# 2023 Lancaster University

# SCC 201 Lab Week 6

**Instructions:**

This week you will work on Schema Refinement, Normal Forms, JDBC and SQL.

## TASK1: Schema refinement and normal forms

1) Please find the FDs for the given relational schema using Armstrong's Axiom.
2) Please find the Normalisation Level
3) Please convert the relational schema into 3NF if it's not in 3NF.

| ID | TYPE | SALARY | ADDRESS | WEIGHT |
|---|---|---|---|---|
| 123 | Light | 11,230 | LA1 1WW | 40 |
| 124 | Light | 12,300 | LA1 1TY | 40 |
| 125 | Heavy | 9,000 | LE3 7GN | 100 |
| 126 | Heavy | 9,000 | LA1 1WW | 100 |

## TASK2: Schema refinement and normal forms

1) Let R be S(A,B,C,D,E,F,G), also let R has a set of FDs {A->B, B->C, C->DEF}. Then, prove that AG is a key for R, decompose the relation and bring the Normalisation level to 3NF.

**TASK3: JDBC**

1) Open Text Editor on your computer.

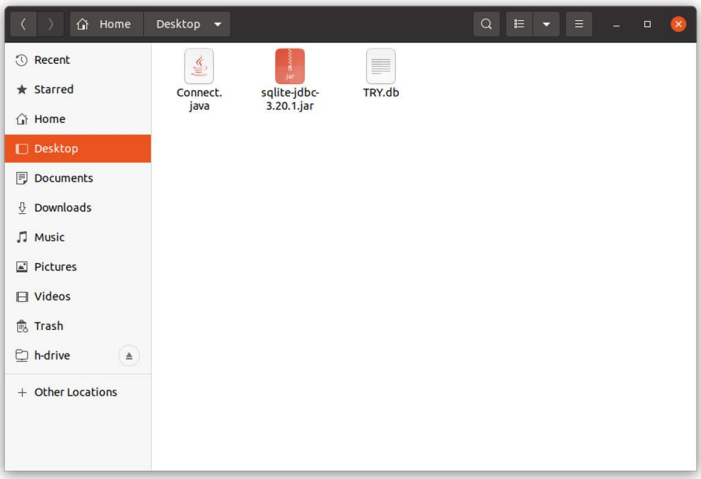
2) Type the following, and save the file as Connect.java.

```java
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
public class Connect {
    /**
     * Connect to a sample database a
     */
    public Connection  connect() {
        Connection conn = null;

        try {
            // db parameters
            String url = "jdbc:sqlite:TRY.db";
            // create a connection to the database
            conn = DriverManager.getConnection(url);
            System.out.println("Connection to SQLite has been established.");

        } catch (SQLException e) {
            System.out.println(e.getMessage());
        } finally {
            try {
                if (conn != null) {
                    conn.close();
                }
            } catch (SQLException ex) {
                System.out.println(ex.getMessage());
            }
        }
        return conn;
    }
    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        Connect myC = new Connect();
        Connection c = myC.connect();
    }
}
```

3) Download SQLite Driver 3.20.1 from moodle.

4) Download TRY.db from moodle.

5) Place these three files under the same directory e.g.:




6) Open the terminal and proceed to the folder.

7) Type **java -classpath ".:sqlite-jdbc-3.20.1.jar" Connect.java**

and press enter.

8) Observe


**TASK4: JDBC**

In this task, you will create a JAVA application that:

1) Receives a DATABASE name from a user.
2) Creates the DATABASE using the JDBC-SQLITE driver **(Please see slide #121 from Week 5 Material, there is an example!)**
3) Receives a table name from a user.
4) Creates the table with three attributes <STID INTEGER, STAGE INTEGER, STNAME VARCHAR(10) and STAGE is the PRIMARY KEY>.
5) And closes.

**TASK5:Write the SQL queries for the given Relational Schema**

**In the below ESSN is an SSN, PNO is a PNUMBER, DNO is a DNUMBER**

EMPLOYEE(NAME, SSN, BDATE, ADDRESS, SALARY, SUPERSSN, *DNO*)

DEPARTMENT(DNAME, DNUMBER, *MGRSSN*, MGRSTARTDATE)

DEPT_LOCATIONS(*DNUMBER*, DLOCATION)

PROJECT(PNAME, PNUMBER, PLOCATION, *DNUM*)

WORKSON(*ESSN, PNO*, HOURS)

DEPENDENT(*ESSN*, DEPENDENT_NAME, SEX, BDATE, RELATIONSHIP)

Q1) Retrieve the rows from the employee.

Q2) Retrieve salaries of employees who are living in the address “Lancaster.”

Q3) Retrieve the names of employees working in a department located in “London.”

Q4) Retrieve the names of employees who work on all projects.

Q5) Retrieve the names of the Projects located in “Kahramanmaras”.

Q6) Retrieve the Department Name whose manager’s name is “Sofiia” or “Ryan”.

Q7) Retrieve the Employee Name whose Dependents name is “Zsofia” or “Denver”.

**TASK6 (Challenging queries. Provided for your future reference.. Do not expect such questions in the exam!):**

**Write the SQL queries for the given Relational Schema**

**In the below ESSN is an SSN, PNO is a PNUMBER, DNO is a DNUMBER**

EMPLOYEE(NAME, SSN, BDATE, ADDRESS, SALARY, SUPERSSN, *DNO*)

DEPARTMENT(DNAME, DNUMBER, *MGRSSN*, MGRSTARTDATE)

DEPT_LOCATIONS(*DNUMBER*, DLOCATION)

PROJECT(PNAME, PNUMBER, PLOCATION, *DNUM*)

WORKSON(*ESSN, PNO*, HOURS)

DEPENDENT(*ESSN*, DEPENDENT_NAME, SEX, BDATE, RELATIONSHIP)

Q1: List the names of managers with at least one dependent

Q2: List the names of employees who do not work on a project controlled by department no 5.

Q3: Find the SUM of the salaries of all employees, the max salary, min salary and average salary.

Q4: Find the SUM of the salaries of all employees, the max salary, min salary and average salary for research department.

Q5: Find the number of employees in research department.

Q6: Find the number of distinct salary values for Employees.

Q7: List the names of employees who do not work on all projects controlled by department no 5.

Q8: List the names of employees who do not have supervisors (IS NULL checks for null values!)

Q9: Display the names of the employees who do not practice birth control (more than 5 children).

Q10: For each department, retrieve the department number, number of employees in that department and the average salary.

Q11: For each project, retrieve the project number, the project name, and the number of employees who work for that project.

Q12: For each project, retrieve the project number, the project name, and the number of employees who work for that project.

Q13: For each project on which more than two employees work, retrieve the project number, the project name, and the number of employees who work on the project.

Q15: For each department with more than five employees, retrieve the department number and the number of its employees who are making more than 40000.