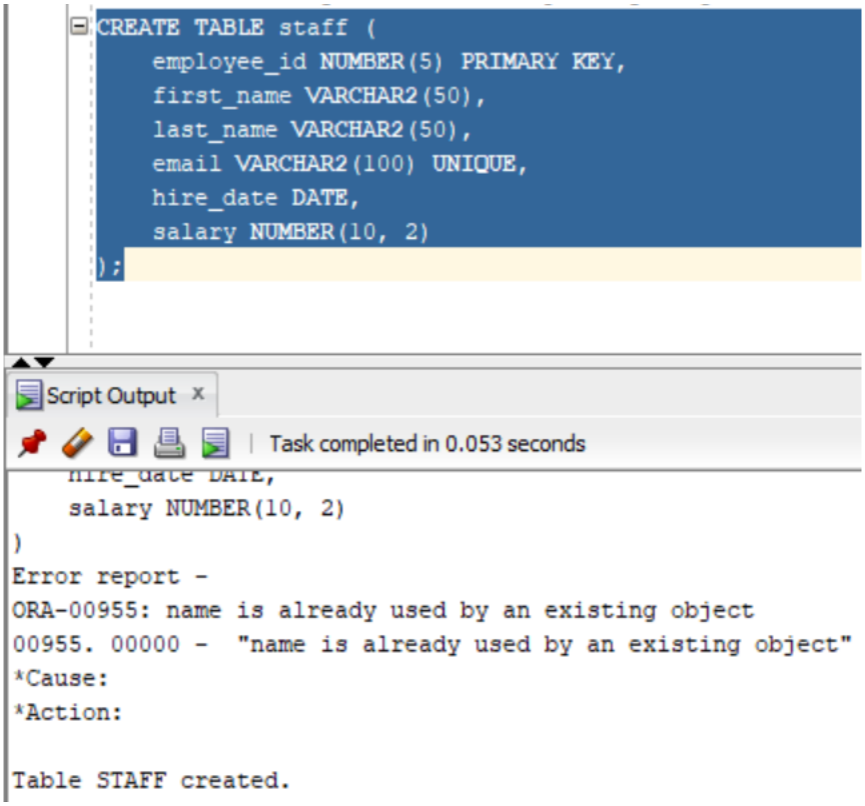
Constraints In Oracle

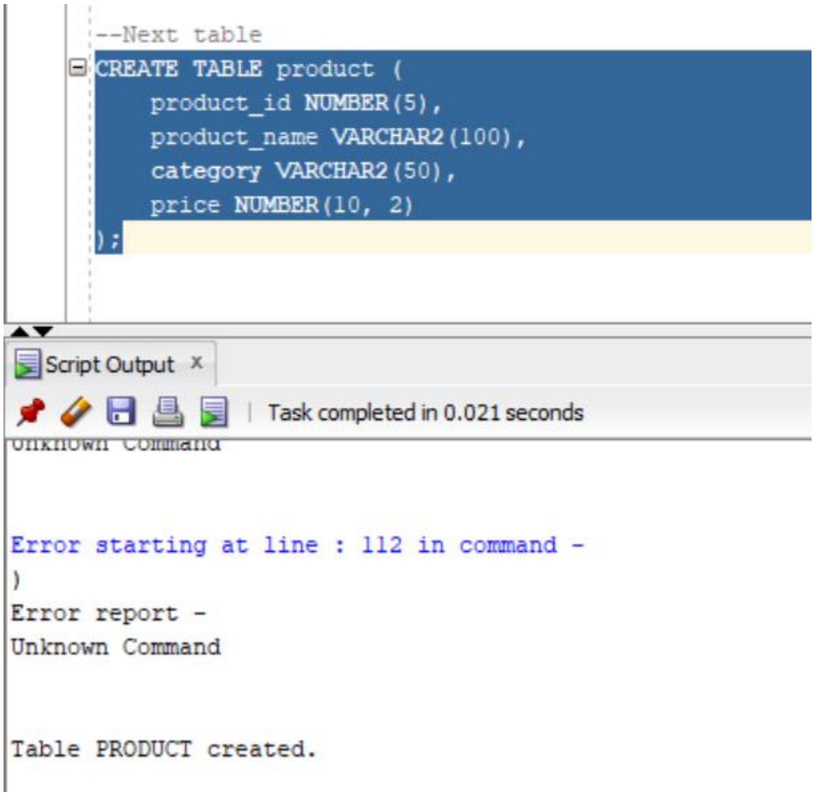
- Primary key
- Composite key
- Foreign Key
- Check
- Not Null
- Unique Key
- Default



```
CREATE TABLE staff (
    employee_id NUMBER(5) PRIMARY KEY,
    first_name VARCHAR2(50),
    last_name VARCHAR2(50),
    email VARCHAR2(100) UNIQUE,
    hire_date DATE,
    salary NUMBER(10, 2)
);
```

```
    hire_date DATE,
    salary NUMBER(10, 2)
)
Error report -
ORA-00955: name is already used by an existing object
00955. 00000 -  "name is already used by an existing object"
*Cause:
*Action:

Table STAFF created.
```

Here we have use normal method to create table with primary key and unique key. But, how we will add if we forgot to add while creating the table.

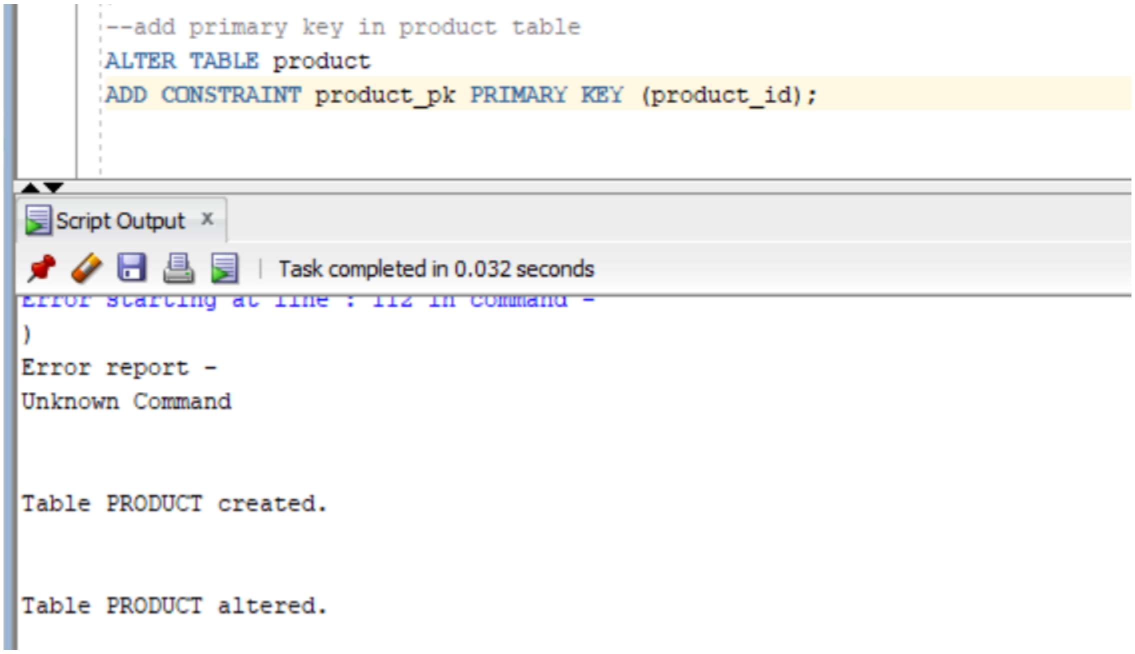


```
--Next table
CREATE TABLE product (
    product_id NUMBER(5),
    product_name VARCHAR2(100),
    category VARCHAR2(50),
    price NUMBER(10, 2)
);
```

Task completed in 0.021 seconds

```
Unknown Command


Error starting at line : 112 in command -
)
Error report -
Unknown Command


Table PRODUCT created.
```

Now let’s **add primary key using constraint**:

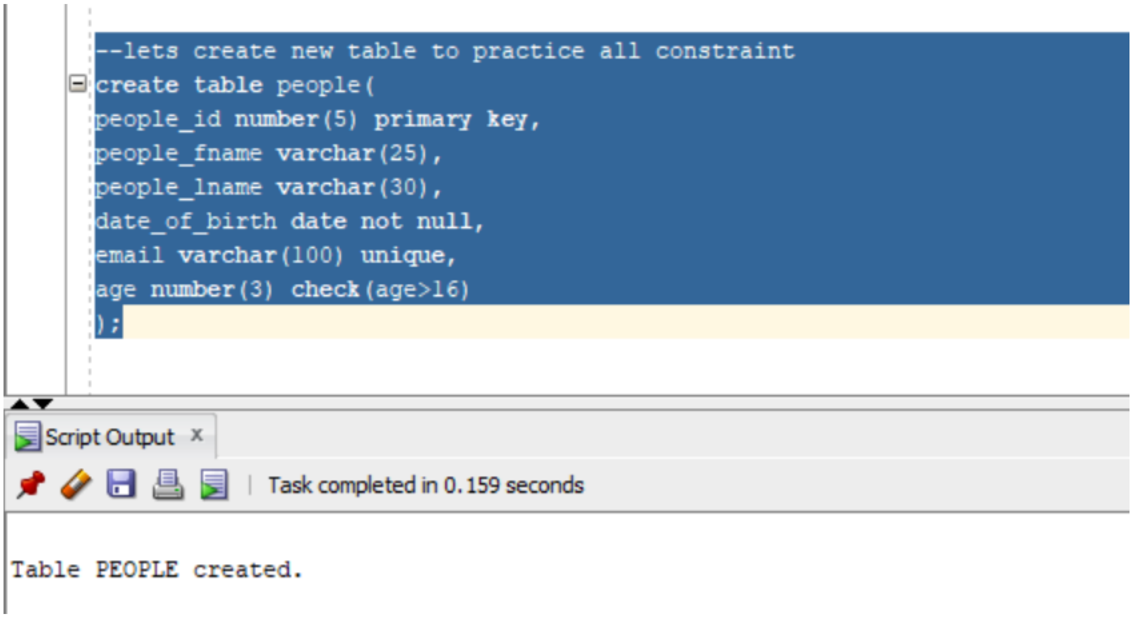


```
--add primary key in product table
ALTER TABLE product
ADD CONSTRAINT product_pk PRIMARY KEY (product_id);
```

Task completed in 0.032 seconds

```
Error starting at line : 112 in command -
)
Error report -
Unknown Command


Table PRODUCT created.


Table PRODUCT altered.
```

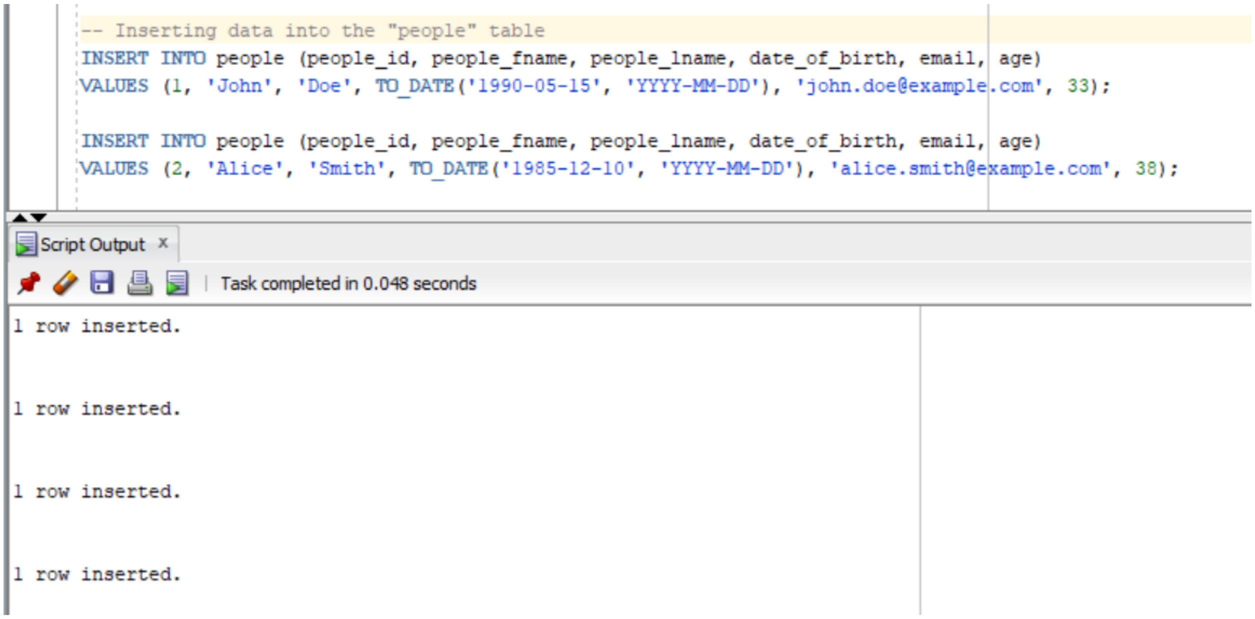
Primary key, foreign key, Unique, check, not null



```sql
--lets create new table to practice all constraint
create table people(
people_id number(5) primary key,
people_fname varchar(25),
people_lname varchar(30),
date_of_birth date not null,
email varchar(100) unique,
age number(3) check(age>16)
);
```

```
Task completed in 0.159 seconds

Table PEOPLE created.
```

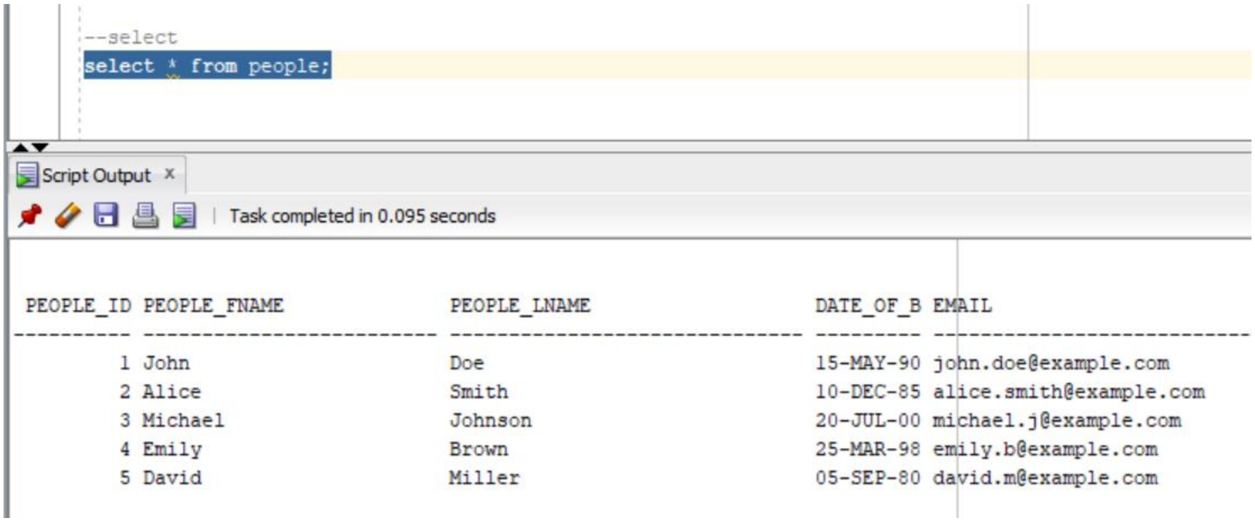
Let’s insert data in it: **insert (DML)**



```sql
-- Inserting data into the "people" table
INSERT INTO people (people_id, people_fname, people_lname, date_of_birth, email, age)
VALUES (1, 'John', 'Doe', TO_DATE('1990-05-15', 'YYYY-MM-DD'), 'john.doe@example.com', 33);

INSERT INTO people (people_id, people_fname, people_lname, date_of_birth, email, age)
VALUES (2, 'Alice', 'Smith', TO_DATE('1985-12-10', 'YYYY-MM-DD'), 'alice.smith@example.com', 38);
```

```
Task completed in 0.048 seconds
1 row inserted.


1 row inserted.


1 row inserted.


1 row inserted.
```

Now, lets see few queries as revise what we practice in previously:

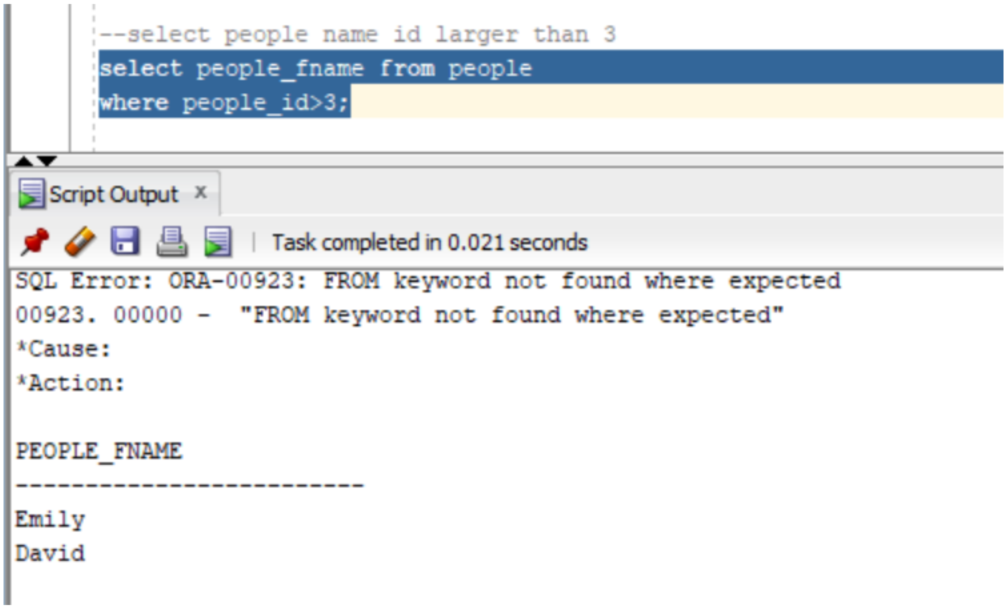


```
--select
select * from people;
```

Task completed in 0.095 seconds

```
 PEOPLE_ID PEOPLE_FNAME              PEOPLE_LNAME                   DATE_OF_B EMAIL
---------- ------------------------- ------------------------------ --------- ---------------------------
         1 John                      Doe                            15-MAY-90 john.doe@example.com
         2 Alice                     Smith                          10-DEC-85 alice.smith@example.com
         3 Michael                   Johnson                        20-JUL-00 michael.j@example.com
         4 Emily                     Brown                          25-MAR-98 emily.b@example.com
         5 David                     Miller                         05-SEP-80 david.m@example.com
```

I only need people's name whose id is above than 5:

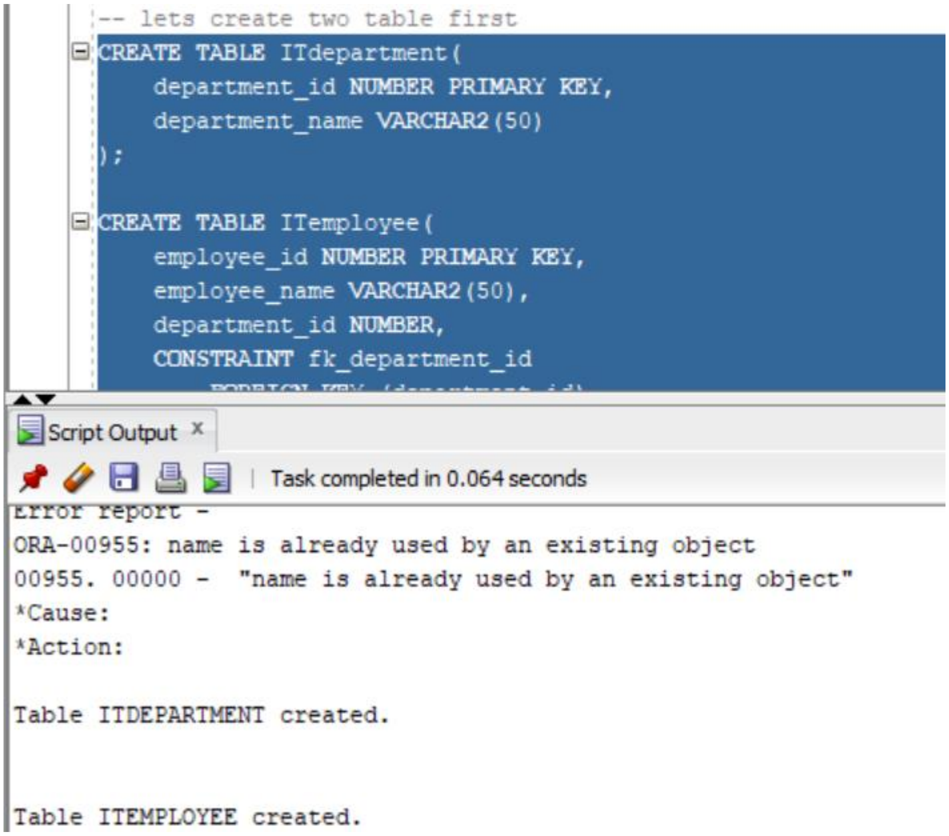


```
--select people name id larger than 3
select people_fname from people
where people_id>3;
```

Task completed in 0.021 seconds

```
SQL Error: ORA-00923: FROM keyword not found where expected
00923. 00000 -  "FROM keyword not found where expected"
*Cause:
*Action:

PEOPLE_FNAME
-------------------------
Emily
David
```

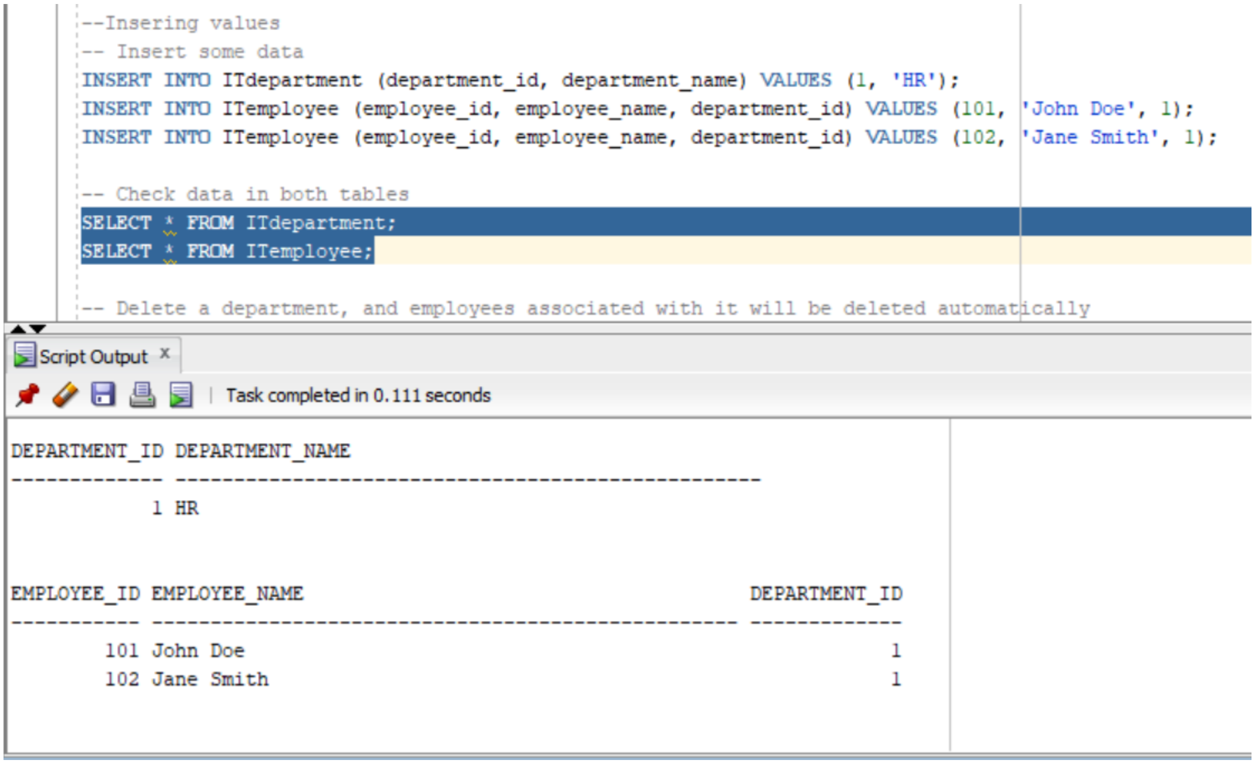
Delete Cascade



```
-- lets create two table first
CREATE TABLE ITdepartment(
    department_id NUMBER PRIMARY KEY,
    department_name VARCHAR2(50)
);

CREATE TABLE ITemployee(
    employee_id NUMBER PRIMARY KEY,
    employee_name VARCHAR2(50),
    department_id NUMBER,
    CONSTRAINT fk_department_id
        FOREIGN KEY (department_id)
```

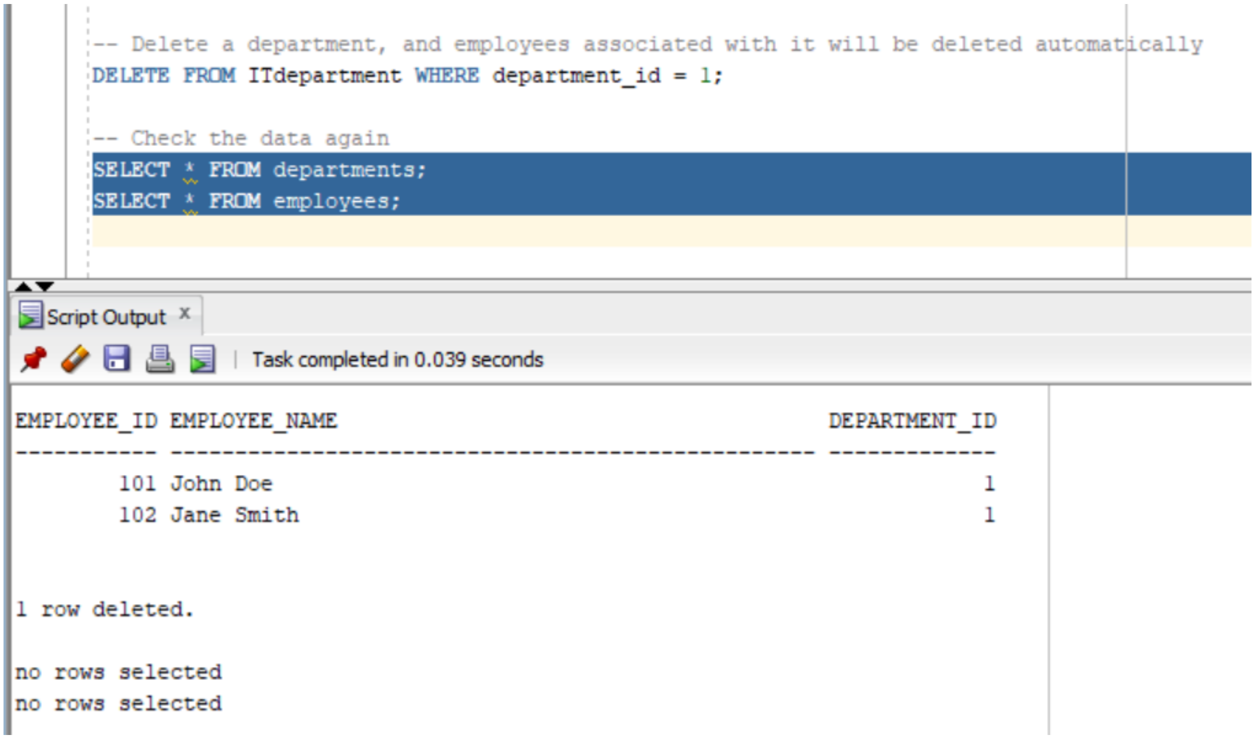
```
Error report -
ORA-00955: name is already used by an existing object
00955. 00000 -  "name is already used by an existing object"
*Cause:
*Action:

Table ITDEPARTMENT created.


Table ITEMPLOYEE created.
```

Two table created, lets insert some values in both table:



```
--Insering values
-- Insert some data
INSERT INTO ITdepartment (department_id, department_name) VALUES (1, 'HR');
INSERT INTO ITemployee (employee_id, employee_name, department_id) VALUES (101, 'John Doe', 1);
INSERT INTO ITemployee (employee_id, employee_name, department_id) VALUES (102, 'Jane Smith', 1);

-- Check data in both tables
SELECT * FROM ITdepartment;
SELECT * FROM ITemployee;

-- Delete a department, and employees associated with it will be deleted automatically
```

```
DEPARTMENT_ID DEPARTMENT_NAME
------------- --------------------------------------------------
            1 HR


EMPLOYEE_ID EMPLOYEE_NAME                                      DEPARTMENT_ID
----------- -------------------------------------------------- -------------
        101 John Doe                                                       1
        102 Jane Smith                                                     1
```



```
-- Delete a department, and employees associated with it will be deleted automatically
DELETE FROM ITdepartment WHERE department_id = 1;

-- Check the data again
SELECT * FROM departments;
SELECT * FROM employees;
```

```
EMPLOYEE_ID EMPLOYEE_NAME                                      DEPARTMENT_ID
----------- -------------------------------------------------- -------------
        101 John Doe                                                       1
        102 Jane Smith                                                     1


1 row deleted.

no rows selected
no rows selected
```

This is known as Delete cascade.