Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Department of Information Technology

NAME: Ayush Vinod Upadhyay
ROLL No: I025
Sap ID: 60003220131
BRANCH: Information Technology
Batch: I1-1

S.Y. BTech (IT)
SUB: DBMS LAB
Experiment No: 2
Study of Basic SQL commands

Table:

- Database object that holds user's data
- Spreadsheet where cells corresponds to column if a table having a specific data type

Data Types in ORACLE:

Char (Size):

- User to store character string value of fixed length determined by size parameter.
- Up to 2000 bytes
- If inserted value is shorter than the size defined for it then it will padded with spaces.

Varchar2(Size):

- Used to store variable length alphanumeric data.
- Data can vary in the number of chars, but length cannot exceed 4000 bytes.
- The inserted value is not padded with spaces.

Date:

- Used to represent date and time
- Standard format is DD-MON-YY i.e. 26- jul -07

Integer:

- It can store upto 38 digits.

Number(p,d)

- Fixed point number, with user-specified precision of *p* digits, with *d* digits to the right of decimal point.

CREATE:

Syntax:

Create table < table name > (< columnname1>) <data type> (<size>), <columnname2> <data type> (<size>)…);

Defines each column of the table uniquely
Each column has a minimum of three attributes;

1. Name
2. Data type
3. Size

Example:

Create Table Bank (AccNum char(10),
FirstName Varchar2(20), LastName Varchar2 (20),
Address Varchar2(50), City Varchar2 (10), PinCode int);

INSERT:

- Stores / loads data /values into the table column with a one to one relationship
- If there are lese values being described than there are columns in the table then it is mandatory to indicate the column name and it corresponding value.

Syntax:

- **To insert data into all columns:**

INSERT INTO <Table Name> VALUES (<expression 1>, <expression 2>);

- **To insert data into some selected columns:** INSERT INTO <Table Name> (<column name1>, <Column name2>…) VALUES (<expression 1>,<expression 2>);

Example:

INSERT INTO
BANK(AccNum,FirstName,LastName,Address,City,PinCode)
VALUES('A11','Vijay','Pandey','Bandra','Mumbai',400051);

SELECT:

Used to view the data stored in a table.

Syntax:

```
SELECT * FROM <table name>;

e.g. SELECT * FROM Bank;
```

The table data is filtered by three ways:- •
Selected columns and all rows

- Selected rows and all columns
- Selected columns and selected rows

Selecting a specific column:

Syntax:

```
SELECT <column name1>, <column name2 > From <Table name>;
```

Example: SELECT AccNo, First name, FROM Bank;

Selecting specific rows:

Syntax:

```
SELECT * From <Table Name> WHERE <Condition>
```

Example:

Display the information of account holders from Bank whose account number is SB001

```
SELECT * FROM BANK WHERE AccNo = ‘A11’;
```

Selecting a specific row and specific column:

Syntax:

```
SELECT <column name1>, <column name 2>
FROM <Table name >
WHERE<condition>
```

E.g. Display the AccNo, First name and Last name from Bank where the Pin code is 400051.

```
SELECT AccNo, First Name, Last Name
FROM Bank
WHERE Pin code =400051
```

Eliminating Duplicate Rows While Displaying:

Syntax:
SELECT DISTINCT< column name1>, <column name2 > FROM <table name>;

SELECT DISTINCT:
Displays only unique values and eliminates rows that have exactly the same content in each column.

Syntax:
SELECT DISTINCT * FROM <table name>;

Example:
Eliminate the duplicate Pin code when retrieving data from the Pin code column of the table Bank
SELECT DISTICT Pin code
FROM Bank;

Example:
Select only unique rows from the bank
SELECT DISTINCT * FROM Bank;

DELETE: Used to remove either all rows or used of rows from a table, remove all rows

Syntax:
DELETE FROM <table name>;
e.g. delete all rows from the table bank
DELETE FROM Bank;

Remove specific rows:
Syntax:
DELETE FROM <table name> WHERE <condition>;

Example:
Delete rows from the table bank where the value in the pin code field is 400051

DELETE FROM Bank WHERE pin code = 400051.

UPDATE: Used to change or modify data values in a table Updating all rows

Syntax:
UPDATE <table name>
SET <column name>= <expression >,
<Column name >=<expression >;

Example:
Change the city from Navi Mumbai to Bombay
UPDATE Bank SET City = 'Bombay';

Updating records on condition:
Syntax:
Update <table name >
SET <column name>=<expression>, <column
name>=<expression> WHERE <condition>; **Example:**

Update the table Bank change the contents of the field First name to Mr. Vijay From the record identified by the field AccNo containing the value A12;

UPDATE BANK SET First Name = 'Mr. Vijay'
WHERE AccNo = 'A12';

Truncating a table: Truncate table statement is fast, a non-logged method of deleting all rows in the table

Syntax:
TRUNCATE TABLE <table name>
Example:
TRUNCATE TABLE BANK;

Destroying a table:
Syntax:
DROP TABLE <table name>;
Example:
Destroy the table Bank and all the data stored in it.
DROP TABLE BANK;

---EXERCISE--

Implement the SQL statements for the following questions

1. **Create the given tables and insert the respective values in the tables.**

Customer Master : CUST		Movie Master : MOVIE		Invoice Transaction : INVOICE	
Column Name	Format	Column Name	Format	Column Name	Format
Cust_id	Varchar2(3)	Movie_no	Integer	Inv_no	Varchar2(3)
Lname	Varchar2(15)	Title	Varchar2(15)	Movie_no	Integer
Fname	Varchar2(15)	Type	Varchar2(10)	Cust_id	Varchar2(3)
Area	Varchar2(2)	Star	Varchar2(25)	Issue_date	Date
Phone_no	Integer	Price	Number(8,2)	Return_date	Date

2. **Insert the following data in the respective tables Data for CUSTOMER table:**

Cust_id	Lname	Fname	Area	Phone_no
A01	Border	Allan	SA	723622
A02	Shields	Tina	Mo	123784
A03	Kumar	Ravi	BI	545621
A04	Rai	Sunita	CH	983724
A05		Sachin	DR	253489
A06	Smith	James	WA	634672

Data for MOVIE table:

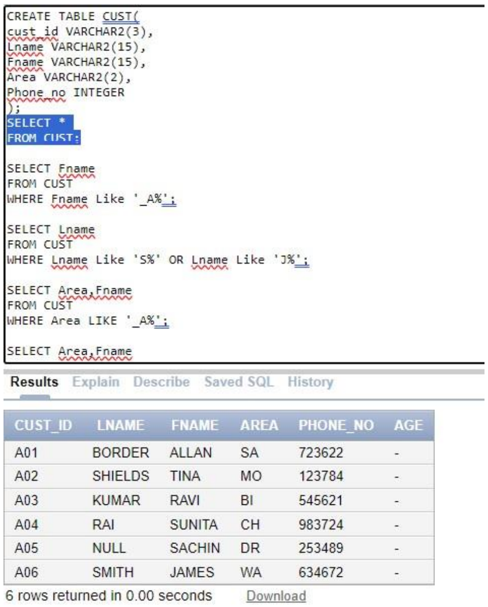
Movieno	Title	Type	Star	Price
1	Carry On doctor	Comedy	Leslie Phollips	175.00
2	The Firm	Thriller	Tom Cruise	200.00
3	Pretty Woman	Romance	Richard Gere	150.55
4	Home Alone	Comedy	Macaulay Culkin	150.00
5	The Fugitive	Thriller	Harison Ford	200.00
6	Coma	Suspense	Michael Douglas	100.00
7	Dracula	Horror	Gary Oldman	150.25
8	Quick Change	Comedy	Bill Murray	190.00

Data for INVOICE Table:

Inv_no	Movie_no	Cust_id	Issue_date	Return_date
I01	4	A01	23-jul-95	28-jul-95
I02	3	A02	12-aug-95	15-aug-95
I03	1	A02	10-sep-95	16-sep-95
I04	6	A03	23-jul-95	24-jul-95
I05	7	A04	28-jul-95	29-jul-95
I06	2	A06	01-sep-95	04-sep-95
I07	9	A05	07-aug-95	08-aug-95
I08	9	A01	18-aug-95	22-sep-95
I09	5	A03	06-jul-95	09-jul-95
I10	8	A06	02-aug-95	05-aug-95

3. **Referring to the above tables and data solve the following queries.**

   a. Print the entire customer table.

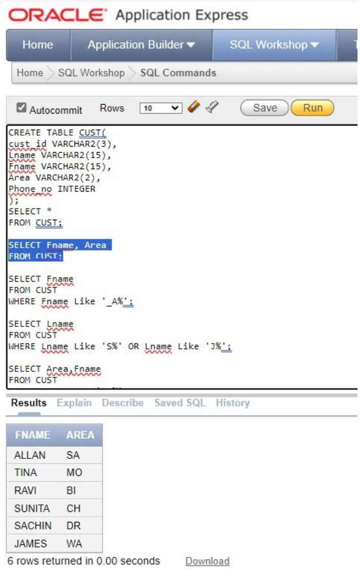


```
CREATE TABLE CUST(
cust_id VARCHAR2(3),
Lname VARCHAR2(15),
Fname VARCHAR2(15),
Area VARCHAR2(2),
Phone_no INTEGER
);
SELECT *
FROM CUST;

SELECT Fname
FROM CUST
WHERE Fname Like '_A%';

SELECT Lname
FROM CUST
WHERE Lname Like 'S%' OR Lname Like 'J%';

SELECT Area,Fname
FROM CUST
WHERE Area LIKE '_A%';

SELECT Area,Fname
```

CUST_ID	LNAME	FNAME	AREA	PHONE_NO	AGE
A01	BORDER	ALLAN	SA	723622	-
A02	SHIELDS	TINA	MO	123784	-
A03	KUMAR	RAVI	BI	545621	-
A04	RAI	SUNITA	CH	983724	-
A05	NULL	SACHIN	DR	253489	-
A06	SMITH	JAMES	WA	634672	-

6 rows returned in 0.00 seconds

b. Retrieve the list of fname and the area of all the customers.



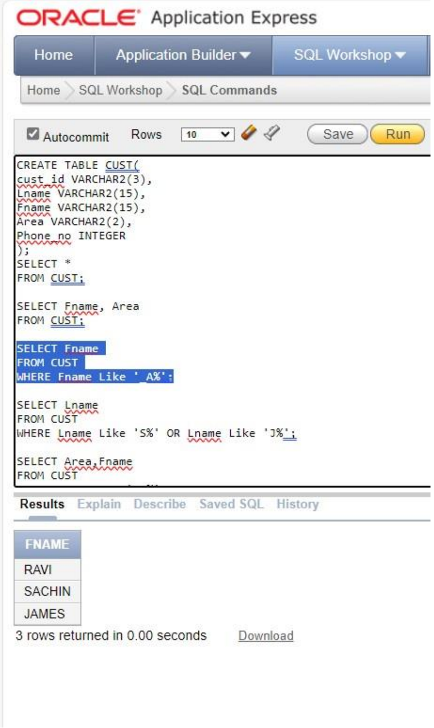
☑ Autocommit Rows 10

```
CREATE TABLE CUST(
cust_id VARCHAR2(3),
Lname VARCHAR2(15),
Fname VARCHAR2(15),
Area VARCHAR2(2),
Phone_no INTEGER
);
SELECT *
FROM CUST;

SELECT Fname, Area
FROM CUST;

SELECT Fname
FROM CUST
WHERE Fname Like '_A%';

SELECT Lname
FROM CUST
WHERE Lname Like 'S%' OR Lname Like 'J%';

SELECT Area,Fname
FROM CUST
```

FNAME	AREA
ALLAN	SA
TINA	MO
RAVI	BI
SUNITA	CH
SACHIN	DR
JAMES	WA

6 rows returned in 0.00 seconds

c. Find the names of all the customers having ‘a’ as the second letter in fname.



☑ Autocommit Rows 10

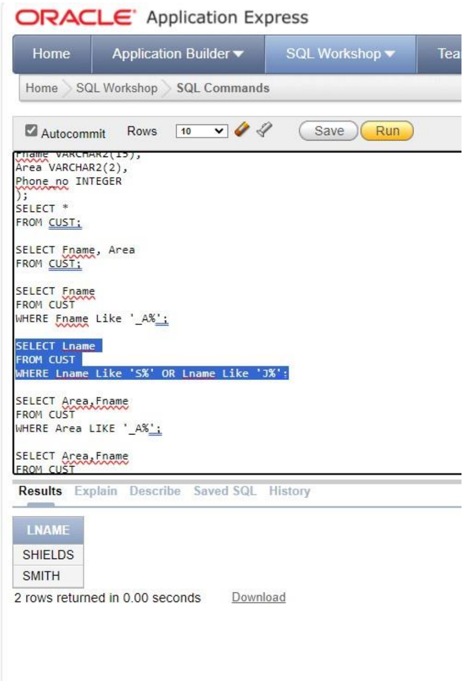
```
CREATE TABLE CUST(
cust_id VARCHAR2(3),
Lname VARCHAR2(15),
Fname VARCHAR2(15),
Area VARCHAR2(2),
Phone_no INTEGER
);
SELECT *
FROM CUST;

SELECT Fname, Area
FROM CUST;

SELECT Fname
FROM CUST
WHERE Fname Like '_A%';

SELECT Lname
FROM CUST
WHERE Lname Like 'S%' OR Lname Like 'J%';

SELECT Area,Fname
FROM CUST
```

Results

FNAME
RAVI
SACHIN
JAMES

3 rows returned in 0.00 seconds

d. Find the lname of all customers that begin with ‘S’ or ‘J’.



☑ Autocommit Rows 10

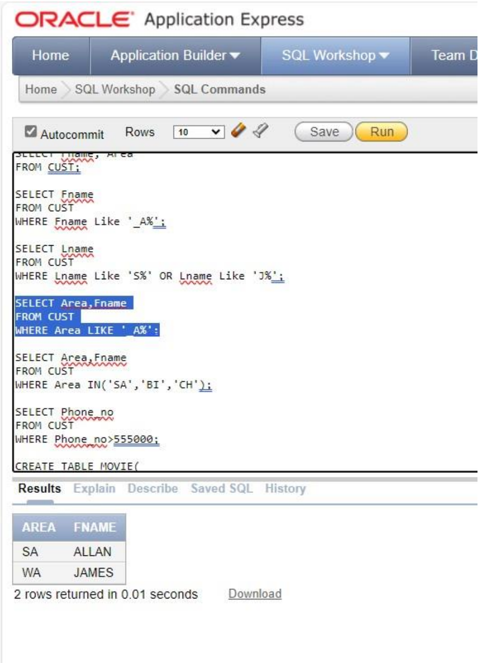
```
Fname VARCHAR2(15),
Area VARCHAR2(2),
Phone_no INTEGER
);
SELECT *
FROM CUST;

SELECT Fname, Area
FROM CUST;

SELECT Fname
FROM CUST
WHERE Fname Like '_A%';

SELECT Lname
FROM CUST
WHERE Lname Like 'S%' OR Lname Like 'J%';

SELECT Area,Fname
FROM CUST
WHERE Area LIKE '_A%';

SELECT Area,Fname
FROM CUST
```

Results

LNAME
SHIELDS
SMITH

2 rows returned in 0.00 seconds

e. Find out the customers who stay in an area whose 2nd letter is ‘a’.

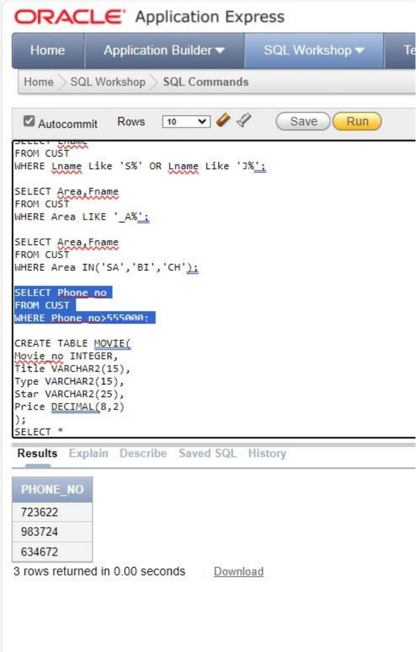


```
SELECT Fname, Area
FROM CUST;

SELECT Fname
FROM CUST
WHERE Fname Like '_A%';

SELECT Lname
FROM CUST
WHERE Lname Like 'S%' OR Lname Like 'J%';

SELECT Area,Fname
FROM CUST
WHERE Area LIKE '_A%';

SELECT Area,Fname
FROM CUST
WHERE Area IN('SA','BI','CH');

SELECT Phone_no
FROM CUST
WHERE Phone_no>555000;

CREATE TABLE MOVIE(
```

AREA	FNAME
SA	ALLAN
WA	JAMES

2 rows returned in 0.01 seconds

f. Print the list of customers whose phone numbers are greater than 555000.

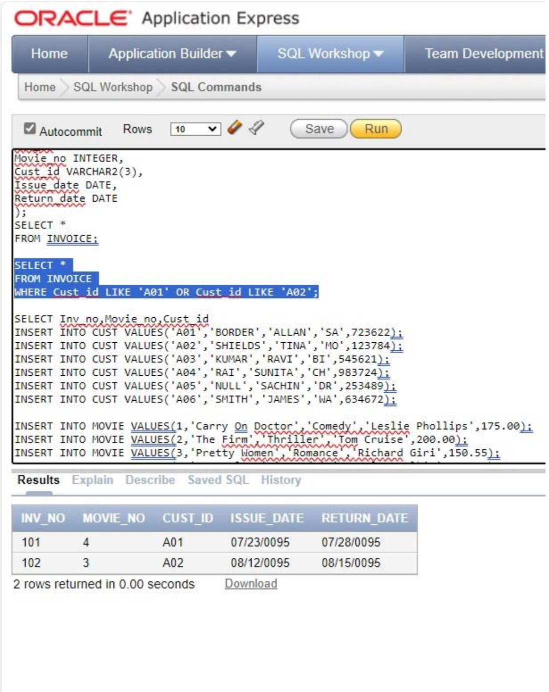


```
SELECT Lname
FROM CUST
WHERE Lname Like 'S%' OR Lname Like 'J%';

SELECT Area,Fname
FROM CUST
WHERE Area LIKE '_A%';

SELECT Area,Fname
FROM CUST
WHERE Area IN('SA','BI','CH');

SELECT Phone_no
FROM CUST
WHERE Phone_no>555000;

CREATE TABLE MOVIE(
Movie_no INTEGER,
Title VARCHAR2(15),
Type VARCHAR2(15),
Star VARCHAR2(25),
Price DECIMAL(8,2)
);
SELECT *
```

PHONE_NO
723622
983724
634672

3 rows returned in 0.00 seconds

g. Display the invoice table information for cust_id 'A01' and 'A02'.

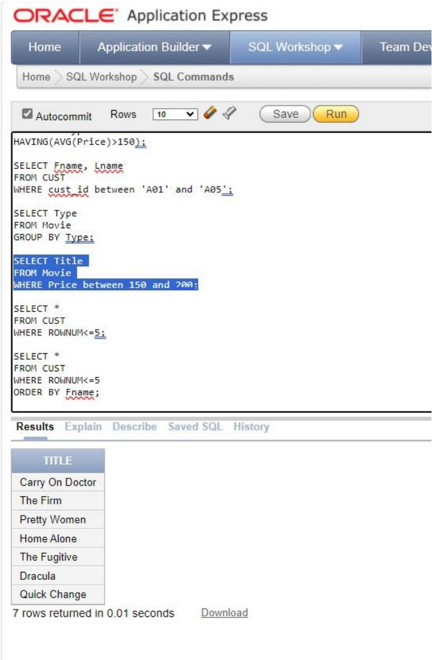


```
Movie_no INTEGER,
Cust_id VARCHAR2(3),
Issue_date DATE,
Return_date DATE
);
SELECT *
FROM INVOICE;

SELECT *
FROM INVOICE
WHERE Cust_id LIKE 'A01' OR Cust_id LIKE 'A02';

SELECT Inv_no,Movie_no,Cust_id
INSERT INTO CUST VALUES('A01','BORDER','ALLAN','SA',723622);
INSERT INTO CUST VALUES('A02','SHIELDS','TINA','MO',123784);
INSERT INTO CUST VALUES('A03','KUMAR','RAVI','BI',545621);
INSERT INTO CUST VALUES('A04','RAI','SUNITA','CH',983724);
INSERT INTO CUST VALUES('A05','NULL','SACHIN','DR',253489);
INSERT INTO CUST VALUES('A06','SMITH','JAMES','WA',634672);

INSERT INTO MOVIE VALUES(1,'Carry On Doctor','Comedy','Leslie Phollips',175.00);
INSERT INTO MOVIE VALUES(2,'The Firm','Thriller','Tom Cruise',200.00);
INSERT INTO MOVIE VALUES(3,'Pretty Women','Romance','Richard Giri',150.55);
```

INV_NO	MOVIE_NO	CUST_ID	ISSUE_DATE	RETURN_DATE
101	4	A01	07/23/0095	07/28/0095
102	3	A02	08/12/0095	08/15/0095

2 rows returned in 0.00 seconds

h. Find the movies whose price is greater than 150 and less than or equal to 200.

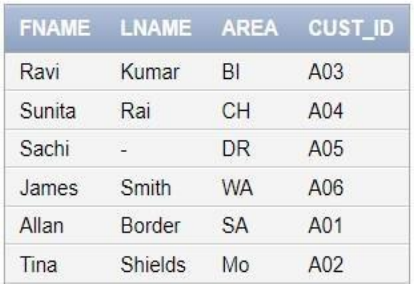


```
HAVING(AVG(Price)>150);

SELECT Fname, Lname
FROM CUST
WHERE cust_id between 'A01' and 'A05';

SELECT Type
FROM Movie
GROUP BY Type;

SELECT Title
FROM Movie
WHERE Price between 150 and 200;

SELECT *
FROM CUST
WHERE ROWNUM<=5;

SELECT *
FROM CUST
WHERE ROWNUM<=5
ORDER BY Fname;
```

TITLE
Carry On Doctor
The Firm
Pretty Women
Home Alone
The Fugitive
Dracula
Quick Change

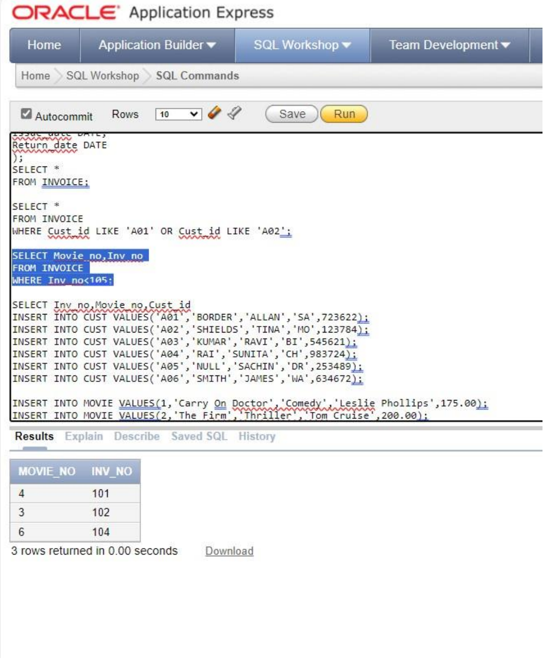
7 rows returned in 0.01 seconds

i. List all details of customers without phone numbers.



FNAME	LNAME	AREA	CUST_ID
Ravi	Kumar	BI	A03
Sunita	Rai	CH	A04
Sachi	-	DR	A05
James	Smith	WA	A06
Allan	Border	SA	A01
Tina	Shields	Mo	A02

j. List the movie_no and inv_no of customers having inv_no less than ‘I05’ from Invoice table.



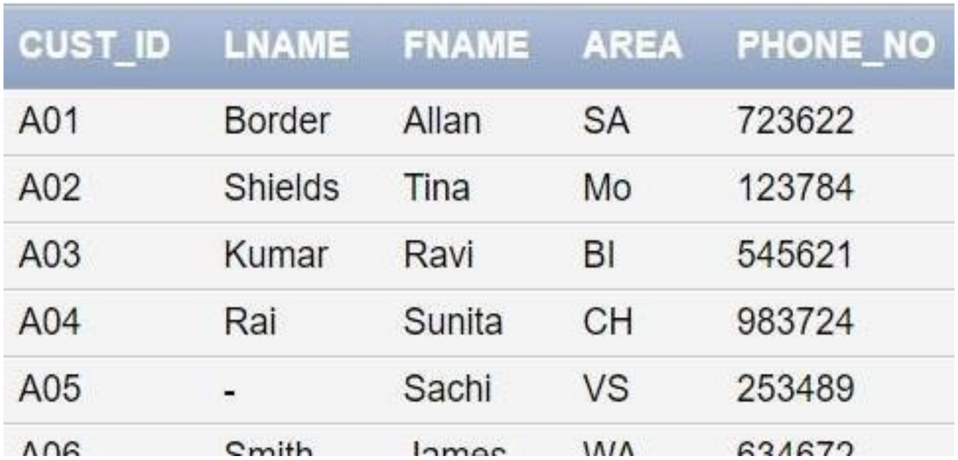
```
Issue_date DATE,
Return_date DATE
);
SELECT *
FROM INVOICE;

SELECT *
FROM INVOICE
WHERE Cust_id LIKE 'A01' OR Cust_id LIKE 'A02';

SELECT Movie_no,Inv_no
FROM INVOICE
WHERE Inv_no<105;

SELECT Inv_no,Movie_no,Cust_id
INSERT INTO CUST VALUES('A01','BORDER','ALLAN','SA',723622);
INSERT INTO CUST VALUES('A02','SHIELDS','TINA','MO',123784);
INSERT INTO CUST VALUES('A03','KUMAR','RAVI','BI',545621);
INSERT INTO CUST VALUES('A04','RAI','SUNITA','CH',983724);
INSERT INTO CUST VALUES('A05','NULL','SACHIN','DR',253489);
INSERT INTO CUST VALUES('A06','SMITH','JAMES','WA',634672);

INSERT INTO MOVIE VALUES(1,'Carry On Doctor','Comedy','Leslie Phollips',175.00);
INSERT INTO MOVIE VALUES(2,'The Firm','Thriller','Tom Cruise',200.00);
```

MOVIE_NO	INV_NO
4	101
3	102
6	104

3 rows returned in 0.00 seconds

k. Change the area of cust_id ‘A05’ to ‘VS’

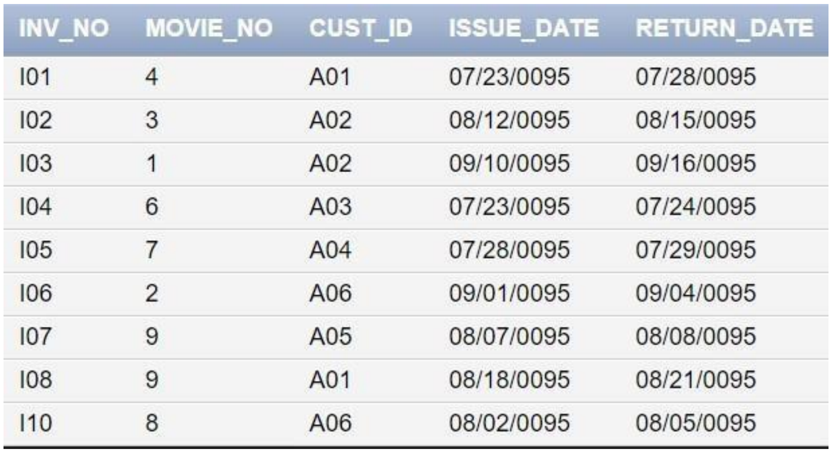
```
UPDATE CUST SET Area='VS' WHERE Cust_id='A05';
SELECT * FROM CUST;
```



CUST_ID	LNAME	FNAME	AREA	PHONE_NO
A01	Border	Allan	SA	723622
A02	Shields	Tina	Mo	123784
A03	Kumar	Ravi	BI	545621
A04	Rai	Sunita	CH	983724
A05	-	Sachi	VS	253489
A06	Smith	James	WA	634672

l. Modify the return date of invoice number ‘I08’ to 21-08-95’.

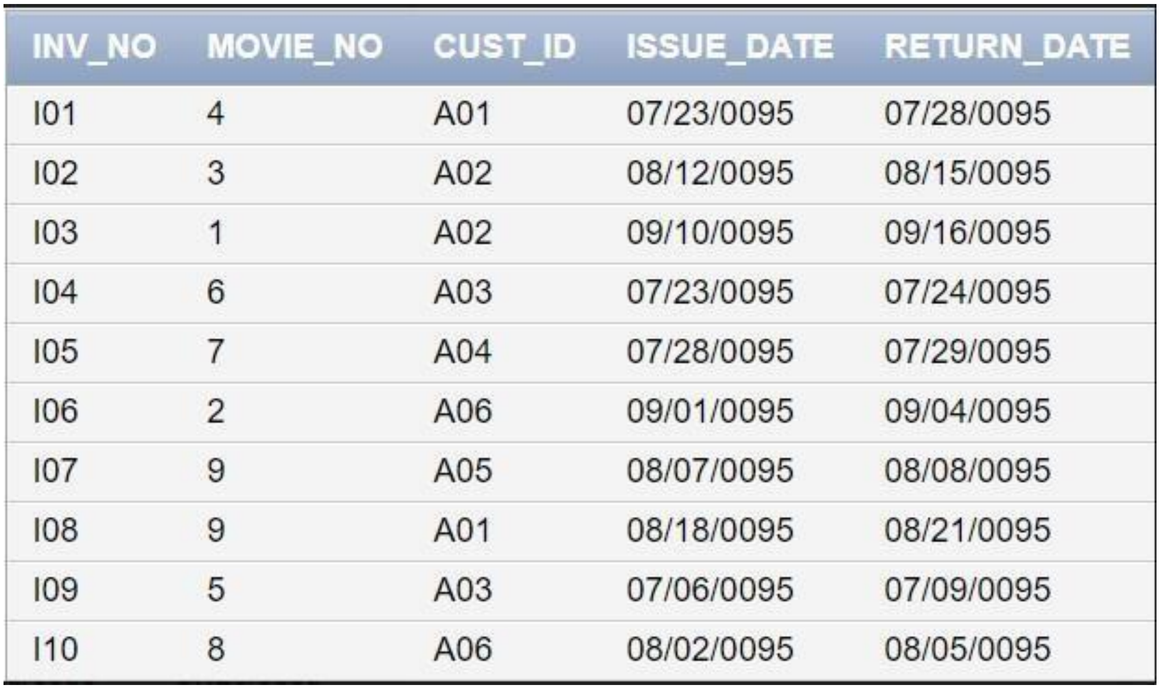
```
UPDATE INVOICE SET Return_date ='08-21-95' WHERE Inv_no='I08'
SELECT * FROM INVOICE
```



INV_NO	MOVIE_NO	CUST_ID	ISSUE_DATE	RETURN_DATE
I01	4	A01	07/23/0095	07/28/0095
I02	3	A02	08/12/0095	08/15/0095
I03	1	A02	09/10/0095	09/16/0095
I04	6	A03	07/23/0095	07/24/0095
I05	7	A04	07/28/0095	07/29/0095
I06	2	A06	09/01/0095	09/04/0095
I07	9	A05	08/07/0095	08/08/0095
I08	9	A01	08/18/0095	08/21/0095
I10	8	A06	08/02/0095	08/05/0095

Delete all the records having return date before 10th July ‘95.

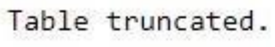
```
DELETE FROM  INVOICE WHERE Return_date <'07-10-95'
```



INV_NO	MOVIE_NO	CUST_ID	ISSUE_DATE	RETURN_DATE
I01	4	A01	07/23/0095	07/28/0095
I02	3	A02	08/12/0095	08/15/0095
I03	1	A02	09/10/0095	09/16/0095
I04	6	A03	07/23/0095	07/24/0095
I05	7	A04	07/28/0095	07/29/0095
I06	2	A06	09/01/0095	09/04/0095
I07	9	A05	08/07/0095	08/08/0095
I08	9	A01	08/18/0095	08/21/0095
I09	5	A03	07/06/0095	07/09/0095
I10	8	A06	08/02/0095	08/05/0095

Truncate the MOVIE table

```
TRUNCATE TABLE MOVIE;
```



```
Table truncated.
```