Lab 07 – Database Application Development

Objective:

In this lab students learn:

- How to connect to an Oracle server from a C++ program.
- How to write and execute SQL queries in a C++ program.

Individual Submission:

Your submission will be a single text-based .cpp file including your C++ program for the Database Application lab.

```
L07_ID_LASTNAME.cpp
```

Your submission needs to have your username and password in the .cpp file. **If any other username is used or left empty, will be marked '0'.**

Lab Requirements and Submission:

In this lab, you need to write a C++ program to execute the following queries and display the result returned by each query. For the output format, see the sample output.

1. Display Employee Number, First Name, Last Name, Phone Number, and Extension of all Employees who work in San Francisco. See the following Sample output. (Sort the report according to the employee number)

2. Display Employee Number, Last Name, Phone Number, and Extension for all managers. (You can use column reportsto to find the managers' employee number)

Your program will output only the following reports.

Remove any outputs (messages) from your program that you have used for testing you code such as printing "The connection is successful".

```
------------------------ Report 1 (Employee Report) ----------------------------------
Employee ID   First Name         Last Name          Phone              Extension
-----------  -----------------  -----------------  ---------------  ---------
1002          Diane              Murphy             +1 650 219 4782   x5800
1056          Mary               Patterson          +1 650 219 4782   x4611
1076          Jeff               Firrelli           +1 650 219 4782   x9273
1143          Anthony            Bow                +1 650 219 4782   x5428
1165          Leslie             Jennings           +1 650 219 4782   x3291
1166          Leslie             Thompson           +1 650 219 4782   x4065

------------------------ Report 2 (Manager Report) -----------------------------------
Employee ID   First Name         Last Name          Phone              Extension
-----------  -----------------  -----------------  ---------------  ---------
1002          Diane              Murphy             +1 650 219 4782   x5800
1056          Mary               Patterson          +1 650 219 4782   x4611
1088          William            Patterson          +61 2 9264 2451   x4871
1102          Gerard             Bondur             +33 14 723 4404   x5408
1143          Anthony            Bow                +1 650 219 4782   x5428
1621          Mami               Nishi              +81 33 224 5000   x101
```

Lab Instruction:

```
#include <iostream>
#include <occi.h>

using oracle::occi::Environment;
using oracle::occi::Connection;
using namespace oracle::occi;
using namespace std;

int main(void) {
	Environment* env = nullptr;
	Connection* conn = nullptr;
	Statement* stmt = nullptr;
	ResultSet* rs = nullptr;
	int num;
	string str;
	string user = "";
	string pass = " ";
		string constr = "myoracle12c.senecacollege.ca:1521/oracle12c";

	try {
		env = Environment::createEnvironment(Environment::DEFAULT);
		conn = env->createConnection(user, pass, constr);


		Statement* stmt = conn->createStatement();

		ResultSet* rs = stmt->executeQuery("SELECT employeenumber, firstname,
			lastname , phone, extension FROM retailemployees, retailoffices
			WHERE retailemployees.officecode = retailoffices.officecode and
```

```
                    LOWER(city) = 'san francisco' ORDER by employeenumber");

             cout << "-------------------------Report 1 (Employee Report)----------
--------------------" << endl;
             cout << "Employee ID First Name Last Name Phone Extension" << endl;
             cout << " ---------- ---------- --------- ----- ---------" << endl;

             while (rs->next()) {
                    int employeenumber = rs->getInt(1);
                    string fname = rs->getString(2);
                    string lname = rs->getString(3);
                    string phone = rs->getString(4);
                    string extension = rs->getString(5);

                    cout << " ";
                    cout.width(12);
                    cout.setf(ios::left);
                    cout << employeenumber;
                    cout << " ";
                    cout.width(14);
                    cout.setf(ios::left);
                    cout << fname;
                    cout.width(8);
                    cout << lname;
                    cout.setf(ios::left);
                    cout.width(4);
                    cout << "";
                    cout << phone;
                    cout.width(8);
                    cout << " ";
                    cout << extension << endl;
             } stmt = conn->createStatement();

             rs = stmt->executeQuery("SELECT DISTINCT(M.employeenumber),
M.firstname,M.lastname, RO.phone, M.extension FROM
                    retailemployees M LEFT OUTER JOIN retailoffices RO ON
M.officecode = RO.officecode JOIN retailemployees
                    EMP ON M.employeenumber = EMP.reportsto WHERE EMP.reportsto is
not NULL
                    ORDER BY M.employeenumber");
                    cout << "\n-------------------------Report 2 (Manager Report) -
----------------------------" << endl;
             cout << "Employee ID First Name Last Name Phone Extension" << endl;
             cout << " ---------- ---------- --------- ----- ---------" << endl;

             while (rs->next()) {
                    int employeenumber = rs->getInt(1);
                    string fname = rs->getString(2);
                    string lname = rs->getString(3);
                    string phone = rs->getString(4);
                    string extension = rs->getString(5);

                    cout << " "; cout.width(12);
                    cout.setf(ios::left);
                    cout << employeenumber;
                    cout << " ";
                    cout.width(14);
                    cout.setf(ios::left);
```

```
                        cout << fname;
                        cout.width(8);
                        cout << lname;
                        cout.setf(ios::left);
                        cout.width(4);
                        cout << "";
                        cout << phone;
                        cout.width(8);
                        cout << " ";
                        cout << extension << endl;
                } env->terminateConnection(conn);
                Environment::terminateEnvironment(env);
        }
        catch (SQLException& sqlExcp) {
                cout << "error";
                cout << sqlExcp.getErrorCode() << ": " << sqlExcp.getMessage();
        }
        return 0;
}
```