



# UNIVERSITY OF CHITTAGONG

## Department of Computer Science and Engineering

Session: 2021-2022

4th semester

Assignment No. : 04  
Course Title : Database Systems  
Course Code No. : CSE-413

Submitted to:

**Dr. Rudra Pratap Deb Nath**

Associate Professor

Department of Computer Science and Engineering  
University of Chittagong

Submitted by:

**Sanzid Islam Mahi**

ID: 22701065

Department of Computer Science and Engineering  
University of Chittagong

Date: Jul 13, 2024

# Oracle9i SQL-I

## Chapter 9

### Practice 9

1. Create the DEPT table based on the following table instance chart. Place the syntax in a script called `lab9_1.sql`, then execute the statement in the script to create the table. Confirm that the table is created

Column Name	ID	NAME
Key Type		
Nulls/Unique		
FK Table		
FK Column		
Data type	NUMBER	VARCHAR2
Length	7	25

Name	Null?	Type
ID		NUMBER(7)
NAME		VARCHAR2(25)

#### Solution:

```
1 CREATE TABLE dept (  
2     ID INT (7) ,  
3     NAME VARCHAR (25)  
4 );  
5  
6 DESCRIBE DEPT ;
```

#### Output:

#	Field	Type	Null
1	id	int	YES
2	name	varchar(25)	YES

2. Populate the DEPT table with data from the DEPARTMENTS table. Include only columns that you need.

#### Solution:

```
1 INSERT INTO dept  
2 SELECT department_id , department_name  
3 FROM departments ;
```

3. Create the EMP table based on the following table instance chart. Place the syntax in a script called `lab9_3.sql`, and then execute the statement in the script to create the table. Confirm that the table is created.

#### Solution:

Column Name	ID	LAST_NAME	FIRST_NAME	DEPT_ID
Key Type				
Nulls/Unique				
FK Table				
FK Column				
Data type	NUMBER	VARCHAR2	VARCHAR2	NUMBER
Length	7	25	25	7

Name	Null?	Type
ID		NUMBER(7)
LAST_NAME		VARCHAR2(25)
FIRST_NAME		VARCHAR2(25)
DEPT_ID		NUMBER(7)

```

1 CREATE TABLE emp(
2     id int (7),
3     last_name varchar(25),
4     first_name varchar(25),
5     dept_id int (7)
6 );
7
8 DESCRIBE emp;

```

**Output:**

#	Field	Type	Null
1	id	int	YES
2	last_name	varchar(30)	YES
3	first_name	varchar(25)	YES
4	dept_id	int	YES

4. Modify the EMP table to allow for longer employee last names. Confirm your modification.

Name	Null?	Type
ID		NUMBER(7)
LAST_NAME		VARCHAR2(50)
FIRST_NAME		VARCHAR2(25)
DEPT_ID		NUMBER(7)

**Solution:**

```

1 ALTER TABLE emp
2 MODIFY last_name VARCHAR(50);
3
4 DESCRIBE emp;

```

**Output:**

#	Field	Type	Null
1	id	int	YES
2	last_name	varchar(50)	YES
3	first_name	varchar(25)	YES
4	dept_id	int	YES

  

TABLE_NAME
DEPT
EMP

5. Confirm that both the DEPT and EMP tables are stored in the data dictionary.  
(Hint: USER\_TABLES)

**Solution:**

```
1 SELECT table_name
2 FROM information_schema.tables
3 WHERE table_name IN ('dept', 'emp');
```

**Output:**

#	TABLE_NAME
1	dept
2	emp

6. Create the EMPLOYEES2 table based on the structure of the EMPLOYEES table. Include only the EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME, SALARY, and DEPARTMENT\_ID columns. Name the columns in your new table ID, FIRST\_NAME, LAST\_NAME, SALARY, and DEPT\_ID, respectively.

**Solution:**

```
1 CREATE TABLE employees2 AS
2 SELECT employee_id id, first_name, last_name, salary,
3        department_id dept_id
4 FROM employees;
```

7. Drop the EMP table.

**Solution:**

```
1 DROP TABLE emp;
```

8. Rename the EMPLOYEES2 table as EMP.

**Solution:**

```
1 RENAME TABLE employees2 TO emp;
```

9. Add a comment to the DEPT and EMP table definitions describing the tables. Confirm your additions in the data dictionary.

**Solution:**

```
1 ALTER TABLE dept
2 COMMENT = 'This table stores department information';
3
4 ALTER TABLE emp
5 COMMENT = 'This table stores employee information';
6
7 SELECT table_name, table_comment
8 FROM information_schema.tables
9 WHERE table_name IN ('dept', 'emp');
```

**Output:**

#	TABLE_NAME	TABLE_COMMENT
1	dept	This table stores department information
2	emp	This table stores employee information

10. Drop the FIRST\_NAME column from the EMP table. Confirm your modification by checking the description of the table.

**Solution:**

```
1 ALTER TABLE emp
2 DROP COLUMN first_name;
3
4 DESCRIBE emp;
```

**Output:**

#	Field	Type	Null
1	id	int unsigned	NO
2	first_name	varchar(20)	YES
3	salary	decimal(8,2)	NO
4	dept_id	int unsigned	YES

11. In the EMP table, mark the DEPT\_ID column as UNUSED. Confirm your modification by checking the description of the table.

**Solution:**

```
1 ALTER TABLE emp
2 SET UNUSED COLUMN dept_id;
3
4 DESCRIBE emp;
```

12. Drop all the UNUSED columns from the EMP table. Confirm your modification by checking the description of the table.

**Solution:**

# Chapter 10

## Practice 10

1. Add a table-level PRIMARY KEY constraint to the EMP table on the ID column. The constraint should be named at creation. Name the constraint `my_emp_id_pk`.

**Hint:** The constraint is enabled as soon as the ALTER TABLE command executes successfully.

**Solution:**

**Output:**

2. Create a PRIMARY KEY constraint to the DEPT table using the ID column. The constraint should be named at creation. Name the constraint `my_deptid_pk`.

**Hint:** The constraint is enabled as soon as the ALTER TABLE command executes successfully.

**Solution:**

3. Add a column DEPT\_ID to the EMP table. Add a foreign key reference on the EMP table that ensures that the employee is not assigned to a nonexistent department. Name the constraint `my_emp_dept_id_fk`.

**Solution:**

4. Confirm that the constraints were added by querying the USER\_CONSTRAINTS view. Note the types and names of the constraints. Save your statement text in a file called `lab10_4.sql`.

CONSTRAINT_NAME	C
MY_DEPT_ID_PK	P
SYS_C002541	C
MY_EMP_ID_PK	P
MY_EMP_DEPT_ID_FK	R

**Solution:**

5. Display the object names and types from the USER\_OBJECTS data dictionary view for the EMP and DEPT tables. Notice that the new tables and a new index were created.

**Solution:**

6. Modify the EMP table. Add a COMMISSION column of NUMBER data type, precision 2, scale 2. Add a constraint to the commission column that ensures that a commission value is greater than zero.

**Solution:**

## Chapter 8

### Practice 8