**Lab 4. Database Objects**

1. **Create the Customer table with the following columns.**

Customerid Number(5)

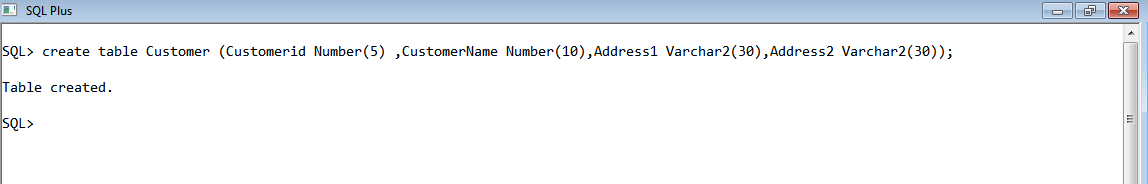
CustomerName Number(10)

Address1 Varchar2(30)

Address2 Varchar2(30)

Solution :

CREATE TABLE CUSTOMER (CUSTOMERID NUMBER(5) ,CUSTOMERNAME NUMBER(10),ADDRESS1 VARCHAR2(30),ADDRESS2 VARCHAR2(30));

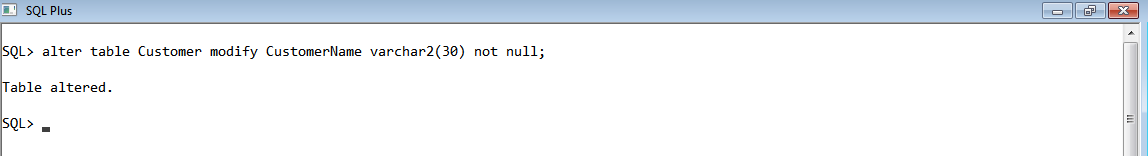


**2. Modify the Customer table CustomerName column of datatype with Varchar2(30).**

**CustomerName should not accept Nulls.**

Solution :

ALTER TABLE CUSTOMER MODIFY CUSTOMERNAME VARCHAR2(30) NOT NULL;



**3. Add the following Columns to the Customer table.**

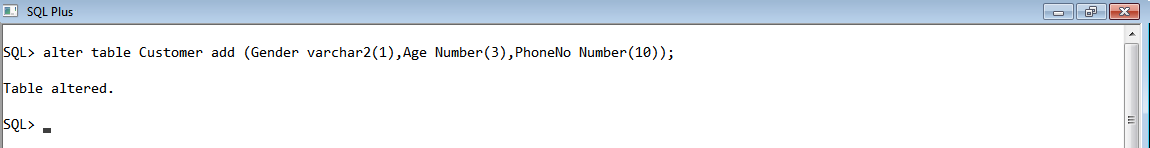
Gender Varchar2(1)

Age Number(3)

PhoneNo Number(10)

Solution :

ALTER TABLE CUSTOMER ADD (GENDER VARCHAR2(1),AGE NUMBER(3),PHONENO NUMBER(10));



**4. Insert rows with the following data in to the Customer table.**

Insert into customer values: (1000, ‘Allen’, ‘#115 Chicago’, ‘#115 Chicago’, ‘M’, ‘25,

7878776’)

In similar manner, add the below records to the Customer table:

 1000, Allen, #115 Chicago, #115 Chicago, M, 25, 7878776

 1001, George, #116 France, #116 France, M, 25, 434524

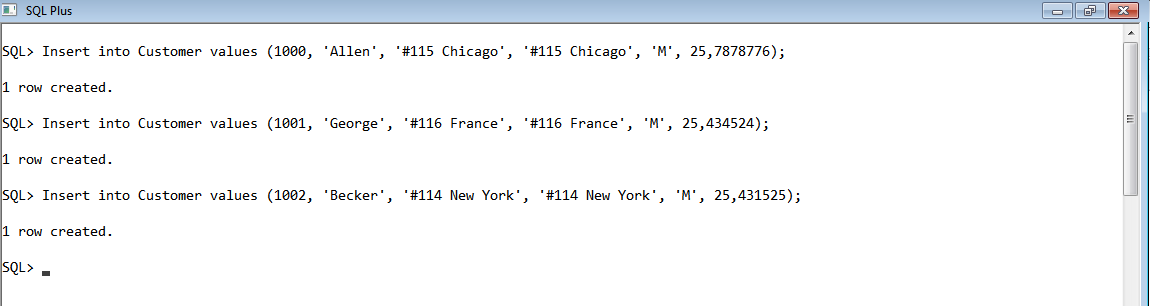
 1002, Becker, #114 New York, #114 New York, M, 45, 431525

Solution :

INSERT INTO CUSTOMER VALUES (1000, 'ALLEN', '#115 CHICAGO', '#115 CHICAGO', 'M', 25,7878776);

INSERT INTO CUSTOMER VALUES (1001, 'GEORGE', '#116 FRANCE', '#116 FRANCE', 'M', 25,434524);

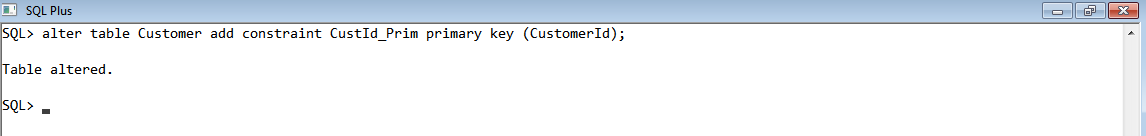
INSERT INTO CUSTOMER VALUES (1002, 'BECKER', '#114 NEW YORK', '#114 NEW YORK', 'M', 25,431525);



**5. Add the Primary key constraint for Customerld with the name Custld\_Prim.**

Solution :

ALTER TABLE CUSTOMER ADD CONSTRAINT CUSTID\_PRIM PRIMARY KEY (CUSTOMERID);

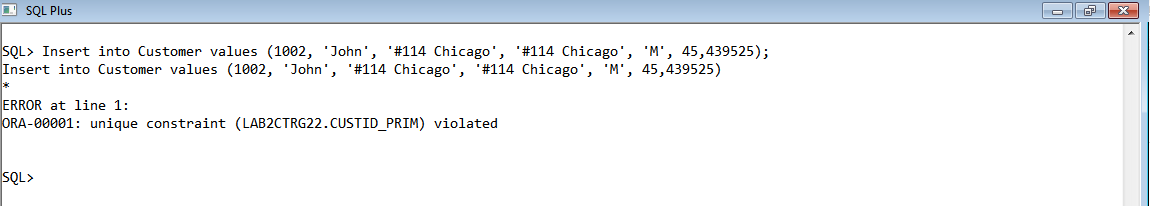


**6. Insert the row given below in the Customer table and see the message generated**

**by the Oracle server.**

**1002, John, #114 Chicago, #114 Chicago, M, 45, 439525**

Solution : INSERT INTO CUSTOMER VALUES (1002, 'JOHN', '#114 CHICAGO', '#114 CHICAGO', 'M', 45,439525);



**7. Disable the constraint on CustomerId, and insert the following data:**

** 1002, Becker, #114 New York, #114 New york , M, 45, 431525**

** 1003, Nanapatekar, #115 India, #115 India , M, 45, 431525**

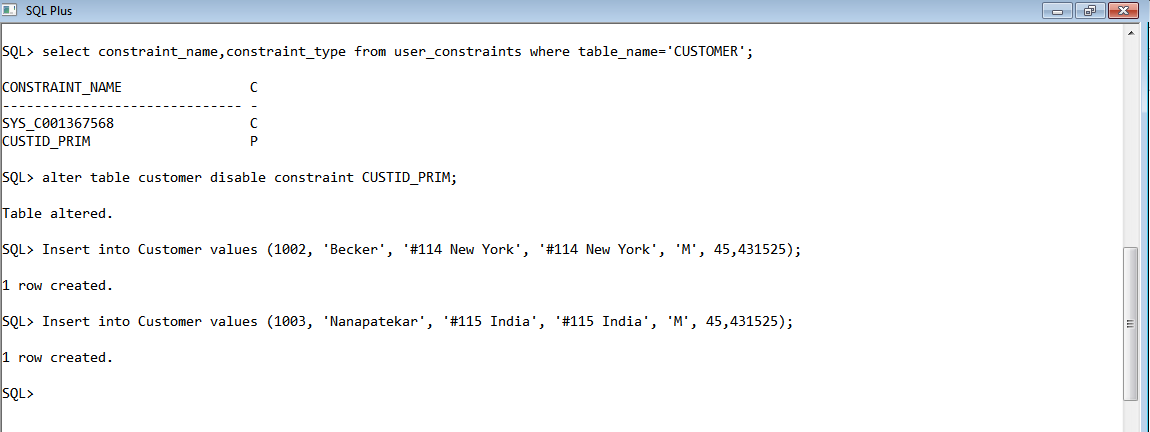
Solution :

select constraint\_name,constraint\_type from user\_constraints where table\_name='CUSTOMER';

alter table customer disable constraint CUSTID\_PRIM;

Insert into Customer values (1002, 'Becker', '#114 New York', '#114 New York', 'M', 45,431525);

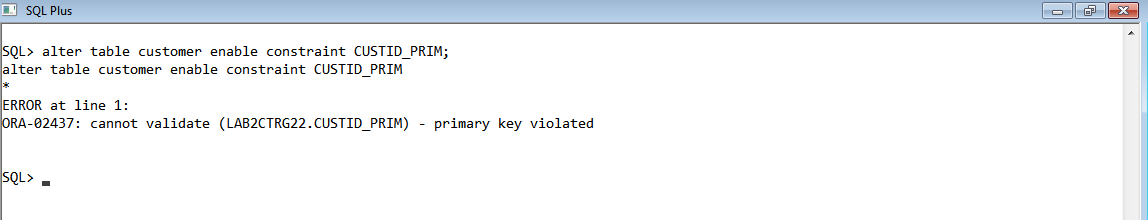
Insert into Customer values (1003, 'Nanapatekar', '#115 India', '#115 India', 'M', 45,431525);



**8. Enable the constraint on CustomerId of the Customer table, and see the message**

**generated by the Oracle server.**

Solution : ALTER TABLE CUSTOMER ENABLE CONSTRAINT CUSTID\_PRIM;



**9. Drop the constraint Custld\_Prim on CustomerId and insert the following Data. Alter**

**Customer table, drop constraint Custid\_Prim.**

** 1002, Becker, #114 New York, #114 New york , M, 45, 431525, 15000.50**

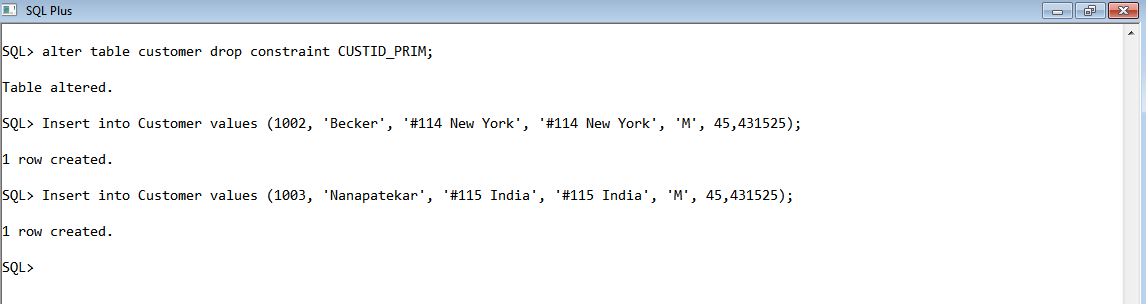
** 1003, Nanapatekar, #115 India, #115 India , M, 45, 431525, 20000.50**

Solution :

ALTER TABLE CUSTOMER DROP CONSTRAINT CUSTID\_PRIM;

INSERT INTO CUSTOMER VALUES (1002, 'BECKER', '#114 NEW YORK', '#114 NEW YORK', 'M', 45,431525);

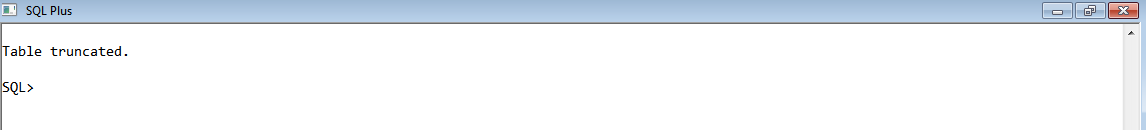
INSERT INTO CUSTOMER VALUES (1003, 'NANAPATEKAR', '#115 INDIA', '#115 INDIA', 'M', 45,431525);



**10. Delete all the existing rows from Customer table, and let the structure remain itself**

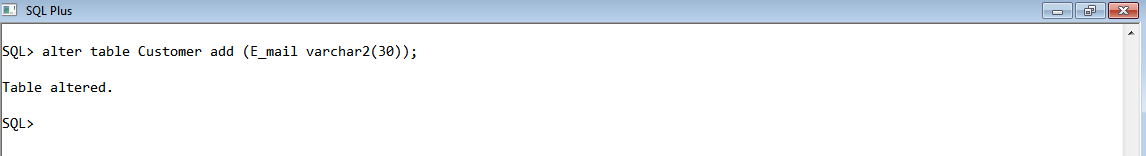
**using TRUNCATE statement.**

Solution : Truncate Table Customer;



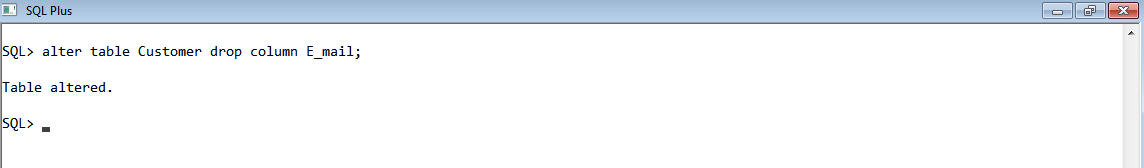
**11. In the Customer table, add a column E\_mail.**

Solution : ALTER TABLE CUSTOMER ADD (E\_MAIL VARCHAR2(30));



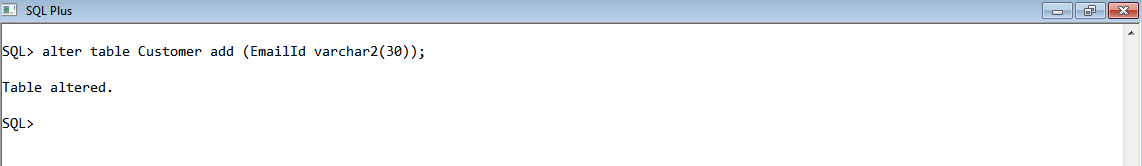
**12. Drop the E\_mail column from Customer table.**

Solution : ALTER TABLE CUSTOMER DROP COLUMN E\_MAIL;



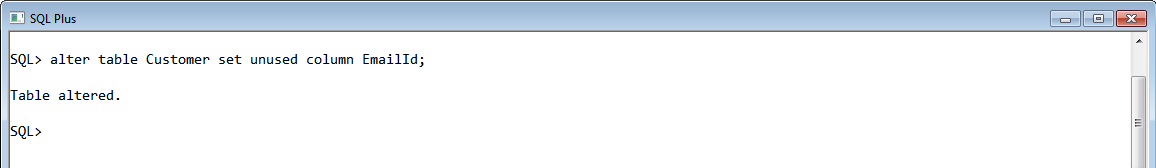
**13. Add a new column EmailId to Customer table.**

Solution : ALTER TABLE CUSTOMER ADD (EMAILID VARCHAR2(30));



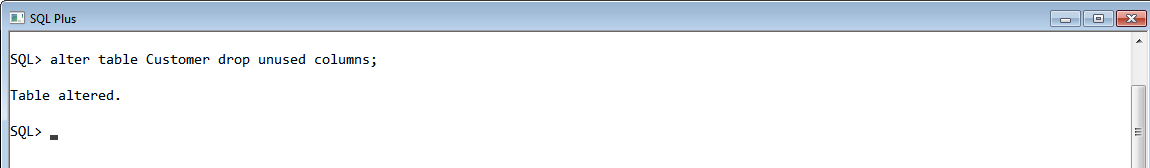
**14. Mark EmailId column as unused before dropping it.**

Solution : ALTER TABLE CUSTOMER SET UNUSED COLUMN EMAILID;



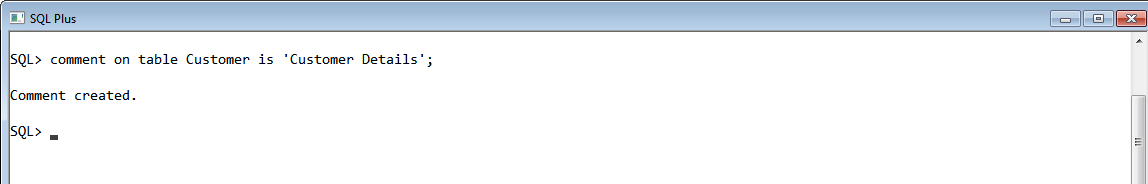
**15. Drop the unused EmailId column from the Customer table.**

Solution : ALTER TABLE CUSTOMER DROP UNUSED COLUMNS;



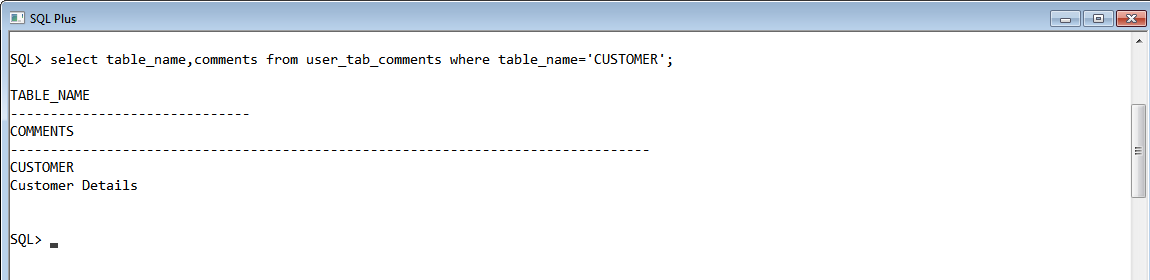
**16. Define the COMMENT ‘Customers Details’ for Customer table.**

Solution : COMMENT ON TABLE CUSTOMER IS 'CUSTOMER DETAILS';



**17. Use Data Dictionary USER\_TAB\_COMMENTS to view the created comment.**

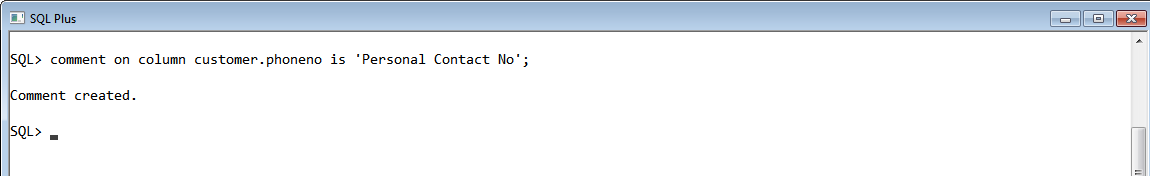
Solution : SELECT TABLE\_NAME,COMMENTS FROM USER\_TAB\_COMMENTS WHERE TABLE\_NAME='CUSTOMER';



**18. Define the COMMENT ‘Personal Contact no’ for the phoneno column of the**

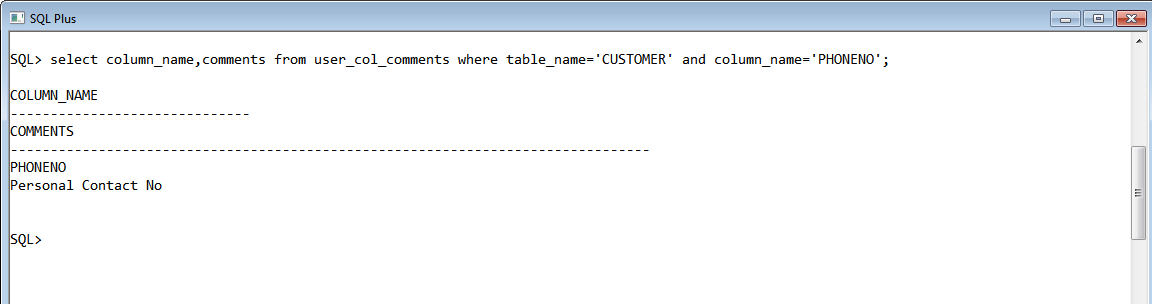
**Customer table.**

Solution : COMMENT ON COLUMN CUSTOMER.PHONENO IS 'PERSONAL CONTACT NO';



**19. Use Data Dictionary USER\_COL\_COMMENTS to view the created comment.**

Solution : SELECT COLUMN\_NAME,COMMENTS FROM USER\_COL\_COMMENTS WHERE TABLE\_NAME='CUSTOM ER' AND COLUMN\_NAME='PHONENO';



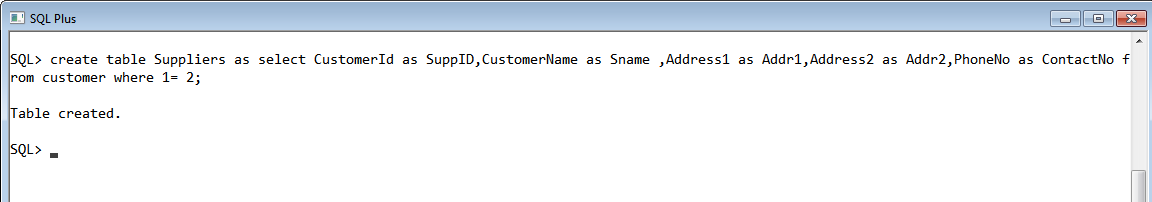
**20. Create the Suppliers table based on the structure of the Customer table. Include**

**only the CustomerId, CustomerName, Address1, Address2, and phoneno columns.**

**Name the columns in the new table as SuppID, SName, Addr1, Addr2, and Contactno**

**respectively.**

Solution : CREATE TABLE SUPPLIERS AS SELECT CUSTOMERID AS SUPPID,CUSTOMERNAME AS SNAME ,ADDRESS1 AS ADDR1,ADDRESS2 AS ADDR2,PHONENO AS CONTACTNO FROM CUSTOMER WHERE 1= 2;



**21. Drop the above table and recreate the following table with the name**

CustomerMaster.

CustomerId Number(5) Primary key(Name of constraint is CustId\_PK)

CustomerName Varchar2(30) Not Null

Addressl Varchar2(30) Not Null

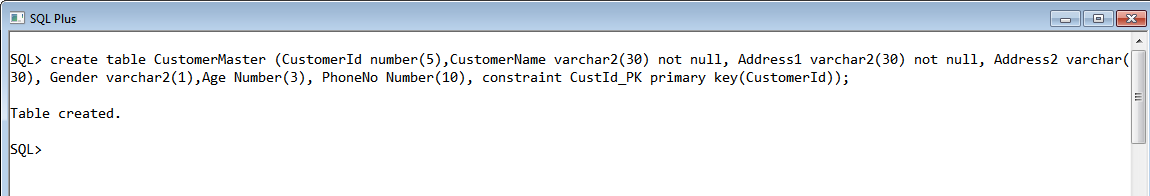
Address2 Varchar2(30)

Gender Varchar2(l)

Age Number(3)

PhoneNo Number(10)

Solution : CREATE TABLE CUSTOMERMASTER (CUSTOMERID NUMBER(5),CUSTOMERNAME VARCHAR2(30) NOT NULL, ADDRESS1 VARCHAR2(30) NOT NULL, ADDRESS2 VARCHAR(30), GENDER VARCHAR2(1),AGE NUMBER(3), PHONENO NUMBER(10), CONSTRAINT CUSTID\_PK PRIMARY KEY(CUSTOMERID));



**22. Create the AccountsMaster table with the following Columns. Use sequence to**

**generate Account number**

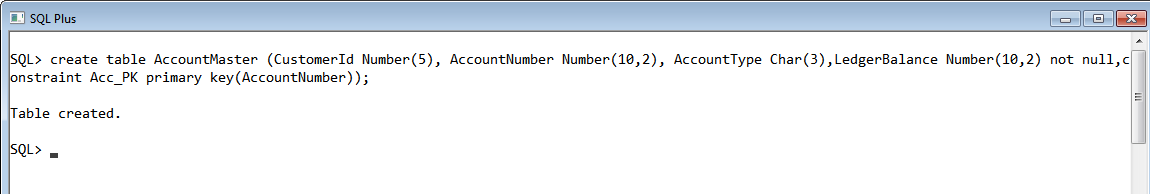
Customerld Number(5)

AccountNumber Number(10,2) Primary key(Name of constraint is Acc\_PK)

AccountType Char(3)

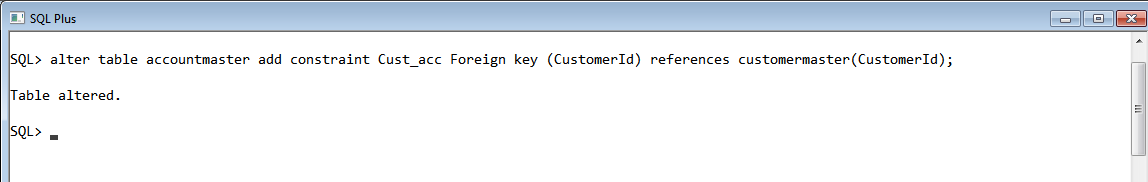
LedgerBalance Number(10,2) Not Null

Solution : CREATE TABLE ACCOUNTMASTER (CUSTOMERID NUMBER(5), ACCOUNTNUMBER NUMBER(10,2), ACCOUNTTYPE CHAR(3), LEDGERBALANCE NUMBER(10,2) NOT NULL,CONSTRAINT ACC\_PK PRIMARY KEY(ACCOUNTNUMBER));



**23. Relate AccountsMaster table and CustomerMaster table through Customerld**

**column with the constraint name Cust\_acc.**

Solution : ALTER TABLE ACCOUNTMASTER ADD CONSTRAINT CUST\_ACC FOREIGN KEY (CUSTOMERID) REFERENCES CUSTOMERMASTER(CUSTOMERID);

**24. Insert the following rows to the CustomerMaster table:**

 1000, Allen, #115 Chicago, #115 Chicago, M, 25, 7878776

 1001, George, #116 France, #116 France, M, 25, 434524

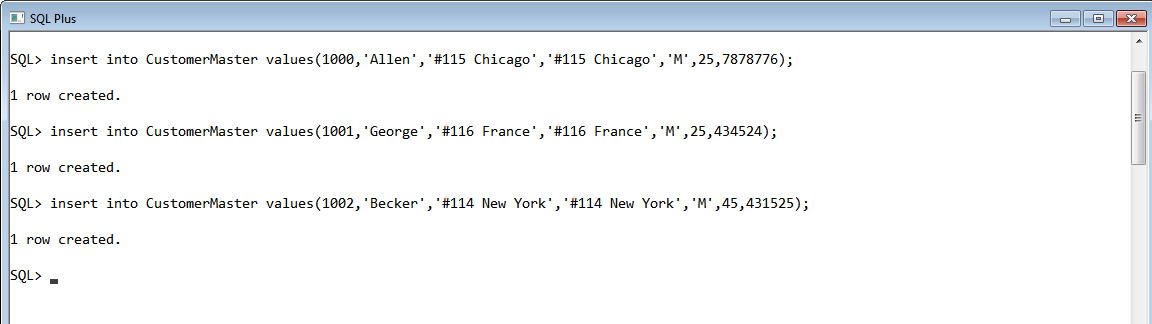
 1002, Becker, #114 New York, #114 New York, M, 45, 431525

Solution :

insert into CustomerMaster values(1000,'Allen','#115 Chicago','#115 Chicago','M',25,7878776);

insert into CustomerMaster values(1001,'George','#116 France','#116 France','M',25,434524);

insert into CustomerMaster values(1002,'Becker','#114 New York','#114 New York','M',45,431525);

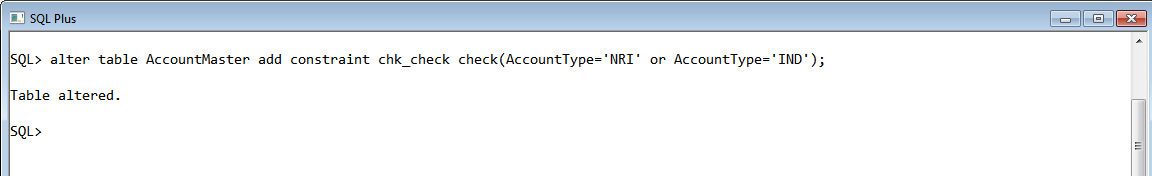


**25. Modify the AccountMaster table with the Check constraint to ensure AccountType**

**should be either NRI or IND.**

Solution :

ALTER TABLE ACCOUNTMASTER ADD CONSTRAINT CHK\_CHECK CHECK(ACCOUNTTYPE='NRI' OR ACCOUNTTYPE='IND');



**26. Insert 5 rows into the AccountsMaster table:**

Solution :

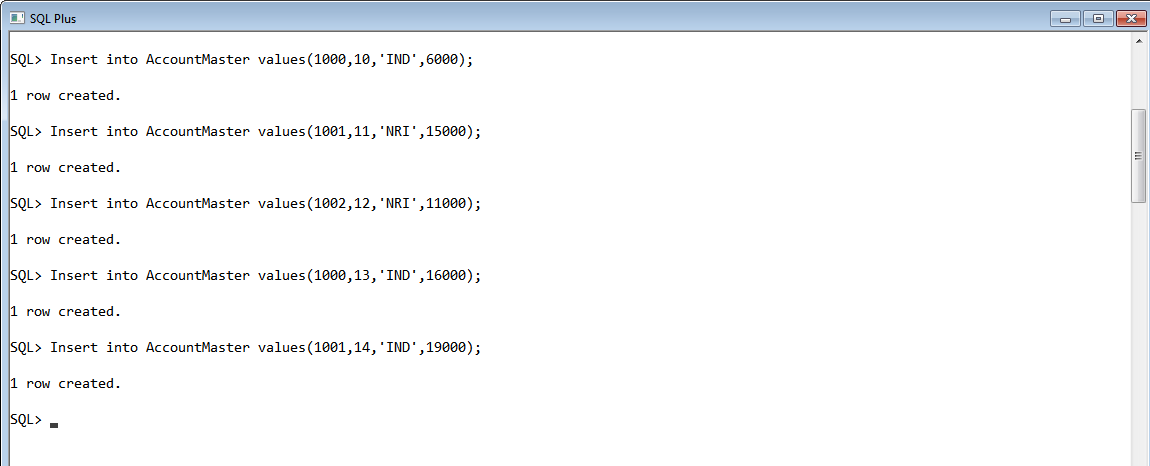
Insert into AccountMaster values(1000,10,'IND',6000);

Insert into AccountMaster values(1001,11,'NRI',15000);

Insert into AccountMaster values(1002,12,'NRI',11000);

Insert into AccountMaster values(1000,13,'IND',16000);

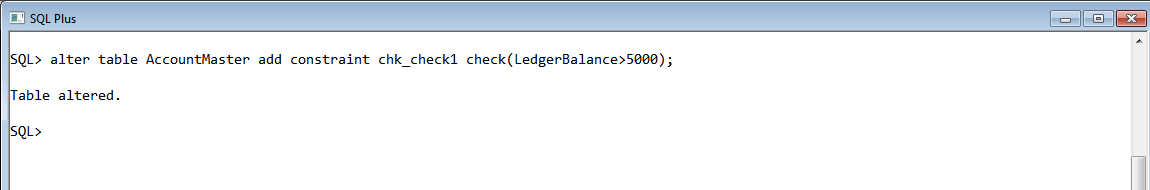
Insert into AccountMaster values(1001,14,'IND',19000);



**27. Modify the AccountsMaster table keeping a Check constraint with the name**

**Balance\_Check for the Minimum Balance which should be greater than 5000.**

Solution : ALTER TABLE ACCOUNTMASTER ADD CONSTRAINT CHK\_CHECK1 CHECK(LEDGERBALANCE>5000);



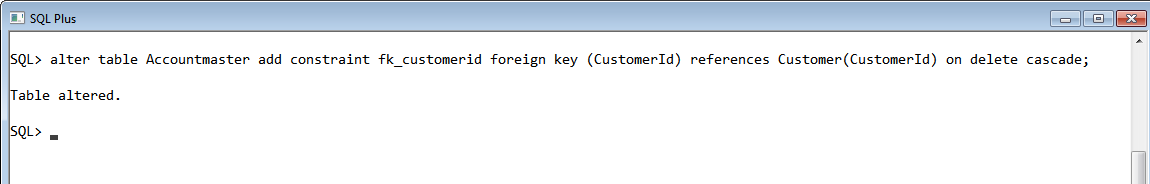
**28. Modify the AccountsMaster table such that if Customer is deleted from Customer**

**table then all his details should be deleted from AccountsMaster table.**

Solution :

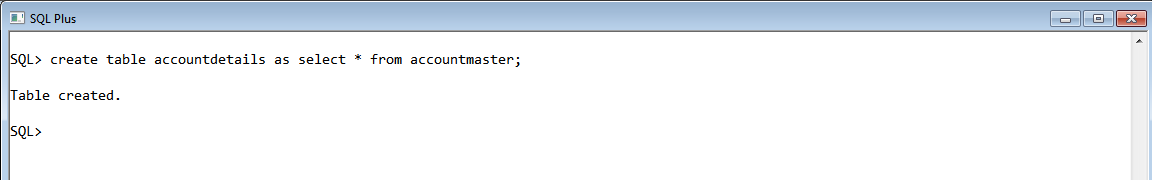
ALTER TABLE CUSTOMER MODIFY CUSTOMERID NUMBER(5) UNIQUE;

ALTER TABLE ACCOUNTMASTER ADD CONSTRAINT FK\_CUSTOMERID FOREIGN KEY (CUSTOMERID) REFERENCES CUSTOMER(CUSTOMERID) ON DELETE CASCADE;



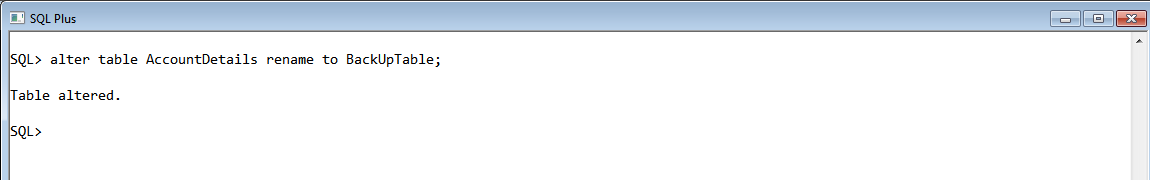
**29. Create Backup copy for the AccountsMaster table with the name ‘AccountDetails’.**

Solution : Create Table Accountdetails As Select \* From Accountmaster;



**30. Change the name of the AccountDetails table to ‘BackUpTable’ table.**

Solution : ALTER TABLE ACCOUNTDETAILS RENAME TO BACKUPTABLE;



**31. Create a view ‘Acc\_view’ with columns Customerld, CustomerName,**

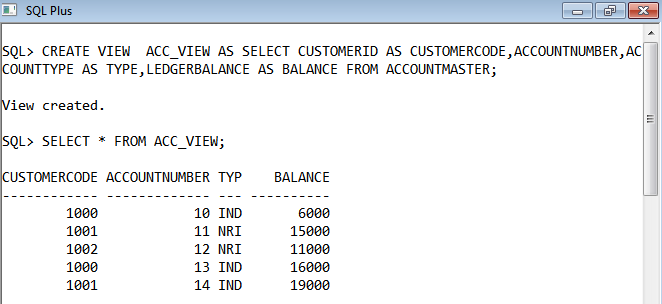
**AccountNumber, AccountType, and LedgerBalance from AccountsMaster. In the view**

**Acc\_view, the column names should be CustomerCode, AccountHolderName,**

**AccountNumber, Type, and Balance for the respective columns from AccountsMaster**

**table.**

Solution : CREATE VIEW ACC\_VIEW AS SELECT CUSTOMERID AS CUSTOMERCODE,ACCOUNTNUMBER,AC COUNTTYPE AS TYPE,LEDGERBALANCE AS BALANCE FROM ACCOUNTMASTER;



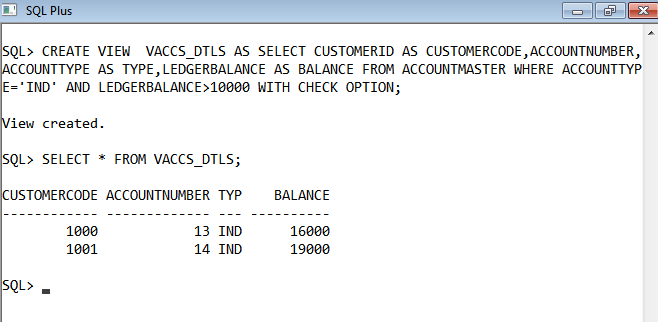
**32. Create a view on AccountsMaster table with name vAccs\_Dtls. This view should**

**list all customers whose AccountType is ‘IND’ and their balance amount should not be**

**less than 10000. Using this view any DML operation should not violate the view**

**conditions.**

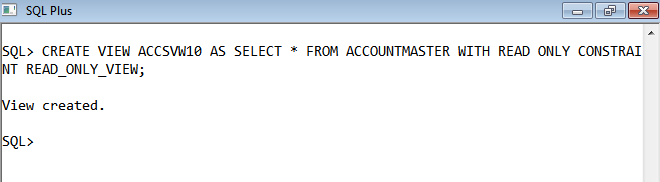
Solution : CREATE VIEW VACCS\_DTLS AS SELECT CUSTOMERID AS CUSTOMERCODE,ACCOUNTNUMBER,ACCOUNTTYPE AS TYPE,LEDGERBALANCE AS BALANCE FROM ACCOUNTMASTER WHERE ACCOUNTTYPE='IND' AND LEDGERBALANCE>10000 WITH CHECK OPTION;



**33. Create a view accsvw10 which will not allow DML statement against it.**

Solution : CREATE VIEW ACCSVW10 AS SELECT \* FROM ACCOUNTMASTER WITH READ ONLY CONSTRAINT READ\_ONLY\_VIEW;

View created.



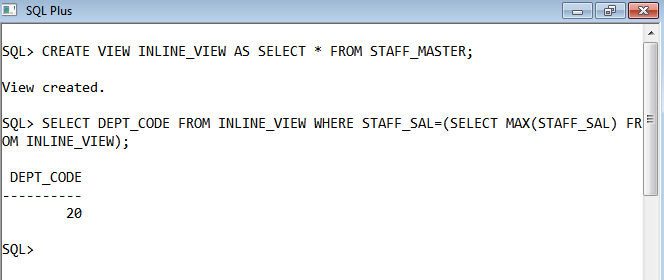
**34. Display the department from Staff table which has the highest salary by using**

**Inline View.**

Solution :

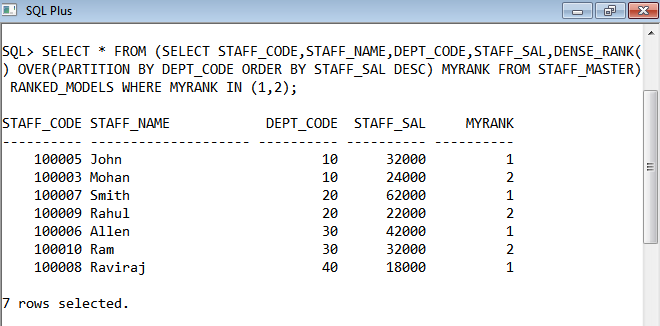
CREATE VIEW INLINE\_VIEW AS SELECT \* FROM STAFF\_MASTER;

SQL> SELECT DEPT\_CODE FROM INLINE\_VIEW WHERE STAFF\_SAL=(SELECT MAX(STAFF\_SAL) FROM INLINE\_VIEW);



**35. List the top two highest earning employees in each department.**

Solution : SELECT \* FROM (SELECT STAFF\_CODE,STAFF\_NAME,DEPT\_CODE,STAFF\_SAL,DENSE\_RANK() OVER(PARTITION BY DEPT\_CODE ORDER BY STAFF\_SAL DESC) MYRANK FROM STAFF\_MASTER) RANKED\_MODELS WHERE MYRANK IN (1,2);



**36. Create a Sequence with the name Seq\_Dept on Deptno column of Dept table. It**

**should start from 40 and stop at 200. Increment parameter for the sequence**

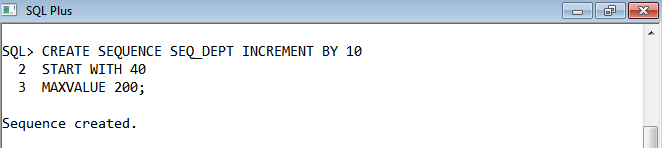
**Seq\_Dept should be in step of 10.**

Solution :

CREATE SEQUENCE SEQ\_DEPT INCREMENT BY 10

START WITH 40

MAXVALUE 200;

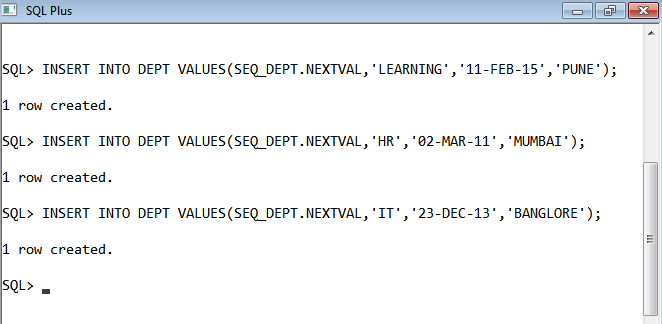


**37. Insert three sample rows by using the above sequence in Dept table.**

Solution : INSERT INTO DEPT VALUES(SEQ\_DEPT.NEXTVAL,'LEARNING','11-FEB-15','PUNE');

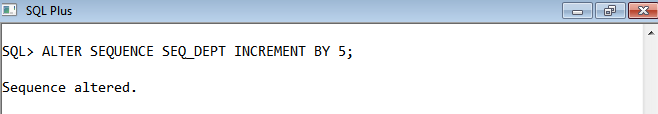
SQL> INSERT INTO DEPT VALUES(SEQ\_DEPT.NEXTVAL,'HR','02-MAR-11','MUMBAI');

SQL> INSERT INTO DEPT VALUES(SEQ\_DEPT.NEXTVAL,'IT','23-DEC-13','BANGLORE');



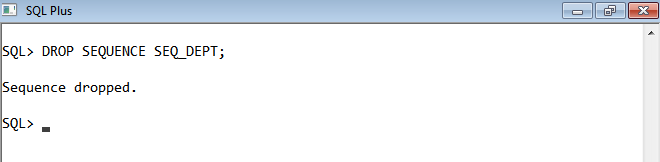
**38. Alter the above specified sequence with an increment by 5.**

Solution : ALTER SEQUENCE SEQ\_DEPT INCREMENT BY 5;



**39. Drop the Seq\_Dept sequence.**

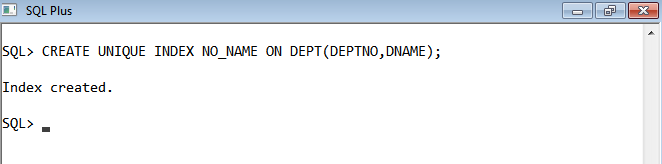
Solution : DROP SEQUENCE SEQ\_DEPT;



**40. Create a Unique index with the name No\_Name on DeptNo and Dname of Dept**

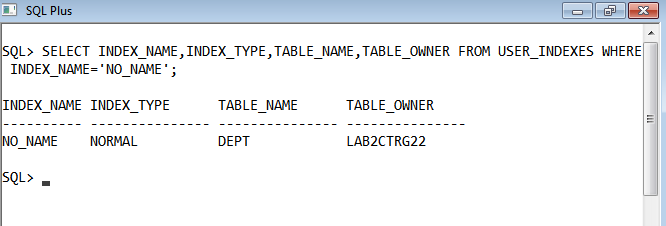
**table.**

Solution : CREATE UNIQUE INDEX NO\_NAME ON DEPT(DEPTNO,DNAME);



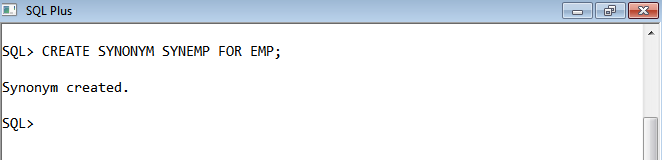
**41. Get information on the index No\_Name from the Data Dictionary.**

Solution : SELECT INDEX\_NAME,INDEX\_TYPE,TABLE\_NAME,TABLE\_OWNER FROM USER\_INDEXES WHERE INDEX\_NAME=’NO\_NAME’;



**42. Create public synonym synEmp for the EMP table.**

Solution : SELECT INDEX\_NAME,INDEX\_TYPE,TABLE\_NAME,TABLE\_OWNER FROM USER\_INDEXES WHERE INDEX\_NAME='NO\_NAME';



**43. Get Information on synonym synEmp from the Data Dictionary.**

Solution : SELECT SYNONYM\_NAME,TABLE\_OWNER,TABLE\_NAME FROM USER\_SYNONYMS WHERE SYNONYM\_NAME='SYNEMP';

