

EXERCISE-1 Creating and Managing Tables

OBJECTIVE

After the completion of this exercise, students should be able to do the following:

- Create tables
- Describing the data types that can be used when specifying column definition
- Alter table definitions
- Drop, rename, and truncate tables

NAMING RULES

Table names and column names:

- Must begin with a letter
- Must be 1-30 characters long
- Must contain only A-Z, a-z, 0-9, _, \$, and #
- Must not duplicate the name of another object owned by the same user
- Must not be an oracle server reserve words
- 2 different tables should not have same name.
- Should specify a unique column name.
- Should specify proper data type along with width
- Can include “not null” condition when needed. By default it is ‘null’.

The CREATE TABLE Statement

Table: Basic unit of storage; composed of rows and columns

Syntax: 1 Create table table_name (column_name1 data_type (size)
column_name2 data_type (size)....);

Syntax: 2 Create table table_name (column_name1 data_type (size) constraints,
column_name2 data_type constraints ...);

Example:

```
Create table employees ( employee_id number(6), first_name varchar2(20), ..job_id varchar2(10),
CONSTRAINT emp_emp_id_pk PRIMARY KEY (employee_id));
```

Tables Used in this course

Creating a table by using a Sub query

SYNTAX

```
// CREATE TABLE table_name(column_name type(size)...);
```

Create table table_name as select column_name1,column_name2,.....column_namen from
table_name where predicate;

AS Subquery

Subquery is the select statement that defines the set of rows to be inserted into the new table.

Example

```
Create table dept80 as select employee_id, last_name, salary*12 Annsal, hire_date  
from employees where dept_id=80;
```

The ALTER TABLE Statement

The ALTER statement is used to

- Add a new column
- Modify an existing column
- Define a default value to the new column
- Drop a column
- To include or drop integrity constraint.

SYNTAX

```
ALTER TABLE table_name ADD /MODIFY(Column_name type(size));
```

```
ALTER TABLE table_name DROP COLUMN (Column_nname);
```

ALTER TABLE ADD CONSTRAINT Constraint_name PRIMARY KEY (Colum_Name);

Example:

```
Alter table dept80 add (jod_id varchar2(9));  
Alter table dept80 modify (last_name varchar2(30));  
Alter table dept80 drop column job_id;
```

NOTE: Once the column is dropped it cannot be recovered.

DROPPING A TABLE

- All data and structure in the table is deleted.
- Any pending transactions are committed.
- All indexes are dropped.
- Cannot roll back the drop table statement.

Syntax:

```
Drop table tablename;
```

Example:

```
Drop table dept80;
```

RENAMING A TABLE

To rename a table or view.

Syntax

```
RENAME old_name to new_name
```

Example:

Rename dept to detail_dept;

TRUNCATING A TABLE

Removes all rows from the table.

Releases the storage space used by that table.

Syntax

TRUNCATE TABLE *table_name*;

Example:

TRUNCATE TABLE copy_emp;

Find the Solution for the following:

Create the following tables with the given structure.

EMPLOYEES TABLE

NAME	NULL?	TYPE
Employee_id	Not null	Number(6)
First_Name		Varchar(20)
Last_Name	Not null	Varchar(25)
Email	Not null	Varchar(25)
Phone_Number		Varchar(20)
Hire_date	Not null	Date
Job_id	Not null	Varchar(10)
Salary		Number(8,2)
Commission_pct		Number(2,2)
Manager_id		Number(6)
Department_id		Number(4)

DEPARTMENT TABLE

NAME	NULL?	TYPE
Dept_id	Not null	Number(6)
Dept_name	Not null	Varchar(20)
Manager_id		Number(6)
Location_id		Number(4)

JOB_GRADE TABLE

NAME	NULL?	TYPE
Grade_level		Varchar(2)
Lowest_sal		Number

Highest_sal	Number
-------------	--------

LOCATION TABLE

NAME	NULL?	TYPE
Location_id	Not null	Number(4)
St_addr		Varchar(40)
Postal_code		Varchar(12)
City	Not null	Varchar(30)
State_province		Varchar(25)
Country_id		Char(2)

1. Create the DEPT table based on the DEPARTMENT following the table instance chart below. 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

create table Department (Dept_id number(7), name varchar2(25));

2. Create the EMP table based on the following instance chart. Confirm that the table is created.

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

create table EMP (ID Number(7) primary key, Last_Name Varchar2(25) Not Null, First_Name Varchar2(25), Dept-ID Number(7));

3. Modify the EMP table to allow for longer employee last names. Confirm the modification.(Hint: Increase the size to 50)

Alter table EMP Modify (Last_Name Varchar2(50));

Column	Data Type
ID	Number(7)
Last_Name	Varchar2(30)
First_Name	Varchar2(25)
Dept-ID	Number(7)

Column	Data Type
ID	Number(7)
Last_Name	Varchar2(25)
First_Name	Varchar2(25)
Dept-ID	Number(7)

Column	Data Type
Name	Varchar2(25)

Column	Type
ID	Number(7)
Last_Name	Varchar2(50)
First_Name	Varchar2(25)
Salary	Number
Dept_ID	Number(7)

4. Create the EMPLOYEES2 table based on the structure of EMPLOYEES table. Include Only the Employee_id, First_name, Last_name, Salary and Dept_id columns. Name the columns Id, First_name, Last_name, salary and Dept_id respectively.

```
create table employees2 as select Id, First_Name,
Last_Name, salary, Dept-ID from EMP where l=2
```

5. Drop the EMP table.

```
Drop Table EMP;
```

6. Rename the EMPLOYEES2 table as EMP.

```
Rename employees2 to EMP;
```

7. Add a comment on DEPT and EMP tables. Confirm the modification by describing the table.

```
Comment on Table Dept is 'This is department
table';
Comment on Table EMP is 'This is EMP Table';
```

8. Drop the First_name column from the EMP table and confirm it.

```
Alter Table EMP drop column First_Name;
```

column	dataType
ID	Number(7)
Last_Name	Varchar2(50)
Dept-ID	Number(7)

Evaluation Procedure	Marks awarded
Query(5)	5
Execution (5)	5
Viva(5)	5
Total (15)	15
Faculty Signature	<u>Dm</u> 8/9/25