CSE5011 –

Database Systems and Design

Cycle Sheet – I

(Ex-1 and Ex-2)

Rahul Pathak

21MAI0069

Vellore Institute of Technology

Exercise-I:

- **Table Structure:**
  - **Employee table:**

```
create table employee
(
    first_name varchar(15),
    mid_name char(2),
    last_name varchar(15),
    SSN_number char(9),
    birthday date,
    address varchar(50),
    sex char(1),
    salary number(7),
    supervisor_SSN char(9),
    department_number number(5)
);
```

Results Explain Describe Saved SQL History

```
Table created.
```

0.24 seconds

  - **Department table:**

```
create table department
(
    department_name varchar(15),
    department_number number(5),
    manager_SSN char(9),
    manager_start_date date
);
```

Results Explain Describe Saved SQL History

```
Table created.
```

0.02 seconds

- **Project table:**

```
create table project
(
    project_name varchar(15),
    project_number number(5),
    project_location varchar(15),
    department_number number(5)
);
```

Results Explain Describe Saved SQL History

```
Table created.

0.00 seconds
```

1. **Insert the data given above in both employee, department, and project tables.**

- Employee Table

```
insert into Employee values
('Doug', 'E', 'Gilbert', 123, '09-JUN-1968', 'Chennai', 'M', 80000, null, 1);
insert into Employee values
('Joyce', '', 'PAN', 124, '07-FEB-1973', 'Vellore', 'F', 70000, null, 1);
insert into Employee values
('Frankin', 'T', 'Wong', 125, '08-DEC-1972', 'Delhi', 'M', 40000, 123, 2);
insert into Employee values
('Jennifer', 'S', 'Wallace', 564, '20-JUN-1983', 'Chennai', 'F', 43000, 123, 2);
insert into Employee values
('John', 'B', 'Smith', 678, '09-JAN-1987', 'Madurai', 'M', 30000, 124, 1);
insert into Employee values
('Ramesh', 'K', 'Narayan', 234, '15-SEP-1985', 'Bangalore', 'M', 38000, 124, 3);
```

Results Explain Describe Saved SQL History

```
1 row(s) inserted.

0.00 seconds
```

```
select * from employee
```

Results Explain Describe Saved SQL History

FIRST_NAME	MID_NAME	LAST_NAME	SSN_NUMBER	BIRTHDAY	ADDRESS	SEX	SALARY	SUPERVISOR_SSN	DEPARTMENT_NUMBER
Doug	E	Gilbert	123	09-JUN-68	Chennai	M	80000	-	1
Joyce	-	PAN	124	07-FEB-73	Vellore	F	70000	-	5
Frankin	T	Wong	125	08-DEC-72	Delhi	M	40000	123	2
Jennifer	S	Wallace	564	20-JUN-83	Chennai	F	43000	123	2
John	B	Smith	678	09-JAN-87	Madurai	M	30000	124	1
Ramesh	K	Narayan	234	15-SEP-85	Bangalore	M	38000	124	3

6 rows returned in 0.16 seconds CSV Export

- Department Table

```
insert into department values('Administration', 2, 564, '03-Jan-2012');
insert into department values('Headquarter', 1, 678, '16-Dec-2014');
insert into department values('Finance', 3, 234, '18-May-2013');
insert into department values('IT', 4, 123, '12-Jun-2015');
```

Results Explain Describe Saved SQL History

```
1 row(s) inserted.

0.00 seconds
```

```
select * from department
```

Results Explain Describe Saved SQL History

DEPARTMENT_NAME	DEPARTMENT_NUMBER	MANAGER_SSN	MANAGER_START_DATE	PHNO
Administration	2	564	03-JAN-12	1234567890
Headquarter	1	678	16-DEC-14	1234567890
Finance	3	234	18-MAY-13	1234567890
IT	4	123	12-JUN-15	1234567890

4 rows returned in 0.02 seconds CSV Export

- Project Table

```
insert into project values('ProjectA', 3388, 'Delhi', 1);
insert into project values('ProjectB', 1945, 'Hyderabad', 1);
insert into project values('ProjectC', 6688, 'Chennai', 2);
insert into project values('ProjectD', 2423, 'Chennai', 2);
insert into project values('ProjectE', 7745, 'Bangalore', 3);
```

Results Explain Describe Saved SQL History

```
1 row(s) inserted.
```

0.02 seconds

```
select * from project
```

Results Explain Describe Saved SQL History

PROJECT_NAME	PROJECT_NUMBER	PROJECT_LOCATION	DEPARTMENT_NUMBER
ProjectA	3388	Delhi	1
ProjectB	1945	Hyderabad	1
ProjectC	6688	Chennai	2
ProjectD	2423	Chennai	2
ProjectE	7745	Bangalore	3

5 rows returned in 0.14 seconds CSV Export

2. Display all the employees' information.

```
select * from employee;
```

Results Explain Describe Saved SQL History

FIRST_NAME	MID_NAME	LAST_NAME	SSN_NUMBER	BIRTHDAY	ADDRESS	SEX	SALARY	SUPERVISOR_SSN	DEPARTMENT_NUMBER
Doug	E	Gilbert	123	09-JUN-68	Chennai	M	80000	-	1
Joyce	-	PAN	124	07-FEB-73	Vellore	F	70000	-	1
Frankin	T	Wong	125	08-DEC-72	Delhi	M	40000	123	2
Jennifer	S	Wallace	564	20-JUN-83	Chennai	F	43000	123	2
John	B	Smith	678	09-JAN-87	Madurai	M	30000	124	1
Ramesh	K	Narayan	234	15-SEP-85	Bangalore	M	38000	124	3

6 rows returned in 0.00 seconds CSV Export

3. **Display Employee name along with his SSN and Supervisor SSN.**

```
select first_name, last_name, SSN_number, supervisor_SSN from employee;
```

Results Explain Describe Saved SQL History

FIRST_NAME	LAST_NAME	SSN_NUMBER	SUPERVISOR_SSN
Doug	Gilbert	123	-
Joyce	PAN	124	-
Frankin	Wong	125	123
Jennifer	Wallace	564	123
John	Smith	678	124
Ramesh	Narayan	234	124

6 rows returned in 0.00 seconds CSV Export

4. **Display the employee names whose bdate is '20-JUN-1983'.**

```
select first_name, last_name from employee where birthday='20-JUN-1983';
```

Results Explain Describe Saved SQL History

FIRST_NAME	LAST_NAME
Jennifer	Wallace

1 rows returned in 0.00 seconds CSV Export

5. **Display salary of the employees without duplications.**

```
select distinct salary from employee;
```

Results Explain Describe Saved SQL History

SALARY
38000
43000
30000
40000
80000
70000

6 rows returned in 0.00 seconds CSV Export

6. **Display the MgrSSN, MgrStartDate of the manager of 'Finance' department.**

```
select Manager_SSN, Manager_start_date from department
where department_name = 'Finance';
```

Results Explain Describe Saved SQL History

MANAGER_SSN	MANAGER_START_DATE
234	18-MAY-13

1 rows returned in 0.00 seconds CSV Export

7. **Modify the department number of an employee having fname as 'Joyce' to 5.**

```
update employee
set department_number = 5
where first_name = 'Joyce'
```

Results Explain Describe Saved SQL History

1 row(s) updated.

0.00 seconds

```
update employee
set department_number = 5
where first_name = 'Joyce';

select * from employee
```

Results Explain Describe Saved SQL History

FIRST_NAME	MID_NAME	LAST_NAME	SSN_NUMBER	BIRTHDAY	ADDRESS	SEX	SALARY	SUPERVISOR_SSN	DEPARTMENT_NUMBER
Doug	E	Gilbert	123	09-JUN-68	Chennai	M	80000	-	1
Joyce	-	PAN	124	07-FEB-73	Vellore	F	70000	-	5
Frankin	T	Wong	125	08-DEC-72	Delhi	M	40000	123	2
Jennifer	S	Wallace	564	20-JUN-83	Chennai	F	43000	123	2
John	B	Smith	678	09-JAN-87	Madurai	M	30000	124	1
Ramesh	K	Narayan	234	15-SEP-85	Bangalore	M	38000	124	3

6 rows returned in 0.00 seconds CSV Export

8. **Alter Table department add column DepartmentPhoneNum of NUMBER data type and insert values into this column only.**

```
alter table department add DepartmentPhoneNum Number(10);
```

Results Explain Describe Saved SQL History

Table altered.

0.02 seconds

```
alter table department add DepartmentPhoneNum Number(10);

select * from department;
```

Results Explain Describe Saved SQL History

DEPARTMENT_NAME	DEPARTMENT_NUMBER	MANAGER_SSN	MANAGER_START_DATE	DEPARTMENTPHONENUM
Administration	2	564	03-JAN-12	-
Headquarter	1	678	16-DEC-14	-
Finance	3	234	18-MAY-13	-
IT	4	123	12-JUN-15	-

4 rows returned in 0.00 seconds CSV Export

```
alter table department add DepartmentPhoneNum Number(10);

select * from department;

update department
set DepartmentPhoneNum = 1234567890
```

Results Explain Describe Saved SQL History

4 row(s) updated.

0.02 seconds

```
alter table department add DepartmentPhoneNum Number(10);

update department
set DepartmentPhoneNum = 1234567890

select * from department
```

Results Explain Describe Saved SQL History

DEPARTMENT_NAME	DEPARTMENT_NUMBER	MANAGER_SSN	MANAGER_START_DATE	DEPARTMENTPHONENUM
Administration	2	564	03-JAN-12	1234567890
Headquarter	1	678	16-DEC-14	1234567890
Finance	3	234	18-MAY-13	1234567890
IT	4	123	12-JUN-15	1234567890

4 rows returned in 0.00 seconds CSV Export

9. Alter table department to modify the size of DepartmentPhoneNum.

```
alter table department modify DepartmentPhoneNum Number(12);
```

Results Explain Describe Saved SQL History

```
Table altered.
```

0.04 seconds

10. Modify the field name DepartmentPhoneNum of departments table to PhNo.

```
alter table department rename column DepartmentPhoneNum to PhNo
```

Results Explain Describe Saved SQL History

```
Table altered.
```

0.01 seconds

```
alter table department rename column DepartmentPhoneNum to PhNo

select * from department
```

Results Explain Describe Saved SQL History

DEPARTMENT_NAME	DEPARTMENT_NUMBER	MANAGER_SSN	MANAGER_START_DATE	PHNO
Administration	2	564	03-JAN-12	1234567890
Headquarter	1	678	16-DEC-14	1234567890
Finance	3	234	18-MAY-13	1234567890
IT	4	123	12-JUN-15	1234567890

4 rows returned in 0.00 seconds CSV Export

11. Rename Table Department as DEPT.

```
rename department to DEPT
```

Results Explain Describe Saved SQL History

```
Statement processed.

0.01 seconds
```

```
rename department to DEPT

select * from dept
```

Results Explain Describe Saved SQL History

DEPARTMENT_NAME	DEPARTMENT_NUMBER	MANAGER_SSN	MANAGER_START_DATE	PHNO
Administration	2	564	03-JAN-12	1234567890
Headquarter	1	678	16-DEC-14	1234567890
Finance	3	234	18-MAY-13	1234567890
IT	4	123	12-JUN-15	1234567890

4 rows returned in 0.01 seconds CSV Export

12. Alter Table department remove column PhNo.

```
alter table DEPT drop column PhNo
```

Results Explain Describe Saved SQL History

```
Table dropped.
```

0.19 seconds

```
alter table DEPT drop column PhNo
select * from dept
```

Results Explain Describe Saved SQL History

DEPARTMENT_NAME	DEPARTMENT_NUMBER	MANAGER_SSN	MANAGER_START_DATE
Administration	2	564	03-JAN-12
Headquarter	1	678	16-DEC-14
Finance	3	234	18-MAY-13
IT	4	123	12-JUN-15

4 rows returned in 0.00 seconds CSV Export

13. Create a table COPYOFDEPT as a copy of the table DEPT.

```
create table copyOfDept as select * from dept
```

Results Explain Describe Saved SQL History

```
Table created.
```

0.28 seconds

```
create table copyOfDept as select * from dept

select * from copyOfDept
```

Results Explain Describe Saved SQL History

DEPARTMENT_NAME	DEPARTMENT_NUMBER	MANAGER_SSN	MANAGER_START_DATE
Administration	2	564	03-JAN-12
Headquarter	1	678	16-DEC-14
Finance	3	234	18-MAY-13
IT	4	123	12-JUN-15

4 rows returned in 0.05 seconds CSV Export

14. Delete all the rows from COPYOF DEPT table.

```
delete from copyOfDept
```

Results Explain Describe Saved SQL History

```
4 row(s) deleted.
```

0.00 seconds

```
delete from copyOfDept

select * from copyOfDept
```

Results Explain Describe Saved SQL History

no data found

15. Remove COPYOF DEPT table.

```
drop table copyOfDept
```

Results Explain Describe Saved SQL History

```
Table dropped.
```

0.11 seconds

Exercise-2:

Exercise-2 deals with the constraints, so we are going to add constraint to the tables as described below:

- Employee table:

Table Name: Employee

Attribute	Data Type	Constraint
First Name	Varchar (15)	Not Null
Mid Name	Char(2)	
Last Name	Varchar (15)	Not Null
SSN Number	Char (9)	Primary Key
Birthday	Date	
Address	Varchar (50)	
Sex	Char(1)	Sex In (M,F,m,f)
Salary	Number (7)	Default 800
Supervisor SSN	Char (9)	Foreign Key Employee (SSN) on delete set null
Department number	Number(5)	Foreign key to department number of department table on delete cascade

---

- We will alter the existing table using **alter** commands. But before, we check if there exists any constraint on the table.

```
select * from USER_CONSTRAINTS where TABLE_NAME = 'EMPLOYEE';
```

Results Explain Describe Saved SQL History

no data found

- There are no constraints on the table, so, we add the above mentioned constraints

```
alter table employee modify (first_name varchar(15) constraint con_notnull not null);
alter table employee modify (last_name varchar(15) constraint con_notnull2 not null);
alter table employee add constraint con_pk1 primary key(SSN_number);
alter table employee add constraint con_in1 check(Sex in ('M', 'F', 'm', 'f'));
alter table employee modify (salary Number(7) constraint con_notnull3 not null);
alter table employee modify salary default (800);
alter table employee add constraint con_fk1 foreign key (SUPERVISOR_SSN) references employee(SSN_NUMBER) on delete set null;
```

Results Explain Describe Saved SQL History

```
Table altered.
```

0.11 seconds

```
select * from USER_CONSTRAINTS where TABLE_NAME = 'EMPLOYEE';
```

Results Explain Describe Saved SQL History

OWNER	CONSTRAINT_NAME	CONSTRAINT_TYPE	TABLE_NAME	SEARCH_CONDITION	R_OWNER	R_CONSTRAINT_NAME	DELETE_RULE	STATUS	DEFERRABLE	DEFERRED	VALIDATED	GENERATED
SYS	CON_NOTNULL	C	EMPLOYEE	"FIRST_NAME" IS NOT NULL	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME
SYS	CON_NOTNULL2	C	EMPLOYEE	"LAST_NAME" IS NOT NULL	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME
SYS	CON_PK1	P	EMPLOYEE	-	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME
SYS	CON_IN1	C	EMPLOYEE	Sex in ('M', 'F', 'm', 'f')	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME
SYS	CON_NOTNULL3	C	EMPLOYEE	"SALARY" IS NOT NULL	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME
SYS	CON_FK1	R	EMPLOYEE	-	SYS	CON_PK1	SET NULL	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME

- While adding foreign key constraint in **DEPARTMENT_NUMBER** column we get the following error, it is due to the primary key or unique key does not present in referenced table-column i.e., **department table DEPARTMENT_NUMBER column**

```
alter table employee add constraint con_fk2 foreign key (DEPARTMENT_NUMBER) references department(DEPARTMENT_NUMBER) on delete cascade;
```

Results Explain Describe Saved SQL History

```
ORA-02270: no matching unique or primary key for this column-list
```

0.10 seconds

- So, to make it work, we add **primary key** constraint to **department table DEPARTMENT_NUMBER column** first then we add **foreign key** constraint in **employee table DEPARTMENT_NUMBER column.**

```
alter table department add constraint con_pk2 primary key(DEPARTMENT_NUMBER);
select * from USER_CONSTRAINTS where TABLE_NAME = 'DEPARTMENT';
```

Results Explain Describe Saved SQL History

OWNER	CONSTRAINT_NAME	CONSTRAINT_TYPE	TABLE_NAME	SEARCH_CONDITION	R_OWNER	R_CONSTRAINT_NAME	DELETE_RULE	STATUS	DEFERRABLE	DEFERRED	VALIDATED	GENERATED
SYS	CON_PK2	P	DEPARTMENT	-	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME

1 rows returned in 0.11 seconds CSV Export

```
alter table employee add constraint con_fk2 foreign key (DEPARTMENT_NUMBER) references department(DEPARTMENT_NUMBER) on delete cascade;
```

Results Explain Describe Saved SQL History

```
ORA-02298: cannot validate (SYS.CON_FK2) - parent keys not found
```

0.03 seconds

- Again, we get another error regarding **parent keys not found** – this is due to the **unmatched values in parent key column and foreign key column**. So, we **compare** the values in both the tables employee and department and **make an update** to match the values in both primary key and foreign key column.

```
select * from department
```

Results Explain Describe Saved SQL History

DEPARTMENT_NAME	DEPARTMENT_NUMBER	MANAGER_SSN	MANAGER_START_DATE	PHNO
Administration	2	564	03-JAN-12	1234567890
Headquarter	1	678	16-DEC-14	1234567890
Finance	3	234	18-MAY-13	1234567890
IT	4	123	12-JUN-15	1234567890

4 rows returned in 0.01 seconds CSV Export

```
select * from department
select * from employee
```

Results Explain Describe Saved SQL History

FIRST_NAME	MID_NAME	LAST_NAME	SSN_NUMBER	BIRTHDAY	ADDRESS	SEX	SALARY	SUPERVISOR_SSN	DEPARTMENT_NUMBER
Doug	E	Gilbert	123	09-JUN-68	Chennai	M	80000	-	1
Joyce	-	PAN	124	07-FEB-73	Vellore	F	70000	-	5
Frankin	T	Wong	125	08-DEC-72	Delhi	M	40000	123	2
Jennifer	S	Wallace	564	20-JUN-83	Chennai	F	43000	123	2
John	B	Smith	678	09-JAN-87	Madurai	M	30000	124	1
Ramesh	K	Narayan	234	15-SEP-85	Bangalore	M	38000	124	3

6 rows returned in 0.02 seconds CSV Export

```
select * from department
select * from employee

update employee
set department_number = null
where department_number > 4;
```

Results Explain Describe Saved SQL History

```
1 row(s) updated.
```

0.08 seconds

- Now we alter table to see the changes

```
alter table employee add constraint con_fk2 foreign key (DEPARTMENT_NUMBER) references department(DEPARTMENT_NUMBER) on delete cascade;
```

Results Explain Describe Saved SQL History

```
Table altered.
```

0.03 seconds

```
alter table employee add constraint con_fk2 foreign key (DEPARTMENT_NUMBER) references department(DEPARTMENT_NUMBER) on delete cascade;
select * from USER_CONSTRAINTS where TABLE_NAME = 'EMPLOYEE';
```

Results Explain Describe Saved SQL History

OWNER	CONSTRAINT_NAME	CONSTRAINT_TYPE	TABLE_NAME	SEARCH_CONDITION	R_OWNER	R_CONSTRAINT_NAME	DELETE_RULE	STATUS	DEFERRABLE	DEFERRED	VALIDATED	GENERATED
SYS	CON_NOTNULL	C	EMPLOYEE	"FIRST_NAME" IS NOT NULL	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME
SYS	CON_NOTNULL2	C	EMPLOYEE	"LAST_NAME" IS NOT NULL	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME
SYS	CON_PK1	P	EMPLOYEE	-	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME
SYS	CON_IN1	C	EMPLOYEE	Sex in ('M', 'F', 'm', 'f')	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME
SYS	CON_NOTNULL3	C	EMPLOYEE	"SALARY" IS NOT NULL	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME
SYS	CON_FK1	R	EMPLOYEE	-	SYS	CON_PK1	SET NULL	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME
SYS	CON_FK2	R	EMPLOYEE	-	SYS	CON_PK2	CASCADE	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME

7 rows returned in 0.02 seconds CSV Export

- <u>Department Table</u>

Table Name : Department

Attribute	Data type	Constraint
Department Name	Varchar(15)	Not Null
Department number	INT(5)	Primary key
Manager SSN	Char (9)	Foreign key-Employee (SSN) on delete set null
Manage start date	Date	

- We already applied **primary key** to **DEPARTMENT_NUMBER**, now we alter table to add more constraints as described.

```
alter table department modify (DEPARTMENT_NAME varchar(15) constraint con_notnull4 not null);
alter table department add constraint con_fk3 foreign key (MANAGER_SSN) references employee(SSN_NUMBER) on delete set null;
```

Results Explain Describe Saved SQL History

```
Table altered.
```

0.05 seconds

```
alter table department modify (DEPARTMENT_NAME varchar(15) constraint con_notnull4 not null);
alter table department add constraint con_fk3 foreign key (MANAGER_SSN) references employee(SSN_NUMBER) on delete set null;

select * from USER_CONSTRAINTS where TABLE_NAME = 'DEPARTMENT';
```

Results Explain Describe Saved SQL History

OWNER	CONSTRAINT_NAME	CONSTRAINT_TYPE	TABLE_NAME	SEARCH_CONDITION	R_OWNER	R_CONSTRAINT_NAME	DELETE_RULE	STATUS	DEFERRABLE	DEFERRED	VALIDATED	GENERATED	B
SYS	CON_PK2	P	DEPARTMENT	-	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME	-
SYS	CON_NOTNULL4	C	DEPARTMENT	"DEPARTMENT_NAME" IS NOT NULL	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME	-
SYS	CON_FK3	R	DEPARTMENT	-	SYS	CON_PK1	SET NULL	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME	-

3 rows returned in 0.00 seconds CSV Export

- Dept_locations Table

Table Name : Dept_locations

Attribute	Data type	Constraint
Department Number	Number(5)	Department (dep no) onDelete Cascade
Department Location	Varchar (15)	

- We do not have **dept_locations** table, so we create table with constraints

```
create table dept_locations
(
   dept_no Number(5),
   dept_loc varchar(15),
   constraint con_fk4 foreign key(dept_no)
   references department(DEPARTMENT_NUMBER)
   on delete cascade
)
```

Results Explain Describe Saved SQL History

```
Table created.

0.18 seconds
```

```
select * from USER_CONSTRAINTS where TABLE_NAME = 'DEPT_LOCATIONS';
```

Results Explain Describe Saved SQL History

OWNER	CONSTRAINT_NAME	CONSTRAINT_TYPE	TABLE_NAME	SEARCH_CONDITION	R_OWNER	R_CONSTRAINT_NAME	DELETE_RULE	STATUS	DEFERRABLE	DEFERRED	VALIDATED	GENERATED
SYS	CON_FK4	R	DEPT_LOCATIONS	-	SYS	CON_PK2	CASCADE	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME

1 rows returned in 0.06 seconds CSV Export

- Project Table

Table Name: Project

Attribute	Data type	Constraint
Project Name	Varchar2(15)	Not Null
Project number	Number(5)	Primary key
Project Location	Varchar2(50)	
Department Number	Number(5)	Foreign Key –Department (dep no) on delete set null

---

- We already have a Project Table from ex-1. Here we alter table to add constraints as described

```
alter table project modify (PROJECT_NAME varchar2(15) constraint con_notnull5 not null);
alter table project add constraint con_pk3 primary key(PROJECT_NUMBER);
alter table project add constraint con_fk5 foreign key (DEPARTMENT_NUMBER) references department(DEPARTMENT_NUMBER) on delete set null;
```

Results Explain Describe Saved SQL History

```
Table altered.
```

0.02 seconds

```
select * from USER_CONSTRAINTS where TABLE_NAME = 'PROJECT';
```

Results Explain Describe Saved SQL History

OWNER	CONSTRAINT_NAME	CONSTRAINT_TYPE	TABLE_NAME	SEARCH_CONDITION	R_OWNER	R_CONSTRAINT_NAME	DELETE_RULE	STATUS	DEFERRABLE	DEFERRED	VALIDATED	GENERATED	B
SYS	CON_NOTNULL5	C	PROJECT	"PROJECT_NAME" IS NOT NULL	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME	-
SYS	CON_PK3	P	PROJECT	-	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME	-
SYS	CON_FK5	R	PROJECT	-	SYS	CON_PK2	SET NULL	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	USER NAME	-

3 rows returned in 0.03 seconds CSV Export

- Works_on Table

Table Name: Works_On

The combination of Employee SSN and Project Number must be a Primary Key

Attribute	Data type	Constraint
Employee SSN	Char (9)	Foreign Key Employee (SSN) on delete cascade
Project number	INT(5)	Foreign Key project (Pnumber) on delete cascade
Hours	Decimal (3,1)	Not null

- We do not have Works_on Table. So, we create table with described constraints.

```
create table works_on
(
   employee_SSN char(9),
   project_no number(5),
   hours decimal(3,1) not null,
   primary key(employee_SSN, project_no),
   foreign key(employee_SSN) references employee(SSN_NUMBER)
   on delete cascade,
   foreign key(project_no) references project(PROJECT_NUMBER)
   on delete cascade
)
```

Results Explain Describe Saved SQL History

```
Table created.

0.11 seconds
```

```
select * from USER_CONSTRAINTS where TABLE_NAME = 'WORKS_ON';
```

Results Explain Describe Saved SQL History

OWNER	CONSTRAINT_NAME	CONSTRAINT_TYPE	TABLE_NAME	SEARCH_CONDITION	R_OWNER	R_CONSTRAINT_NAME	DELETE_RULE	STATUS	DEFERRABLE	DEFERRED	VALIDATED	GENERATED	B
SYS	SYS_C004166	C	WORKS_ON	"HOURS" IS NOT NULL	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	GENERATED NAME	-
SYS	SYS_C004167	P	WORKS_ON	-	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	GENERATED NAME	-
SYS	SYS_C004168	R	WORKS_ON	-	SYS	CON_PK1	CASCADE	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	GENERATED NAME	-
SYS	SYS_C004169	R	WORKS_ON	-	SYS	CON_PK3	CASCADE	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	GENERATED NAME	-

4 rows returned in 0.06 seconds CSV Export

- <u>Dependent Table</u>

Name: Dependent

The combination of Employee SSN and Dependent Name must be a Primary Key.

Attribute	Datatype	Constraint
Employee	Char (9)	Foreign Key- Employee (SSN) on Delete Cascade
Dependent Name	Varchar(15)	
Sex	Char(1)	Check Sex in (M,F,m,f)
Birthday	Date	
Relationship	Varchar(8)	

- We also do not have dependent table. So, we create table with described constraints.

```
create table dependent
(
  employee_SSN char(9),
  depn_name varchar(15),
  sex char(1) check(sex in('M','F','m','f')),
  birthday date,
  relationship varchar(8),
  primary key(employee_SSN, depn_name),
  foreign key(employee_SSN) references employee(SSN_NUMBER)
  on delete cascade
)
```

Results Explain Describe Saved SQL History

```
Table created.
```

0.05 seconds

```
select * from USER_CONSTRAINTS where TABLE_NAME = 'DEPENDENT';
```

Results Explain Describe Saved SQL History

OWNER	CONSTRAINT_NAME	CONSTRAINT_TYPE	TABLE_NAME	SEARCH_CONDITION	R_OWNER	R_CONSTRAINT_NAME	DELETE_RULE	STATUS	DEFERRABLE	DEFERRED	VALIDATED	GENERATED	B
SYS	SYS_C004170	C	DEPENDENT	sex in('M','F','m','f')	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	GENERATED NAME	-
SYS	SYS_C004171	P	DEPENDENT	-	-	-	-	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	GENERATED NAME	-
SYS	SYS_C004172	R	DEPENDENT	-	SYS	CON_PK1	CASCADE	ENABLED	NOT DEFERRABLE	IMMEDIATE	VALIDATED	GENERATED NAME	-

3 rows returned in 0.01 seconds CSV Export

- <u>Data insert for dept_locations</u>

Data for table - Dept_Locations

Dep No	D Location
1	Houston
1	Chicago
2	New York
2	San Francisco
3	Salt Lake City
4	Stafford
4	Bellaire
5	Sugarland
5	Houston

- Since **dept_locations** table is **linked** with **foreign key** to **department** table, we need to insert some dummy data in department table first to avoid error inserting all the given data to dept_locations.

```
insert into department values('Production', 5, 125, '29-SEP-21')
```

Results Explain Describe Saved SQL History

1 row(s) inserted.

0.01 seconds

```
select * from department
```

Results Explain Describe Saved SQL History

DEPARTMENT_NAME	DEPARTMENT_NUMBER	MANAGER_SSN	MANAGER_START_DATE
Finance	3	234	18-MAY-13
IT	4	123	12-JUN-15
Administration	2	564	03-JAN-12
Headquarter	1	678	16-DEC-14
Production	5	125	29-SEP-21

5 rows returned in 0.00 seconds CSV Export

```
insert into dept_locations values(1, 'Houston')
insert into dept_locations values(1, 'Chicago')
insert into dept_locations values(2, 'New York')
insert into dept_locations values(2, 'San Francisco')
insert into dept_locations values(3, 'Salt Lake City')
insert into dept_locations values(4, 'Stafford')
insert into dept_locations values(4, 'Bellaire')
insert into dept_locations values(5, 'Sugarland')
insert into dept_locations values(5, 'Houston')
```

Results Explain Describe Saved SQL History

1 row(s) inserted.

0.00 seconds

```
select * from dept_locations
```

Results Explain Describe Saved SQL History

DEP_NO	DEP_LOC
1	Houston
1	Chicago
2	New York
2	San Francisco
3	Salt Lake City
4	Stafford
4	Bellaire
5	Sugarland
5	Houston

9 rows returned in 0.01 seconds CSV Export

- Data insert for works_on

Data for Table - Works_On

ESSN	Pno	Hours
123456789	3388	32.5
123456789	1945	7.5
666884444	3388	40.0
453453453	77	20.0
453453453	22	20.0
333445555	77	10.0
333445555	6688	10.0
333445555	43	35.0
333445555	22	28.5
999887777	1	11.5
999887777	12	13.0
543216789	22	17.0
554433221	1945	21.5

- Since works_on **ESSN** is linked with **foreign key** to **Employee(SSN)** and **Pno** is linked with **foreign key** to **Project(Pnumber)** we must insert some **dummy data to employee table and project table** first to avoid error inserting above data.

```
insert into employee values('Robert', 'D', 'Junior', 123456789, '04-APR-65', 'West Burbank CA 91506, USA', 'M', 90000, 123, 5);
insert into employee values('Chris', '', 'Evans', 666884444, '13-JUN-81', 'Beverly Hills, CA 90212, USA', 'M', 91000, 123456789, 5);
insert into employee values('Chris', '', 'Hemsworth', 453453453, '11-AUG-83', '9255 W Sunset Blvd, USA', 'M', 92000, 666884444, 1);
insert into employee values('Elizabeth', '', 'Olsen', 333445555, '16-FEB-89', 'Sherman Oaks, California, USA.', 'F', 93000, 453453453, 1);
insert into employee values('Scarlett', '', 'Johansson', 999887777, '22-NOV-84', 'Manhattan, New York City, USA', 'F', 94000, 123456789, 1);
insert into employee values('Mark ', 'A', 'Ruffalo', 543216789, '22-NOV-67', 'Beverly Hills, USA', 'M', 89000, 453453453, 2);
insert into employee values('Tom', '', 'Hiddleston', 554433221, '09-FEB-81', 'Beverly Hills, USA', 'M', 88000, null, 2);
```

Results Explain Describe Saved SQL History

```
1 row(s) inserted.

0.03 seconds
```

```
select * from employee
```

Results Explain Describe Saved SQL History

FIRST_NAME	MID_NAME	LAST_NAME	SSN_NUMBER	BIRTHDAY	ADDRESS	SEX	SALARY	SUPERVISOR_SSN	DEPARTMENT_NUMBER
Frankin	T	Wong	125	08-DEC-72	Delhi	M	40000	123	2
Jennifer	S	Wallace	564	20-JUN-83	Chennai	F	43000	123	2
John	B	Smith	678	09-JAN-87	Madurai	M	30000	124	1
Ramesh	K	Narayan	234	15-SEP-85	Bangalore	M	38000	124	3
Chris	-	Evans	666884444	13-JUN-81	Beverly Hills, CA 90212, USA	M	91000	123456789	5
Chris	-	Hemsworth	453453453	11-AUG-83	9255 W Sunset Blvd, USA	M	92000	666884444	1
Elizabeth	-	Olsen	333445555	16-FEB-89	Sherman Oaks, California, USA.	F	93000	453453453	1
Scarlett	-	Johansson	999887777	22-NOV-84	Manhattan, New York City, USA	F	94000	123456789	1
Mark	A	Ruffalo	543216789	22-NOV-67	Beverly Hills, USA	M	89000	453453453	2
Tom	-	Hiddleston	554433221	09-FEB-81	Beverly Hills, USA	M	88000	-	2

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds CSV Export

```
insert into project values('ProjectF', 77, 'Mumbai', 5)
insert into project values('ProjectG', 22, 'Sikkim', 5)
insert into project values('ProjectH', 43, 'Pune', 4)
insert into project values('ProjectI', 1, 'Patna', 3)
insert into project values('ProjectG', 12, 'Chandigarh', 3)
```

Results Explain Describe Saved SQL History

```
1 row(s) inserted.

0.00 seconds
```

```
select * from project
```

Results Explain Describe Saved SQL History

PROJECT_NAME	PROJECT_NUMBER	PROJECT_LOCATION	DEPARTMENT_NUMBER
ProjectA	3388	Delhi	1
ProjectB	1945	Hyderabad	1
ProjectC	6688	Chennai	2
ProjectD	2423	Chennai	2
ProjectE	7745	Bangalore	3
ProjectF	77	Mumbai	5
ProjectG	22	Sikkim	5
ProjectH	43	Pune	4
ProjectI	1	Patna	3
ProjectG	12	Chandigarh	3

10 rows returned in 0.00 seconds CSV Export

```
insert into works_on values(123456789, 3388, 32.5)
insert into works_on values(123456789, 1945, 7.5)
insert into works_on values(666884444, 3388, 40.0)
insert into works_on values(453453453, 77, 20.0)
insert into works_on values(453453453, 22, 20.0)
insert into works_on values(333445555, 77, 10.0)
insert into works_on values(333445555, 6688, 10.0)
insert into works_on values(333445555, 43, 35.0)
insert into works_on values(333445555, 22, 28.5)
insert into works_on values(999887777, 1, 11.5)
insert into works_on values(999887777, 12, 13.0)
insert into works_on values(543216789, 22, 17.0)
insert into works_on values(554433221, 1945, 21.5)
```

Results Explain Describe Saved SQL History

```
1 row(s) inserted.
```

0.00 seconds

```
select * from works_on
```

Results Explain Describe Saved SQL History

EMPLOYEE_SSN	PROJECT_NO	HOURS
123456789	3388	32.5
123456789	1945	7.5
666884444	3388	40
453453453	77	20
453453453	22	20
999887777	12	13
543216789	22	17
554433221	1945	21.5
333445555	77	10
333445555	6688	10
333445555	43	35
333445555	22	28.5
999887777	1	11.5

13 rows returned in 0.00 seconds CSV Export

- Data insert for Dependent

Data for Table - Dependent

ESSN	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	05-Apr-76	Daughter
333445555	Theodore	M	25-Oct-73	Son
333445555	Joy	F	03-May-48	Spouse
987654321	Abner	M	29-Feb-32	Spouse
123456789	Alice	F	31-Dec-78	Daughter
123456789	Elizabeth	F	05-may-57	Spouse

- Since **Dependent** table is linked with **foreign key** to **Employee(SSN)** we need to insert **dummy data to employee** first to avoid errors while inserting above data

```
insert into employee values('Tom', '', 'Holland', 987654321, '01-JUN-96', 'Beverly Hills, USA', 'M', 87000, 333445555, 3);
```

Results Explain Describe Saved SQL History

```
1 row(s) inserted.
```

```
select * from employee
```

Results Explain Describe Saved SQL History

FIRST_NAME	MID_NAME	LAST_NAME	SSN_NUMBER	BIRTHDAY	ADDRESS	SEX	SALARY	SUPERVISOR_SSN	DEPARTMENT_NUMBER
Frankin	T	Wong	125	08-DEC-72	Delhi	M	40000	123	2
Jennifer	S	Wallace	564	20-JUN-83	Chennai	F	43000	123	2
John	B	Smith	678	09-JAN-87	Madurai	M	30000	124	1
Ramesh	K	Narayan	234	15-SEP-85	Bangalore	M	38000	124	3
Chris	-	Evans	666884444	13-JUN-81	Beverly Hills, CA 90212, USA	M	91000	123456789	5
Chris	-	Hemsworth	453453453	11-AUG-83	9255 W Sunset Blvd, USA	M	92000	666884444	1
Elizabeth	-	Olsen	333445555	16-FEB-89	Sherman Oaks, California, USA.	F	93000	453453453	1
Scarlett	-	Johansson	999887777	22-NOV-84	Manhattan, New York City, USA	F	94000	123456789	1
Mark	A	Ruffalo	543216789	22-NOV-67	Beverly Hills, USA	M	89000	453453453	2
Tom	-	Hiddleston	554433221	09-FEB-81	Beverly Hills, USA	M	88000	-	2
Tom	-	Holland	987654321	01-JUN-96	Beverly Hills, USA	M	87000	333445555	3
Doug	E	Gilbert	123	09-JUN-68	Chennai	M	80000	-	1
Joyce	-	PAN	124	07-FEB-73	Vellore	F	70000	-	-
Robert	D	Junior	123456789	04-APR-65	West Burbank CA 91506, USA	M	90000	123	5

14 rows returned in 0.00 seconds CSV Export

```
insert into dependent values(333445555, 'Alice', 'F', '05-Apr-76', 'Daughter')
insert into dependent values(333445555, 'Theodore', 'M', '25-Oct-73', 'Son')
insert into dependent values(333445555, 'Joy', 'F', '03-May-48', 'Spouse')
insert into dependent values(987654321, 'Abner', 'M', '29-Feb-32', 'Spouse')
insert into dependent values(123456789, 'Alice', 'F', '31-Dec-78', 'Daughter')
insert into dependent values(123456789, 'Elizabeth', 'F', '05-may-57', 'Spouse')
```

Results Explain Describe Saved SQL History

```
1 row(s) inserted.
```

0.00 seconds

```
select * from dependent
```

Results | Explain | Describe | Saved SQL | History

EMPLOYEE_SSN	DEPN_NAME	SEX	BIRTHDAY	RELATIONSHIP
333445555	Alice	F	05-APR-76	Daughter
333445555	Theodore	M	25-OCT-73	Son
333445555	Joy	F	03-MAY-48	Spouse
987654321	Abner	M	29-FEB-32	Spouse
123456789	Alice	F	31-DEC-78	Daughter
123456789	Elizabeth	F	05-MAY-57	Spouse

6 rows returned in 0.00 seconds CSV Export

Execute the following Query on the Db to display and discuss the integrity constraints violated by any of the following operations

1. Insert ('Robert', 'F', 'Scott', '943775543', '21-JUN-42', '2365 Newcastle Rd, Bellaire, TX', M, 58000, '888665555', 1) into EMPLOYEE.

```
insert into employee values('Robert', 'F', 'Scott', '943775543', '21-JUN-42', '2365 Newcastle Rd, Bellaire, TX', 'M', 58000, '888665555', 1)
```

Results Explain Describe Saved SQL History

```
ORA-02291: integrity constraint (LAB2.CON_FK1) violated - parent key not found
```

- Here we get an **integrity constraint violated – parent key not found error.** This is due to the referencing column does not find the keys in the parent column, in other words primary key column not having a value referenced by foreign key column. In this case the value **'888665555'** referenced by **SUPERVISOR_SSN** column is not present in the primary key column **SSN_NUMBER** as we can see from table below.

```
select * from employee
```

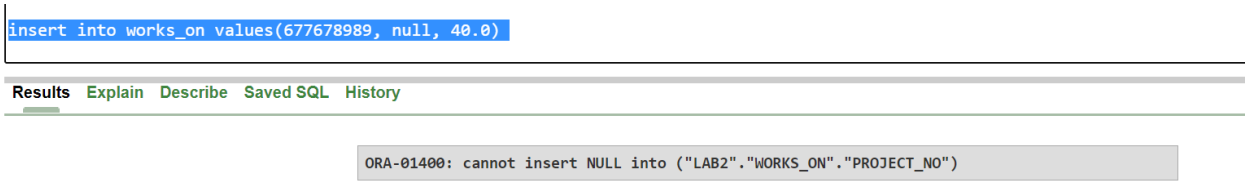
Results Explain Describe Saved SQL History

FIRST_NAME	MID_NAME	LAST_NAME	SSN_NUMBER	BIRTHDAY	ADDRESS	SEX	SALARY	SUPERVISOR_SSN	DEPARTMENT_NUMBER
Frankin	T	Wong	125	08-DEC-72	Delhi	M	40000	123	2
Jennifer	S	Wallace	564	20-JUN-83	Chennai	F	43000	123	2
John	B	Smith	678	09-JAN-87	Madurai	M	30000	124	1
Ramesh	K	Narayan	234	15-SEP-85	Bangalore	M	38000	124	3
Chris	-	Evans	666884444	13-JUN-81	Beverly Hills, CA 90212, USA	M	91000	123456789	5
Chris	-	Hemsworth	453453453	11-AUG-83	9255 W Sunset Blvd, USA	M	92000	666884444	1
Elizabeth	-	Olsen	333445555	16-FEB-89	Sherman Oaks, California, USA.	F	93000	453453453	1
Scarlett	-	Johansson	999887777	22-NOV-84	Manhattan, New York City, USA	F	94000	123456789	1
Mark	A	Ruffalo	543216789	22-NOV-67	Beverly Hills, USA	M	89000	453453453	2
Tom	-	Hiddleston	554433221	09-FEB-81	Beverly Hills, USA	M	88000	-	2
Tom	-	Holland	987654321	01-JUN-96	Beverly Hills, USA	M	87000	333445555	3
Doug	E	Gilbert	123	09-JUN-68	Chennai	M	80000	-	1
Joyce	-	PAN	124	07-FEB-73	Vellore	F	70000	-	-
Robert	D	Junior	123456789	04-APR-65	West Burbank CA 91506, USA	M	90000	123	5

14 rows returned in 0.00 seconds CSV Export

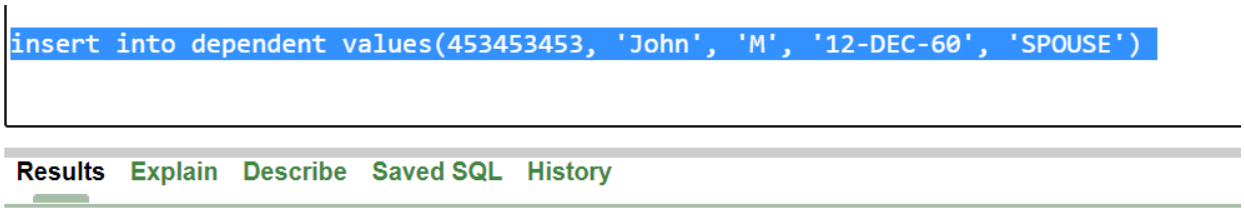
- Thus, it avoids inserting values into the referenced column that is not present in the parent column.

2. Insert ('677678989', null, '40.0') into WORKS_ON.



- Error statement: **cannot insert NULL into("WORKS_ON"."PROJECT_NO")** which indicates that **PROJECT_NO Column** of **WORKS_ON table** is a **NOT NULL** column that does not accepts null values.
- Although we have not defined PROJECT_NO column as NOT NULL column, it is NOT NULL due to the **primary key** associated with the PROJECT_NO column.
- Here in the above insert we are trying to insert null value into that column which violates the Not Null constraint on this table.
- Thus, it avoids inserting null into the primary key column which does not allow nulls.

3. Insert ('453453453', 'John', M, '12-DEC-60', 'SPOUSE') into DEPENDENT



```
select * from dependent
```

Results Explain Describe Saved SQL History

EMPLOYEE_SSN	DEPN_NAME	SEX	BIRTHDAY	RELATIONSHIP
333445555	Alice	F	05-APR-76	Daughter
333445555	Theodore	M	25-OCT-73	Son
333445555	Joy	F	03-MAY-48	Spouse
987654321	Abner	M	29-FEB-32	Spouse
123456789	Alice	F	31-DEC-78	Daughter
123456789	Elizabeth	F	05-MAY-57	Spouse
453453453	John	M	12-DEC-60	SPOUSE

7 rows returned in 0.00 seconds CSV Export

- Data inserted with a success – thus it does not violate any integrity constraints.

4. Delete the WORKS_ON tuples with ESSN= '333445555'.

```
delete from works_on where EMPLOYEE_SSN = 333445555
```

Results Explain Describe Saved SQL History

```
4 row(s) deleted.

0.01 seconds
```

```
select * from works_on
```

Results Explain Describe Saved SQL History

EMPLOYEE_SSN	PROJECT_NO	HOURS
123456789	3388	32.5
123456789	1945	7.5
666884444	3388	40
453453453	77	20
453453453	22	20
999887777	12	13
543216789	22	17
554433221	1945	21.5
999887777	1	11.5

9 rows returned in 0.02 seconds CSV Export

- Data deleted with a success – 4 tuples related to SSN_NUMBER were deleted – this does not violate any integrity constraints.

5. Modify the MGRSSN and MGRSTARTDATE of the DEPARTMENT tuple with DNUMBER=5 to '123456789' and '01-OCT-88', respectively

```
update department
set MANAGER_SSN = 123456789,
MANAGER_START_DATE = '01-OCT-88'
where DEPARTMENT_NUMBER = 5
```

Results Explain Describe Saved SQL History

1 row(s) updated.

0.00 seconds

```
select * from department
```

Results Explain Describe Saved SQL History

DEPARTMENT_NAME	DEPARTMENT_NUMBER	MANAGER_SSN	MANAGER_START_DATE
Finance	3	234	18-MAY-13
IT	4	123	12-JUN-15
Administration	2	564	03-JAN-12
Headquarter	1	678	16-DEC-14
Production	5	123456789	01-OCT-88

5 rows returned in 0.01 seconds CSV Export

- The information is updated successfully – Thus no integrity constraint violated.

Alter the tables to

1. Add Foreign Keys using Alter Table [if not done earlier].

   - This activity is done earlier.
   - All the foreign keys were added using alter table command only in the existing table.

2. Drop Foreign key defined on SuperSSN and add it using Alter table command.

   - SuperSSN is associated with Employee table.
   - First, we find the foreign key constraint name applied to SuperSSN column

```
select * from USER_CONS_COLUMNS where TABLE_NAME = 'EMPLOYEE' and COLUMN_NAME = 'SUPERVISOR_SSN'
```

Results Explain Describe Saved SQL History

OWNER	CONSTRAINT_NAME	TABLE_NAME	COLUMN_NAME	POSITION
LAB2	CON_FK1	EMPLOYEE	SUPERVISOR_SSN	1

1 rows returned in 0.11 seconds CSV Export

   - Then we drop the constraint.

```
alter table EMPLOYEE drop constraint CON_FK1;
```

Results Explain Describe Saved SQL History

```
Table dropped.

0.50 seconds
```

- We add the constraint again using alter command.

```
alter table employee add constraint new_fk1 foreign key (SUPERVISOR_SSN) references employee(SSN_NUMBER) on delete set null;
```

Results Explain Describe Saved SQL History

```
Table altered.

0.00 seconds
```

- Again, we view the constraint with new name.

```
select * from USER_CONS_COLUMNS where TABLE_NAME = 'EMPLOYEE' and COLUMN_NAME = 'SUPERVISOR_SSN'
```

Results Explain Describe Saved SQL History

OWNER	CONSTRAINT_NAME	TABLE_NAME	COLUMN_NAME	POSITION
LAB2	NEW_FK1	EMPLOYEE	SUPERVISOR_SSN	1

1 rows returned in 0.00 seconds CSV Export

3. Make name of Project as Unique and sex of employee as not null.

- Adding constraint Unique to name column of Project table:

```
alter table project add constraint project_unique UNIQUE (PROJECT_NAME)
```

Results Explain Describe Saved SQL History

```
Table altered.

0.03 seconds
```

```
select * from USER_CONS_COLUMNS where TABLE_NAME = 'PROJECT' and COLUMN_NAME = 'PROJECT_NAME'
```

Results Explain Describe Saved SQL History

OWNER	CONSTRAINT_NAME	TABLE_NAME	COLUMN_NAME	POSITION
LAB2	CON_NOTNULL5	PROJECT	PROJECT_NAME	-
LAB2	PROJECT_UNIQUE	PROJECT	PROJECT_NAME	1

2 rows returned in 0.00 seconds CSV Export

- Add not null constraint to sex column of employee table:

```
alter table employee modify (sex constraint sex_notnull not null);
```

Results Explain Describe Saved SQL History

```
Table altered.
```

0.09 seconds

```
select * from USER_CONS_COLUMNS where TABLE_NAME = 'EMPLOYEE' and COLUMN_NAME = 'SEX'
```

Results Explain Describe Saved SQL History

OWNER	CONSTRAINT_NAME	TABLE_NAME	COLUMN_NAME	POSITION
LAB2	CON_IN1	EMPLOYEE	SEX	-
LAB2	SEX_NOTNULL	EMPLOYEE	SEX	-

2 rows returned in 0.00 seconds CSV Export

4. Make Address as a new type containing door no, street, city, State, Continent.

- Creating an address type object containing door_no, street, city , state, and continent

```
create type address as OBJECT
(
  door_no number(5),
  street varchar(30),
  city varchar(20),
  state varchar(20),
  continent varchar(10)
)
```

Results Explain Describe Saved SQL History

```
Type created.
```

0.08 seconds

- Now we alter table to add column of this **type** object

```
create type address as OBJECT
(
  door_no number(5),
  street varchar(30),
  city varchar(20),
  state varchar(20),
  continent varchar(10)
)

alter table employee add ADDRESS_TYPE address;
```

Results Explain Describe Saved SQL History

```
Table altered.
```

0.16 seconds

- Now we update one row to map some value to existing row

```
update employee e
set e.ADDRESS_TYPE = address(1, 'abc', 'vellore', 'tamilnadu', 'asia')
where e.SSN_NUMBER = 125
```

Results Explain Describe Saved SQL History

```
1 row(s) updated.
```

0.07 seconds

- Since we have an object column in our table, we have to write complex select statement with dot(.) operator to view values or we can create a view and call view directly to list all columns.

```
create view employee_with_object(first_name, mid_name, last_name, ssn_number,
address, sex, salary, sup_ssn, dept_no, door_no, street, city, state, continent)   as
select e.first_name, e.mid_name, e.last_name, e.ssn_number,
e.address, e.sex, e.salary, e.supervisor_ssn, e.department_number,
e.address_type.door_no, e.address_type.street, e.address_type.city,
e.address_type.state, e.address_type.continent from
employee e
```

Results Explain Describe Saved SQL History

```
View created.
```

0.07 seconds

```
select * from employee_with_object
```

Results Explain Describe Saved SQL History

FIRST_NAME	MID_NAME	LAST_NAME	SSN_NUMBER	ADDRESS	SEX	SALARY	SUP_SSN	DEPT_NO	DOOR_NO	STREET	CITY	STATE	CONTINENT
Frankin	T	Wong	125	Delhi	M	40000	123	2	1	abc	vellore	tamilnadu	asia
Jennifer	S	Wallace	564	Chennai	F	43000	123	2	-	-	-	-	-
John	B	Smith	678	Madurai	M	30000	124	1	-	-	-	-	-
Ramesh	K	Narayan	234	Bangalore	M	38000	124	3	-	-	-	-	-
Chris	-	Evans	666884444	Beverly Hills, CA 90212, USA	M	91000	123456789	5	-	-	-	-	-
Chris	-	Hemsworth	453453453	9255 W Sunset Blvd, USA	M	92000	666884444	1	-	-	-	-	-
Elizabeth	-	Olsen	333445555	Sherman Oaks, California, USA.	F	93000	453453453	1	-	-	-	-	-
Scarlett	-	Johansson	999887777	Manhattan, New York City, USA	F	94000	123456789	1	-	-	-	-	-
Mark	A	Ruffalo	543216789	Beverly Hills, USA	M	89000	453453453	2	-	-	-	-	-
Tom	-	Hiddleston	554433221	Beverly Hills, USA	M	88000	-	2	-	-	-	-	-
Tom	-	Holland	987654321	Beverly Hills, USA	M	87000	333445555	3	-	-	-	-	-
Doug	E	Gilbert	123	Chennai	M	80000	-	1	-	-	-	-	-
Joyce	-	PAN	124	Vellore	F	70000	-	-	-	-	-	-	-
Robert	D	Junior	123456789	West Burbank CA 91506, USA	M	90000	123	5	-	-	-	-	-

14 rows returned in 0.01 seconds CSV Export

5. Make salary of employee to accept real values

```
alter table employee modify salary Real;
```

Results Explain Describe Saved SQL History

```
Table altered.
```

0.05 seconds

- Here the REAL is an alias for FLOAT data type

```
select TABLE_NAME, COLUMN_NAME, DATA_TYPE, DATA_LENGTH, DATA_PRECISION
from ALL_TAB_COLUMNS where TABLE_NAME = 'EMPLOYEE' and COLUMN_NAME = 'SALARY' and OWNER = 'LAB2'
```

Results Explain Describe Saved SQL History

TABLE_NAME	COLUMN_NAME	DATA_TYPE	DATA_LENGTH	DATA_PRECISION
EMPLOYEE	SALARY	FLOAT	22	63

1 rows returned in 0.01 seconds CSV Export