $80,000.

```
SELECT DISTINCT A.aname FROM aircraft A
WHERE A.aid IN
      (SELECT C.aid
      FROM certified C, employees E
      WHERE C.eid = E.eid AND NOT EXISTS
                              ( SELECT * FROM employees E1
                              WHERE E1.eid = E.eid AND E1.salary < 80000 ))
```

Solutions

1-B) For each pilot who is certified for more than three aircraft, find the eid and the maximum crusingrange of the aircraft for which she or he is certified.

```
SELECT C.eid, MAX(A.crusingrange)
FROM certified C, aircraft A
WHERE C.aid = A.aid
GROUP BY C.eid
HAVING COUNT(*) > 3
```

Solutions

1-C) Identify the routes that can be piloted by every pilot who makes more than $100,000.

```
SELECT DISTINCT F.origin, F.destination
FROM flights F
WHERE NOT EXISTS
                    (SELECT * FROM employees E
                     WHERE E.salary > 100000 AND
                     NOT EXISTS
                            (SELECT * FROM aircraft A, certified C
                             WHERE A.crusingrange > F.distance
                             AND E.eid = C.eid
                             AND A.aid = C.aid))
```

Solutions

1-D) Print the enames of pilots who can operate planes with cruisingrange greater than 3000 miles but are not certified on any Boeing aircraft.

```
SELECT DISTINCT E.ename FROM employees E
WHERE E.eid IN (
        (SELECT C.eid FROM certified C
        WHERE EXISTS
                (SELECT A.aid FROM aircraft A
                WHERE A.aid = C.aid AND A.crusingrange > 3000 )
                AND NOT EXISTS
                        (SELECT A1.aid
                        FROM aircraft A1
                        WHERE A1.aid = C.aid
                                AND A1.aname LIKE 'Boeing%' )))
```

Solutions

1-E) Compute the difference between the average salary of a pilot and the average salary of all employees (including pilots).

```
SELECT Temp1.avg - Temp2.avg
FROM (SELECT AVG (E.salary) AS avg
        FROM employees E
        WHERE E.eid IN
                        (SELECT DISTINCT C.eid
                         FROM certified C )
        ) AS Temp1,
       (SELECT AVG (E1.salary) AS avg
        FROM employees E1 ) AS Temp2
```

Exercise - 2

Consider the following relational schema. An employee can work in more than one department; the pct time field of the Works relation shows the percent-age of time that a given employee works in a given department.

Emp(*eid:* `integer`, *ename:* `string`, *age:* `integer`, *salary:* `real`)
Works(*eid:* `integer`, *did:* `integer`, *pct_time:* `integer`)
Dept(*did:* `integer`, *dname:* `string`, *budget:* `real`, *managerid:* `integer`)

Write each of the following queries in SQL.

Exercise-2

A. Print the name of each employee whose salary exceeds the budget of all of the departments that he or she works in.
B. Find the managerids of managers who manage only departments with budgets greater than $1 million.
C. Find the enames of managers who manage the departments with the largest budgets.
D. If a manager manages more than one department, he or she controls the sum of all the budgets for those departments. Find the managerids of managers who control more than $5 million.

Solutions..

2-A) Print the name of each employee whose salary exceeds the budget of all of the departments that he or she works in.

```
SELECT E.ename
FROM emp E
WHERE E.salary > ALL (SELECT D.budget
                      FROM dept D, works W
                      WHERE E.eid = W.eid AND D.did = W.did)
```

Solutions..

2-B) Find the managerids of managers who manage only departments with budgets greater than $1 million.

```
SELECT DISTINCT D.managerid
FROM dept D
WHERE 1000000 < ALL (SELECT D2.budget
                     FROM dept D2
                     WHERE D2.managerid = D.managerid )
```

Solutions..

2-C) Find the enames of managers who manage the departments with the largest budgets.

```
SELECT E.ename
FROM emp E
WHERE E.eid IN
                    (SELECT D.managerid
                     FROM dept D
                     WHERE D.budget = (SELECT MAX(D2.budget) FROM dept D2))
```

Solutions..

2-D) If a manager manages more than one department, he or she controls the sum of all the budgets for those departments. Find the managerids of managers who control more than $5 million.

```
SELECT D.managerid
FROM dept D
WHERE 5000000 < (SELECT SUM (D2.budget)
                 FROM dept D2
                 WHERE D2.managerid = D.managerid )
```

PHP-MySQLi

PHP application needs to communicate with a database server

- **Connecting to the database server**
- **Querying the database**
- **Displaying results on a web form**
- **Taking user input through a web form**
- **Querying the database using user-input as a parameter**
- Inserting the user-input into the database (next week)

MySQLi

- MySQLi stands for MySQL improved
- It is an API that provides functions to connect to the database, send queries and parameters to the database and collect results.

Interaction with a database using MySQLi can be categorized into 4 main stages:

1. Connecting to a database
2. Firing a query
3. Collecting its results
4. Closing the connection to the database

Documentation: http://php.net/manual/en/book.mysqli.php

Stage 1: Connect to a database

- Download **my_connect.php** from Learn.

Open ▾ | my_connect.php ~/Desktop/lab5 | Save

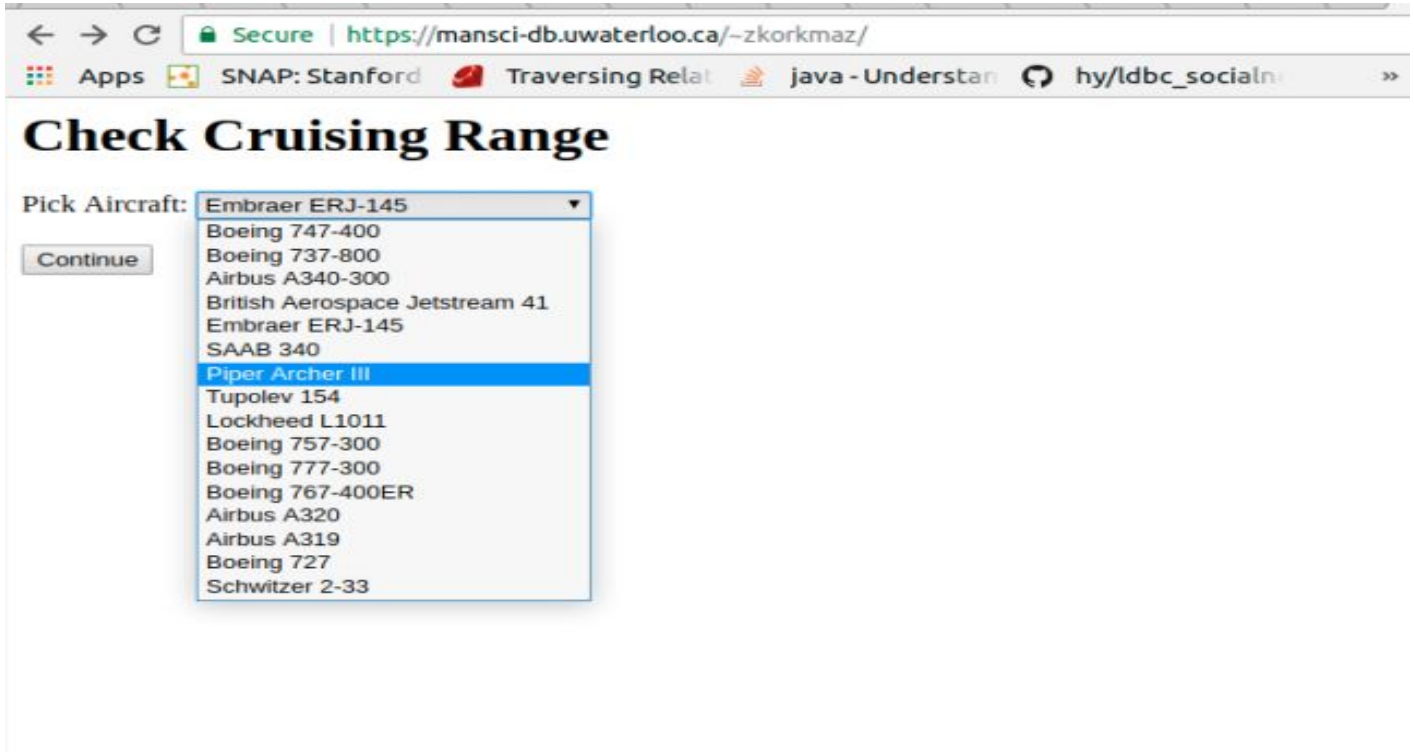
```php
<?php
// Function to obtain mysqli connection.
function get_mysqli_conn()
{
$dbhost = 'localhost';
$dbuser = 'username';
$dbpassword = 'password';
$dbname = 'username';
$mysqli = new mysqli($dbhost, $dbuser, $dbpassword, $dbname);
if ($mysqli->connect_errno)
{
echo 'Failed to connect to MySQL: (' . $mysqli->connect_errno . ') ' . $mysqli->connect_error;
}
return $mysqli;
}
?>
```

Connection Parameters

Connection Variable

Exercise-3

Application: Check cruising range of an aircraft



Step 1: Connect to a db

- Download **index.php** from Learn
- Include my_connect.php in index.php
- Assign connection variable

Open ▾ index.php ~/Desktop/lab5

```
1  <body>
2  <h1>Check Cruising Range</h1>
3
4  <form action="cruisingrange.php" method="get">
5
6  <?php
7  // Enable error logging:
8  error_reporting(E_ALL ^ E_NOTICE);
9  // mysqli connection via user-defined function
10 include ('./my_connect.php');
11 $mysqli = get_mysqli_conn();
12 ?>
13
14 <?php
15 // SQL statement
16 $sql = "SELECT a.aid, a.aname "
17         . "FROM aircraft a";
18
19 // Prepared statement, stage 1: prepare
20 $stmt = $mysqli->prepare($sql);
21
22 // Prepared statement, stage 2: execute
23 $stmt->execute();
24
25 // Bind result variables
26 $stmt->bind_result($aircraft_id, $aircraft_name);
27
28 /* fetch values */
29 echo '<label for="aid">Pick Aircraft: </label>';
30 echo '<select name="aid">';
31 while ($stmt->fetch())
32 {
33 printf ('<option value="%s">%s</option>', $aircraft_id, $aircraft_name);
34 }
35 echo '</select><br>';
36
37 /* close statement and connection*/
38 $stmt->close();
39 $mysqli->close();
40 ?>
41
42 <br>
43 <input type="submit" value="Continue"/>
44 </br>
45 </form>
46 </body>
```

Step 2: Fire a query

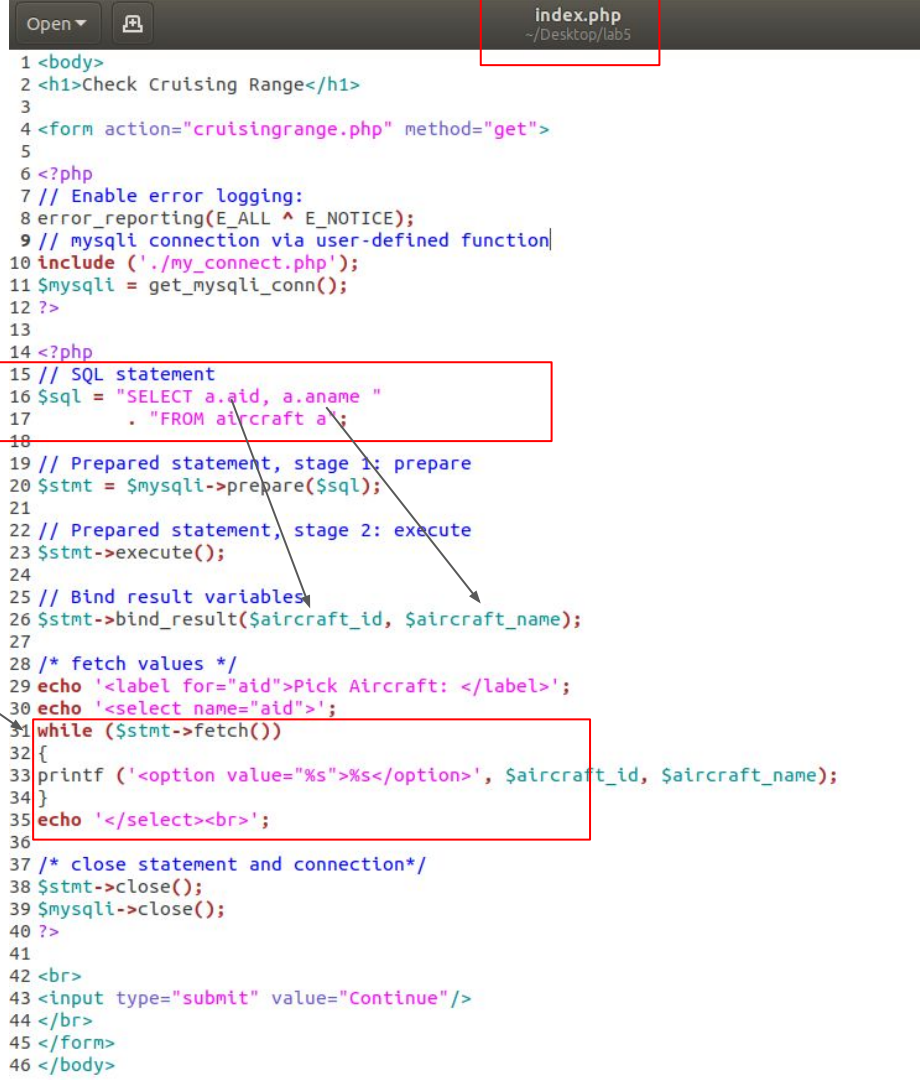
- Index page needs to prompt the user to select the name of the aircraft to check its cruising range.
  - Populate drop-down list with all aircraft names (fetch from database)
- Prepare statement.
- Execute query.

Open ▾ index.php ~/Desktop/lab5

```
1 <body>
2 <h1>Check Cruising Range</h1>
3
4 <form action="cruisingrange.php" method="get">
5
6 <?php
7 // Enable error logging:
8 error_reporting(E_ALL ^ E_NOTICE);
9 // mysqli connection via user-defined function
10 include ('./my_connect.php');
11 $mysqli = get_mysqli_conn();
12 ?>
13
14 <?php
15 // SQL statement
16 $sql = "SELECT a.aid, a.aname "
17         . "FROM aircraft a";
18
19 // Prepared statement, stage 1: prepare
20 $stmt = $mysqli->prepare($sql);
21
22 // Prepared statement, stage 2: execute
23 $stmt->execute();
24
25 // Bind result variables
26 $stmt->bind_result($aircraft_id, $aircraft_name);
27
28 /* fetch values */
29 echo '<label for="aid">Pick Aircraft: </label>';
30 echo '<select name="aid">';
31 while ($stmt->fetch())
32 {
33 printf ('<option value="%s">%s</option>', $aircraft_id, $aircraft_name);
34 }
35 echo '</select><br>';
36
37 /* close statement and connection*/
38 $stmt->close();
39 $mysqli->close();
40 ?>
41
42 <br>
43 <input type="submit" value="Continue"/>
44 </br>
45 </form>
46 </body>
```

Step 3: Collect Results

- Bind selected columns to PHP variables
- Fetch values by iterating through the executed statement's results



Step 3: Collect Results

- The name of the drop-down list is “aid”. When the user hits the “continue” button, **its value** will be stored as “aid”.

Step 3: Collect Results

- The name of the drop-down list is “aid”. When the user hits the “continue” button, its value will be stored as “aid” and redirect to **cruisingrange.php**

Open ▾ index.php ~/Desktop/lab5

```
1 <body>
2 <h1>Check Cruising Range</h1>
3
4 <form action="cruisingrange.php" method="get">
5
6 <?php
7 // Enable error logging:
8 error_reporting(E_ALL ^ E_NOTICE);
9 // mysqli connection via user-defined function
10 include ('./my_connect.php');
11 $mysqli = get_mysqli_conn();
12 ?>
13
14 <?php
15 // SQL statement
16 $sql = "SELECT a.aid, a.aname "
17         . "FROM aircraft a";
18
19 // Prepared statement, stage 1: prepare
20 $stmt = $mysqli->prepare($sql);
21
22 // Prepared statement, stage 2: execute
23 $stmt->execute();
24
25 // Bind result variables
26 $stmt->bind_result($aircraft_id, $aircraft_name);
27
28 /* fetch values */
29 echo '<label for="aid">Pick Aircraft: </label>';
30 echo '<select name="aid">';
31 while ($stmt->fetch())
32 {
33 printf ('<option value="%s">%s</option>', $aircraft_id, $aircraft_name);
34 }
35 echo '</select><br>';
36
37 /* close statement and connection*/
38 $stmt->close();
39 $mysqli->close();
40 ?>
41
42 <br>
43 <input type="submit" value="Continue"/>
44 </br>
45 </form>
46 </body>
```

Step 4: Close Connection

- Close the SQL query statement ($sql).
- Close the connection ($mysqli) if there are no further transactions in this PHP file

Open ▾ index.php ~/Desktop/lab5

```
1 <body>
2 <h1>Check Cruising Range</h1>
3
4 <form action="cruisingrange.php" method="get">
5
6 <?php
7 // Enable error logging:
8 error_reporting(E_ALL ^ E_NOTICE);
9 // mysqli connection via user-defined function
10 include ('./my_connect.php');
11 $mysqli = get_mysqli_conn();
12 ?>
13
14 <?php
15 // SQL statement
16 $sql = "SELECT a.aid, a.aname "
17         . "FROM aircraft a";
18
19 // Prepared statement, stage 1: prepare
20 $stmt = $mysqli->prepare($sql);
21
22 // Prepared statement, stage 2: execute
23 $stmt->execute();
24
25 // Bind result variables
26 $stmt->bind_result($aircraft_id, $aircraft_name);
27
28 /* fetch values */
29 echo '<label for="aid">Pick Aircraft: </label>';
30 echo '<select name="aid">';
31 while ($stmt->fetch())
32 {
33 printf ('<option value="%s">%s</option>', $aircraft_id, $aircraft_name);
34 }
35 echo '</select><br>';
36
37 /* close statement and connection*/
38 $stmt->close();
39 $mysqli->close();
40 ?>
41
42 <br>
43 <input type="submit" value="Continue"/>
44 </br>
45 </form>
46 </body>
```

How to use selected aircraft (user input) to find its cruising range?

Step 1: Connect to a db

- Download cruisingrange.php from Learn.
- Include my_connect.php in index.php
- Assign connection variable

Open ▾ | cruisingrange.php | mansci-db.uwaterloo.ca ~/public_html | Save

```
<body>
<h1>Cruising Range Information</h1>

<?php
// Enable error logging:
error_reporting(E_ALL ^ E_NOTICE);
// mysqli connection via user-defined function

include('./my_connect.php');
$mysqli = get_mysqli_conn();

// SQL statement
$sql = "SELECT a.crusingrange, a.aname "
. "FROM aircraft a "
. "WHERE a.aid = ?";

// Prepared statement, stage 1: prepare
$stmt = $mysqli->prepare($sql);

// Prepared statement, stage 2: bind and execute
$aid = $_GET['aid'];
// "i" for integer, "d" for double, "s" for string, "b" for blob
$stmt->bind_param('i', $aid);
$stmt->execute();

// Bind result variables
$stmt->bind_result($aircraft_cruisingrange, $aircraft_name);

/* fetch values */
if ($stmt->fetch())
{
printf ('Cruising Range for Aircraft %s is %s', $aircraft_name, $aircraft_cruisingrange);
//echo 'Cruising Range for Aircraft ' .$aircraft_name.' is: ' . $aircraft_cruisingrange;
}
//else
//{
//echo 'Record not found';
//}

/* close statement and connection*/
$stmt->close();
$mysqli->close();
?>
</body>

```

Step 2: Fire a query

- The MySQL server supports using anonymous positional placeholder with ?
  - Use ? for parameters (later filled in code or user-input)
- Prepare statement.

Open ▾ | cruisingrange.php | mansci-db.uwaterloo.ca ~/public_html | Save

```
1 <body>
2 <h1>Cruising Range Information</h1>
3
4 <?php
5 // Enable error logging:
6 error_reporting(E_ALL ^ E_NOTICE);
7 // mysqli connection via user-defined function
8
9 include('./my_connect.php');
10 $mysqli = get_mysqli_conn();
11
12 // SQL statement
13 $sql = "SELECT a.crusingrange, a.aname "
14 . "FROM aircraft a "
15 . "WHERE a.aid = ?";
16
17 // Prepared statement, stage 1: prepare
18 $stmt = $mysqli->prepare($sql);
19
20 // Prepared statement, stage 2: bind and execute
21 $aid = $_GET['aid'];
22 // "i" for integer, "d" for double, "s" for string, "b" for blob
23 $stmt->bind_param('i', $aid);
24 $stmt->execute();
25
26 // Bind result variables
27 $stmt->bind_result($aircraft_cruisingrange, $aircraft_name);
28
29 /* fetch values */
30 if ($stmt->fetch())
31 {
32 printf ('Cruising Range for Aircraft %s is %s', $aircraft_name, $aircraft_cruisingrange);
33 //echo 'Cruising Range for Aircraft ' .$aircraft_name.' is: ' . $aircraft_cruisingrange;
34 }
35 //else
36 //{
37 //echo 'Record not found';
38 //}
39
40 /* close statement and connection*/
41 $stmt->close();
42 $mysqli->close();
43 ?>
44 </body>
45
```

“aid” fetched from index.php

Step 2: Fire a query

- The MySQL server supports using anonymous positional placeholder with ?
  - Use ? for parameters (later filled in code or user-input)
- Prepare statement.
- Bind PHP variables to MySQL parameters
  - Should be done for all parameters that come from user-inputs

Open ▾ | cruisingrange.php | mansci-db.uwaterloo.ca ~/public_html | Save

```
1 <body>
2 <h1>Cruising Range Information</h1>
3
4 <?php
5 // Enable error logging:
6 error_reporting(E_ALL ^ E_NOTICE);
7 // mysqli connection via user-defined function
8
9 include('./my_connect.php');
10 $mysqli = get_mysqli_conn();
11
12 // SQL statement
13 $sql = "SELECT a.crusingrange, a.aname "
14 . "FROM aircraft a "
15 . "WHERE a.aid = ?";
16
17 // Prepared statement, stage 1: prepare
18 $stmt = $mysqli->prepare($sql);
19
20 // Prepared statement, stage 2: bind and execute
21 $aid = $_GET['aid'];
22 // "i" for integer, "d" for double, "s" for string, "b" for blob
23 $stmt->bind_param('i', $aid);
24 $stmt->execute();
25
26 // Bind result variables
27 $stmt->bind_result($aircraft_cruisingrange, $aircraft_name);
28
29 /* fetch values */
30 if ($stmt->fetch())
31 {
32 printf ('Cruising Range for Aircraft %s is %s', $aircraft_name, $aircraft_cruisingrange);
33 //echo 'Cruising Range for Aircraft ' .$aircraft_name.' is: ' . $aircraft_cruisingrange;
34 }
35 //else
36 //{
37 //echo 'Record not found';
38 //}
39
40 /* close statement and connection*/
41 $stmt->close();
42 $mysqli->close();
43 ?>
44 </body>
45
```

Step 3: Collect Results

- Bind selected columns to PHP variables
- Fetch values

cruisingrange.php
mansci-db.uwaterloo.ca ~/public_html

```
<body>
<h1>Cruising Range Information</h1>

<?php
// Enable error logging:
error_reporting(E_ALL ^ E_NOTICE);
// mysqli connection via user-defined function

include('./my_connect.php');
$mysqli = get_mysqli_conn();

// SQL statement
$sql = "SELECT a.crusingrange, a.aname "
. "FROM aircraft a "
. "WHERE a.aid = ?";

// Prepared statement, stage 1: prepare
$stmt = $mysqli->prepare($sql);

// Prepared statement, stage 2: bind and execute
$aid = $_GET['aid'];
// "i" for integer, "d" for double, "s" for string, "b" for blob
$stmt->bind_param('i', $aid);
$stmt->execute();

// Bind result variables
$stmt->bind_result($aircraft_cruisingrange, $aircraft_name);

/* fetch values */
if ($stmt->fetch())
{
printf ('Cruising Range for Aircraft %s is %s', $aircraft_name, $aircraft_cruisingrange);
//echo 'Cruising Range for Aircraft '.$aircraft_name.' is: '. $aircraft_cruisingrange;
}
//else
//{
//echo 'Record not found';
//}

/* close statement and connection*/
$stmt->close();
$mysqli->close();
?>
</body>
```

Step 4: Close Connection

Open ▾ | cruisingrange.php | mansci-db.uwaterloo.ca ~/public_html | Save

```
<body>
<h1>Cruising Range Information</h1>

<?php
// Enable error logging:
error_reporting(E_ALL ^ E_NOTICE);
// mysqli connection via user-defined function

include('./my_connect.php');
$mysqli = get_mysqli_conn();

// SQL statement
$sql = "SELECT a.crusingrange, a.aname "
. "FROM aircraft a "
. "WHERE a.aid = ?";

// Prepared statement, stage 1: prepare
$stmt = $mysqli->prepare($sql);

// Prepared statement, stage 2: bind and execute
$aid = $_GET['aid'];
// "i" for integer, "d" for double, "s" for string, "b" for blob
$stmt->bind_param('i', $aid);
$stmt->execute();

// Bind result variables
$stmt->bind_result($aircraft_cruisingrange, $aircraft_name);

/* fetch values */
if ($stmt->fetch())
{
printf ('Cruising Range for Aircraft %s is %s', $aircraft_name, $aircraft_cruisingrange);
//echo 'Cruising Range for Aircraft ' .$aircraft_name.' is: ' . $aircraft_cruisingrange;
}
//else
//{
//echo 'Record not found';
//}

/* close statement and connection*/
$stmt->close();
$mysqli->close();
?>
</body>
```

Exercise-4 (Next Week)

Application: Find the aircrafts that a pilot is certified for and have cruising ranges greater than a specified value.

- List all employees in drop-down list
- Input a cruising range

} You need to bind two parameters!!

- Use the selected pilot and cruising range to list the aircrafts that she or he is certified for.
    - If the result is empty -> print “No record found”