California State University, San Bernardino School of Computer Science & Engineering
CSE572 Summer 2019 -Database Systems
LAB 03– CREATE/ALTER TABLE & TABLE CONSTRAINTS

For this lab exercise, you will deal with four tables for the CoyoteCorp DB.

Table Name	Attributes	Primary Key	Description
EMPLOYEES		empNo	An individual who works in CoyoteCorp
	empNo		Unique id, format 9999
	fname		First name of the employee
	lname		Last name of the employee
	address		Home address of the employee
	sex		Gender of the employee F- female or M-male
	salary		Yearly salary of the employee, format 999999, salary cannot be lower than $12,000.
	position		Job role of the employee in CoyoteCorp – clerk, programmer, manager, sales representative, account representative, dba
	deptNo		Department number that this employee works for, format 99 An employee can work in only one department and a department can have more than one employee. Every employee must work in a department. Every department must have at least 1 employee. Not every employee will manage a department. An employee need not work on a project. But every project must have at least one employee working on it.
DEPTARTMENTS		deptNo	A functional division within CoyoteCorp
	deptNo		Unique id, format 99
	deptName		Name of the functional division – IT, Sales, Accounting, Marketing, Administration
	Mgr		Employee number of the manager of the department A department must have a manager.
PROJECTS		projNo	Piece of planned work or an activity that is finished over a period of time
	projNo		Unique id, format 99
	projName		Name of panned work/activity – Computeration, ProductX, ProductY, etc
	deptNum		Department that controls the project – format 99 A department controls many projects but a project can be controlled by only one department. A department need not control a project. Every project must be controlled by a department.
EMP_PROJ		(empNo, projNo)	Details of the hours worked by the employee on each project
			Number of hours spent by the employee in the project

EMP Data

empNo	fname	lname	address	sex	salary	position	deptNo
1000	Steven	King	731 Fondren, Houston, TX	M	30000	Programmer	60
1007	Diana	Lorentz	638 Voss, Bellaire, TX	F	24000	Clerk	20
2002	Pat	Fay	3321 Castle, Spring, TX	F	15000	Sales Representative	80
1760	Jonathan	Taylor	561 Rice, Houston, TX	M	60000	Manager	20
1740	Ellen	Abel	890 Stone, Houston, TX	F	65000	Manager	60
2060	William	Gietz	450 Berry, Bellaire, TX	M	65000	Manager	80
2000	Jennifer	Whalen	980 Fire Oak, Humble, TX	F	28000	Clerk	60
1444	Peter	Vargas	975 Dallas, Houston, TX	M	20000	Sales Representative	80

DEPT Data

deptNumber	deptName	Mgr
20	Marketing	1760
60	IT	1740
80	Sales	2060

PROJ Data

projNumber	projName	deptNum
10	Product X	20
20	Product Y	20
30	Computerization	60
40	Product Z	80
50	Mobile Apps	60

EMP_PROJ Data

empNo	projNo	hoursWorked
1000	30	32.5
1000	50	7.5
2002	10	40.0
1444	20	20.0
1760	10	5.0
1760	20	10.0
1740	50	15.0
2060	40	12.0

NOTES:

CREATE TABLE <tablename>
(column datatype [DEFAULT expr] [column-level constraint],
…
[table_constraint] [, …]);

where

- Table names and column names must begin with a letter and be 1-30 characters long.
- Names must contain only A-Z, a-z, 0-9, _(underscore), $, # (legal characters, but their use is discouraged.)
- Names must not duplicate the name of another object owned by the same Oracle server user.
- Names cannot be an Oracle server reserved word.
- Datatype is the column's data type and length

Data Type	Description
varchar2(size)	Variable-length character data, specify maximum size; min size is 1; max size is 4000.
char (size)	Fixed-length character data of length size bytes (default min size is 1; max size is 2000).
number(p,s)	Variable-length numeric data having precision p and scale s. Precision is total number of decimal digits (1-38) and scale is the number of digits to the right of the decimal point ;
date	Date and time values to the nearest second between January 1, 4712 B.C. and December 31, 9999 A.D. Date data is stored in fixed-length fields of seven bytes each, corresponding to century, year, month, day, hour, minute, and second. For input and output of dates, the standard Oracle date format is DD-MON-YY, as follows: '13-NOV-92' To enter dates that are not in standard Oracle date format, use the TO_DATE function with a format mask.
long	Variable-length character data type up to 2 gigabytes
clob	Character data type up to 4 gigabytes
raw(size)	Raw binary data of length size (max size must be specified. Max size is 2000.)
long raw	Raw binary data of variable length up to 2 gigabytes
blob	Binary data up to 4 gigabytes
bfile	Binary data stored in an external file; up to 4 gigabytes
rowid	A 64 base number system representing the unique address of a row in its table

- A long column is not copied when a table is created using a subquery.
- A long column cannot be included in a GROUP BY or an ORDER BY clause.
- Only one long column can be used per table.
- No constraints can be defined on a long column.
- Use a clob column rather than a long column.
- DEFAULT expr: specifies a default value if a value is omitted in the INSERT statement; can be a literal, an expression, or a SQL function, such as SYSDATE and USER, but the value cannot be the name of another column or a pseudocolumn, such as NEXTVAL or CURRVAL. The default expression must match the data type of the column.
- column is the name of the column
- column_constraint is an integrity constraint as part of the column definition and references a single column; can define any type of integrity constraint
- table_constraint is an integrity constraint as part of the table definition

CONSTRAINT [constraint_name] constraint type,

where

constraint_name is the name of the constraint and use the "tablename_columnname_##" must be provided otherwise Oracle server generates a name for it with SYS_Cn

where ## can either be NN (not null), PK (primary key), FK (foreign key), UK (unique key), CK (check constraint)

constraint_type	description	example
NOT NULL	Specified only at column level not at table level	CREATE TABLE employees (… last_name varchar2(25) CONSTRAINT employee_last_name_NN NOT NULL, …);
UNIQUE	Requires that no two rows of a table have duplicate values in the specified column; allows input of NULLs unless NOT NULL is defined Can be defined at the column level Or at the table level Oracle server enforces unique constraint by implicitly creating a unique index on the column	CREATE TABLE employees (… email varchar2(25) CONSTRAINT employee_email_UK UNIQUE, …); CREATE TABLE employees (… email varchar2(25), …. CONSTRAINT employee_email_UK UNIQUE(email),);
PRIMARY KEY	Creates a primary key for the table; only one PK per table; Enforces uniqueness for the column(s) and no column	CREATE TABLE employees (employee_id NUMBER(6) CONSTRAINT employee_employee_id_PK PRIMARY KEY, …);

	that is part of key can contain NULL If the primary key consists only of one column, it can be defined at the column level or at the table level If the primary key is a composite key of more than one attributes, then define the PK constraint must be defined at the table level only.	CREATE TABLE employees (employee_id NUMBER(6), …. CONSTRAINT employee_employee_id_PK PRIMARY KEY (employee_id), …); CREATE TABLE works_on (employee_id NUMBER(6), project_id NUMBER(4), …. CONSTRAINT works_on_employee_id_project_id_PK PRIMARY KEY (employee_id, project_id), …);
CHECK	Defines a condition that each row must satisfy; the condition can use the same construct as query condition with the following exceptions: References to CURRVAL, NEXTVAL, LEVEL, and ROWNUM pseudocolumns ; Calls to SYSTDATE, UID, UER, and USERENV	CREATE TABLE employees (…. salary NUMBER(8,2) CONSTRAINT employees_salary_CK CHECK (salary > 0), …);

	functions; Queries that refer to other values in other rows Can be defined at the column or table level.	

1. Add the FOREIGN KEY CONSTRAINTS after creating the table with other contraints. To add a FOREIGN KEY CONSTRAINT

ALTER TABLE <table_name>
ADD CONSTRAINT constraint_name_FK FOREIGN KEY(column) REFERENCES
<parent_table_name> (primary_key_of_parent_table) [ON DELETE CASCADE | ON DELETE SET NULL};

where

- <table_name> is the name of the table
- constraint_name is the name of the constraint in the form 'table_name_column_name
- REFERENCES identifies the table and column in the parent table
- ON DELETE CASCADE indicates that when the row in the parent table is deleted, the dependent rows in the child table will also be deleted.
- ON DELETE SET NULL converts foreign key values to NULL when the parent value is removed. The default behavior is the RESTRICT RULE, which disallows the update or deletion of referenced data.

  WITHOUT the ON DELETE CASCADE or the ON DELETE SET NULL options, the row in the parent table cannot be deleted if it is referenced in the child table.

The FOREIGN KEY is defined in the child table and the table containing the referenced column is the parent table.

EXAMPLE:

```
ALTER  TABLE employees(
  ADD CONSTRAINT employees_dno_FK FOREIGN KEY (dno)  REFERENCES
        department(department_id);
```

If not every employee must work in a department, and when we delete a department tuple, then you can issue

```
ALTER  TABLE employees(
ADD CONSTRAINT employees_dno_FK FOREIGN KEY (dno)  REFERENCES
        department(department_id)
         ON DELETE SET NULL;
```

If every employee must work in a department and we really want to delete a department tuple, you will need first to transfer the employee to another department (by doing an UPDATE command) before you can delete the original department that this employee belongs to!

WHAT TO DO for LAB03 -- must be done by Tuesday, August 26, 2019.

1. Log on to orafarm.cse.csusb.edu using SQL Developer.
2. Use the SQL Developer to create the four tables EMP, DEPT, PROJ, EMP_PROJ with the appropriate constraints except FOREIGN KEY constraints. For filename, use CREATE.sql accordingly.
3. Import data from the csv files to EMP, DEPT, and PROJ tables.
4. Check that all data for EMP, DEPT, and PROJ tables has been imported.
5. In SQL Developer add the FOREIGN KEY constraints for each of the EMP, DEPT, PROJ and EMP_PROJ tables using Alter statement. For filename, use ALTER.sql accordingly.
6. Use the substitution & method of INSERT command to populate EMP_PROJ table.
   INSERT INTO EMP_PROJ VALUES ('&empNo', '&projNo', &hoursWorked);

   NOTE: enclose &empNo in ' ' if the datatype is a string – VARCHAR2 or CHAR
   If empNo is NUMBER datatype then do not enclose &empNo in ' '!
7. Check that all data for EMP_PROJ table has been entered.
8. Use the alter table to add a new column named email in the employees table.
9. Try to execute the following insert statement and explain what is happening.
   INSERT INTO EMPLOYEES VALUES (1172, 'joe', 'Calvert', '672 White Pine, Austin, TX', 'X', 10000, 10) ;
10. Please necessary corrections on the insert statement above.
11. **Please compile all your statements in one file and submit to Blackboard.**