L09-DBS301-Create alter and more

You will **create tables** first, then **add / modify /remove some columns** and finally **add / modify / remove some constraints** in this lab.

1. Create table SALESREP and load it with data from table EMPLOYEEStable. Use only the equivalent columns from EMPLOYEE as shown below. (Do NOT create this table from scratch), AND only for people in department 80.

*SALESREP*

Column Type

RepId NUMBER (6)

FName VARCHAR2(20)

LName VARCHAR2(25)

Phone# VARCHAR2(20) ALL these columns’ data types match ones

Salary NUMBER(8,2) in table EMPLOYEES

Commission NUMBER(2,2)

**You will have exactly 3 rows here**.

Answer:

CREATE TABLE SALESREP (RepId, FName, LName NOT NULL, Phone#, Salary, Commission)

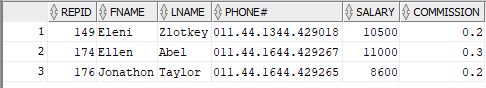
AS (SELECT Employee\_id, First\_Name, Last\_Name, Phone\_Number, Salary, Commission\_Pct

FROM employees

WHERE department\_id = 80);

1.JPG

SELECT \* FROM SALESREP;



2a. Create CUST table.

CREATE TABLE CUST

(CUST# NUMBER(6),

CUSTNAME VARCHAR2(30),

CITY VARCHAR2(20),

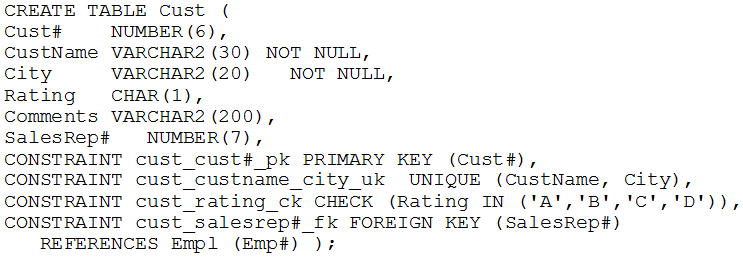
RATING CHAR(1),

COMMENTS VARCHAR2(200),

SALESREP# NUMBER(7)

)

The constraints were left off in the above. The constraints shown below are what would normally be applied as shown. These were applied at the table level



CREATE TABLE CUST( CUST# NUMBER(6),

CUSTNAME VARCHAR2(30) NOT NULL,

CITY VARCHAR2(20) NOT NULL,

RATING CHAR(1),

COMMENTS VARCHAR2(200),

SALESREP# NUMBER(7),

CONSTRAINT cust\_cust#\_pk PRIMARY KEY(Cust#),

CONSTRAINT cust\_custname\_city\_uk UNIQUE(CustName, City),

CONSTRAINT cust\_rating\_ck CHECK (Rating IN('A', 'B', 'C', 'D')),

CONSTRAINT cust\_salesrep#\_fk FOREIGN KEY(SalesRep#)

REFERENCES employees(employee\_id)

);

Load the table with these values in the chart. Some of the inserts have been done for you.

See below the chart

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CUST#** | **CUSTNAME** | **CITY** | **RAT** | **SALESREP#** |
| 501 | ABC LTD. | Montreal | C | 201 |
| 502 | Black Giant | Ottawa | B | 202 |
| 503 | Mother Goose | London | B | 202 |
| 701 | BLUE SKY LTD | Vancouver | B | 102 |
| 702 | MIKE and SAM Inc. | Kingston | A | 107 |
| 703 | RED PLANET | Mississauga | C | 107 |
| 717 | BLUE SKY LTD | Regina | D | 102 |

7 rows selected.

NOTE: Caution that copying from WORD will create errors if WORD is using quotes that look like ‘this’

SQL needs straight quotes like 'this'

These have been corrected for you

INSERT INTO CUST VALUES (501, 'ABC LTD', 'Montreal', 'C', NULL, 201)

Insert the rest of the rows. Here is a few but not all of them done for you

INSERT INTO CUST values (701, 'MIKE and SAM inc.', 'Kingston', 'A', NULL, 102);

INSERT INTO CUST values (703, 'RED PLANET', 'Mississauga', 'C', NULL, 107);

INSERT INTO CUST values (717, 'blue sky ltd', 'Regina', 'D', NULL, 102);

Answer:

INSERT INTO CUST VALUES(501, 'ABC LTD', 'Montreal', 'C', NULL, 201);

INSERT INTO CUST VALUES(502, 'Black Giant', 'Ottawa', 'B', NULL, 202);

INSERT INTO CUST VALUES(503, 'Mother Goose', 'London', 'B', NULL, 202);

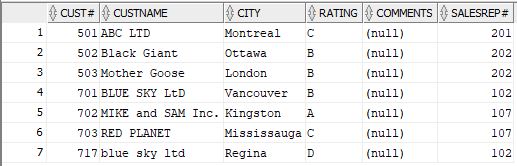
INSERT INTO CUST VALUES(701, 'BLUE SKY LtD', 'Vancouver', 'B', NULL, 102);

INSERT INTO CUST VALUES(702, 'MIKE and SAM Inc.', 'Kingston', 'A', NULL, 107);

INSERT INTO CUST VALUES(703, 'RED PLANET', 'Mississauga', 'C', NULL, 107);

INSERT INTO CUST VALUES(717, 'blue sky ltd', 'Regina', 'D', NULL, 102);

SELECT \* FROM CUST;



2b. Create table GOODCUST **from table CUST** by using following columns (do NOT create this table from scratch), but only if their rating is A or B.

*GOODCUST*

Column Type

CustId NUMBER (6)

Name VARCHAR2(30)

Location VARCHAR2(20) 🡪 ALL these columns’ data types match ones

RepId NUMBER(7) in table CUST

**🡪 You will have exactly 4 rows here.**

|  |  |  |  |
| --- | --- | --- | --- |
| **CUSTID** | **NAME** | **LOCATION** | **REPID** |
| 502 | Black Giant | Ottawa | 202 |
| 503 | Mother Goose | London | 202 |
| 504 | BLUE SKY LTD | Vancouver | 202 |
| 701 | MIKE and SAM inc. | Kingston | 10 |

Answer:

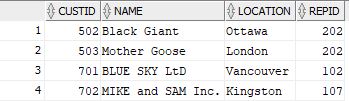
CREATE TABLE GOODCUST (CustId, Name, Location, Repid)

AS SELECT cust#, custname, city, salesrep#

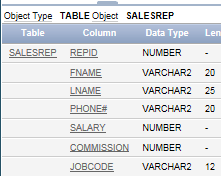
FROM CUST

WHERE rating IN('A','B');

SELECT \* FROM GOODCUST;



3. Now add new column to table SALESREP called JobCode that will be of variable character type with maximum length of 12. Do a DESC SALESREP to ensure it executed

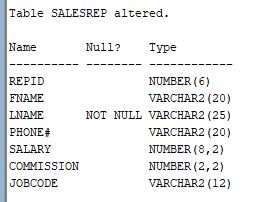


Answer:

ALTER TABLE SALESREP

ADD JobCode VARCHAR2(12);

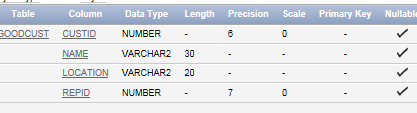
DESC SALESREP;



4. Declare column Salary in table SALESREP as mandatory one and

Column Location in table GOODCUST as optional one. You can see location is already optional.

GODCUST before looks like the following.



AFTER the change it would look as follows:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Column** | **Data Type** | **Length** | **Precision** | **Scale** | **Primary Key** | **Nullable** | **Default** | **Comment** |
| [SALESREP](javascript:ret_Column('RON.SALESREP');) | [REPID](javascript:ret_Column('REPID');) | NUMBER | - | 6 | 0 | 1 | - | - | - |
|  | [FNAME](javascript:ret_Column('FNAME');) | VARCHAR2 | 37 | - | - | - | nullable | - | - |
|  | [LNAME](javascript:ret_Column('LNAME');) | VARCHAR2 | 25 | - | - | - | - | - | - |
|  | [PHONE#](javascript:ret_Column('PHONE#');) | VARCHAR2 | 20 | - | - | - | nullable | - | - |
|  | [SALARY](javascript:ret_Column('SALARY');) | NUMBER | - | 8 | 2 | - | - | - | - |
|  | [COMMISSION](javascript:ret_Column('COMMISSION');) | NUMBER | - | 2 | 2 | - | nullable | - | - |
|  | [JOBCODE](javascript:ret_Column('JOBCODE');) | VARCHAR2 | 12 | - | - | - | nullable |  |  |

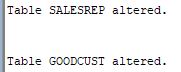
Answer:

ALTER TABLE SALESREP

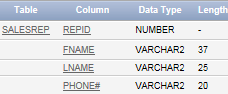
MODIFY Salary NOT NULL;

ALTER TABLE GOODCUST

MODIFY Location NULL;



5. Lengthen FNAME in SALESREP to 37. The result of a DESCIBE should show it happening

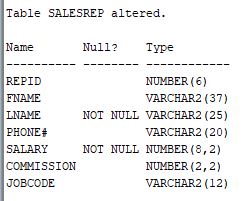


Answer:

ALTER TABLE SALESREP

MODIFY FNAME VARCHAR2(37);

DESC SALESREP;



You can only decrease the size or length of NAME in GOODCUST to the maximum length of data already stored. Do it by using SQL and not by looking at each entry and counting the characters. May take two SQL statements

Answer:

SELECT MAX(LENGTH(NAME))

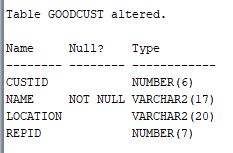
FROM GOODCUST;

8.JPG

ALTER TABLE GOODCUST

MODIFY name VARCHAR2(17);

DESC GOODCUST;



6. Now get rid of the column JobCode in table SALESREP in a way that will not affect daily performance.

Answer:

ALTER TABLE SALESREP

SET UNUSED(JobCode);

10.JPG

7. Declare PK constraints in both new tables 🡪RepId and CustId

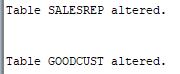
Answer:

ALTER TABLE SALESREP

ADD CONSTRAINT sale\_rep\_id\_pk PRIMARY KEY(Repid);

ALTER TABLE GOODCUST

ADD CONSTRAINT good\_cust\_id\_pk PRIMARY KEY(Custid);



8. Declare UK constraints in both new tables 🡪 Phone# and Name

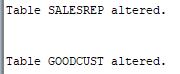
Answer:

ALTER TABLE SALESREP

ADD CONSTRAINT sale\_phone\_uk UNIQUE(Phone#);

ALTER TABLE GOODCUST

ADD CONSTRAINT good\_name\_uk UNIQUE(Name);



9. Restrict amount of Salary column to be in the range [6000, 12000] and Commission to be not more than 50%.

Answer:

ALTER TABLE SALESREP

ADD CONSTRAINT sale\_salary\_ck CHECK((salary BETWEEN 6000 AND 12000)

AND (commission < 0.5));

12.JPG

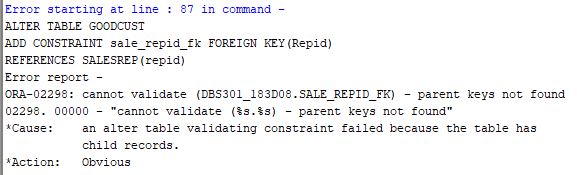
10. Ensure that only valid RepId numbers from table SALESREP may be entered in the table GOODCUST. Why this statement has failed?

Answer:

ALTER TABLE GOODCUST

ADD CONSTRAINT sale\_repid\_fk FOREIGN KEY(Repid)

REFERENCES SALESREP(repid);



‘RepId’ can only be entered in goodcust table because goodcust table has rep\_id as a foreign key.

11. Firstly write down the values for RepId column in table GOODCUST and then make all these values blank. Now redo the question 10. Was it successful?

Answer:

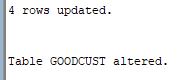
UPDATE GOODCUST

SET REPID = NULL;

ALTER TABLE GOODCUST

ADD CONSTRAINT sale\_repid\_fk FOREIGN KEY(Repid)

REFERENCES SALESREP(repid);



12. Disable this FK constraint now and enter old values for RepId in table GOODCUST and save them. Then try to enable your FK constraint. What happened?

Answer:

ALTER TABLE GOODCUST

DISABLE CONSTRAINT sale\_repid\_fk;

UPDATE GOODCUST

SET REPID = 202

WHERE CUSTID = 502;

UPDATE GOODCUST

SET REPID = 202

WHERE CUSTID = 503;

UPDATE GOODCUST

SET REPID = 102

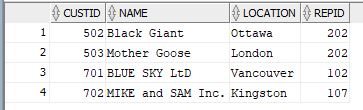
WHERE CUSTID = 701;

UPDATE GOODCUST

SET REPID = 107

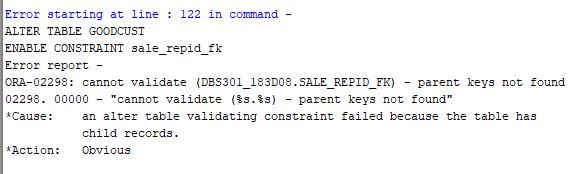
WHERE CUSTID = 702;

SELECT \* FROM GOODCUST;



ALTER TABLE GOODCUST

ENABLE CONSTRAINT sale\_repid\_fk;



13. Get rid of this FK constraint. Then modify your CK constraint from question 9 to allow Salary amounts from 5000 to 15000.

Answer:

ALTER TABLE GOODCUST

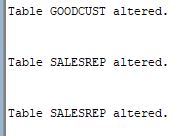
DROP CONSTRAINT sale\_repid\_fk;

ALTER TABLE SALESREP

DROP CONSTRAINT sale\_salary\_ck;

ALTER TABLE SALESREP

ADD CONSTRAINT sale\_salary\_ck CHECK((salary BETWEEN 5000 AND 15000) AND (commission < 0.5));

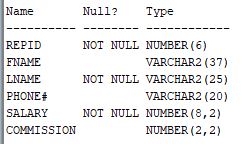


14. Describe both new tables SALESREP and GOODCUST and then show all constraints

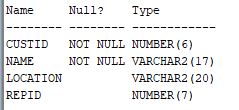
for these two tables by running the following query:

Answer:

DESC SALESREP;



DESC GOODCUST;



SELECT constraint\_name, constraint\_type, search\_condition, table\_name

FROM user\_constraints

WHERE table\_name IN ('SALESREP','GOODCUST')

ORDER BY 4 , 2

