CPSC 304 Project Cover Page

Milestone #: 4

Date: August 9th, 2023

Group Number: 36

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Cole Rowell	52077435	chr2002	colerowell11@gmail.com
Nicholas Fong	78575271	nfong01	nicholasfong1120@gmail.com
Anikait Kapur	83243055	akapu3	anikaitkapur@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Repository Link

https://github.students.cs.ubc.ca/CPSC304-2023S-T2/project_h9f1g_r0u2d_s7t3k

SQL Script

Present in the GitHub repository in Client. The filename is SQL_Script.sql, and you can click on the name for an embedded link.

README.txt

It is included in the repository and outlines the changes that we made to the tech stack.

Project Description

The goal of our initial project was to create a database for an university co-op program, we focused on three major entities: Students, co-op Advisors, and Jobs. Of those major entities we relied on supporting entities to help better visualize the relationships and be better suited for real world application. These included company contacts, student documents, and other information that most co-op programs would like to keep track of such as which students got interviews at specific companies. While we did not create a fully functional co-op board we were able to effectively implement the queries necessary for day-to-day data analysis. For example, we wanted to focus on the student and co-op advisors more specifically, as that is the bulk of the tuples that will be present. Of those queries we implemented the insertion, deletion, and updation of co-op advisor entities, while also adding the functionality to display students advised by specific co-op advisors based on id, finding how many students each co-op advisor is managing, and projection of student attributes.

Final Schema Changes

From the original ER diagrams we turned in, a few changes were made such as turning Student Interviews With Company Contact into a ternary relationship which became Student Interviews With Company Contact For Job with a many-to-many relationship for each element. This was done to more accurately describe what interviews are like in real life. Additionally, we turned the document ISA into a double ISA where Document is at the top with Student Document as one of its sub-elements. This was done to allow Students only to upload Resumes and Cover Letters, while Employers can only upload contracts. Additionally, date and time attributes were added to a lot of relationships in order to provide more detail, this was done as well for all instances of Name as an attribute became FirstName and LastName for the same reasons. Additionally, a few more attributes were added to Contract and Employer to create more functional dependencies. Then from Milestones 2 to 4, the only thing that was changed in our schema was adding age to students to create more interesting queries for the application.

List of Queries:

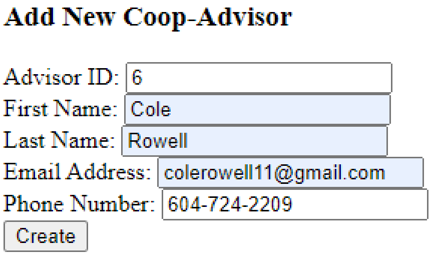
Queries used in the project can be found in the sections below as well as screenshots of sample outputs.

1. Insert
2. Delete
3. Update
4. Select
5. Projection
6. Join
7. Aggregation with Having
8. Aggregation with Group By
9. Nested Aggregation with Group By
10. Division

Insert Function:

```
</form>
<h3>Add New Coop-Advisor</h3>
<form method ="POST" action="index.php">
<input type="hidden" id ="insertCoopAdvisorRequest" name = "insertCoopAdvisorRequest">
  <label for="CoopAdvisorAdvisorID">Advisor ID:</label>
  <input
        type="number"
        id="CoopAdvisorAdvisorID"
        name="CoopAdvisorAdvisorID"
  /><br />

  <label for="CoopAdvisorFirstName">First Name:</label>
  <input
        type="text"
        id="CoopAdvisorFirstName"
        name="CoopAdvisorFirstName"
  /><br />

  <label for="CoopAdvisorLastName">Last Name:</label>
  <input
        type="text"
        id="CoopAdvisorLastName"
        name="CoopAdvisorLastName"
  /><br />

  <label for="CoopAdvisorEmailAddress">Email Address:</label>
  <input
        type="email"
        id="CoopAdvisorEmailAddress"
        name="CoopAdvisorEmailAddress"
  /><br />

  <label for="CoopAdvisorPhoneNumber">Phone Number:</label>
```

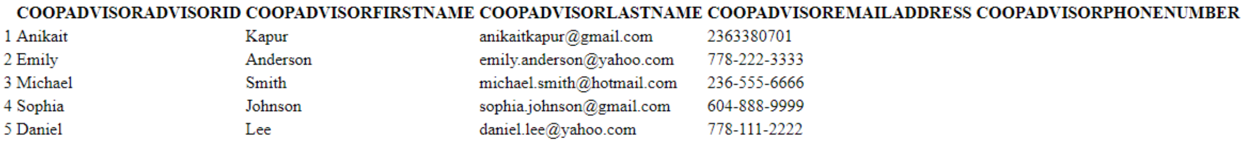
```
    <input
        type="tel"
        id="CoopAdvisorPhoneNumber"
        name="CoopAdvisorPhoneNumber"
    /><br />

    <input type="submit" value="Create" name="insertCoopAdvisor"></p>
</form>

function handleInsertRequestCoopAdvisor() {
        global $db_conn;

        // Getting the values from the user and insert data into the table
        $tuple = array (
            ":bind1" => $_POST['CoopAdvisorAdvisorID'],
            ":bind2" => $_POST['CoopAdvisorFirstName'],
            ":bind3" => $_POST['CoopAdvisorLastName'],
            ":bind4" => $_POST['CoopAdvisorEmailAddress'],
            ":bind5" => $_POST['CoopAdvisorPhoneNumber']
        );

        $alltuples = array (
            $tuple
        );

        executeBoundSQL("insert into CoopAdvisor values (:bind1, :bind2, :bind3, :bind4, :bind5)", $alltuples);
        OCICommit($db_conn);

        // Display the updated Co-op advisor table
        handleDisplayRequest();
    }
```



Add New Coop-Advisor

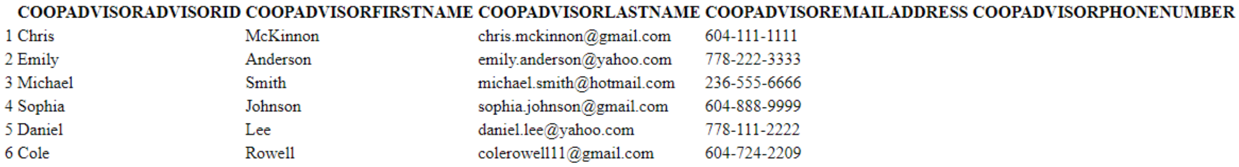
Advisor ID: 6
First Name: Cole
Last Name: Rowell
Email Address: colerowell11@gmail.com
Phone Number: 604-724-2209

Before:



```
  COOPADVISORADVISORID COOPADVISORFIRSTNAME COOPADVISORLASTNAME COOPADVISOREMAILADDRESS COOPADVISORPHONENUMBER
1 Anikait              Kapur                anikaitkapur@gmail.com    2363380701
2 Emily                Anderson             emily.anderson@yahoo.com  778-222-3333
3 Michael              Smith                michael.smith@hotmail.com 236-555-6666
4 Sophia               Johnson              sophia.johnson@gmail.com  604-888-9999
5 Daniel               Lee                  daniel.lee@yahoo.com      778-111-2222
```

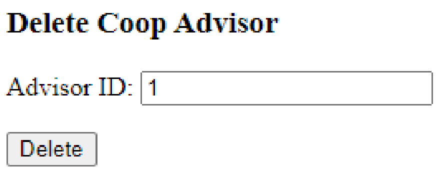
After:



```
  COOPADVISORADVISORID COOPADVISORFIRSTNAME COOPADVISORLASTNAME COOPADVISOREMAILADDRESS COOPADVISORPHONENUMBER
1 Chris                McKinnon             chris.mckinnon@gmail.com  604-111-1111
2 Emily                Anderson             emily.anderson@yahoo.com  778-222-3333
3 Michael              Smith                michael.smith@hotmail.com 236-555-6666
4 Sophia               Johnson              sophia.johnson@gmail.com  604-888-9999
5 Daniel               Lee                  daniel.lee@yahoo.com      778-111-2222
6 Cole                 Rowell               colerowell11@gmail.com    604-724-2209
```

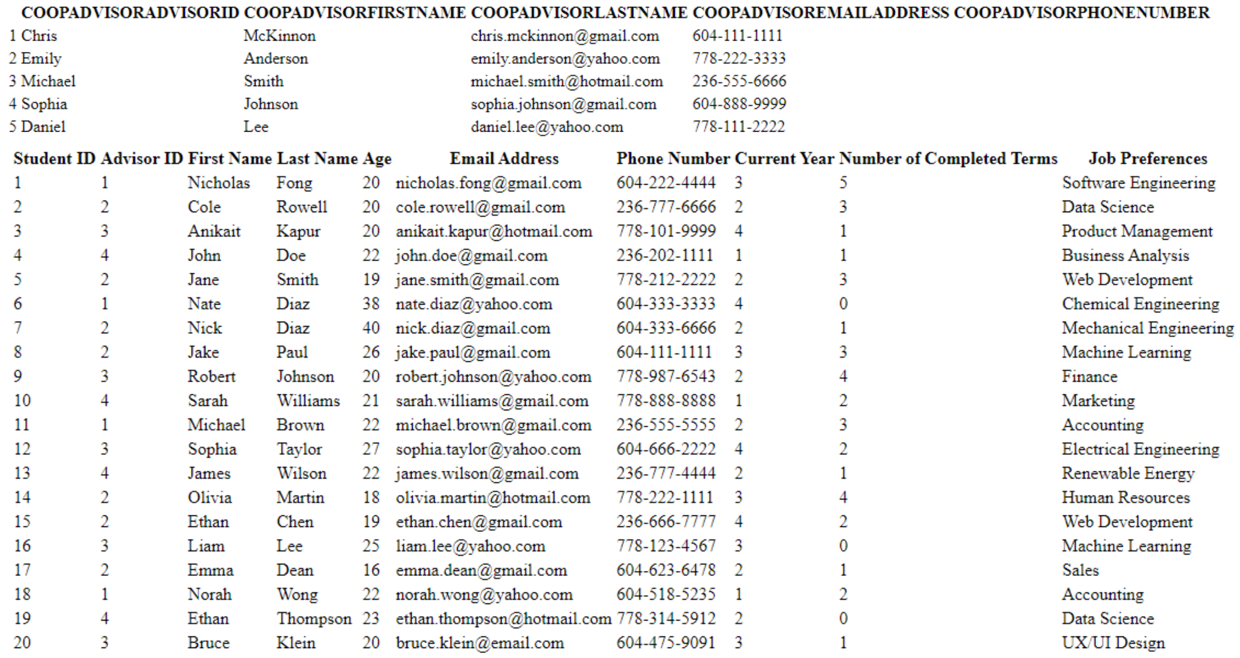
Delete Function:

```
<h3>Delete Coop Advisor</h3>
<form method ="POST" action="index.php">
   <input type="hidden" id ="deleteCoopAdvisorRequest" name = "deleteCoopAdvisorRequest">
   <label for="CoopAdvisorAdvisorID">Advisor ID:</label>
   <input
         type="number"
         id="CoopAdvisorAdvisorID"
         name="CoopAdvisorAdvisorID"
   /><br />
   <p><input type="submit" value="Delete" name="deleteCoopAdvisor"></p>
</form>

function handleDeleteRequestCoopAdvisor(){
         global $db_conn;

         $ID = $_POST['CoopAdvisorAdvisorID'];

         executePlainSQL("DELETE FROM CoopAdvisor WHERE CoopAdvisorAdvisorID = {$ID}");
         OCICommit($db_conn);

         handleDisplayRequest();
         handleStudentRequest();
      }
```



Delete Coop Advisor

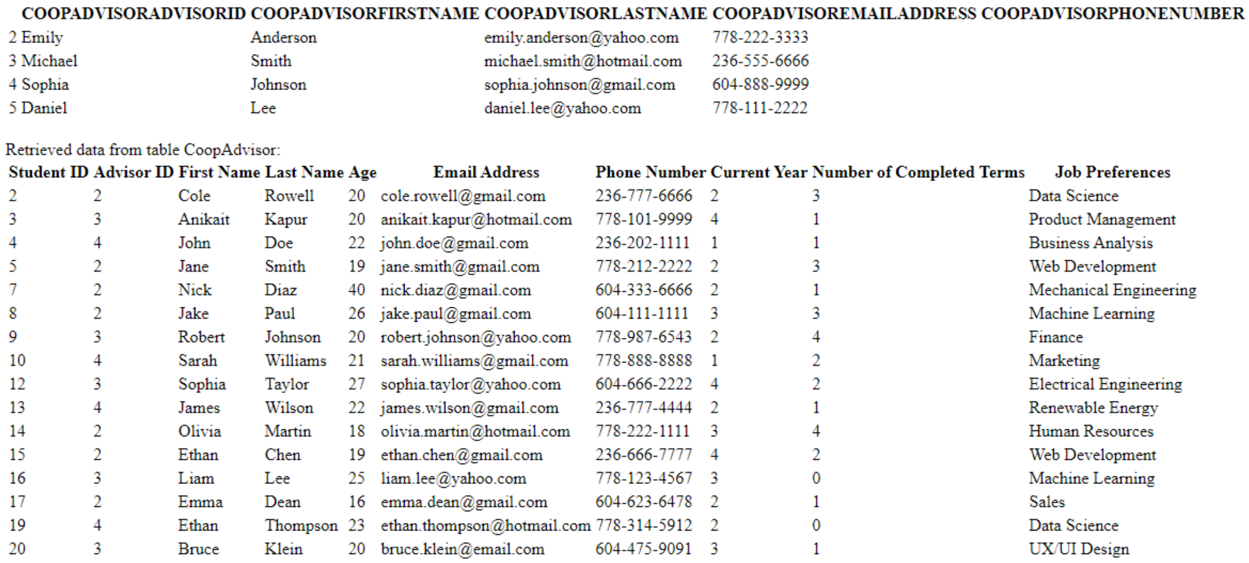
Before:



COOPADVISORADVISORID	COOPADVISORFIRSTNAME	COOPADVISORLASTNAME	COOPADVISOREMAILADDRESS	COOPADVISORPHONENUMBER
1	Chris	McKinnon	chris.mckinnon@gmail.com	604-111-1111
2	Emily	Anderson	emily.anderson@yahoo.com	778-222-3333
3	Michael	Smith	michael.smith@hotmail.com	236-555-6666
4	Sophia	Johnson	sophia.johnson@gmail.com	604-888-9999
5	Daniel	Lee	daniel.lee@yahoo.com	778-111-2222

Student ID	Advisor ID	First Name	Last Name	Age	Email Address	Phone Number	Current Year	Number of Completed Terms	Job Preferences
1	1	Nicholas	Fong	20	nicholas.fong@gmail.com	604-222-4444	3	5	Software Engineering
2	2	Cole	Rowell	20	cole.rowell@gmail.com	236-777-6666	2	3	Data Science
3	3	Anikait	Kapur	20	anikait.kapur@hotmail.com	778-101-9999	4	1	Product Management
4	4	John	Doe	22	john.doe@gmail.com	236-202-1111	1	1	Business Analysis
5	2	Jane	Smith	19	jane.smith@gmail.com	778-212-2222	2	3	Web Development
6	1	Nate	Diaz	38	nate.diaz@yahoo.com	604-333-3333	4	0	Chemical Engineering
7	2	Nick	Diaz	40	nick.diaz@gmail.com	604-333-6666	2	1	Mechanical Engineering
8	2	Jake	Paul	26	jake.paul@gmail.com	604-111-1111	3	3	Machine Learning
9	3	Robert	Johnson	20	robert.johnson@yahoo.com	778-987-6543	2	4	Finance
10	4	Sarah	Williams	21	sarah.williams@gmail.com	778-888-8888	1	2	Marketing
11	1	Michael	Brown	22	michael.brown@gmail.com	236-555-5555	2	3	Accounting
12	3	Sophia	Taylor	27	sophia.taylor@yahoo.com	604-666-2222	4	2	Electrical Engineering
13	4	James	Wilson	22	james.wilson@gmail.com	236-777-4444	2	1	Renewable Energy
14	2	Olivia	Martin	18	olivia.martin@hotmail.com	778-222-1111	3	4	Human Resources
15	2	Ethan	Chen	19	ethan.chen@gmail.com	236-666-7777	4	2	Web Development
16	3	Liam	Lee	25	liam.lee@yahoo.com	778-123-4567	3	0	Machine Learning
17	2	Emma	Dean	16	emma.dean@gmail.com	604-623-6478	2	1	Sales
18	1	Norah	Wong	22	norah.wong@yahoo.com	604-518-5235	1	2	Accounting
19	4	Ethan	Thompson	23	ethan.thompson@hotmail.com	778-314-5912	2	0	Data Science
20	3	Bruce	Klein	20	bruce.klein@email.com	604-475-9091	3	1	UX/UI Design

After:



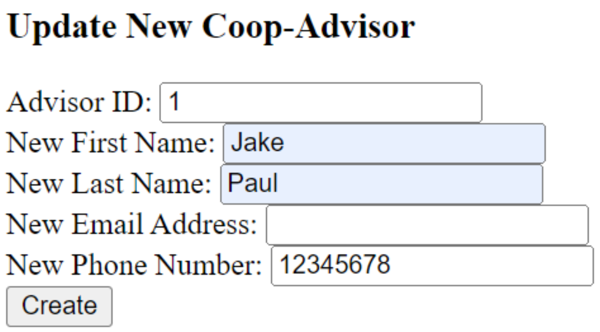
COOPADVISORADVISORID	COOPADVISORFIRSTNAME	COOPADVISORLASTNAME	COOPADVISOREMAILADDRESS	COOPADVISORPHONENUMBER
2 Emily	Anderson	emily.anderson@yahoo.com	778-222-3333	
3 Michael	Smith	michael.smith@hotmail.com	236-555-6666	
4 Sophia	Johnson	sophia.johnson@gmail.com	604-888-9999	
5 Daniel	Lee	daniel.lee@yahoo.com	778-111-2222	

Retrieved data from table CoopAdvisor:

Student ID	Advisor ID	First Name	Last Name	Age	Email Address	Phone Number	Current Year	Number of Completed Terms	Job Preferences
2	2	Cole	Rowell	20	cole.rowell@gmail.com	236-777-6666	2	3	Data Science
3	3	Anikait	Kapur	20	anikait.kapur@hotmail.com	778-101-9999	4	1	Product Management
4	4	John	Doe	22	john.doe@gmail.com	236-202-1111	1	1	Business Analysis
5	2	Jane	Smith	19	jane.smith@gmail.com	778-212-2222	2	3	Web Development
7	2	Nick	Diaz	40	nick.diaz@gmail.com	604-333-6666	2	1	Mechanical Engineering
8	2	Jake	Paul	26	jake.paul@gmail.com	604-111-1111	3	3	Machine Learning
9	3	Robert	Johnson	20	robert.johnson@yahoo.com	778-987-6543	2	4	Finance
10	4	Sarah	Williams	21	sarah.williams@gmail.com	778-888-8888	1	2	Marketing
12	3	Sophia	Taylor	27	sophia.taylor@yahoo.com	604-666-2222	4	2	Electrical Engineering
13	4	James	Wilson	22	james.wilson@gmail.com	236-777-4444	2	1	Renewable Energy
14	2	Olivia	Martin	18	olivia.martin@hotmail.com	778-222-1111	3	4	Human Resources
15	2	Ethan	Chen	19	ethan.chen@gmail.com	236-666-7777	4	2	Web Development
16	3	Liam	Lee	25	liam.lee@yahoo.com	778-123-4567	3	0	Machine Learning
17	2	Emma	Dean	16	emma.dean@gmail.com	604-623-6478	2	1	Sales
19	4	Ethan	Thompson	23	ethan.thompson@hotmail.com	778-314-5912	2	0	Data Science
20	3	Bruce	Klein	20	bruce.klein@email.com	604-475-9091	3	1	UX/UI Design

Update Function:

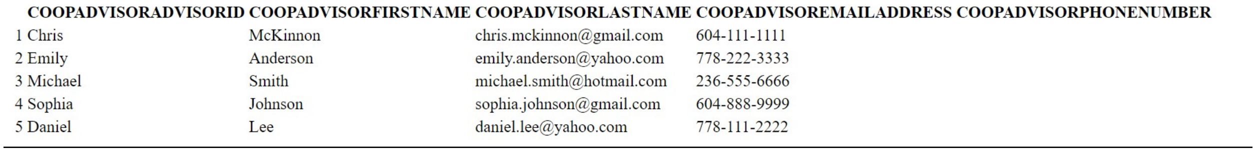
```
<h3>Update New Coop-Advisor</h3>
<form method ="POST" action="index.php">
<input type="hidden" id ="updateCoopAdvisorRequest" name = "updateCoopAdvisorRequest">
   <label for="CoopAdvisorAdvisorID">Advisor ID:</label>
   <input
         type="number"
         id="CoopAdvisorAdvisorID"
         name="CoopAdvisorAdvisorID"
   /><br />

   <label for="CoopAdvisorFirstName">New First Name:</label>
   <input
         type="text"
         id="CoopAdvisorFirstName"
         name="CoopAdvisorFirstName"
   /><br />

   <label for="CoopAdvisorLastName">New Last Name:</label>
   <input
         type="text"
         id="CoopAdvisorLastName"
         name="CoopAdvisorLastName"
   /><br />

   <label for="CoopAdvisorEmailAddress">New Email Address:</label>
   <input
         type="email"
         id="CoopAdvisorEmailAddress"
         name="CoopAdvisorEmailAddress"
   /><br />
   <label for="CoopAdvisorPhoneNumber">New Phone Number:</label>
   <input
         type="tel"
         id="CoopAdvisorPhoneNumber"
         name="CoopAdvisorPhoneNumber"
   /><br />


   <input type="submit" value="Create" name="updateCoopAdvisor"></p>
</form>

 function handleUpdateRequestCoopAdvisor() {
         global $db_conn;

         $newID = $_POST['CoopAdvisorAdvisorID'];
         $newFirstName = $_POST["CoopAdvisorFirstName"];
         $newLastName = $_POST["CoopAdvisorLastName"];
         $newEmail = $_POST["CoopAdvisorEmailAddress"];
         $newPhone = $_POST["CoopAdvisorPhoneNumber"];

         if ($newID == ""){
            echo "Please input Advisor ID";
         } else{
            if(empty($newFirstName)){
               $result = executePlainSQL("SELECT CoopAdvisorFirstName FROM CoopAdvisor WHERE
CoopAdvisorAdvisorID=" . $newID);
               $row = OCI_Fetch_Array($result, OCI_BOTH);
               $newFirstName = $row[0];
            }
             if (empty($newEmail)){
               $result = executePlainSQL("SELECT CoopAdvisorEmailAddress FROM CoopAdvisor WHERE
CoopAdvisorAdvisorID=" . $newID);
               $row = OCI_Fetch_Array($result, OCI_BOTH);
               $newEmail = $row[0];
            }
            if (empty($newLastName)){
               $result = executePlainSQL("SELECT CoopAdvisorLastName FROM CoopAdvisor WHERE
CoopAdvisorAdvisorID=" . $newID);
               $row = OCI_Fetch_Array($result, OCI_BOTH);
               $newLastName = $row[0];
            }
            if (empty($newPhone)){
               $result = executePlainSQL("SELECT CoopAdvisorPhoneNumber FROM CoopAdvisor WHERE
CoopAdvisorAdvisorID=" . $newID);
               $row = OCI_Fetch_Array($result, OCI_BOTH);
               $newPhone = $row[0];
            }
            // you need the wrap the old name and new name values with single quotations
            executePlainSQL("UPDATE CoopAdvisor SET CoopAdvisorFirstName='" .$newFirstName . "',
CoopAdvisorLastName='" .$newLastName ."', CoopAdvisorEmailAddress='" .$newEmail . "',
CoopAdvisorPhoneNumber='" . $newPhone ."' WHERE CoopAdvisorAdvisorID='" . $newID . "'");
            handleDisplayRequest();
         }
         OCICommit($db_conn);

     }
```



Update New Coop-Advisor

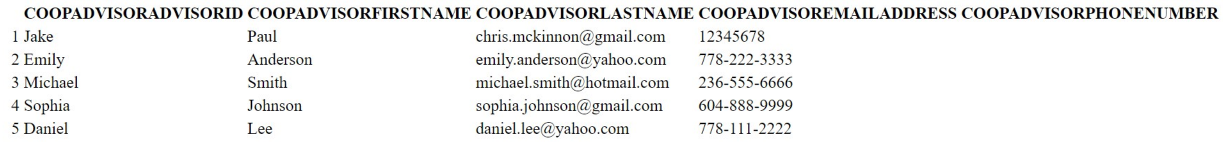
Advisor ID:	1
New First Name:	Jake
New Last Name:	Paul
New Email Address:	
New Phone Number:	12345678

Before:



COOPADVISORADVISORID	COOPADVISORFIRSTNAME	COOPADVISORLASTNAME	COOPADVISOREMAILADDRESS	COOPADVISORPHONENUMBER
1	Chris	McKinnon	chris.mckinnon@gmail.com	604-111-1111
2	Emily	Anderson	emily.anderson@yahoo.com	778-222-3333
3	Michael	Smith	michael.smith@hotmail.com	236-555-6666
4	Sophia	Johnson	sophia.johnson@gmail.com	604-888-9999
5	Daniel	Lee	daniel.lee@yahoo.com	778-111-2222

After:

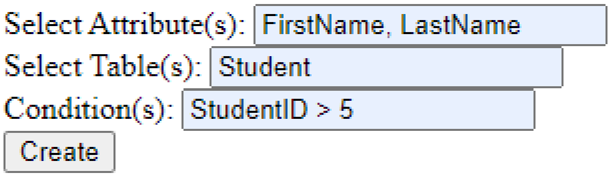


COOPADVISORADVISORID	COOPADVISORFIRSTNAME	COOPADVISORLASTNAME	COOPADVISOREMAILADDRESS	COOPADVISORPHONENUMBER
1	Jake	Paul	chris.mckinnon@gmail.com	12345678
2	Emily	Anderson	emily.anderson@yahoo.com	778-222-3333
3	Michael	Smith	michael.smith@hotmail.com	236-555-6666
4	Sophia	Johnson	sophia.johnson@gmail.com	604-888-9999
5	Daniel	Lee	daniel.lee@yahoo.com	778-111-2222

Selection Function:

```
<h3>Selection</h3>
<form method="GET" action="index.php">
<input type="hidden" id ="selectionRequest" name = "selectionRequest">
   <label for="selection">Select Attribute(s):</label>
   <input
         type="text"
         id="Attribute"
         name="Attribute"
   /><br />

   <label for="From">Select Table(s):</label>
   <input
         type="text"
         id="Table"
         name="Table"
   /><br />

   <label for="Where">Condition(s):</label>
   <input
         type="text"
         id="Condition"
         name="Condition"
   /><br />

   <input type="submit" value="Create" name="select"></p>
</form>
```

```
function handleSelectionRequest() {
        global $db_conn;
        $attribute = $_GET["Attribute"];
        $table = $_GET["Table"];
        $condition = $_GET["Condition"];

        if ($condition == "NONE") {
           $result = executePlainSQL("SELECT $attribute FROM $table");
        } else {
           $result = executePlainSQL("SELECT $attribute FROM $table WHERE $condition");
        }
        printResult($result);
    }
```

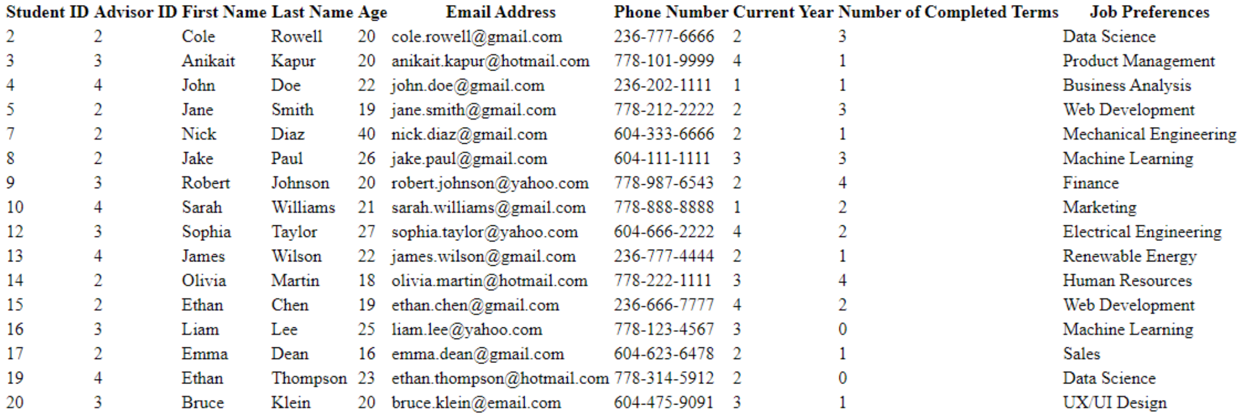


Select Attribute(s): FirstName, LastName

Select Table(s): Student

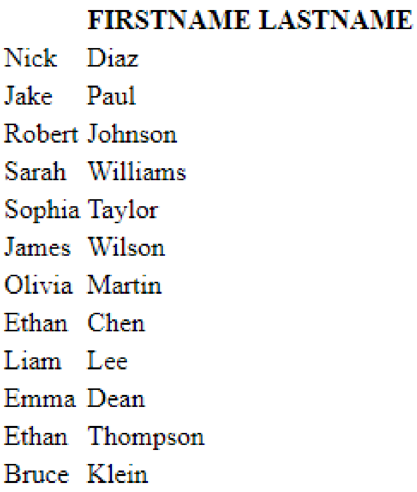
Condition(s): StudentID > 5

Before:



Student ID	Advisor ID	First Name	Last Name	Age	Email Address	Phone Number	Current Year	Number of Completed Terms	Job Preferences
2	2	Cole	Rowell	20	cole.rowell@gmail.com	236-777-6666	2	3	Data Science
3	3	Anikait	Kapur	20	anikait.kapur@hotmail.com	778-101-9999	4	1	Product Management
4	4	John	Doe	22	john.doe@gmail.com	236-202-1111	1	1	Business Analysis
5	2	Jane	Smith	19	jane.smith@gmail.com	778-212-2222	2	3	Web Development
7	2	Nick	Diaz	40	nick.diaz@gmail.com	604-333-6666	2	1	Mechanical Engineering
8	2	Jake	Paul	26	jake.paul@gmail.com	604-111-1111	3	3	Machine Learning
9	3	Robert	Johnson	20	robert.johnson@yahoo.com	778-987-6543	2	4	Finance
10	4	Sarah	Williams	21	sarah.williams@gmail.com	778-888-8888	1	2	Marketing
12	3	Sophia	Taylor	27	sophia.taylor@yahoo.com	604-666-2222	4	2	Electrical Engineering
13	4	James	Wilson	22	james.wilson@gmail.com	236-777-4444	2	1	Renewable Energy
14	2	Olivia	Martin	18	olivia.martin@hotmail.com	778-222-1111	3	4	Human Resources
15	2	Ethan	Chen	19	ethan.chen@gmail.com	236-666-7777	4	2	Web Development
16	3	Liam	Lee	25	liam.lee@yahoo.com	778-123-4567	3	0	Machine Learning
17	2	Emma	Dean	16	emma.dean@gmail.com	604-623-6478	2	1	Sales
19	4	Ethan	Thompson	23	ethan.thompson@hotmail.com	778-314-5912	2	0	Data Science
20	3	Bruce	Klein	20	bruce.klein@email.com	604-475-9091	3	1	UX/UI Design

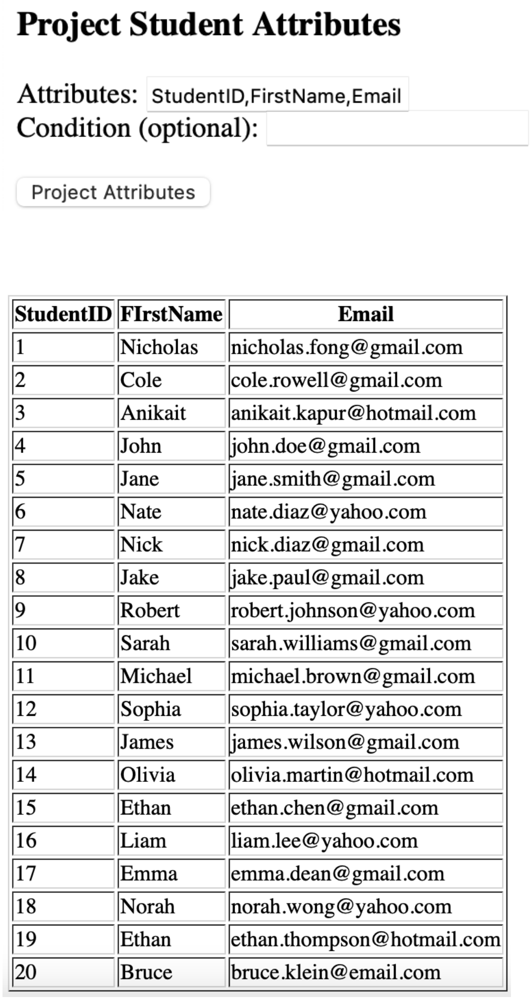
After:



FIRSTNAME	LASTNAME
Nick	Diaz
Jake	Paul
Robert	Johnson
Sarah	Williams
Sophia	Taylor
James	Wilson
Olivia	Martin
Ethan	Chen
Liam	Lee
Emma	Dean
Ethan	Thompson
Bruce	Klein

Projection Function:

```
<h3>Project Student Attributes</h3>
<form method="GET" action="index.php">
    <input type="hidden" name="studentProjectionRequest">
    <label for="attributes">Attributes:</label>
    <input type="text" id="attributes" name="attributes" required>
    <br>
    <label for="condition">Condition (optional):</label>
    <input type="text" id="condition" name="condition">
    <p><input type="submit" value="Project Attributes" name="projectAttributes"></p>
</form>

function handleProjectionRequest() {
            global $db_conn;

            if (isset($_GET['attributes'])) {
                $attributes = $_GET['attributes'];
                $attributesArray = explode(",", $attributes);
                $selectClause = implode(", ", $attributesArray);

                $whereClause = "";
                if (isset($_GET['condition']) && !empty($_GET['condition'])) {
                    $condition = $_GET['condition'];
                    $whereClause = " WHERE $condition";
                }

                $query = "SELECT $selectClause FROM Student $whereClause";

                $stmt = oci_parse($db_conn, $query);
                oci_execute($stmt);

                echo "<table border='1'>
                    <tr>";

                foreach ($attributesArray as $attribute) {
                    echo "<th>$attribute</th>";
                }

                echo "</tr>";

                while ($row = oci_fetch_assoc($stmt)) {
                    echo "<tr>";
                    foreach ($attributesArray as $attribute) {
                        echo "<td>" . $row[strtoupper($attribute)] . "</td>";
                    }
                    echo "</tr>";
                }

                echo "</table>";
            } else {
                echo "Please provide attributes to project.";
            }
        }
```



Project Student Attributes

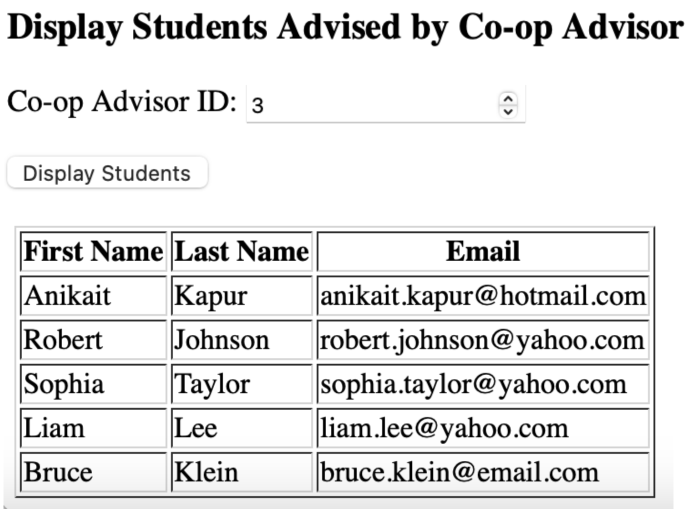
Attributes: StudentID,FirstName,Email
Condition (optional):

StudentID	FIrstName	Email
1	Nicholas	nicholas.fong@gmail.com
2	Cole	cole.rowell@gmail.com
3	Anikait	anikait.kapur@hotmail.com
4	John	john.doe@gmail.com
5	Jane	jane.smith@gmail.com
6	Nate	nate.diaz@yahoo.com
7	Nick	nick.diaz@gmail.com
8	Jake	jake.paul@gmail.com
9	Robert	robert.johnson@yahoo.com
10	Sarah	sarah.williams@gmail.com
11	Michael	michael.brown@gmail.com
12	Sophia	sophia.taylor@yahoo.com
13	James	james.wilson@gmail.com
14	Olivia	olivia.martin@hotmail.com
15	Ethan	ethan.chen@gmail.com
16	Liam	liam.lee@yahoo.com
17	Emma	emma.dean@gmail.com
18	Norah	norah.wong@yahoo.com
19	Ethan	ethan.thompson@hotmail.com
20	Bruce	bruce.klein@email.com

Join Function:

```
<h3>Display Students Advised by Co-op Advisor</h3>
<form method="GET" action="index.php">
   <input type="hidden" name="joinTableRequest">
   <label for="advisorID">Co-op Advisor ID:</label>
   <input type="number" id="advisorID" name="advisorID" required>
   <p><input type="submit" value="Display Students" name="displayStudents"></p>
</form>


function handlejoinTableRequest() {
        global $db_conn;

        if (isset($_GET['advisorID'])) {
           $advisorID = $_GET['advisorID'];

           $query = "SELECT s.FirstName, s.LastName, s.Email
```

```
                      FROM Student s
                      JOIN CoopAdvisor c ON s.AdvisorID = c.CoopAdvisorAdvisorID
                      WHERE c.CoopAdvisorAdvisorID = :advisorID";

            $stmt = oci_parse($db_conn, $query);
            oci_bind_by_name($stmt, ":advisorID", $advisorID);
            oci_execute($stmt);

            echo "<table border='1'>
            <tr>
                <th>First Name</th>
                <th>Last Name</th>
                <th>Email</th>
            </tr>";


            while ($row = oci_fetch_assoc($stmt)) {
                echo "<tr>";
                echo "<td>" . $row['FIRSTNAME'] . "</td>";
                echo "<td>" . $row['LASTNAME'] . "</td>";
                echo "<td>" . $row['EMAIL'] . "</td>";
                echo "</tr>";
            }

            echo "</table>";
        } else {
            echo "Please provide a valid CoopAdvisorAdvisorID.";
        }
    }
}
```



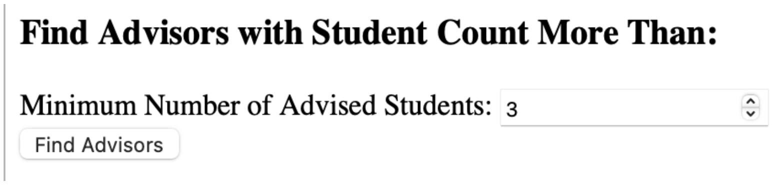
Display Students Advised by Co-op Advisor

Co-op Advisor ID: 3

First Name	Last Name	Email
Anikait	Kapur	anikait.kapur@hotmail.com
Robert	Johnson	robert.johnson@yahoo.com
Sophia	Taylor	sophia.taylor@yahoo.com
Liam	Lee	liam.lee@yahoo.com
Bruce	Klein	bruce.klein@email.com

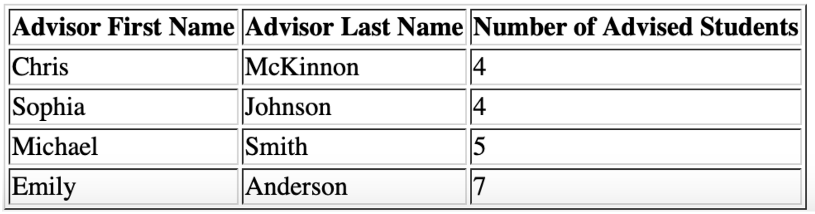
Aggregation with Having:

```
<h3>Find Advisors with Student Count More Than:</h3>
<form method="GET" action="index.php">
    <input type="hidden" name="advisorStudentCountThresholdRequest">
    <label for="numberofStudents">Minimum Number of Advised Students:</label>
    <input type="number" id="numberofStudents" name="numberofStudents" required><br>
    <input type="submit" value="Find Advisors" name="submit">
</form>

 function handleAdvisorStudentCountThresholdRequest() {
           global $db_conn;
             if (isset($_GET['advisorStudentCountThresholdRequest'])) {
              $numberofStudents = $_GET['numberofStudents'];



              $query = "SELECT ca.CoopAdvisorFirstName AS AdvisorFirstName,
                           ca.CoopAdvisorLastName AS AdvisorLastName,
                           COUNT(s.StudentID) AS TotalAdvisedStudents
                        FROM
                           CoopAdvisor ca
                        LEFT JOIN
                           Student s ON ca.CoopAdvisorAdvisorID = s.AdvisorID
                        GROUP BY
                           ca.CoopAdvisorAdvisorID, ca.CoopAdvisorFirstName, ca.CoopAdvisorLastName
                        HAVING
                           COUNT(s.StudentID) > :numberofStudents";

              $stmt = oci_parse($db_conn, $query);
              oci_bind_by_name($stmt, ":numberofStudents", $numberofStudents);
              oci_execute($stmt);

              echo "<table border='1'>
              <tr>
                 <th>Advisor First Name</th>
                 <th>Advisor Last Name</th>
                 <th>Number of Advised Students</th>
              </tr>";

              while ($row = oci_fetch_assoc($stmt)) {
                 echo "<tr><td>" . $row["ADVISORFIRSTNAME"] . "</td><td>" . $row["ADVISORLASTNAME"] .
"</td><td>" . $row["TOTALADVISEDSTUDENTS"] . "</td></tr>";
              }

              echo "</table>";
           }
       }
```



Find Advisors with Student Count More Than:

Minimum Number of Advised Students: 3



Advisor First Name	Advisor Last Name	Number of Advised Students
Chris	McKinnon	4
Sophia	Johnson	4
Michael	Smith	5
Emily	Anderson	7

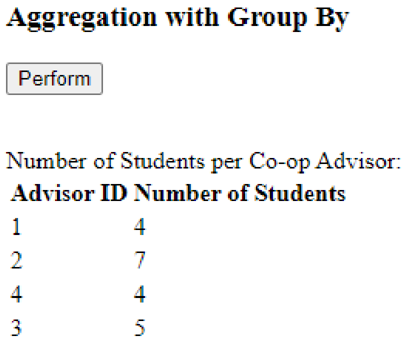
Aggregation with Group By

```
<h3>Aggregation with Group By</h3>
<form method ="GET" action="index.php">
   <input type="hidden" id ="aggregationWithGroupByRequest" name = "aggregationWithGroupByRequest">
   <p><input type="submit" value="Perform" name="aggregationWithGroupBy"></p>
</form>
```

```
function handleAggregationWithGroupBy(){
            global $db_conn;

            $result = executePlainSQL("SELECT AdvisorID, COUNT(*) FROM Student GROUP BY AdvisorID");
            printAggregationWithGroupByResult($result);
        }
```

```
function printAggregationWithGroupByResult($result) { //prints results from a select statement
            echo "<br>Number of Students per Co-op Advisor:<br>";
            echo "<table>";
            echo "<tr><th>Advisor ID</th><th>Number of Students</th></tr>";

            while ($row = OCI_Fetch_Array($result, OCI_BOTH)) {
                echo "<tr><td>" . $row[0] . "</td><td>" . $row[1] . "</td><td>";
            }

            echo "</table>";
        }
```

Output:



```
Aggregation with Group By

Perform

Number of Students per Co-op Advisor:
Advisor ID Number of Students
1          4
2          7
4          4
3          5
```

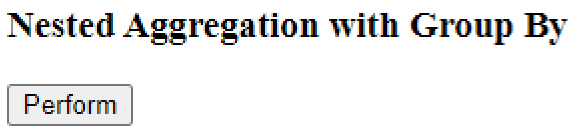
Nested Aggregation with Group By

```
<h3>Nested Aggregation with Group By</h3>
<form method ="GET" action="index.php">
   <input type="hidden" id ="nestedAggregationWithGroupByRequest" name =
"nestedAggregationWithGroupByRequest">
   <p><input type="submit" value="Perform" name="nestedAggregationWithGroupBy"></p>
</form>
```

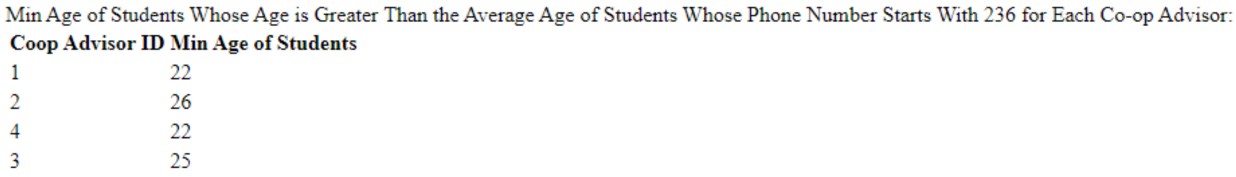
```
function handleNestedAggregationWithGroupBy(){
```

```
            global $db_conn;


            $result = executePlainSQL("SELECT S1.AdvisorID, MIN(Age)
            FROM Student S1
            WHERE S1.Age > (SELECT AVG(S.Age) FROM Student S WHERE PhoneNumber LIKE '236%')
            GROUP BY S1.AdvisorID");
            printNestedAggregationWithGroupByResult($result);
        }
```

```
function printNestedAggregationWithGroupByResult($result) { //prints results from a select statement
            echo "<br>Min Age of Students Whose Age is Greater Than the Average Age of Students Whose Phone
Number Starts With 236 for Each Co-op Advisor:<br>";
            echo "<table>";
            echo "<tr><th>Coop Advisor ID</th><th>Min Age of Students</th></tr>";

            while ($row = OCI_Fetch_Array($result, OCI_BOTH)) {
              //echo "<tr><td>" . $row["ID"] . "</td><td>" . $row["NAME"] . "</td></tr>"; //or just use "echo $row[0]"
              echo "<tr><td>" . $row[0] . "</td><td>" . $row[1] . "</td></tr>";
            }

            echo "</table>";
        }
```



Nested Aggregation with Group By

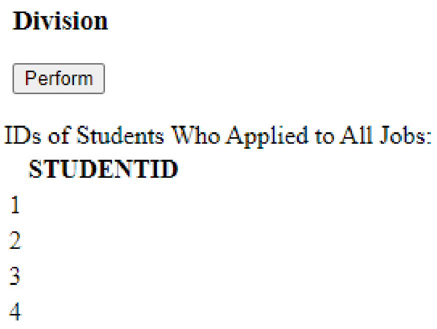


```
Min Age of Students Whose Age is Greater Than the Average Age of Students Whose Phone Number Starts With 236 for Each Co-op Advisor:
Coop Advisor ID Min Age of Students
1               22
2               26
4               22
3               25
```

Division

```
<h3>Division</h3>
<form method="GET" action="index.php">
  <input type="hidden" id ="divisionRequest" name = "divisionRequest">
  <p><input type="submit" value="Perform" name="division"></p>
</form>
```

```
function handleDivisionRequest(){
            global $db_conn;
            echo "<br>IDs of Students Who Applied to All Jobs:<br>";
            $result = executePlainSQL("SELECT StudentID
            FROM Student
            WHERE NOT EXISTS (
              SELECT JobID
              FROM Job
              MINUS
              SELECT JobID
              FROM JobApplication
              WHERE Student.StudentID = JobApplication.StudentID
            )");
            printResult(($result));
        }
```



Division

IDs of Students Who Applied to All Jobs:

STUDENTID
1
2
3
4