DESIGN

The program works by first creating the connection to the required server. It then builds the necessary query string. This is done early in the program to prevent the input and output loop from recreating it each time. The query works by creating a subquery to generate the aggregate values for the enrollment table. Now that the query has the number of students, the average, and the total marks (necessary for course average) it is joined with the class, course, and professor tables. During this join the professor table is filtered by department. This massive table is then grouped by course and year (found by taking a substring of the term to get the aggregate enrollment, section count, and overall average. This is then sorted properly and the correct columns are returned with names for referencing. The program then enters a loop where it reads in data from the user, sets that data in the prepared statement, and prints the output. When the user enters "exit" the loop is broken and everything is closed allowing the program to end safely.

EXECUTION

The execution of the program requires that the two necessary jars are present in the same directory as the executable. Then run

javac CourseInfo.java; java -cp ".:*" CourseInfo

JUSTIFICATION
Efficiency is achieved by preparing the sql query at the start of the program and simply setting the necessary values when they are passed in from the user. This prevents the recreation of this statement each pass through the loop thus speeding up the response of the program. This values returned are correct for the following reasons:

C#: simple table look
Name: value comes from joining the course table with the class table based on equal cno's so the name must match the class
Enrollment: this value comes from summing the number of marks in the enrollment table for that class in that term per section then summing the enrollment for each section in a given year resulting in the total number of students for the course in that year.
#Section: this is found by counting the number of entries in the grouping of enrollment by class number, term, and section for a given year resulting in the number of sections for a year.
Course Ave: This value is found by getting the total of all marks in a section then summing across all sections in a year. This total of all marks for the course in a year is then divided by the total enrollment for a course and year. The result should be the average across all classes that year.
Max Average and Min Average: These are found by taking the max/min of the averages for all sections in a year.

SOURCE CODE

```
import java.util.Properties;
import java.util.Scanner;
import java.sql.*;

public class CourseInfo {
    public static void main(String[] args){
        Connection con = null;
        PreparedStatement stmt = null;
        ResultSet rs = null;
```

```
        try{
            Class.forName("com.ibm.db2.jcc.DB2Driver");
            Properties props = new Properties() {
                {
                    put("currentSchema", "ENROLLMENT");
                    put("user", "db2guest");
                    put("password", "upKellynoisylair");
                }
            };

            con = DriverManager.getConnection("jdbc:db2://linux.student.cs.uwaterloo.ca:50002/cs348",
props);
        } catch (Exception e){
            System.out.println(e);
            System.exit(-1);
        }

        String query = ""
        + "SELECT\n"
            + "class.cno,\n"
            + "course.cname,\n"
            + "SUM(enrollmentData.students) as students,\n"
            + "COUNT(class.section) sections\n,"
            + "SUM(enrollmentData.total)/SUM(enrollmentData.students) as average,\n"
            + "MAX(enrollmentData.average) as maxave,\n"
            + "MIN(enrollmentData.average) as minave\n"
        + "FROM class\n"
        + "INNER JOIN professor\n"
        + "ON class.instructor = professor.eid AND professor.dept = ?\n"
        + "INNER JOIN course\n"
        + "ON class.cno = course.cno\n"
        + "INNER JOIN (\n"
            + "SELECT\n"
                + "enrollment.cno,\n"
                + "enrollment.term,\n"
                + "enrollment.section,\n"
                + "AVG(enrollment.mark) as average,\n"
                + "SUM(enrollment.mark) as total,\n"
                + "COUNT(enrollment.mark) as students\n"
            + "FROM enrollment\n"
            + "GROUP BY\n"
                + "enrollment.cno,\n"
                + "enrollment.term,\n"
                + "enrollment.section)\n"
            + "AS enrollmentData\n"
        + "ON class.cno = enrollmentData.cno AND class.term = enrollmentData.term AND class.section =
enrollmentData.section\n"
        + "WHERE ? <= SUBSTR(class.term, 2) AND ? >= SUBSTR(class.term, 2)\n"
        + "GROUP BY\n"
            + "class.cno,\n"
            + "course.cname,\n"
            + "SUBSTR(class.term, 2)\n"
        + "ORDER BY\n"
            + "SUBSTR(class.term, 2),\n"
            + "class.cno\n"
        ;

        try{
            stmt = con.prepareStatement(query);
        } catch (SQLException e){
            System.out.println(e);
            System.exit(-1);
        }

        while(true){
            Scanner reader = new Scanner(System.in);
            System.out.println("Enter Department ...");
            String dept = reader.nextLine();
            if(dept.equals("exit")){
                break;
            }
            System.out.println("Enter Start ...");
```

```
            int start = reader.nextInt();
            start -= 1900;
            System.out.println("Enter End ...");
            int end = reader.nextInt();
            end -= 1900;

            System.out.println("Year " + start);
            System.out.printf("%-7s %-20s %-10s %-10s %-10s %-12s %-12s\n", "C#", "Name", "Enrollment",
"#Section", "Course Ave", "Max Class Ave", "Min Class Ave");

            try{
                stmt.setString(1, dept);
                stmt.setInt(2, start);
                stmt.setInt(3, end);
                rs = stmt.executeQuery();

                while (rs.next()){
                    String cno = rs.getString("CNO");
                    String name = rs.getString("CNAME");
                    name = (name.length() < 16) ? name : name.substring(0, 16) + " ...";
                    String enrollment = rs.getString("STUDENTS");
                    String section = rs.getString("SECTIONS");
                    String average = rs.getString("AVERAGE");
                    String max = rs.getString("MAXAVE");
                    String min = rs.getString("MINAVE");
                    System.out.printf("%-7s %-20s %-10s %-10s %-10s %-12s %-12s\n", cno, name,
enrollment, section, average, max, min);
                }


                System.out.println("Year " + end);
            } catch (SQLException e){
                System.out.println(e);
                System.exit(-1);
            }
        }
        try{
            stmt.close();
            con.close();
        } catch (SQLException e){
            System.out.println(e);
            System.exit(-1);
        }

    }
}
```

OUTPUT

```
Enter Department ...
CS
Enter Start ...
1989
Enter End ...
1992
Year 89
C#      Name                 Enrollment #Section   Course Ave Max Class Ave Min
Class Ave
CS134   Principles of Co ... 46         1          82         82           82
CS240   Data Structures  ... 44         1          72         72           72
CS246   Software Abstrac ... 44         1          67         67           67
CS342   Concurrent Progr ... 71         1          61         61           61
CS134   Principles of Co ... 170        2          70         75           62
CS240   Data Structures  ... 148        3          62         66           61
CS241   Foundation of Se ... 148        3          73         76           64
CS246   Software Abstrac ... 148        2          67         70           60
```

```
CS134   Principles of Co ... 79         4          64         68          55
CS240   Data Structures  ... 134        2          59         65          53
CS241   Foundation of Se ... 147        3          73         79          68
CS246   Software Abstrac ... 134        2          68         72          63
CS342   Concurrent Progr ... 148        1          72         72          72
CS134   Principles of Co ... 25         1          63         63          63
CS240   Data Structures  ... 13         1          69         69          69
CS246   Software Abstrac ... 13         1          42         42          42
Year 92
Enter Department ...
CS
Enter Start ...
1990
Enter End ...
1993
Year 90
C#      Name                Enrollment #Section   Course Ave Max Class Ave Min
Class Ave
CS134   Principles of Co ... 170        2          70         75          62
CS240   Data Structures  ... 148        3          62         66          61
CS241   Foundation of Se ... 148        3          73         76          64
CS246   Software Abstrac ... 148        2          67         70          60
CS134   Principles of Co ... 79         4          64         68          55
CS240   Data Structures  ... 134        2          59         65          53
CS241   Foundation of Se ... 147        3          73         79          68
CS246   Software Abstrac ... 134        2          68         72          63
CS342   Concurrent Progr ... 148        1          72         72          72
CS134   Principles of Co ... 25         1          63         63          63
CS240   Data Structures  ... 13         1          69         69          69
CS246   Software Abstrac ... 13         1          42         42          42
CS240   Data Structures  ... 21         1          75         75          75
CS241   Foundation of Se ... 21         1          72         72          72
CS246   Software Abstrac ... 21         1          42         42          42
Year 93
Enter Department ...
CS
Enter Start ...
1991
Enter End ...
1995
Year 91
C#      Name                Enrollment #Section   Course Ave Max Class Ave Min
Class Ave
CS134   Principles of Co ... 79         4          64         68          55
CS240   Data Structures  ... 134        2          59         65          53
CS241   Foundation of Se ... 147        3          73         79          68
CS246   Software Abstrac ... 134        2          68         72          63
CS342   Concurrent Progr ... 148        1          72         72          72
CS134   Principles of Co ... 25         1          63         63          63
CS240   Data Structures  ... 13         1          69         69          69
CS246   Software Abstrac ... 13         1          42         42          42
CS240   Data Structures  ... 21         1          75         75          75
CS241   Foundation of Se ... 21         1          72         72          72
CS246   Software Abstrac ... 21         1          42         42          42
CS342   Concurrent Progr ... 26         1          63         63          63
CS348   Introduction to  ... 48         2          75         76          74
CS354   Operating System ... 26         1          76         76          76
CS348   Introduction to  ... 79         2          76         77          74
```

```
Year 95
Enter Department ...
exit
```