**SAP ID: 600032200045**

**Roll no. I011**

**CLASS:IT1**

**NAME: ANISHSHARMA**

**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 |

1. **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 |

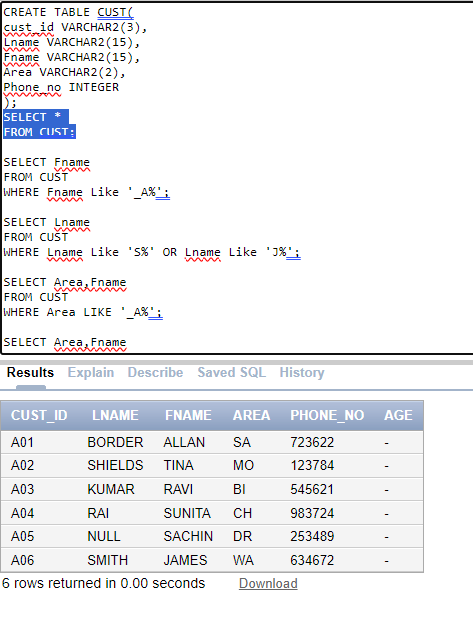
**Data for MOVIE table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Movien o | 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 |

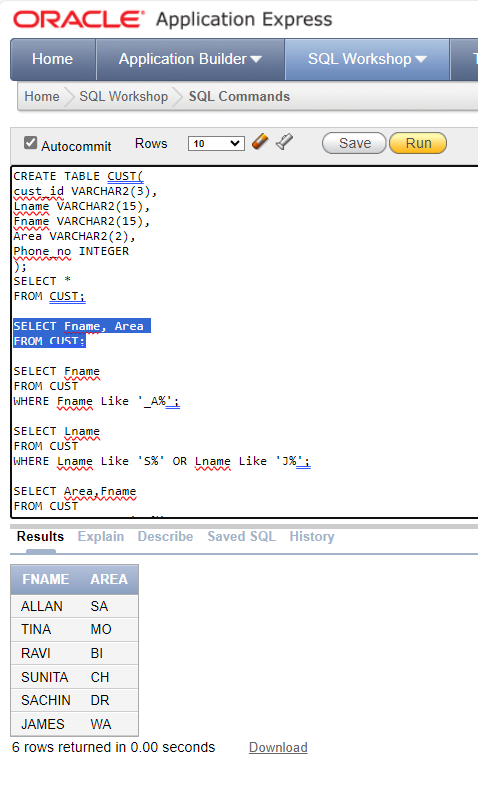
**Data for INVOICE Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |

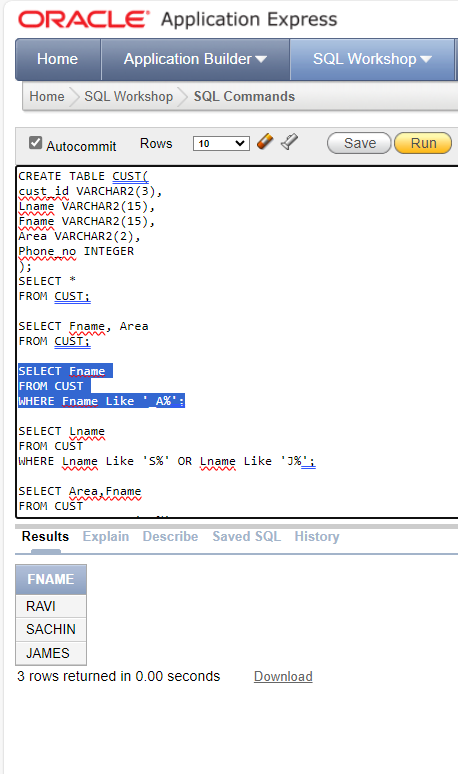
1. **Referring to the above tables and data solve the following queries.** 
   1. Print the entire customer table.



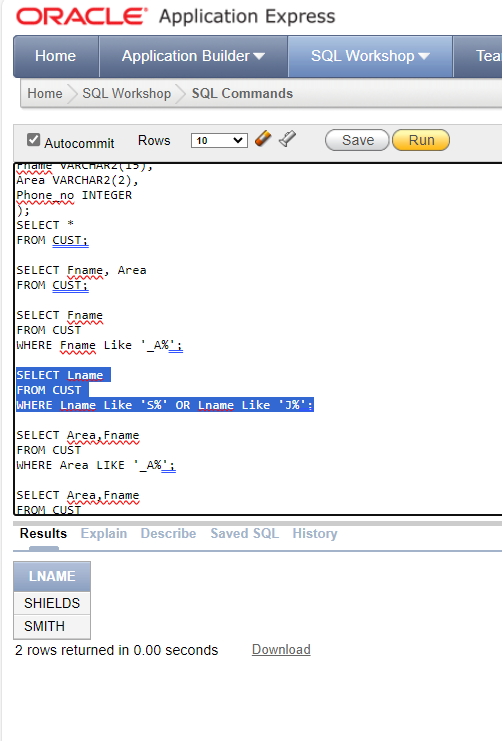
* 1. Retrieve the list of fname and the area of all the customers.



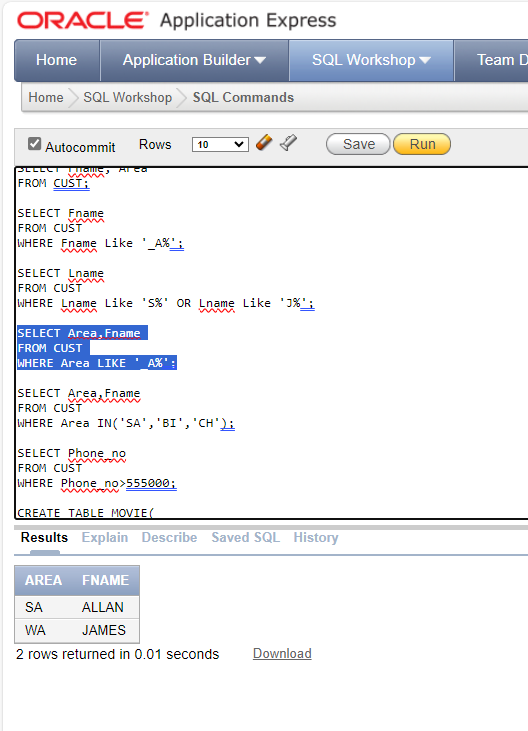
* 1. Find the names of all the customers having ‘a’ as the second letter in fname.



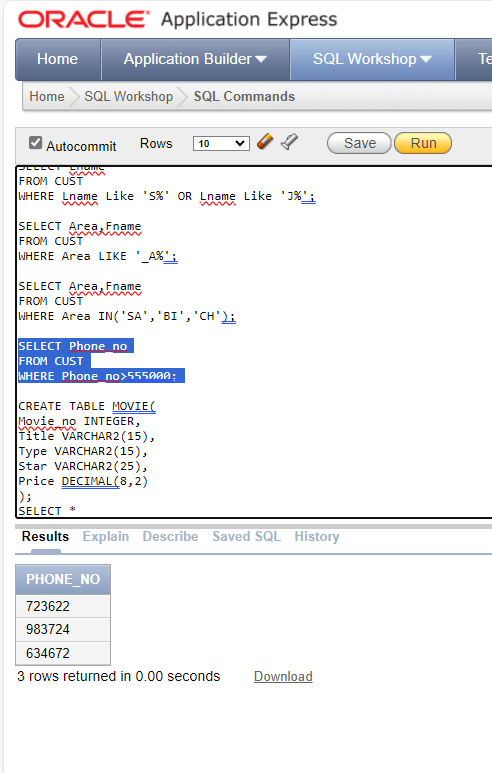
* 1. Find the lname of all customers that begin with ‘S’ or ‘J’.



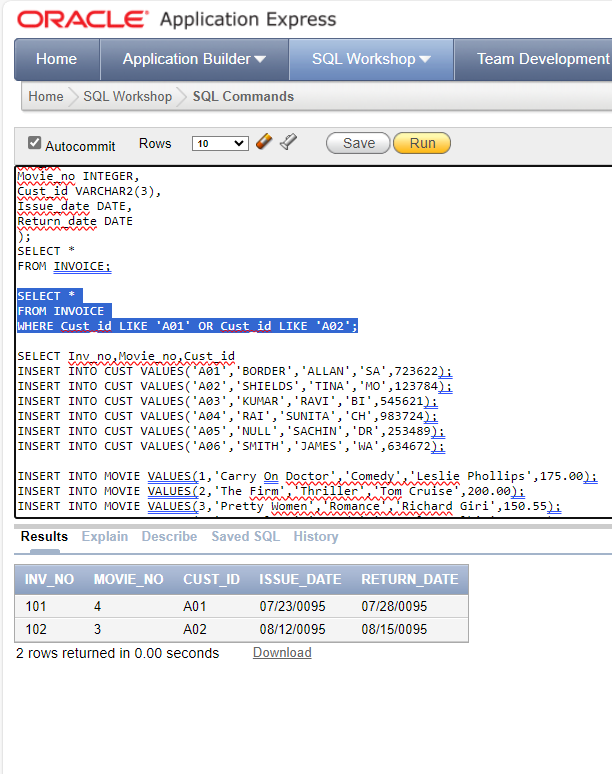
* 1. Find out the customers who stay in an area whose 2nd letter is ‘a’.



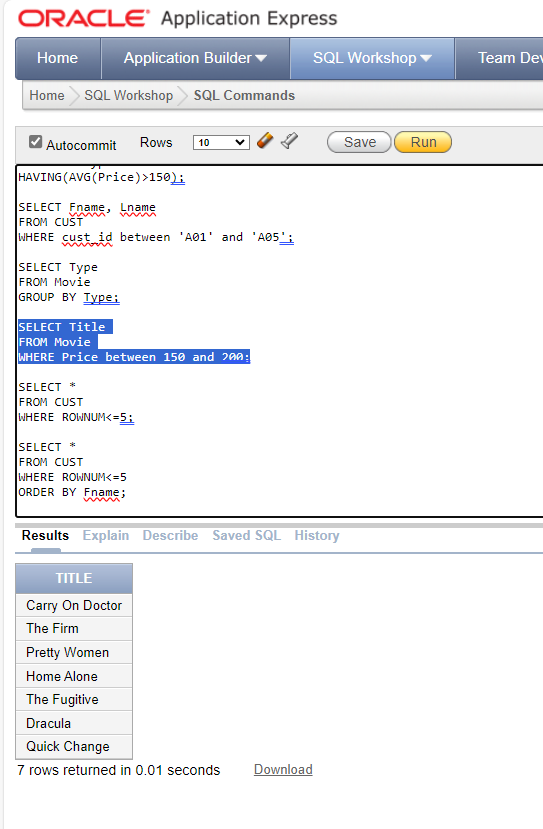
* 1. Print the list of customers whose phone numbers are greater than 555000.



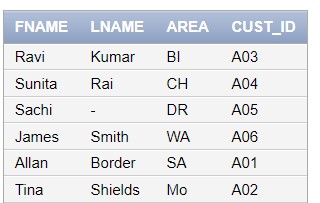
* 1. Display the invoice table information for cust\_id ‘A01’ and ‘A02’.



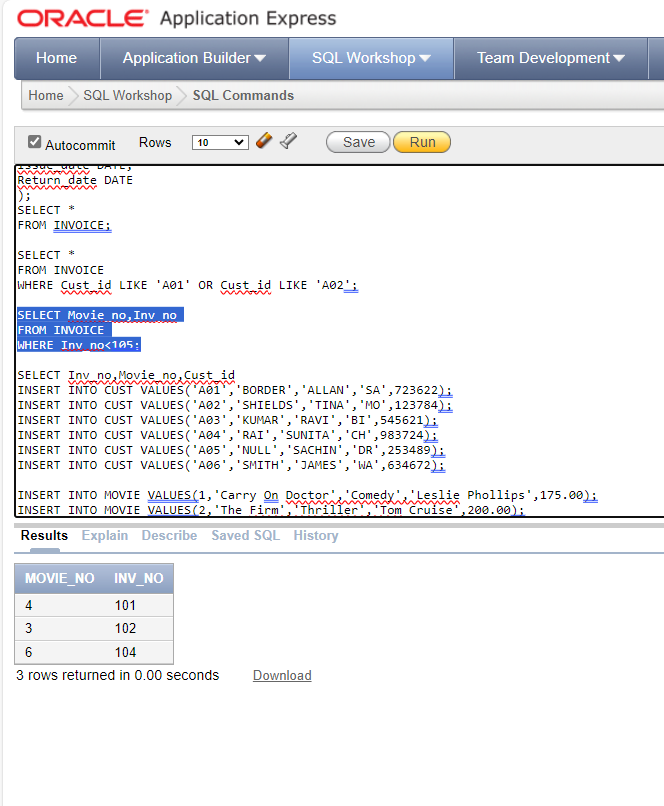
* 1. Find the movies whose price is greater than 150 and less than or equal to 200.



* 1. List all details of customers without phone numbers.



* 1. List the movie\_no and inv\_no of customers having inv\_no less than ‘I05’ from Invoice table.



* 1. Change the area of cust\_id ‘A05’ to ‘VS’

UPDATE CUST SET Area='VS' WHERE Cust\_id='A05';

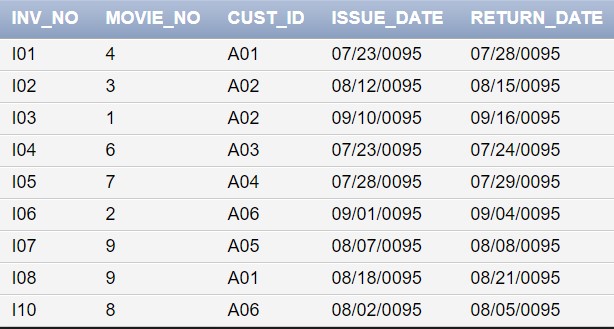
SELECT \* FROM CUST;



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

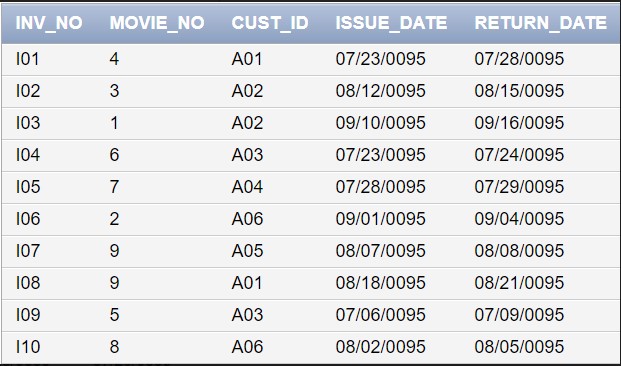
UPDATE INVOICE SET Return\_date ='08-21-95' WHERE Inv\_no='I08'

SELECT \* FROM INVOICE



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

DELETE FROM INVOICE WHERE Return\_date <'07-10-95'



* 1. Truncate the MOVIE table

TRUNCATE TABLE MOVIE;

