

# CS3431: Project Description

## Building a Database Application

### Phase 2: SQL Design and Queries

**Release Date:** Jan. 29, 2019.

**Due Date:** Feb. 8, 2019 (11:59PM - Midnight).

**Teams:** The project is done in teams of two.

#### Description:

In this phase you are required to convert the database model you created for a **Hospital System** to a working relational database using Oracle system.

#### Instructions to access Oracle (the pdf posted in Canvas System)

##### CS3431 Database Systems I Oracle Development Environment

1. Windows PCs: Download the programs putty and pscp to your computer from the following link:  
<http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>.  
  
MACs: Instead of putty, you will use the ssh command in a Terminal window (program is location in the Utilities folder in the Applications folder on your hard drive)
2. Windows PCs: Use putty to connect to the server  
Server: **ccc.wpi.edu** (Nothing else needs to be entered to connect)  
  
Macs: Open a terminal window and run the following command to connect to the server  
**ssh ccc.wpi.edu**
3. Login to your account  
  
username: <your WPI username>  
Passwd: <regular WPI passwd>
4. After the login is successful, execute the following command to setup the correct environment for the database.  
**source /cs/bin/oracle-setup**  
  
*It will print out a message "Setup done. You should be able to connect to the Oracle DB ..."*
5. At the prompt, execute the command below to start up Oracle  
**sqlplus username@CS** (**CS** is capital letters)(**NOT @cs.wpi.edu!**) (Enter your own username)  
*Your Database password will be your username capitalized (all caps).*  
  
*Now you are connected and should have the SQL prompt. You are ready to create tables and use your database.*
6. After the first time you login, change your Oracle SQL password by using the following command (don't forget the semicolon at the end!). Use your own username and password in the command.  
  
**alter user username identified by new\_password;**

**\*\*\*\* Important Note – Use the Screen Command! \*\*\*\***

- If you follow the steps above, and then for any reason your terminal computer crashes or loses connection, your DB connection will hang. **This will require the Admin team in CS to reset your connection. This can be slow and may delay your progress.**
- To avoid that, use the **screen** unix command before connecting to the database, i.e., before issuing “**sqlplus ....**” Command.
- To learn more about the **screen** command, check this link: <https://kb.iu.edu/d/acuy> (Or generally search in Google).

**Copying files between your local hard drive and your CCC account**

Windows: open a command window and use pscp

<ftp://ftp.gnome.org/pub/putty/putty-0.56/html/doc/Chapter5.html>

Macs: open a terminal window and use scp

[https://en.wikipedia.org/wiki/Secure\\_copy](https://en.wikipedia.org/wiki/Secure_copy)

Copying from your computer to the CCC account:

**pscp** filename username@ccc.wpi.edu:/home/username (use scp if using a Mac)

Copying to your computer from the CCC account:

**pscp** username@ccc.wpi.edu:/home/username/filename . (use scp if using a Mac)

Substitute your actual username and filename in the above commands.

---

**Important Note**

Make sure to store the statements you write, e.g., CREATE TABLE statements, or SELECT queries in a file because you may need to re-use them multiple times. (See “**Deliverables**” section below).

### **Part 1 (25 Points): Create the database**

Given the ER diagram and the initial relational model that you created in Phase 1, you are required to do the following:

- Build the database in Oracle that reflects the given requirements in Phase 1. When converting from the ERD to the relational tables, you must follow the rules given in class to create a good database design.
- The database you will create must include:
  - All tables with descriptive names and fields.
  - For the fields, choose the appropriate data types
  - Add the appropriate constraints, which must include (you may add more):
    - Primary Keys and Unique constraints
    - Foreign Key constraints
    - Not Null constraints
    - Domain constraints

***Note: It is up to each team, to either keep using your own ER design from Phase 1, or you may use the design given in the posted solution of Phase 1. Both are accepted as long as your design is correct and captures all the requirements. Therefore, the solution to the queries (Part 2 below) may differ based on your design.***

## **Part 2 (60 Points—5 points each): SQL Queries**

Write SQL queries against the tables you created above to answer the following queries:

Q1: Report the hospital rooms (the room number) that are currently occupied.

Q2: For a given division manager (say, ID = 10), report all regular employees that are supervised by this manager. Display the employees ID, names, and salary.

Q3: For each patient, report the sum of amounts paid by the insurance company for that patient, i.e., report the patients SSN, and the sum of insurance payments over all visits.

**Note: If you keep the insurance coverage as a percentage, then compute this percentage before getting the sum.**

Q4: Report the number of visits done for each patient, i.e., for each patient, report the patient SSN, first and last names, and the count of visits done by this patient.

Q5: Report the room number that has an equipment unit with serial number 'A01-02X'.

Q6: Report the employee who has access to the largest number of rooms. We need the employee ID, and the number of rooms (s)he can access.

**Note: If there are several employees with the same maximum number, then report all of these employees.**

Q7: Report the number of regular employees, division managers, and general managers in the hospital. The output should look like:

Type	Count
-----	
Regular employees	10
Division managers	5
General managers	2

Q8: For patients who have a scheduled future visit (which is part of their most recent visit), report that patient (SSN, and first and last names) and the visit date. Do not report patients who do not have scheduled visit.

Q9: For each equipment type that has more than 3 units, report the equipment type ID, model, and the number of units this type has.

Q10: Report the date of the coming future visit for patient with SSN = 111-22-3333.

**Note: This date should exist in the last (most recent) visit of that patient.**

Q11: For patient with SSN = 111-22-3333, report the doctors (only ID) who have examined this patient more than 2 times.

Q12: Report the equipment types (only the ID) for which the hospital has purchased equipments (units) in both 2010 and 2011. Do not report duplication.

### **Part 3 (15 Points): Populate the Database**

Insert records in the database such that:

1. All the constraints that you defined in Part 1 will be obeyed
2. Each of the above queries must return some records in their results
3. Insert information for at least:
  - a. 10 Patients
  - b. 10 Doctors
  - c. 10 Rooms, at least 3 of these rooms have 2 or more services
  - d. 3 Equipment types
  - e. 3 Equipment units of each type
  - f. At least 5 patients have 2 or more admissions (visits)
  - g. 10 regular employees, 4 division managers, and 2 general managers

**Test your queries against the data that you will insert and make sure they return the expected results.**

### **Grading:**

The maximum grade is 100 Points. No late submissions.

### **Deliverables:**

Each team should deliver two files as follows:

- 1) Text file (.sql) that contains all SQL commands from Part1, Part3, and Part2 above (Put the insert statements before the queries). The file must be executable from SQL using command:  
SQL > @<fileName>

The file must run correctly, which means:

- a) Dropping all tables first using DROP TABLE commands.
- b) Creating all tables in the right order of foreign keys-Primary keys.
- c) Inserting data in all tables in the right order of foreign keys-Primary keys.
- d) Finally, executing queries.

If the file's syntax was not correct (and the file did not run), you will loose 20 points in addition to any other deductions.

Above each SQL query, also write the query in text enclosed between `/* */`. In this case, SQL will treat this text as a comment and will execute normally. For example:

```
/* This query is to select the emp name given its id */  
Select name From emp where Id = 10;
```

### **Important Notes:**

- The .sql file you create is just a regular text file. It does not have any special format. Put the file in the right directory so SQL can reach it. You may write a full path in the command, e.g., “@<full-path>”
  - Check this link about adding comments within the SQL commands (be careful by putting space after “/\*” and before “\*/”, otherwise some commands may execute multiple times, and give you strange errors. That is “/\* ... \*/”  
[https://docs.oracle.com/cd/B28359\\_01/server.111/b28286/sql\\_elements006.htm#SQLRF51086](https://docs.oracle.com/cd/B28359_01/server.111/b28286/sql_elements006.htm#SQLRF51086)
- 2) Any comments or assumptions that you have, you can write them to a separate .doc or .pdf file.
  - 3) Put the two files in a single zip file that will be submitted as below.

### **Submission (Each team give one submission):**

- Submit electronically by the due date via [canvas.wpi.edu](https://canvas.wpi.edu) website. Make sure your two files (the text file .sql, and your report .doc or .pdf) are zipped, and you upload one file.
- Each team submits one copy (from either of the team members).
- ***No hard copy submissions***
- ***The team members' names must be written inside the report file.***