## 4.1: Database Objects

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

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.

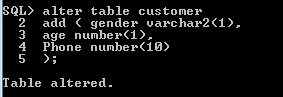


**3.** Add the following Columns to the Customer table.

Gender Varchar2(1)

Age Number(3)

PhoneNo Number(10)

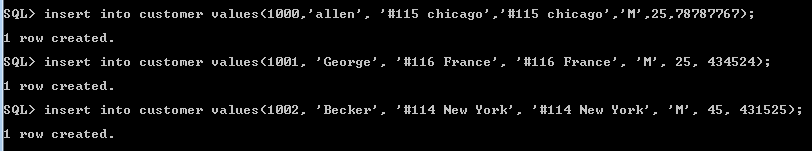


**4.** Insert rows with the following data in to the Customer table.

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

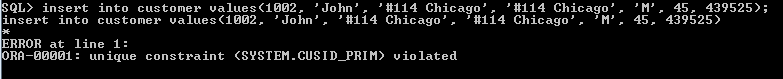


**5.** Add the Primary key constraint for Customerld with the name Custld\_Prim.



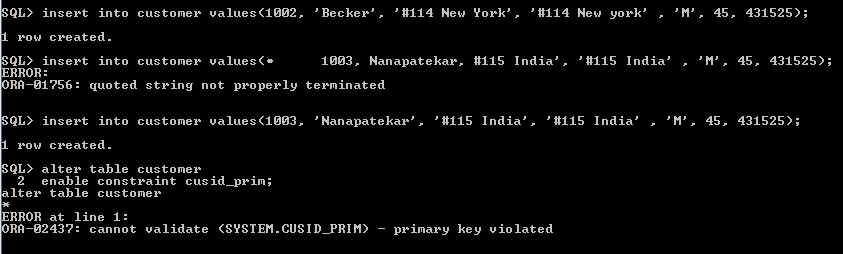
**6.** Insert the row given below in the Customer table and see the message generated by the Oracle server.

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

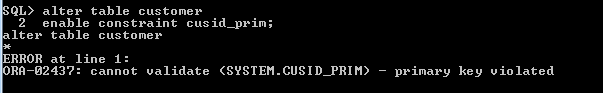


**7.** Disable the constraint on CustomerId, and insert the following data:

* 1002, Becker, #114 New York, #114 New york , M, 45, 431525
* 1003, Nanapatekar, #115 India, #115 India , M, 45, 431525

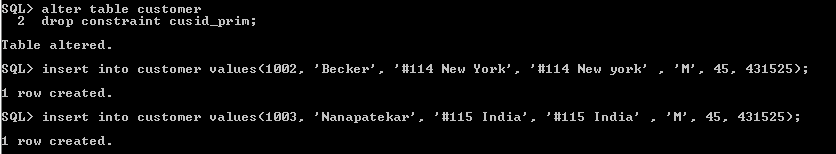


**8.** Enable the constraint on CustomerId of the Customer table, and see the message generated by the Oracle server.

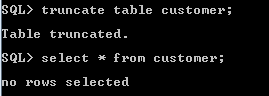


**9.** Drop the constraint Custld\_Prim on CustomerId and insert the following Data. Alter Customer table, drop constraint Custid\_Prim.

* 1002, Becker, #114 New York, #114 New york , M, 45, 431525, 15000.50
* 1003, Nanapatekar, #115 India, #115 India , M, 45, 431525, 20000.50



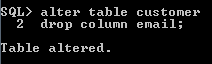
**10.** Delete all the existing rows from Customer table, and let the structure remain itself using TRUNCATE statement.



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



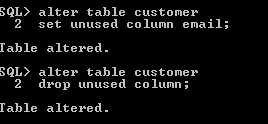
**12.** Drop the E\_mail column from Customer table.



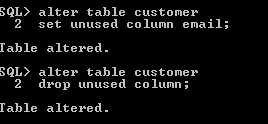
**13.** Add a new column EmailId to Customer table.



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



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



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



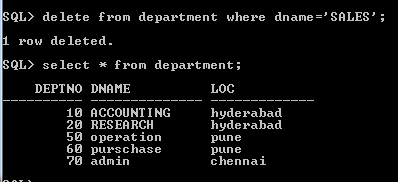
**17.** Use Data Dictionary USER\_TAB\_COMMENTS to view the created comment.



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

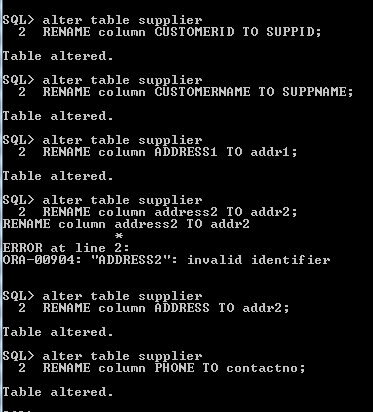


**19.** Use Data Dictionary USER\_COL\_COMMENTS to view the created comment.



**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.



**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

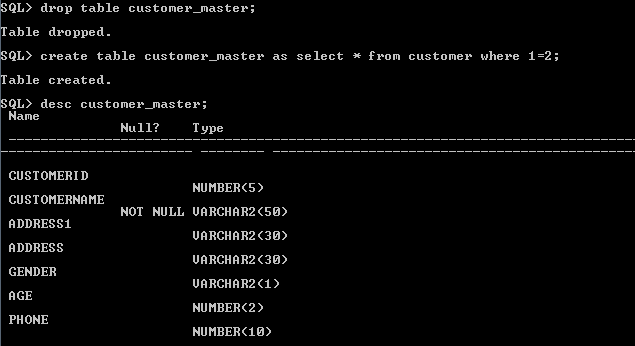
Addressl Varchar2(30) Not Null

Address2 Varchar2(30)

Gender Varchar2(l)

Age Number(3)

PhoneNo Number(10)



**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



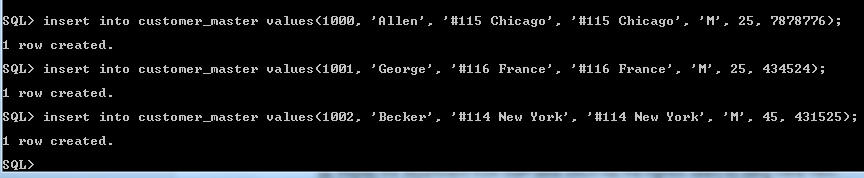


**23.** Relate AccountsMaster table and CustomerMaster table through Customerld column with the constraint name Cust\_acc.

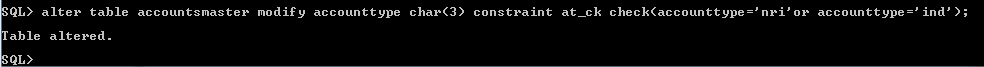


**24.** Insert the following rows to the CustomerMaster table:

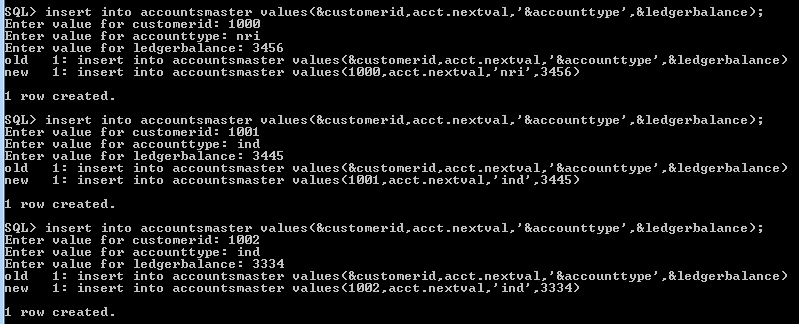
* 1000, Allen, #115 Chicago, #115 Chicago, M, 25, 7878776
* 1001, George, #116 France, #116 France, M, 25, 434524
* 1002, Becker, #114 New York, #114 New York, M, 45, 431525



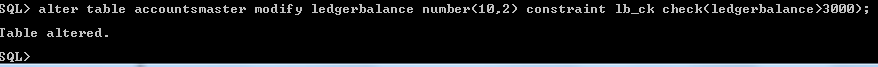
**25.** Modify the AccountMaster table with the Check constraint to ensure AccountType should be either NRI or IND.



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



**27.** Modify the AccountsMaster table keeping a Check constraint with the name Balance\_Check for the Minimum Balance which should be greater than 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.



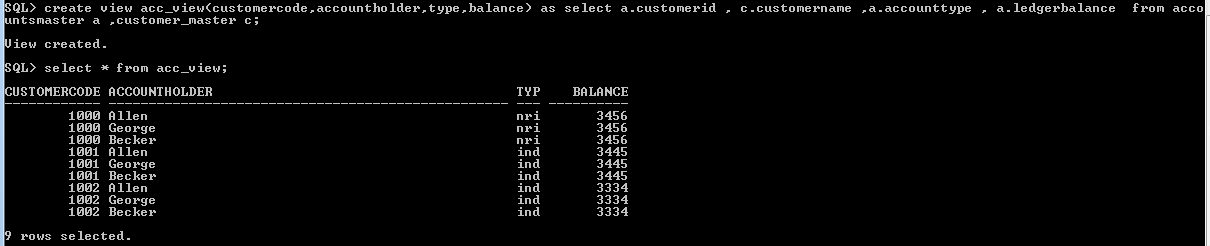
**29.** Create Backup copy for the AccountsMaster table with the name ‘AccountDetails’.



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

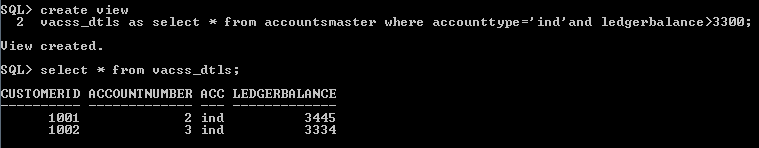


**31.** Create a view ‘Acc\_view’ with columns Customerld, CustomerName, AccountNumber, AccountType, and LedgerBalance from AccountsMaster. In the view Acc\_view, the column names should be CustomerCode, AccountHolderName, AccountNumber, Type, and Balance for the respective columns from AccountsMaster table.

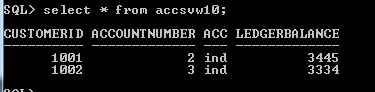


**32.** Create a view on AccountsMaster table with name vAccs\_Dtls. This view should list all customers whose AccountType is ‘IND’ and their balance amount should not be less than 10000. Using this view any DML operation should not violate the view conditions.

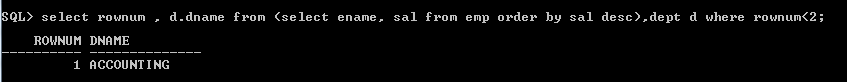
**hint2Hint:** Use the With Check Option constraint.



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



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

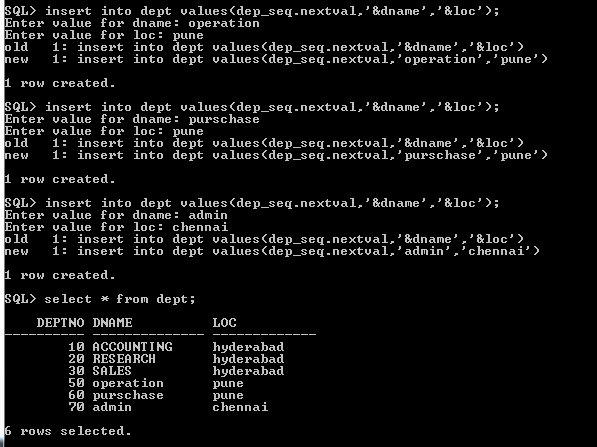


**35.** List the top two highest earning employees in each department.

**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.



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



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



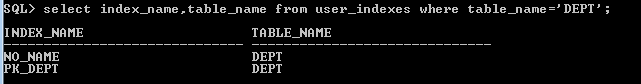
**39.** Drop the Seq\_Dept sequence.



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



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



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



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

