USMAN INSTITUTE OF TECHNOLOGY

Department of Computer Science

CS311 Introduction to Database Systems

Lab#6

Objective:

-Creating and managing tables.

Name of Student: Muhammad Waleed

Roll No: 20B-115-SE **Sec. B**

Date of Experiment:

....

Marks Obtained/Remarks: ______________________________

Signature: ______________________________

THEORY

Database Objects: An Oracle database can contain multiple data structures. Each structure should be outlined in the database design so that it can be created during the build stage of database development.

Table: Stores data

View: Subset of data from one or more tables

Sequence: Generates primary key values

Index: Improves the performance of some queries

Synonym: Gives alternative names to objects

Oracle Table Structures

- Tables can be created at any time, even while users are using the database.
- We do not need to specify the size of any table. The size is ultimately defined by the amount of space allocated to the database as a whole.
- Table structure can be modified online.

Naming Conventions

- Name database tables and columns according to the standard rules for naming any Oracle database object.
- Table names and column names must begin with a letter and can be 1-30 characters long.
- Names must contain only the characters A-Z, a-z, 0-9, _(*underscore*), $, and # (legal characters, but their use is discouraged).
- Names must not duplicate the name of another object owned by the same Oracle Server user.
- Names must not be an Oracle Server reserved word.

Creating and Altering Tables

The CREATE TABLE statement

To create a table, a user must have the CREATE TABLE privilege and a storage area in which to create objects. The database administrator uses data control language (DCL) statements, covered in a later session, to grant privileges to users.

The syntax is as follows:-

CREATE TABLE [*schema* .] *table* (*column datatype* [DEFAULT *expr*] [, …]);

Referencing another user's tables

A *schema* is a collection of objects. *Schema objects* are the logical structures that directly refer to the data in a database. Schema objects include tables, views, synonyms, sequences, stored procedures, indexes, clusters, and database links.

If a table does not belong to the user, the owner's name must be prefixed to the table.

The DEFAULT option

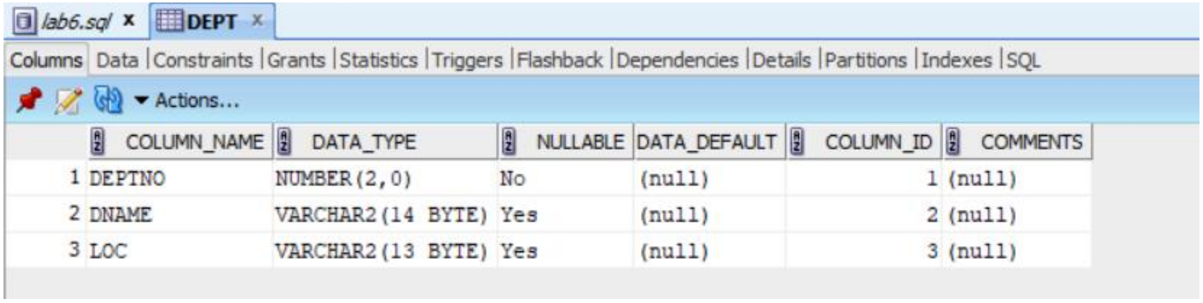
A column can be given a default value by using the DEFAULT option. This option prevents null values from entering the columns if a row is inserted without a value for the column. The default value can be a literal, an expression, or a SQL function, such as SYSDATE and USER, but the value cannot be the name of another column or a pseudocolumn, such as NEXTVAL, or CURRVAL. The default expression must match the datatype of the column.

For example,
… hiredate DATE DEFAULT SYSDATE, …

<u>Example</u>

The following example creates the DEPT table mentioned in the lab. session 01.

```
CREATE    TABLE    dept
(deptno         NUMBER(2),
dname   VARCHAR2(14),
loc       VARCHAR2(13));
```



	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	DEPTNO	NUMBER(2,0)	No	(null)	1	(null)
2	DNAME	VARCHAR2(14 BYTE)	Yes	(null)	2	(null)
3	LOC	VARCHAR2(13 BYTE)	Yes	(null)	3	(null)

Since creating a table is a DDL statement, an automatic commit takes place when this statement is executed.

In order to confirm the creation of the table, issue the DESCRIBE command as discussed in lab session 01.

DESCRIBE DEPT

SQL Data Types

Datatype	Description
VARCHAR2(*size*)	Variable-length character data (A maximum *size* must be specified. Default and minimum *size* is 1; maximum *size* is 4000)
CHAR(*size*)	Fixed-length character data of length *size* bytes (Default and minimum *size* is 1; maximum *size* is 2000)
NUMBER(*p*, *s*)	Number having precision p and scale s (The precision is the total number of decimal digits and the scale is the number of digits to the

	right of the decimal point. The precision can range from 1 to 38 and the scale can range from -84 to 127.)
DATE	Date and time values between January 1, 4712 B.C. and December 31, 9999 A.D.
LONG	Variable length character data up to 2 gigabytes
Datatype	Description
CLOB	Single-byte character data up to 2 gigabytes
RAW(*size*)	Raw binary data of length *size* (A maximum size must be specified. Maximum *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

Table 8.1

CLOB, BLOB and BFILE are the large object data types and can store blocks of unstructured data (such as text, graphics images, video clips and sound wave forms up to 4 gigabytes in size.) LOBs also support random access to data.

Creating a table by using a Subquery

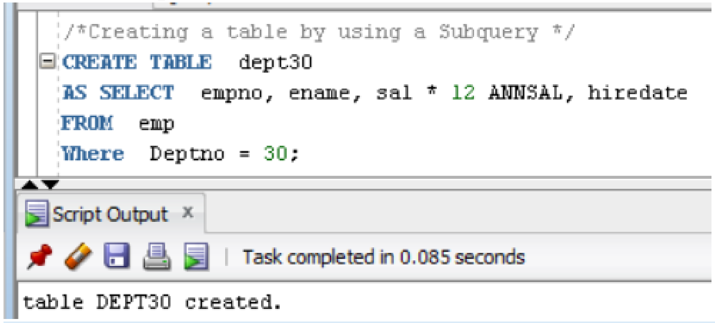
The following example creates a table, DEPT30, that contains details of all employees working in department 30

CREATE TABLE dept30

AS SELECT empno, ename, sal * 12 ANNSAL, hiredate

FROM emp

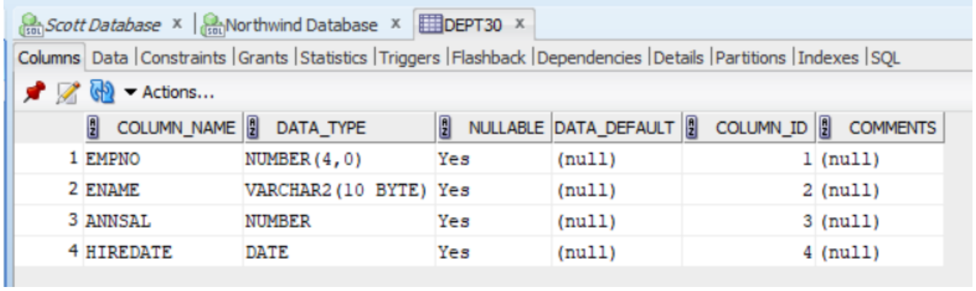
WHERE deptno = 30;



```
/*Creating a table by using a Subquery */
CREATE TABLE  dept30
AS SELECT  empno, ename, sal * 12 ANNSAL, hiredate
FROM  emp
Where  Deptno = 30;
```

Task completed in 0.085 seconds

```
table DEPT30 created.
```



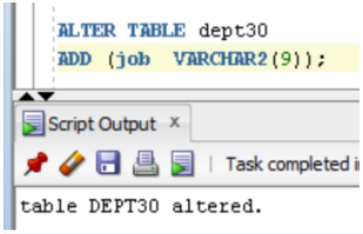
	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	EMPNO	NUMBER(4,0)	Yes	(null)	1	(null)
2	ENAME	VARCHAR2(10 BYTE)	Yes	(null)	2	(null)
3	ANNSAL	NUMBER	Yes	(null)	3	(null)
4	HIREDATE	DATE	Yes	(null)	4	(null)

Altering table structure

The ALTER TABLE statement is used to

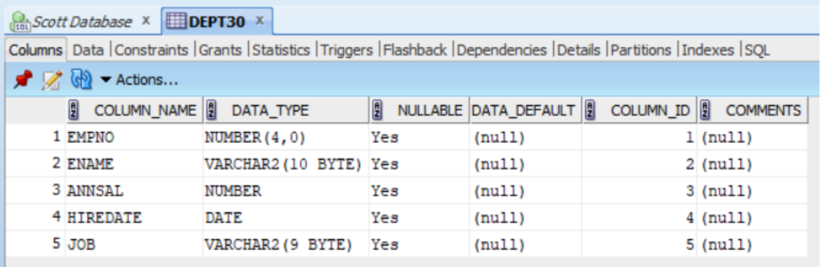
- Add a new column
- Modify an existing column
- Define a default value for the new column

The following example adds a new column to the DEPT30 table:- ALTER TABLE dept30
ADD (job VARCHAR2(9));



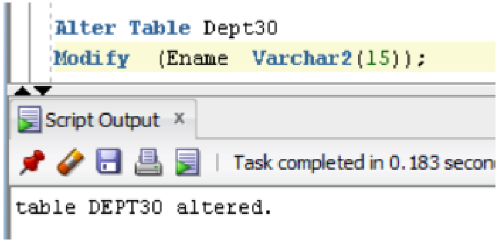
```
ALTER TABLE dept30
ADD (job  VARCHAR2(9));
```

```
table DEPT30 altered.
```



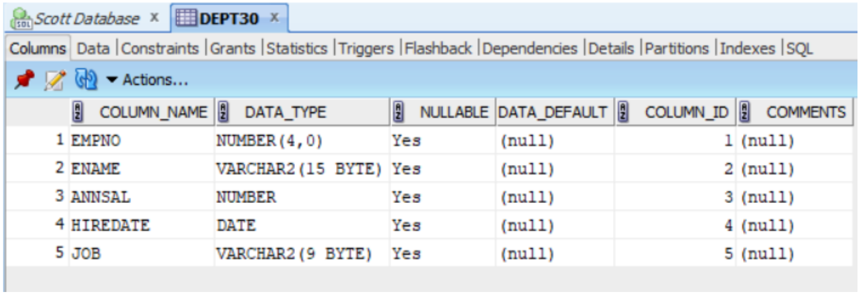
	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	EMPNO	NUMBER(4,0)	Yes	(null)	1	(null)
2	ENAME	VARCHAR2(10 BYTE)	Yes	(null)	2	(null)
3	ANNSAL	NUMBER	Yes	(null)	3	(null)
4	HIREDATE	DATE	Yes	(null)	4	(null)
5	JOB	VARCHAR2(9 BYTE)	Yes	(null)	5	(null)

To modify an existing column, use
ALTER TABLE dept30
MODIFY (ename VARCHAR2(15));



```
Alter Table Dept30
Modify  (Ename  Varchar2(15));
```

```
table DEPT30 altered.
```



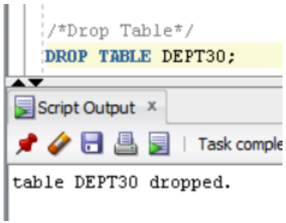
	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	EMPNO	NUMBER(4,0)	Yes	(null)	1	(null)
2	ENAME	VARCHAR2(15 BYTE)	Yes	(null)	2	(null)
3	ANNSAL	NUMBER	Yes	(null)	3	(null)
4	HIREDATE	DATE	Yes	(null)	4	(null)
5	JOB	VARCHAR2(9 BYTE)	Yes	(null)	5	(null)

Dropping a Table

The DROP TABLE statement removes the definition of an Oracle table. The database loses all the data in the table and all the indexes associated with it.

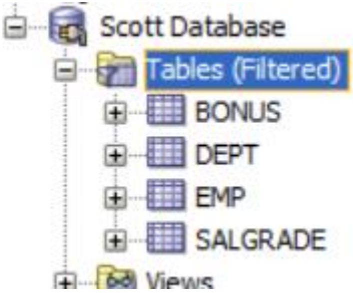
The DROP TABLE statement, once executed, is irreversible. The Oracle Server does not question the action when the statement is issued and the table is immediately dropped. All DDL statements issue a commit, therefore, making the transaction permanent.

To drop the table DEPT30,
DROP TABLE DEPT30;



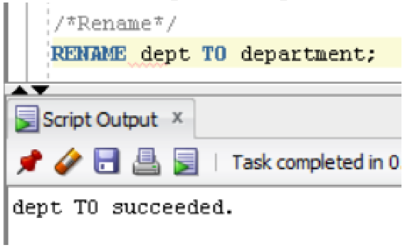
```
/*Drop Table*/
DROP TABLE DEPT30;
```

```
table DEPT30 dropped.
```



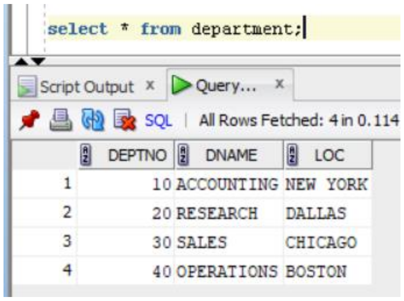
Changing the name of an object

To change the name of a table, view, sequence, or synonym, execute the RENAME statement:-
RENAME dept TO department;



```
/*Rename*/
RENAME dept TO department;
```

```
dept TO succeeded.
```



```
select * from department;
```

	DEPTNO	DNAME	LOC
1	10	ACCOUNTING	NEW YORK
2	20	RESEARCH	DALLAS
3	30	SALES	CHICAGO
4	40	OPERATIONS	BOSTON

What are constraints?

The Oracle Server uses constraints to prevent invalid data entry into tables.
Constraints are used for the following purposes:-

- Enforce rules at the table level whenever a row is inserted, updated, or deleted from that table. The constraint must be satisfied for the operation to succeed.
- Prevent the deletion of a table if there are dependencies from other tables. ▪ Provide rules for Oracle tools, such as Oracle Developer.

The following constraint types valid in Oracle:-

Constraint	Description
NOT NULL	Specifies that this column may not contain a null value
UNIQUE	Specifies a column or combination of columns whose values must be unique for all rows in the table
PRIMARY KEY	Uniquely identifies each row of the table
FOREIGN KEY	Establishes and enforces a foreign key relationship between the column and a column of the referenced table
CHECK	Specifies a condition that must be true

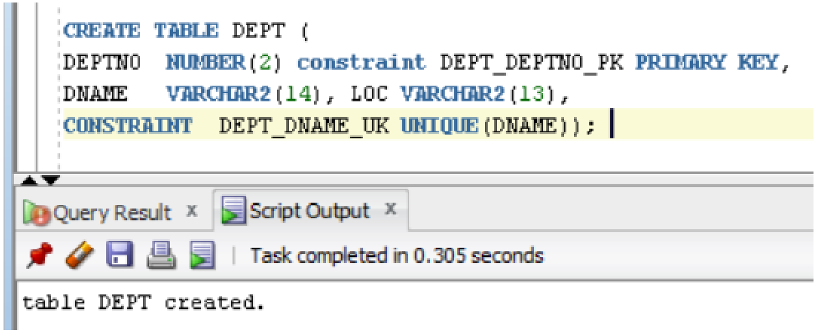
Table 8.2

Constraint Guidelines

- All constraints are stored in the data dictionary
- Name a constraint or the Oracle server will generate a name by using the SYS_Cn format
- Create a constraint
- At the same time as the table is created ○ After the table has been created

The EMP table is being created specifying various constraints:-

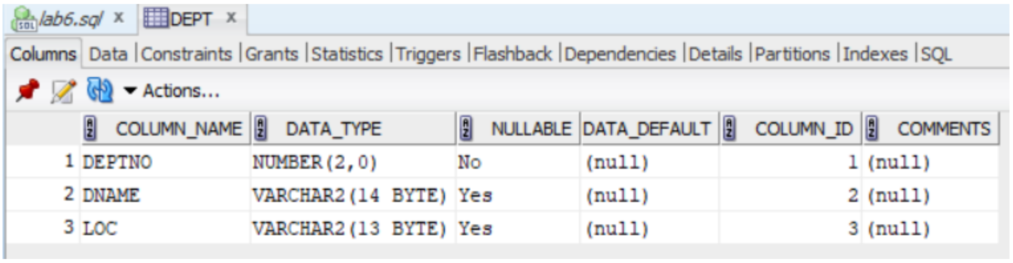
```
CREATE TABLE DEPT (
DEPTNO     NUMBER(2) constraint DEPT_DEPTNO_PK PRIMARY KEY,
DNAME      VARCHAR2(14),     LOC
           VARCHAR2(13),
CONSTRAINT       DEPT_DNAME_UK         UNIQUE(DNAME));
```



```
CREATE TABLE DEPT (
DEPTNO  NUMBER(2) constraint DEPT_DEPTNO_PK PRIMARY KEY,
DNAME   VARCHAR2(14), LOC VARCHAR2(13),
CONSTRAINT  DEPT_DNAME_UK UNIQUE(DNAME));
```

Task completed in 0.305 seconds

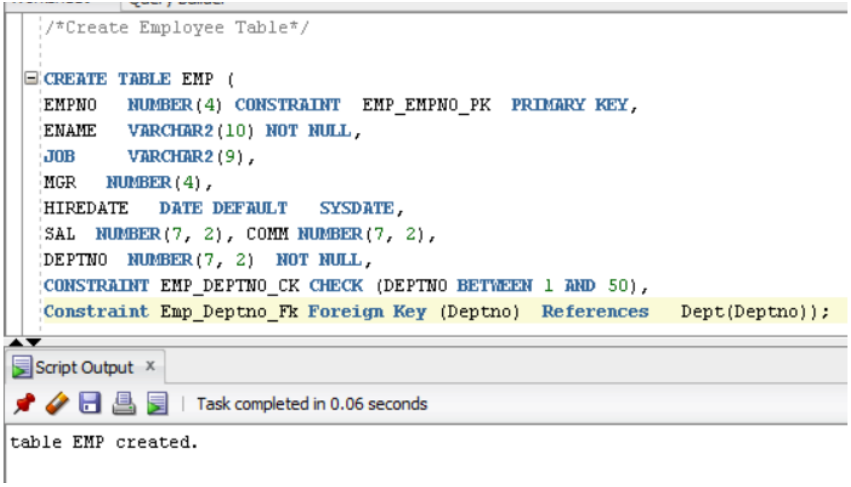
```
table DEPT created.
```


lab6.sql
DEPT
Columns | Data | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL
Actions...

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	DEPTNO	NUMBER(2,0)	No	(null)	1	(null)
2	DNAME	VARCHAR2(14 BYTE)	Yes	(null)	2	(null)
3	LOC	VARCHAR2(13 BYTE)	Yes	(null)	3	(null)

```
CREATE TABLE EMP (
EMPNO        NUMBER(4) CONSTRAINT  EMP_EMPNO_PK  PRIMARY KEY,
ENAME         VARCHAR2(10) NOT NULL,
JOB              VARCHAR2(9),
MGR             NUMBER(4),
HIREDATE   DATE DEFAULT      SYSDATE,
SAL           NUMBER(7,     2),     COMM
NUMBER(7, 2),
DEPTNO       NUMBER(7, 2)         NOT NULL,
CONSTRAINT          EMP_DEPTNO_CK CHECK (DEPTNO BETWEEN 1 AND 50),
CONSTRAINT        EMP_DEPTNO_FK  FOREIGN  KEY  (DEPTNO)    REFERENCES
        DEPT(DEPTNO));
```

Composite primary keys are defined at the table level.

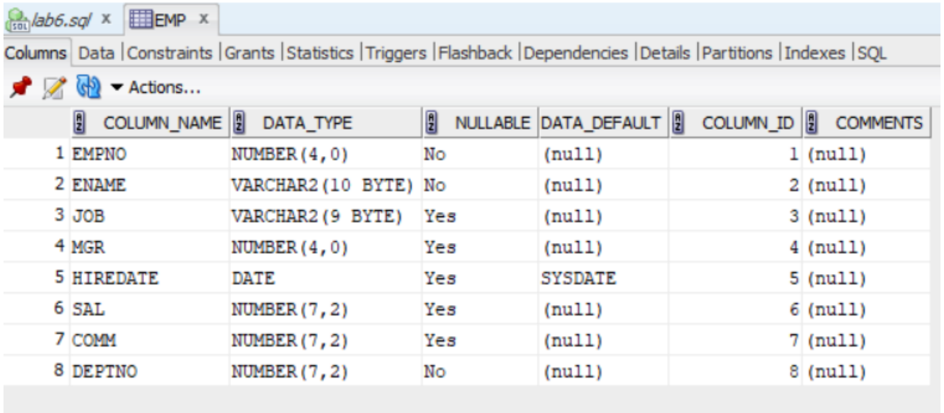


```
/*Create Employee Table*/

CREATE TABLE EMP (
EMPNO   NUMBER(4) CONSTRAINT  EMP_EMPNO_PK  PRIMARY KEY,
ENAME   VARCHAR2(10) NOT NULL,
JOB     VARCHAR2(9),
MGR   NUMBER(4),
HIREDATE   DATE DEFAULT   SYSDATE,
SAL  NUMBER(7, 2), COMM NUMBER(7, 2),
DEPTNO  NUMBER(7, 2)  NOT NULL,
CONSTRAINT EMP_DEPTNO_CK CHECK (DEPTNO BETWEEN 1 AND 50),
Constraint Emp_Deptno_Fk Foreign Key (Deptno)  References   Dept(Deptno));
```

Task completed in 0.06 seconds

```
table EMP created.
```



	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	EMPNO	NUMBER(4,0)	No	(null)	1	(null)
2	ENAME	VARCHAR2(10 BYTE)	No	(null)	2	(null)
3	JOB	VARCHAR2(9 BYTE)	Yes	(null)	3	(null)
4	MGR	NUMBER(4,0)	Yes	(null)	4	(null)
5	HIREDATE	DATE	Yes	SYSDATE	5	(null)
6	SAL	NUMBER(7,2)	Yes	(null)	6	(null)
7	COMM	NUMBER(7,2)	Yes	(null)	7	(null)
8	DEPTNO	NUMBER(7,2)	No	(null)	8	(null)

EXERCISES

Consider the following schema, in the form of normalized relations, to represent information about *employees*, *grades*, *training* and *projects* in an organization.

EMPLOYEE
Empno (eg 6712)
Name
Designation (e.g. *Database Developer*)
Qualification
Joindate

PROJECT
PID (eg P812)
Title
Client
Duration (in weeks)
Status (New, In Progress, Complete)

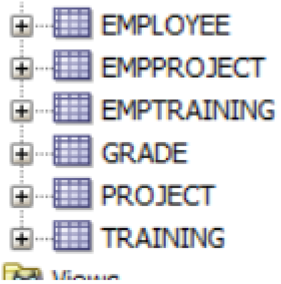
EMP_PROJECT
Empno
PID
Performance (Excellent, Good, Fair, Bad, Poor)

GRADE
Designation
Grade (1-20)
TotalPosts
PostsAvailable (<= TotalPosts)

TRAINING
Tcode (eg T902)
Title
StartDate
EndDate

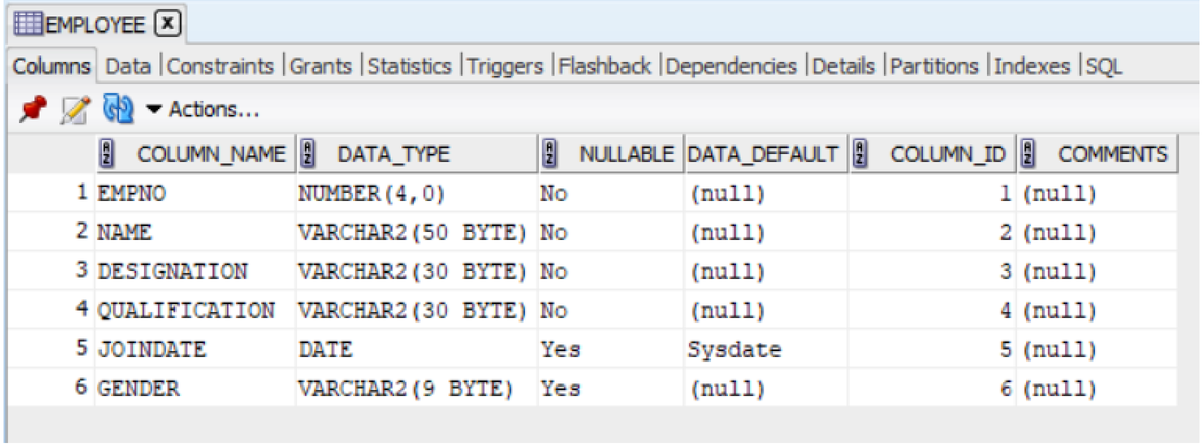
EMP_TRAINING
Empno
Tcode
Attendance (%)

1. Develop a script file **EMPLOYEE.SQL** to create tables for the above schema. Implement all necessary *integrity constraints* including primary and foreign keys. (NOTE: All ***check*** constraints should be at table level)



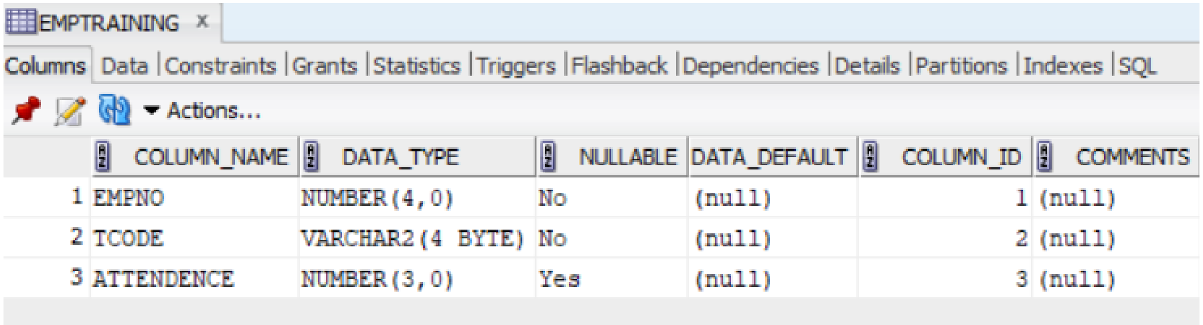
EMPLOYEE
EMPPROJECT
EMPTRAINING
GRADE
PROJECT
TRAINING

2.



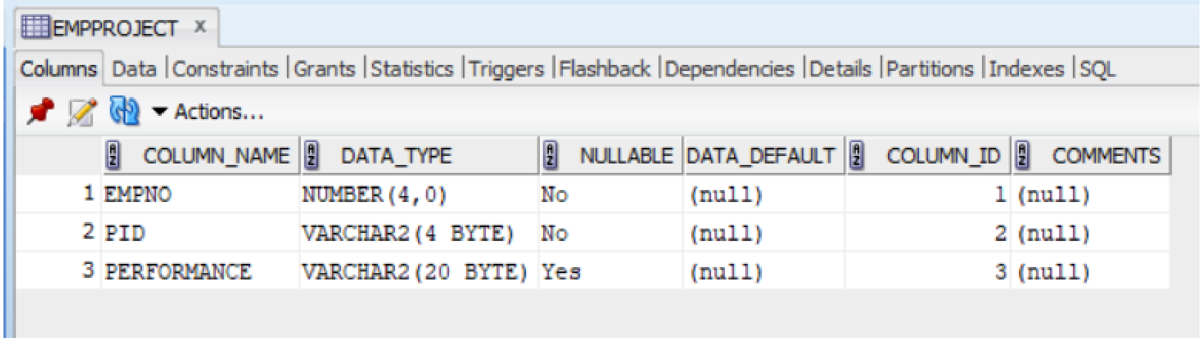
EMPLOYEE
Columns | Data | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL
Actions...

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	EMPNO	NUMBER(4,0)	No	(null)	1	(null)
2	NAME	VARCHAR2(50 BYTE)	No	(null)	2	(null)
3	DESIGNATION	VARCHAR2(30 BYTE)	No	(null)	3	(null)
4	QUALIFICATION	VARCHAR2(30 BYTE)	No	(null)	4	(null)
5	JOINDATE	DATE	Yes	Sysdate	5	(null)
6	GENDER	VARCHAR2(9 BYTE)	Yes	(null)	6	(null)



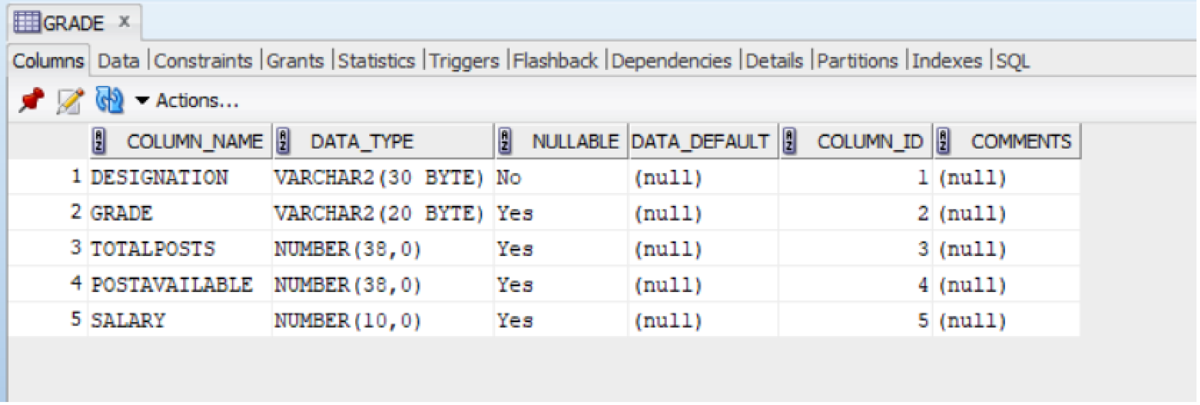
EMPTRAINING
Columns | Data | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL
Actions...

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	EMPNO	NUMBER(4,0)	No	(null)	1	(null)
2	TCODE	VARCHAR2(4 BYTE)	No	(null)	2	(null)
3	ATTENDENCE	NUMBER(3,0)	Yes	(null)	3	(null)

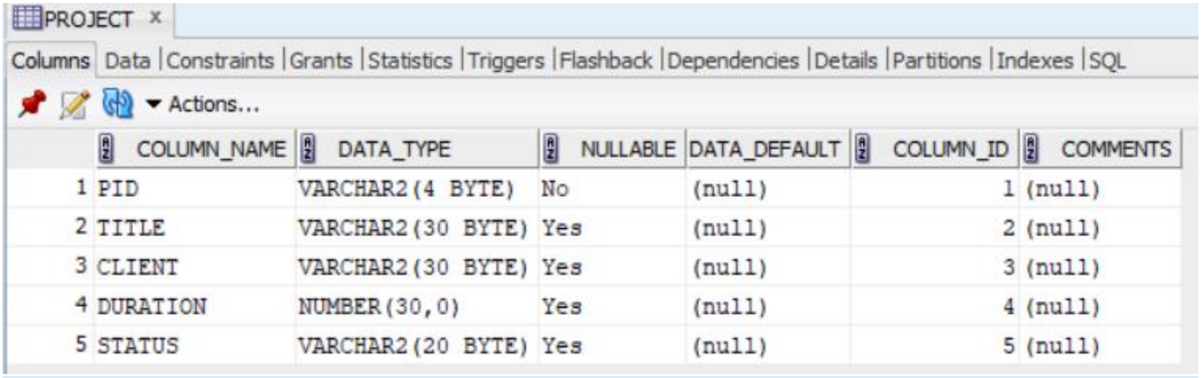


EMPPROJECT
Columns | Data | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL
Actions...

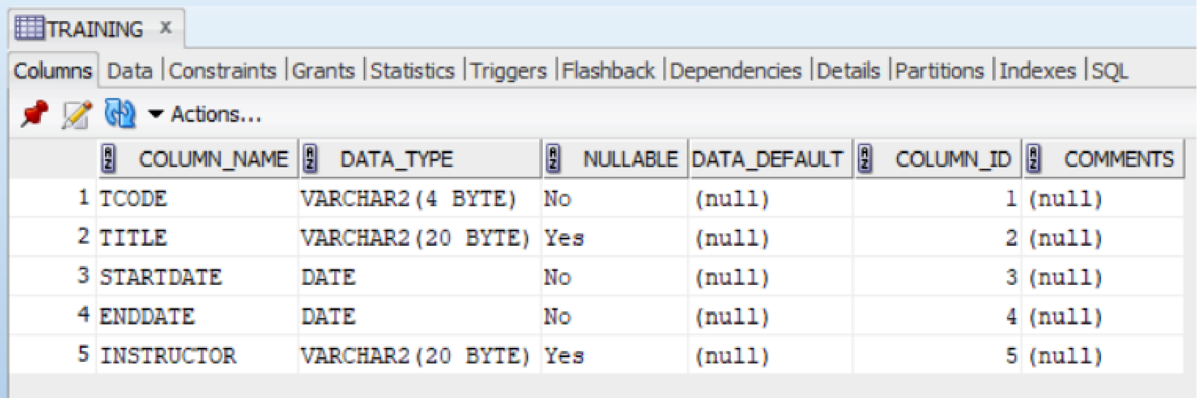
	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	EMPNO	NUMBER(4,0)	No	(null)	1	(null)
2	PID	VARCHAR2(4 BYTE)	No	(null)	2	(null)
3	PERFORMANCE	VARCHAR2(20 BYTE)	Yes	(null)	3	(null)


GRADE ×
Columns | Data | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL
Actions...

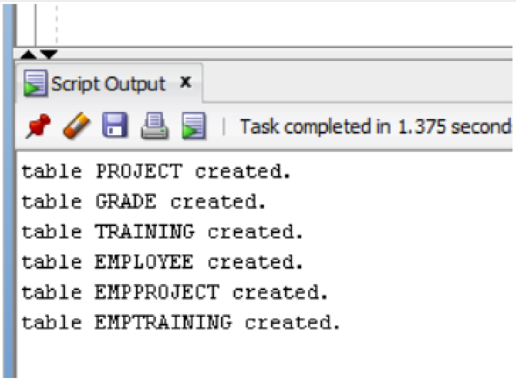
	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	DESIGNATION	VARCHAR2(30 BYTE)	No	(null)	1	(null)
2	GRADE	VARCHAR2(20 BYTE)	Yes	(null)	2	(null)
3	TOTALPOSTS	NUMBER(38,0)	Yes	(null)	3	(null)
4	POSTAVAILABLE	NUMBER(38,0)	Yes	(null)	4	(null)
5	SALARY	NUMBER(10,0)	Yes	(null)	5	(null)


PROJECT ×
Columns | Data | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL
Actions...

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	PID	VARCHAR2(4 BYTE)	No	(null)	1	(null)
2	TITLE	VARCHAR2(30 BYTE)	Yes	(null)	2	(null)
3	CLIENT	VARCHAR2(30 BYTE)	Yes	(null)	3	(null)
4	DURATION	NUMBER(30,0)	Yes	(null)	4	(null)
5	STATUS	VARCHAR2(20 BYTE)	Yes	(null)	5	(null)


TRAINING ×
Columns | Data | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL
Actions...

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	TCODE	VARCHAR2(4 BYTE)	No	(null)	1	(null)
2	TITLE	VARCHAR2(20 BYTE)	Yes	(null)	2	(null)
3	STARTDATE	DATE	No	(null)	3	(null)
4	ENDDATE	DATE	No	(null)	4	(null)
5	INSTRUCTOR	VARCHAR2(20 BYTE)	Yes	(null)	5	(null)

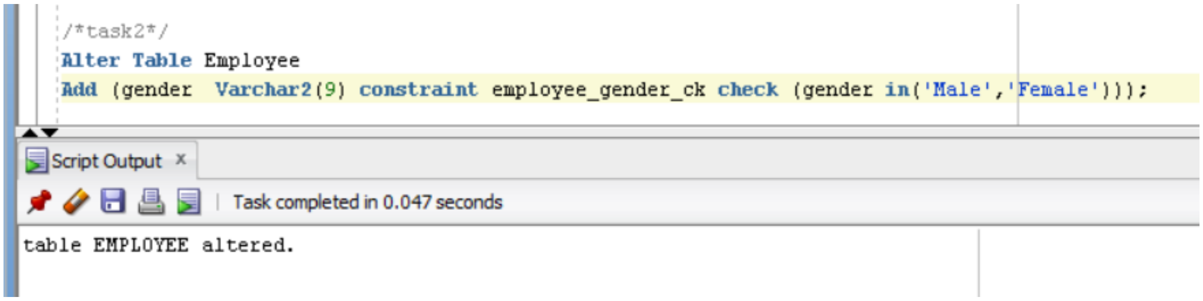


Task completed in 1.375 second

```
table PROJECT created.
table GRADE created.
table TRAINING created.
table EMPLOYEE created.
table EMPPROJECT created.
table EMPTRAINING created.
```

3. Write SQL statements to add

○ *Gender* column to **EMP** table. The only possible values are *Male* and *Female*.

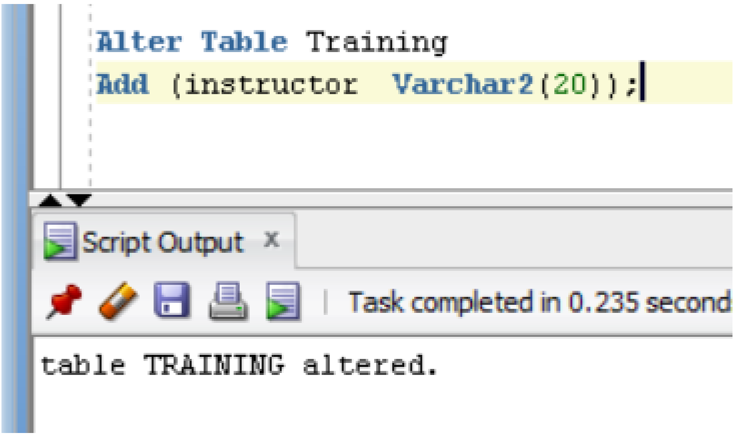


```
/*task2*/
Alter Table Employee
Add (gender  Varchar2(9) constraint employee_gender_ck check (gender in('Male','Female')));
```

Task completed in 0.047 seconds

```
table EMPLOYEE altered.
```

○ *Instructor_Name* column to **TRAINING** table.

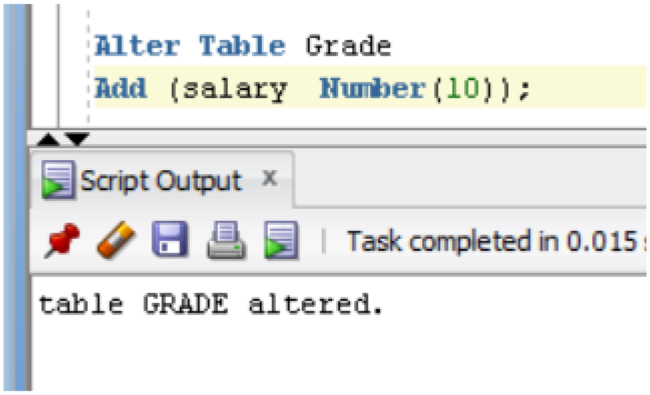


```
Alter Table Training
Add (instructor  Varchar2(20));
```

Task completed in 0.235 second

```
table TRAINING altered.
```

○ *Salary* column to **GRADE** table.



```
Alter Table Grade
Add (salary  Number(10));
```

Task completed in 0.015

```
table GRADE altered.
```

4. What is *database schema*? What are the different objects included in it?

A database schema represents **the logical configuration of all or part of a relational database.** A schema is a collection of objects. Schema objects are the logical structures that directly refer to the data in a database. Schema objects include tables, views, synonyms, sequences, stored procedures, indexes, clusters, and database links.