CALIFORNIA STATE UNIVERSITY, SAN BERNARDINO
SCHOOL OF COMPUTER SCIENCE & ENGINEERING

CSE 572F13: PART 3 PROJECT SPECIFICATIONS – Implementation in ORACLE

Due: December 5, 2013 (Thursday by 10 PM)

This part of the project will implement in ORACLE relational database management system, the normalized relational model from Part II of the project. You will need to prepare a report to be included in your project binder as well as prepare a Powerpoint presentation file that will be used in your presentation.

On Thursday, Nov 21 we will have a lottery on when each group will be doing the **presentation on Nov 26 and Dec 5** . Each presentation will be for 20 minutes.

WHAT TO DO:

You will need to use the normalized relational model from Part 2B of the Project.

1. EVERY TEAM MEMBER: Create a subdirectory PROJECT under CSE572 directory.
2. **TEAM LEADER**: Divide the tables (table creation, data loading) and queries equitably among the team including yourself. **Each member must do** at least one table creation and at least one data loading; and at least three queries. *NOTE: The table creation and data loading must be done on the same table(s) by the same member.*

(See pages 2-4 (Table Definition, Data Loading) of this document for details).

3. TEAM LEADER: Note a listing of problems/challenges encountered (conflicts among team members in design) and summarize the pros/cons and how each problem was resolved. This must be included in the report.

(See page 5(Section V of the report) of this document for proper placement in the report.)

3. TEAM MEMBERS: Provide the files you created for this part of the project to the TEAM Leader who executes the appropriate files in ORACLE.

(See pages 2-4 of this document for details).

4. Submit a Project Report (to be included in the project binder)

A typed report must be turned in for grading. Add this part to the appropriate division in your project binder. If turning in the project binder outside of class, put the project binder in the box provided outside my office, JBH-342.

Please refer to page 6 (Report Sections and Format) of this document for sequencing of pages and format.

5. Team Leader and Members will evaluate each other's performance.

5A. **Team Leader** will evaluate each member of the team based on the evaluation criteria developed in class. The team leader will submit by December 6 (Thurs) 10 PM via email his/her evaluation using the following format.

Format
SUBJECT:
CSE 572: < TEAM name>: PART 3 RATING BY LEADER <name>
BODY:
MEMBER1: <name> -
Criteria item rating & justification
MEMBER2: <name> -
Criteria item rating & justification
MEMBER3: <name> - (if any)
Criteria item rating & justification

5B. Each **TEAM MEMBER** will evaluate the team leader and other team members and must submit the rating on December 6 (Thurs) 6 PM via email using the format below.

Format

SUBJECT: CSE 572: <TEAM name>: PART 3 RATING BY MEMBER <name>
BODY:
LEADER: <name>
Criteria item 1 rating & justification
….
MEMBER1: <name> -
Criteria item rating & justification
….
MEMBER3: <name> - (if any)
Criteria item rating & justification

6. The team will need to prepare powerpoint presentation slides and the team leader submits to me via email a softcopy by **December 6 (Friday) 6 PM. Email Subject: CSE572 <your TEAM NAME> Presentation Slides**

To facilitate the reporting process for part 3 project document, make sure to save all that you do as well as results in a script file. So you can cut and paste from the script files.

Table Definitions

Each person responsible for each table assigned will do the following:

a. Prepare the create table sql file using a text editor. Precede create table command with comment lines as follows:

```
/*
      Project Option: <name of the project option chosen>
      TEAM: <team name>
       CREATED BY: <member's name responsible for this table>
*/
CREATE TABLE ……
( …….
)
```

NOTES:

1. **DO NOT PUT ; at the end of the CREATE TABLE command line**.

2. For the create table sql file use the following naming standard: create_<tablename>.sql where <tablename> must be replaced by the name of the relation that you are creating the definition for. Example: create_Employee.sql

a. Prepare the sql file that will contain the commands that will add a comment about the table.

The comment should include the following information: purpose of the table.
Save this file using the name "comment_table_<tablename>.sql".
For example, comment_table_Employee.sql
Precede COMMENT ON TABLE command with comment lines

/* Project Option: <name of the project option chosen>
TEAM: <team name>
TABLE COMMENT BY: <member's name responsible for this table's comment>
*/
COMMENT ON TABLE …….
For example,

COMMENT ON TABLE Employee IS 'All active employees in the COMPANY database';

b. Add a comment to each of the attribute of the table using the COMMENT ON COLUMN command. The attribute comment should give the purpose of the attribute and an example of the format of data. Your source for this is your final data dictionary from the project – either Part 1 or Part 2. The comments for all attributes must be saved in one file called comment_attributes_<tablename>.sql
Precede COMMENT ON COLUMN command with comment lines

/* Project Option: <name of the project option chosen>
TEAM: <team name>
Comments on Attributes for <table name>
COMMENT BY: <member's name responsible for these comments>
*/

For example,

```
COMMENT ON COLUMN employee.ssn
IS 'Employee's Social Security Number – format 999999999';
COMMENT ON COLUMN employee.name
IS 'Employee's full name- format: firstname MI lastname';
:
:
COMMENT ON COLUMN employee.dno
IS 'Department that employee works in - format: 999';

NOTE:  These definitions should be the same as the one in your data
dictionary info.
```

Data Load

Each person in the team must be able to demonstrate three methods:

Method 1: Insert SQL Command --Normal Insert
Method 2: Insert SQL Command --Using Substitution Variables
Method 3: SQL*LOAD Utility

Provide as many tuples per table that includes **all combinations of input**
NOTE: Care should be taken that data demonstrates good mix of information!
You should also design your data!!!

Choose one table that will use **Method 1: Normal Insert**.

Prepare the insert table sql file using a text editor.
Precede insert table command with comment lines

/* Project Option: <name of the project option chosen>
TEAM: <team name>
Data Load Method 1: Normal Insert – Insert SQL Command
Done By: <member's name responsible for this table>

*/
INSERT ……
NOTE: Suggested naming standard: n_insert_<tablename>.sql
Example: n_insert_Employee.sql

Choose a second table that will use **Method 2: Insert Using Substitution Variables**

Prepare the insert table sql file using a text editor.
Precede insert table command with comment lines
/* Project Option: <name of the project option chosen>
TEAM: <team name>
Data Load Method 2: Insert Using Substitution Variables
Done By: <member's name responsible for this table>
*/
INSERT ……
NOTE: Suggested naming standard: sv_insert_<tablename>.sql
Example: sv_insert_Department.sql

INSERT INTO DEPARTMENTS
VALUES (&department_id, '&department_name', &location_id);

&substitution variable prompts for values and & is a placeholder for the variable value.

Choose a third table that will use **Method 3: SQL*LOAD Utility**

(Please refer to the SQL Loader handout).

Prepare the following files needed for SQL*LOAD.
Need to prepare **.ctl** and **.dat** files using text editor for each table that you will load data into.
You will need to invoke SQLLDR outside ORACLE (sqlplus) but in ORAFARM.
SQLLDR user=<your_oracle username>
control=<tablename with extension .ctl>
data=<tablename with extension .dat>

The system will prompt for your ORACLE password.

The **TEAM MEMBER** who is responsible for the tables and queries assigned must provide the team leader the files that create the tables and load the data. The **TEAM LEADER** executes the appropriate files using the team leader's oracle account. The **TEAM LEADER** will provide select & references privileges to all tables to each member and update, alter, delete privileges only to the team member responsible for the table(s).
[GRANT SELECT, REFERENCES ON <tablename> TO teammember_oracle_userid;]

TEAM:

1. **Before each one starts to submit the create and load files to the team leader**; the team must analyze the dependencies among tables and figure out what constraints should be included in each create table statement.

2. **Before loading data, all constraints for all tables must be present**. The order of creating the tables and loading the data must be described in Part III of the Report for this part of the project..

Queries (Refer to list of queries for your project option)

EXTRA CREDIT(15 pts): **Add any pertinent report or queries not covered by the list of reports/queries required and that will illustrate 'advanced features of SQL/ORACLE' that you learned and would like to 'show off'.** ***State the query/report in English and execute the query in SQL and display the result.***

Each person responsible for each query assigned will do the following.

Prepare the sql statement to implement the query using a text editor.
Precede sql command with comment lines

```
/* English statement for this query as specified in the query list
   Project Option: <name of the project option chosen>
   TEAM: <team name>
   DONE BY: <member's name responsible for this table>
*/
SELECT ……
```

NOTE: Suggested naming standard: query_<number in query list>.sql
Demonstrate the use of variables
Demonstrate the use of formatting,sorting, ...etc
Demonstrate the use of views

The output of the select statement must be in report form
(refer to SQLPLUS FORMAT COMMANDS handout)
I suggest that you put all the format statements in front of the SELECT statement in the sql file described in Section "Queries" above.
Spool the running of the query in a file called spool_<query_number>.lst
'Cut & paste' in a document file (for your report) or in your .ppt file (for your presentation).

Report Sections and Format
Use 1" margin for top, bottom, left and right margins.
All pages must have page numbers in the bottom center of the page.
The sequence of the report is as follows.

I. PART III – ORACLE Implementation Tab Label
II. Title Page (See page 7 of this document for sample format)
NOTES: a. Team member names must be in alphabetical order
b. No footer or page number on this title page.
III. Normalized Relational Model Schema (Revised from Project Part IIB)
IV. Project Tables (presented in sequence of table creation). Must indicate the appropriate alter table commands to reflect the added constraints.
For each table,

- Query the data dictionary table 'USER_TAB_COMMENTS' to display the comment on the table
- Query the data dictionary table 'USER_COL_COMMENTS' to display the comments on the attributes of this table**.**
- Display the Create table command
- Display the structure of the table
- Display the contents of the table in a nice format
- Describe the Data loading done for the table

V. Project Queries (presented in the order listed in the list of queries for the project option.

VI. Comments on Project – Implementation Phase

A. Difficulties you faced in doing this implementation phase and how they were resolved (to be summarized by the Leader based on each member's comments)

B. Likes and dislikes about this part of the project (to be done by each team member as well as by Team Leader; present this section arranged alphabetically by team member's name)

C. What was the most challenging aspect of this part of the design? (to be done by each team member as well as by Team Leader; present this section arranged alphabetically by team member's name)

D. Suggestions on how to improve this part of project (to be done by each team member as well as by Team Leader; present this section arranged alphabetically by team member's name)

PAGINATE EACH PAGE by including a footer in the document as follows (BOLD, font size 8)

CSE572F13: TEAM NAME **PROJECT: PART III** **PAGE # of total pages**

PROJECT PART 3 GRADING by Instructor

This part of the project will be rated by me as follows:

Project Report

1. Presentation …….10 pts
   (format – 3 pts; neatness -2 pts; spelling/grammar- 3 pts, completeness-2 pts)

2. Content ……80 pts
   Table Definitions … 15
   Data Loading ……...15
   Queries ……..... 50
   (correctness, use of views/subqueries/variables, advanced features, readable output)
3. Leader/Member Evaluation ……10 pts

Total Project Report Points …… 100 pts

CALIFORNIA STATE UNIVERSITY, SAN BERNARDINO
SCHOOL OF COMPUTER SCIENCE & ENGINEERING

CSE 572 F13: Database Design Project
PART III – Implementation in Oracle 11g

Title of Project Option
PROJECT OPTION

Name of the Team
TEAM NAME

Lastname, Firstname
TEAM LEADER

TEAM MEMBERS
Adams, Mark
Smith, Clark
Zeta, Pam