FUTURE VISION BIE

One Stop for All Study Materials

& Lab Programs

Future Vision

By K B Hemanth Raj

Scan the QR Code to Visit the Web Page

Or

Visit : https://hemanthrajhemu.github.io

Gain Access to All Study Materials according to VTU, Currently for CSE – Computer Science Engineering...

Join Telegram to get Instant Updates: https://bit.ly/2GKiHnJ

Contact: MAIL: futurevisionbie@gmail.com

INSTAGRAM: www.instagram.com/hemanthraj_hemu/

INSTAGRAM: www.instagram.com/futurevisionbie/

DBMS LABORATORY WITH MINI PROJECT

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2017-2018)

SEMESTER – V

Subject Code:**17CSL58** IA Marks: **40**

Exam Marks: **60** Exam Hours: **03**

Program - 2

Consider the following schema for Order Database:

SALESMAN(Salesman_id, Name, City, Commission)

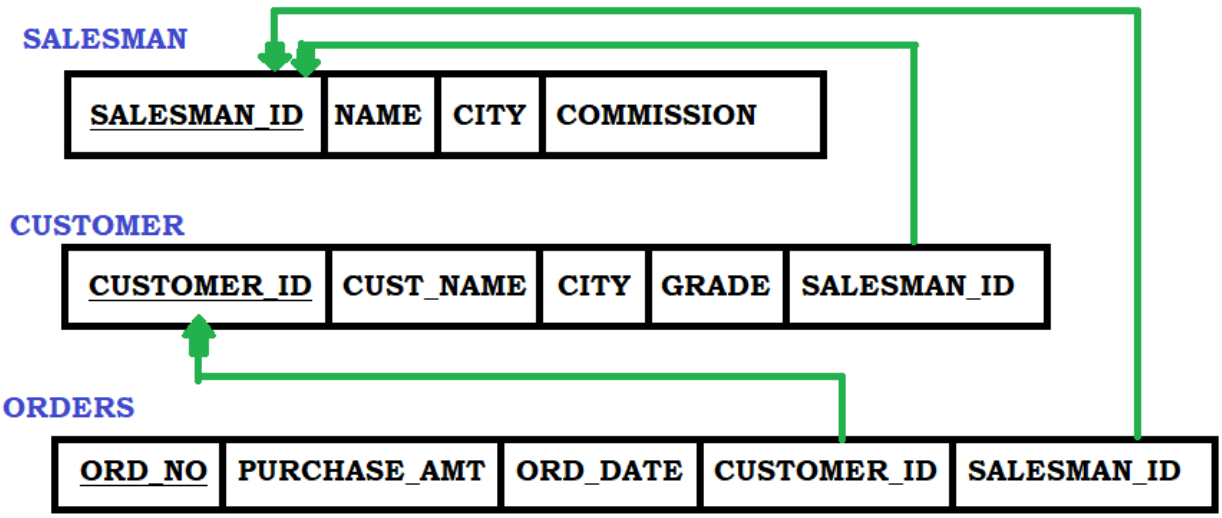
CUSTOMER(Customer_id, Cust_Name, City, Grade, Salesman_id)

ORDERS(Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id)

Write SQL queries to

1. **Count the customers with grades above Bangalore's average.**
2. **Find the name and numbers of all salesman who had more than one customer.**
3. **List all the salesman and indicate those who have and don't have customers in their cities (Use UNION operation.)**
4. **Create a view that finds the salesman who has the customer with the highest order of a day.**
5. **Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted.**

SCHEMA DIAGRAM:



STEPS TO OPEN THE ORACLE DATABASE – 10G EXPRESS EDITION

Step 1: Open the Browser (Preferred Chrome).

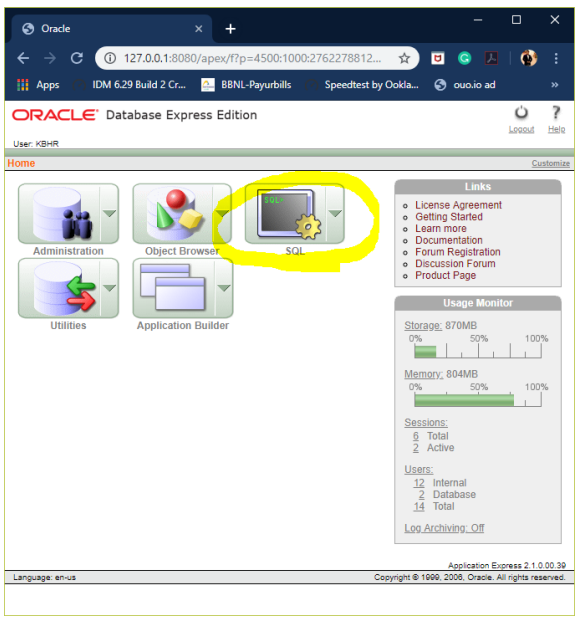
Step 2: http://127.0.0.1:8080/ Enter the link on the browser.

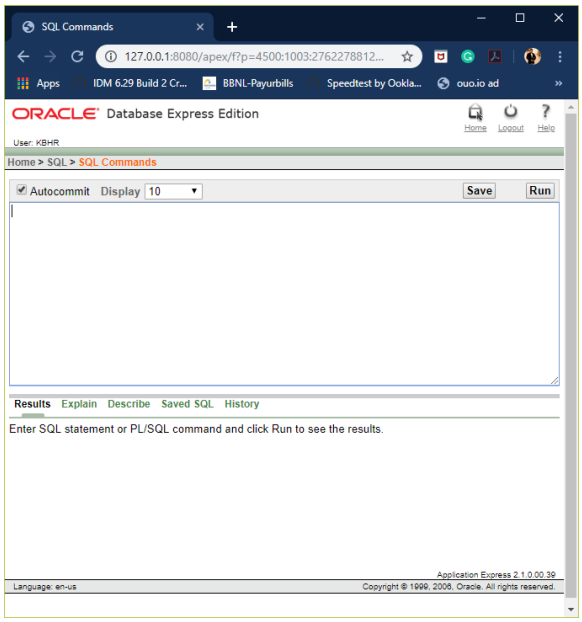
Step 3: login with your id and password (finding difficulty in login in go to the link to know in-depth details

https://hemanthrajhemu.github.io/FutureVisionBIE/WP/5CSE/DBMS_LAB_INFO.html

(Note Username is the system by default & Password is the passkey you entered in the installation)

Step 4: Now click on SQL->SQL Commands. This is the place where we execute the SQL Commands.





Step 5: you are in SQL Command Now you can Create table, create view, Run Queries here & lot more.

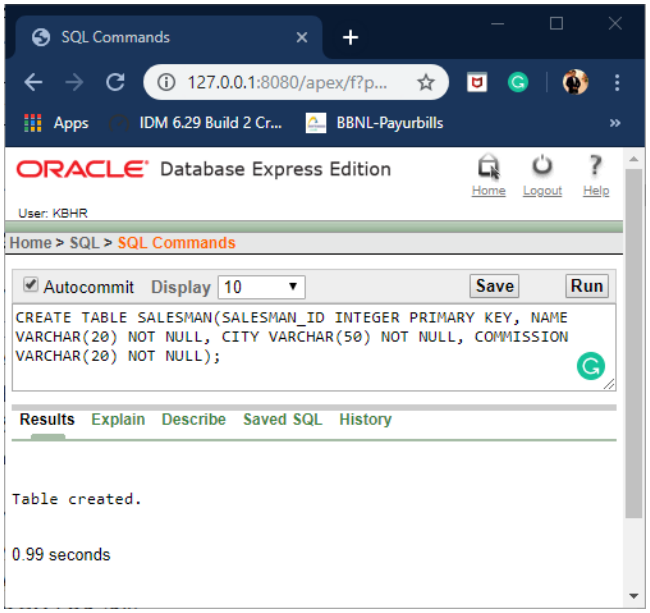
Create Table: (Follow the Schema Diagram in Creating the Data Base)

---

1. **Create Table for SALESMAN**

```
CREATE TABLE SALESMAN
(SALESMAN_ID INTEGER PRIMARY KEY,
NAME VARCHAR(20) NOT NULL,
CITY VARCHAR(50) NOT NULL,
COMMISSION VARCHAR(20) NOT NULL);
```

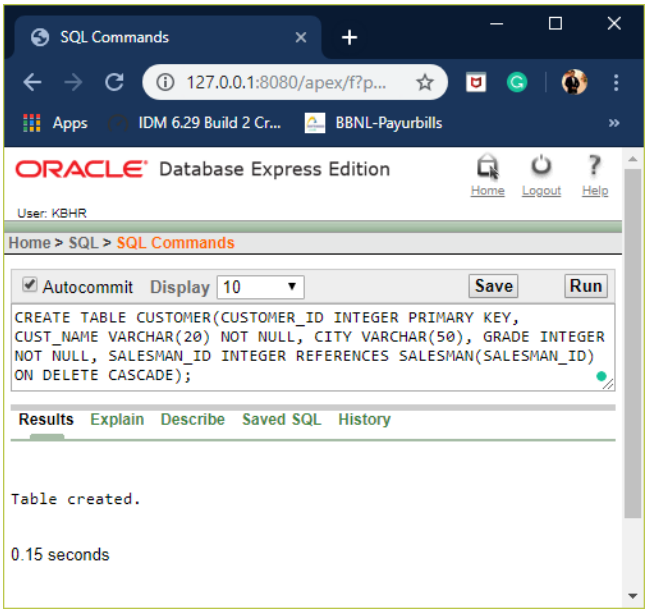
NOW RUN.



2. **Create Table for CUSTOMER**

```
CREATE TABLE CUSTOMER
(CUSTOMER_ID INTEGER PRIMARY KEY,
CUST_NAME VARCHAR(20) NOT NULL,
CITY VARCHAR(50),
GRADE INTEGER NOT NULL,
SALESMAN_ID INTEGER REFERENCES
SALESMAN(SALESMAN_ID)
ON DELETE CASCADE);
```

NOW RUN.



3. **Create Table for ORDERS**

```
CREATE TABLE ORDERS
(ORD_NO INTEGER PRIMARY KEY,
PURCHASED_AMT NUMBER(8,2)
NOT NULL,
ORD_DATE DATE NOT NULL,
CUSTOMER_ID INTEGER
REFERENCES CUSTOMER
(CUSTOMER_ID) ON DELETE CASCADE,
SALESMAN_ID REFERENCES
SALESMAN(SALESMAN_ID)
ON DELETE CASCADE);
```

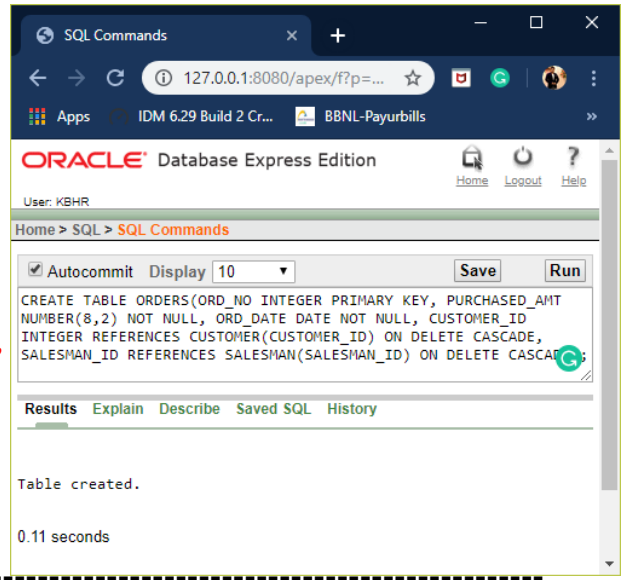
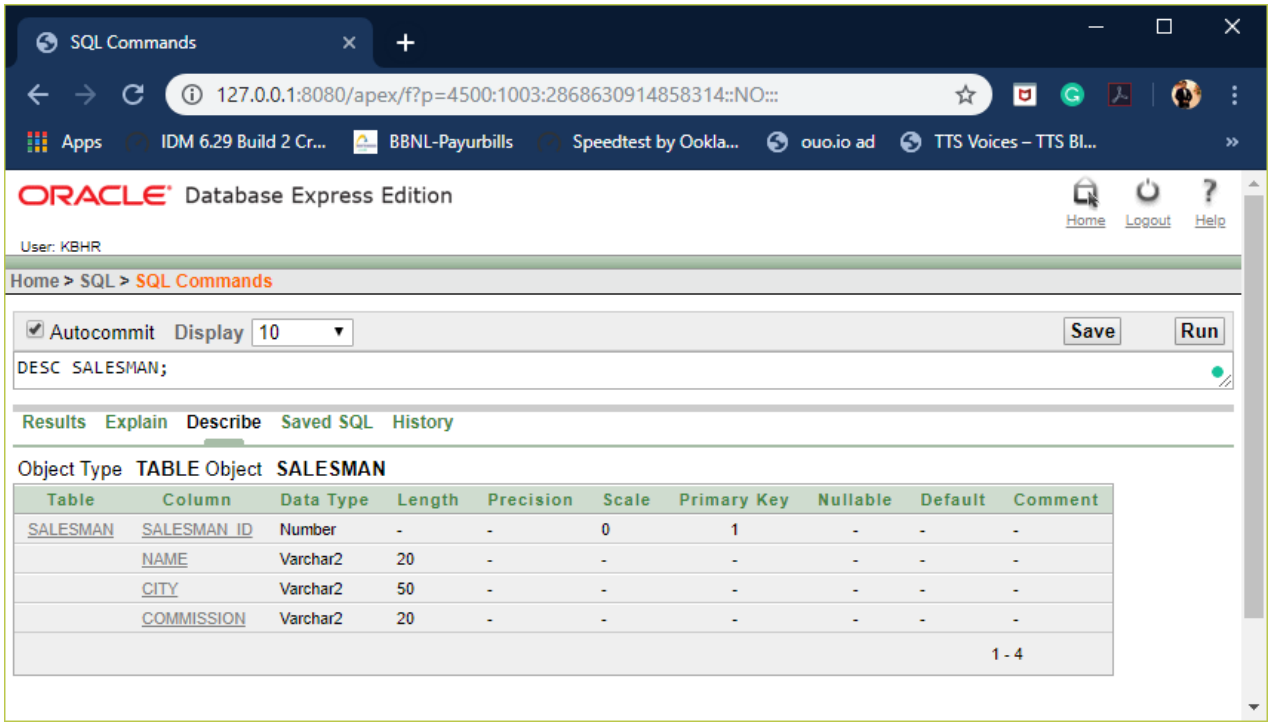


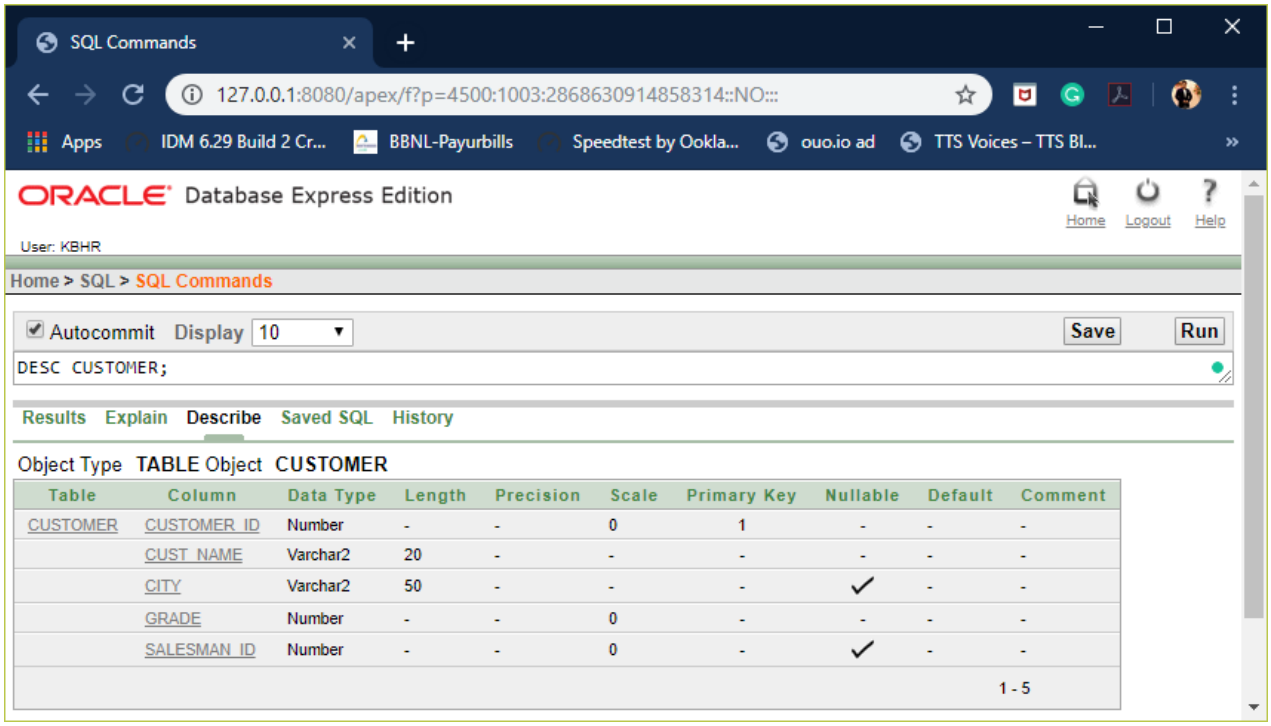
TABLE DESCRIPTION

---

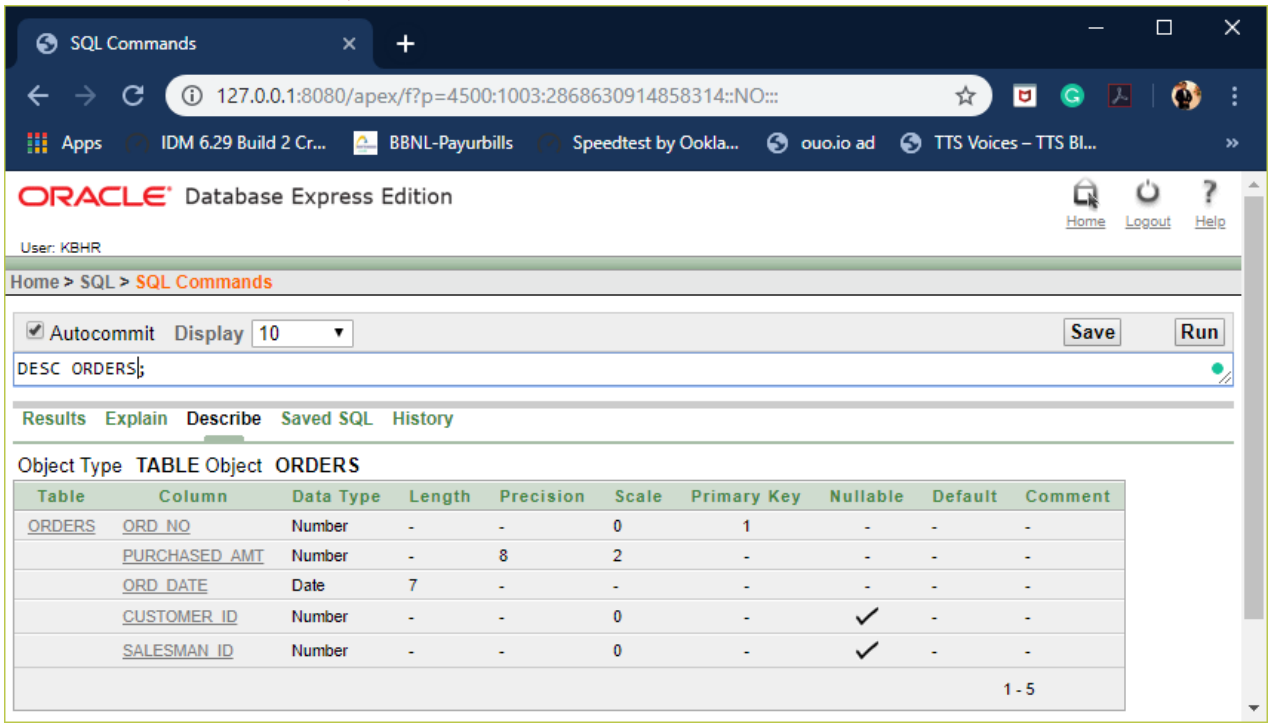
1. DESC SALESMAN;



2. DESC CUSTOMER;



3. DESC ORDERS;



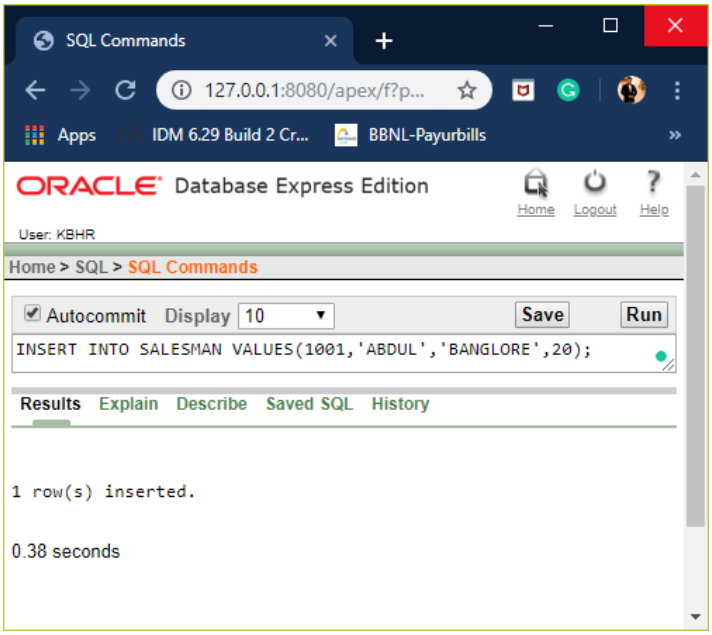
INSERTION OF VALUES TO TABLE

---

1. VALUES INTO SALESMAN;

INSERT INTO SALESMAN
VALUES(<SALESMAN_ID>,<NAME>,<CITY>,<COMMISSION>);

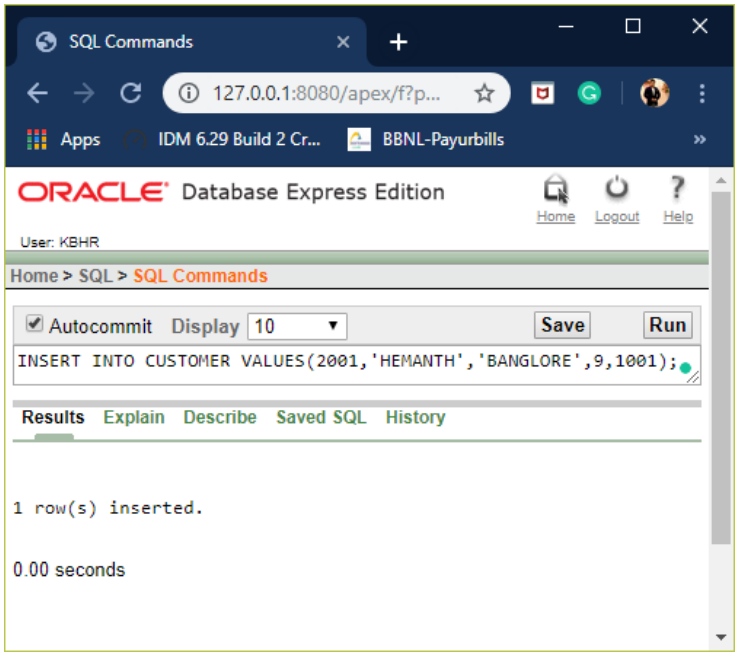
```
INSERT INTO SALESMAN VALUES(1001,'ABDUL','BANGLORE',20);
INSERT INTO SALESMAN VALUES(1002,'PUNITH','BANGLORE',12);
INSERT INTO SALESMAN VALUES(1003,'HARSH','MANGLORE',07);
INSERT INTO SALESMAN VALUES(1004,'HARSHITH','DELHI',26);
INSERT INTO SALESMAN VALUES(1005,'LEELA','BANGLORE',18);
```



2. VALUES INTO CUSTOMER;

INSERT INTO
VALUES(<CUSTOMER_ID>,<CUST_NAME>,<CITY>,<GRADE>,<SALESMAN_ID>);

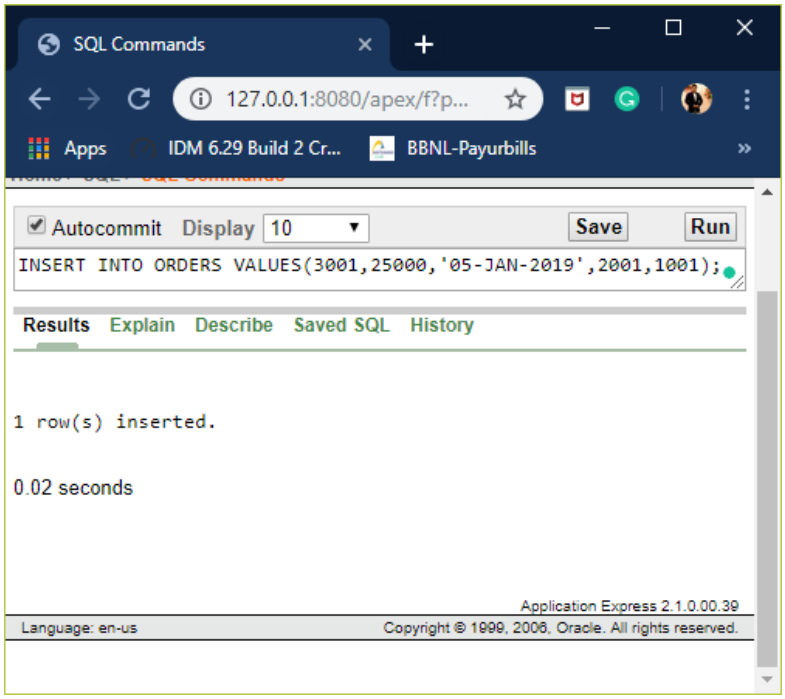
```
INSERT INTO CUSTOMER VALUES(2001,'HEMANTH','BANGLORE',9,1001);
INSERT INTO CUSTOMER VALUES(2002,'RAJ','BANGLORE',7,1001);
INSERT INTO CUSTOMER VALUES(2003,'RAVI','BANGLORE',3,1002);
INSERT INTO CUSTOMER VALUES(2004,'KUMAR','BANGLORE',5,1002);
INSERT INTO CUSTOMER VALUES(2005,'GANESH','MANGLORE',7,1003);
INSERT INTO CUSTOMER VALUES(2006,'VISHNU','MANGLORE',3,1003);
INSERT INTO CUSTOMER VALUES(2007,'SHAH','DELHI',3,1004);
INSERT INTO CUSTOMER VALUES(2008,'KUMAR','DELHI',7,1004);
INSERT INTO CUSTOMER VALUES(2009,'LIRAN','BANGLORE',7,1005);
INSERT INTO CUSTOMER VALUES(2010,'KAVITHA','BANGLORE',8,1005);
```



3. VALUES INTO ORDERS;

INSERT INTO ORDERS VALUES(<ORD_NO>,<PURCHASED_AMT>,<ORD_DATE>,<CUSTOMER_ID>,<SALESMAN_ID>);

INSERT INTO ORDERS VALUES(3001,25000,'05-JAN-2019',2001,1001);
INSERT INTO ORDERS VALUES(3002,5000,'14-FEB-2019',2002,1001);
INSERT INTO ORDERS VALUES(3003,18000,'24-FEB-2019',2003,1002);
INSERT INTO ORDERS VALUES(3004,12000,'26-FEB-2019',2004,1004);
INSERT INTO ORDERS VALUES(3005,7000,'14-MAR-2019',2004,1005);

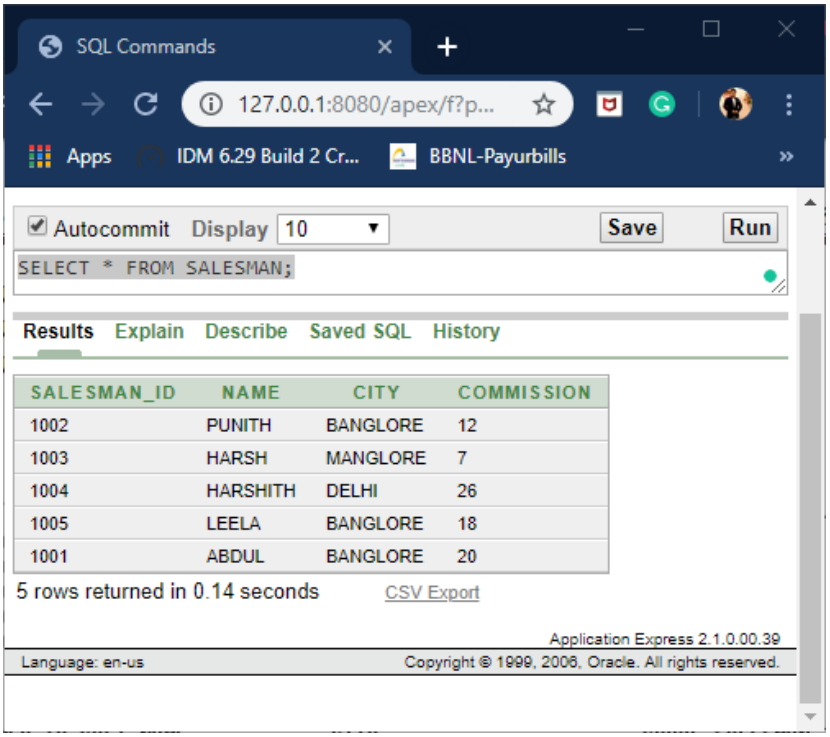


RETRIEVAL OF INSERTED VALUES

---

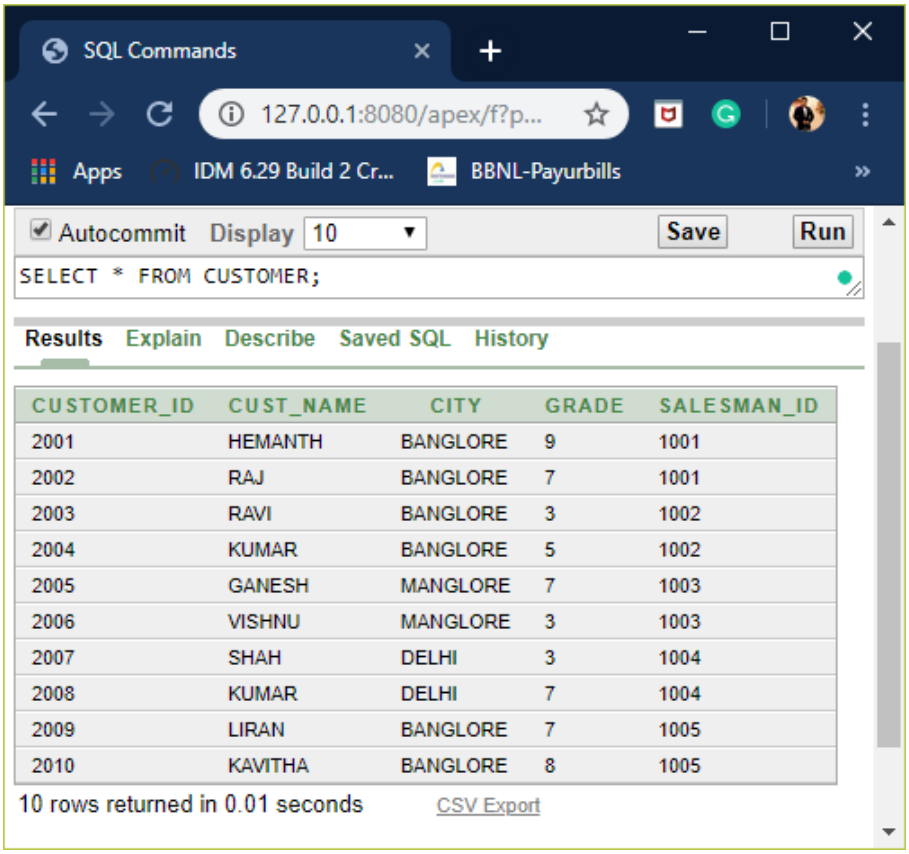
1. **SALESMAN:**

SELECT * FROM SALESMAN;



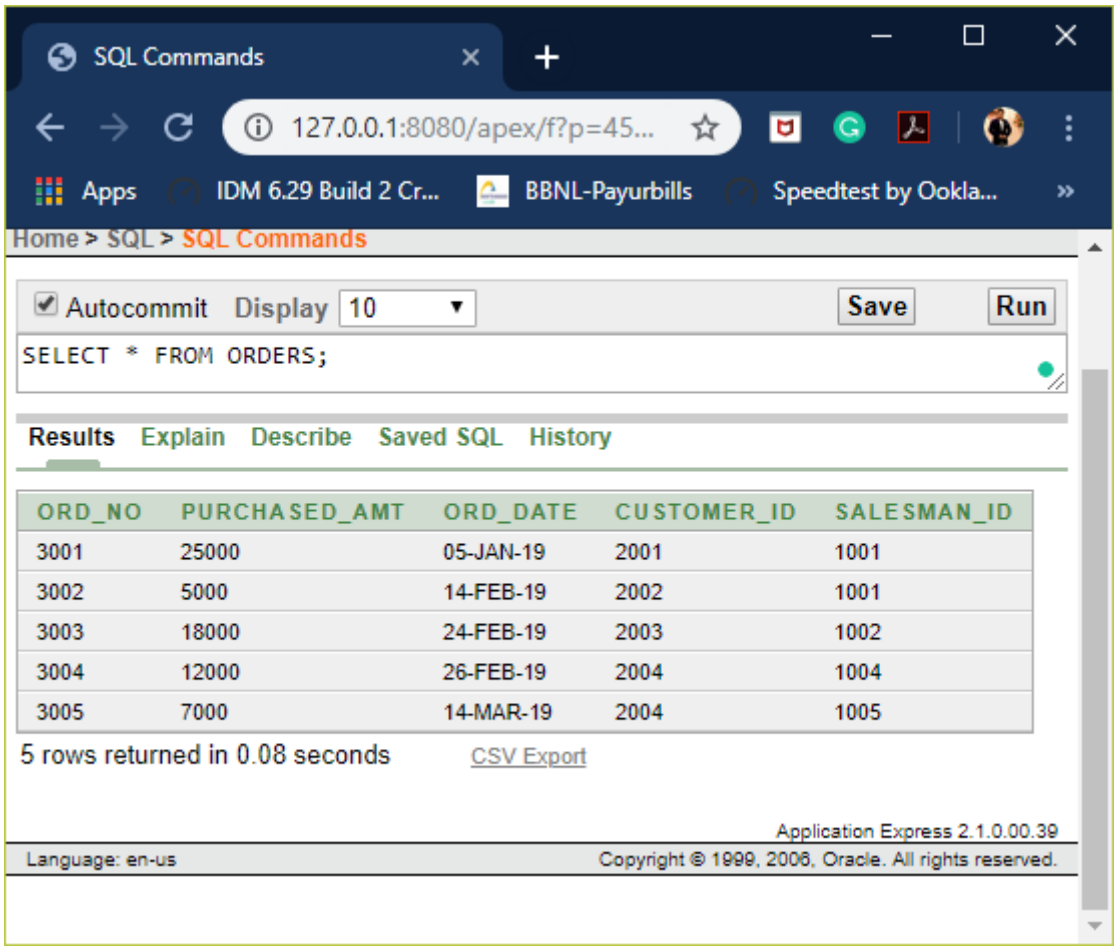
2. **CUSTOMER:**

SELECT * FROM CUSTOMER;



3. ORDERS:

SELECT * FROM ORDERS;

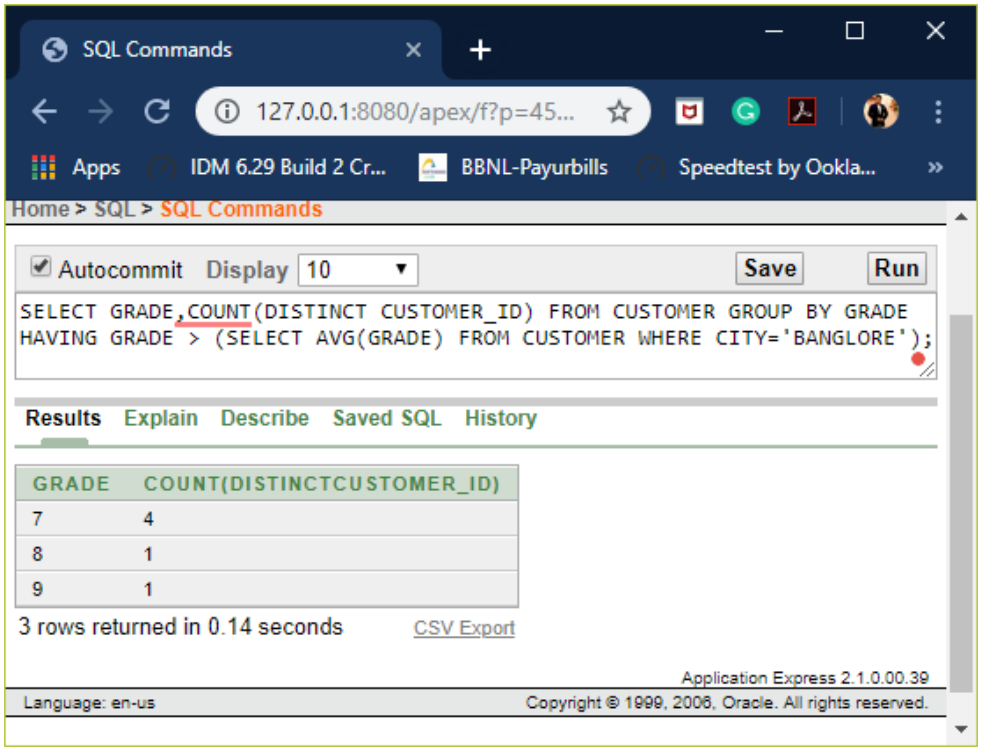


QUERIES

---

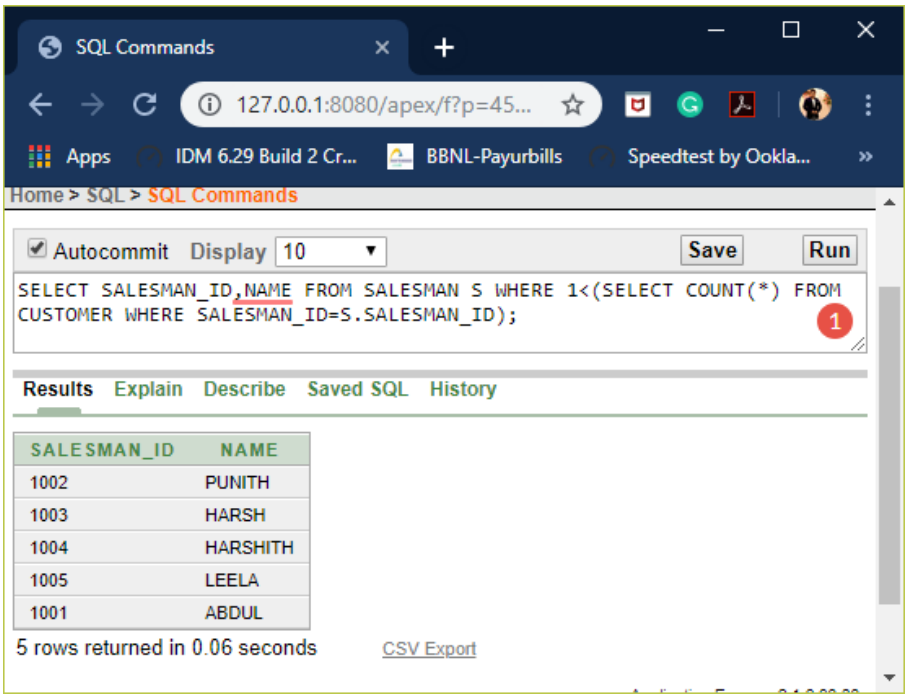
1. **Count the customers with grades above Bangalore's average.**

SELECT GRADE,COUNT(DISTINCT CUSTOMER_ID) FROM CUSTOMER GROUP BY GRADE HAVING GRADE > (SELECT AVG(GRADE) FROM CUSTOMER WHERE CITY='BANGLORE');



2. Find the name and numbers of all salesman who had more than one customer

SELECT SALESMAN_ID, NAME FROM SALESMAN S WHERE 1<(SELECT COUNT(*) FROM CUSTOMER WHERE SALESMAN_ID=S.SALESMAN_ID);

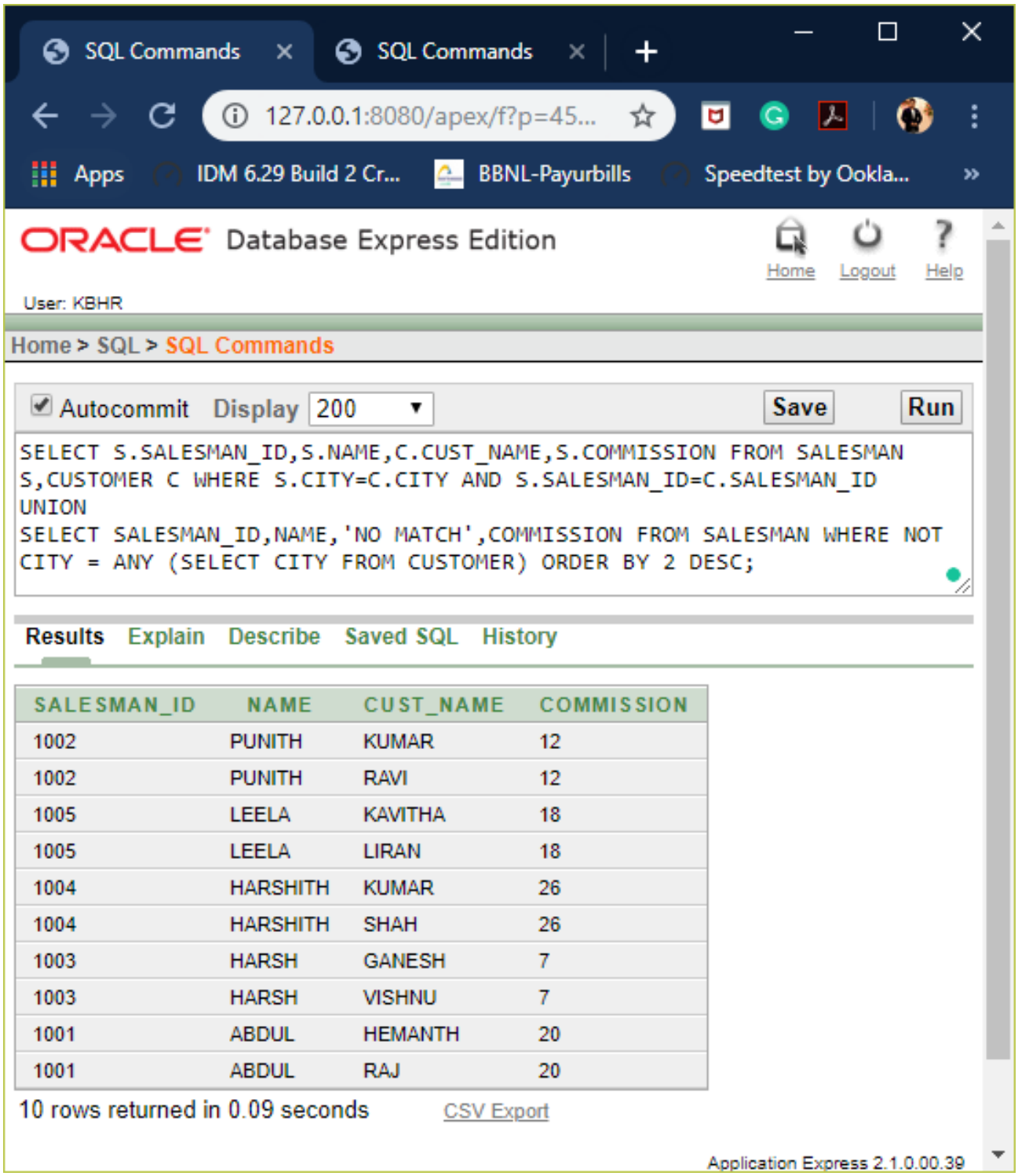


3. List all the salesman and indicate those who have and don't have customers in their cities (Use UNION operation.)

SELECT S.SALESMAN_ID, S.NAME, C.CUST_NAME, S.COMMISSION FROM SALESMAN S, CUSTOMER C WHERE S.CITY=C.CITY AND S.SALESMAN_ID=C.SALESMAN_ID

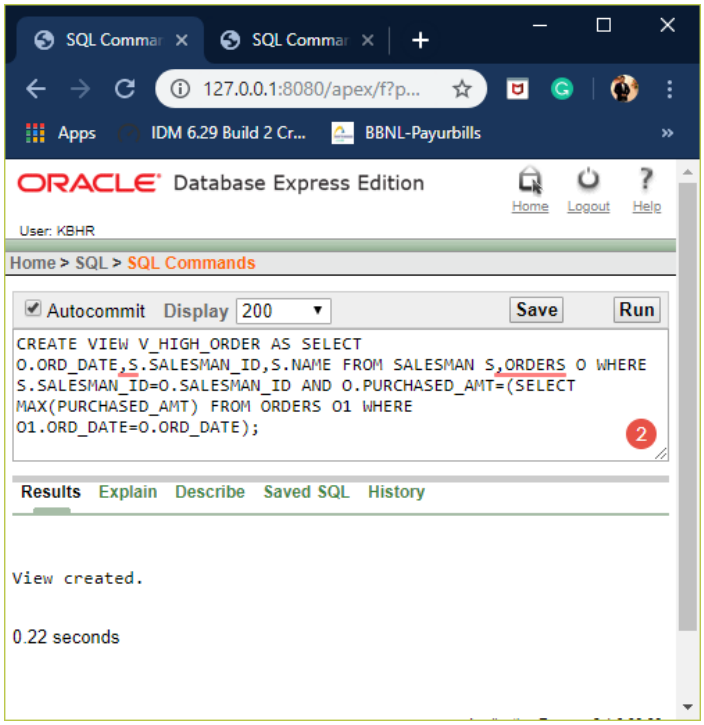
UNION

SELECT SALESMAN_ID, NAME, 'NO MATCH' ,COMMISSION FROM SALESMAN WHERE NOT CITY = ANY (SELECT CITY FROM CUSTOMER) ORDER BY 2 DESC;

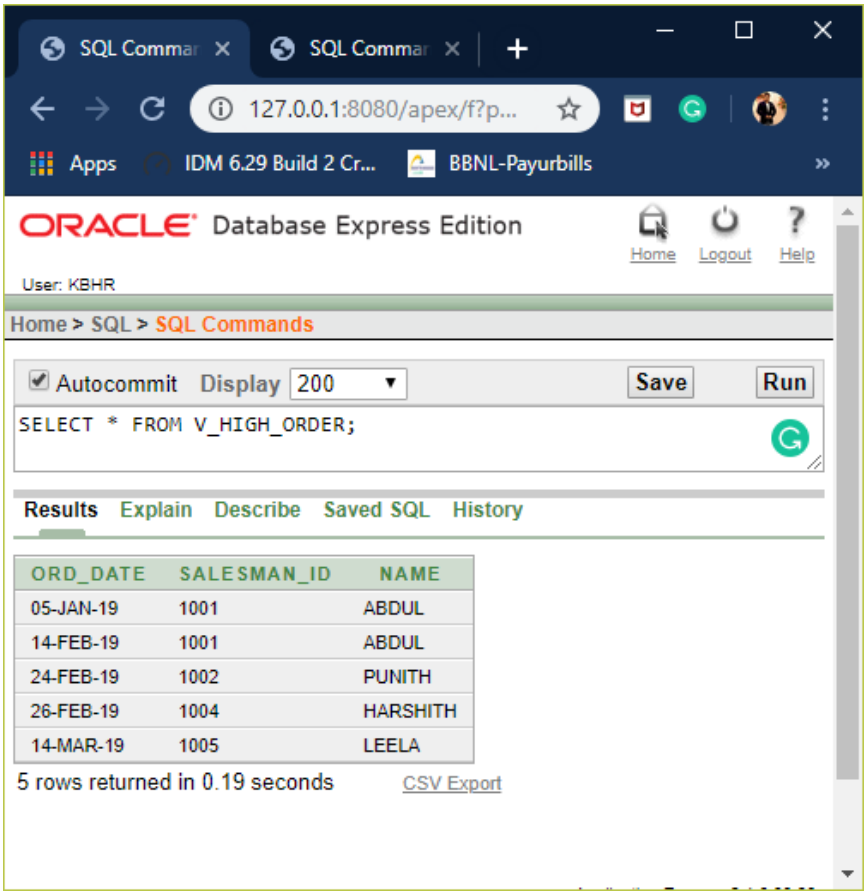


4. Create a view that finds the salesman who has the customer with the highest order of a day.

```
CREATE VIEW V_HIGH_ORDER AS SELECT O.ORD_DATE, S.SALESMAN_ID,
S.NAME FROM SALESMAN S, ORDERS O WHERE
S.SALESMAN_ID=O.SALESMAN_ID AND O.PURCHASED_AMT=(SELECT
MAX(PURCHASED_AMT) FROM ORDERS O1 WHERE
O1.ORD_DATE=O.ORD_DATE);
```

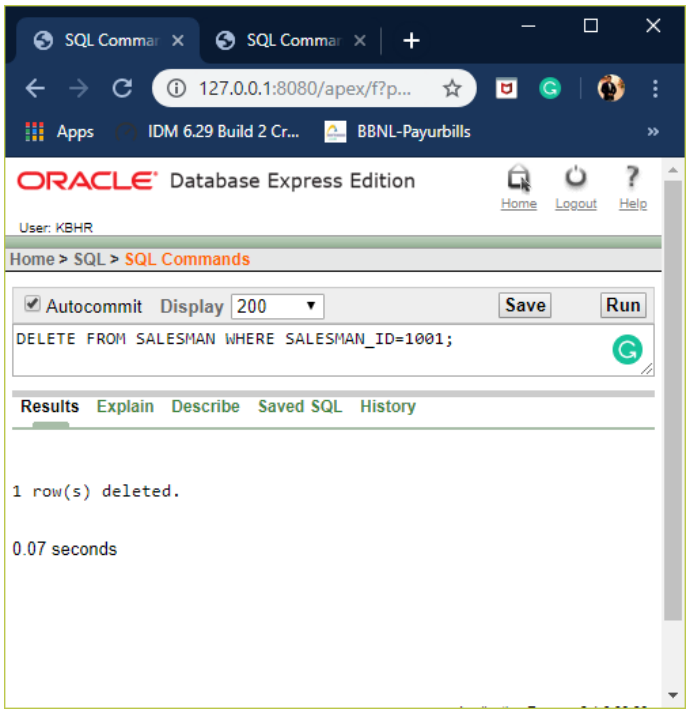



```
SELECT * FROM V_HIGH_ORDER;
```

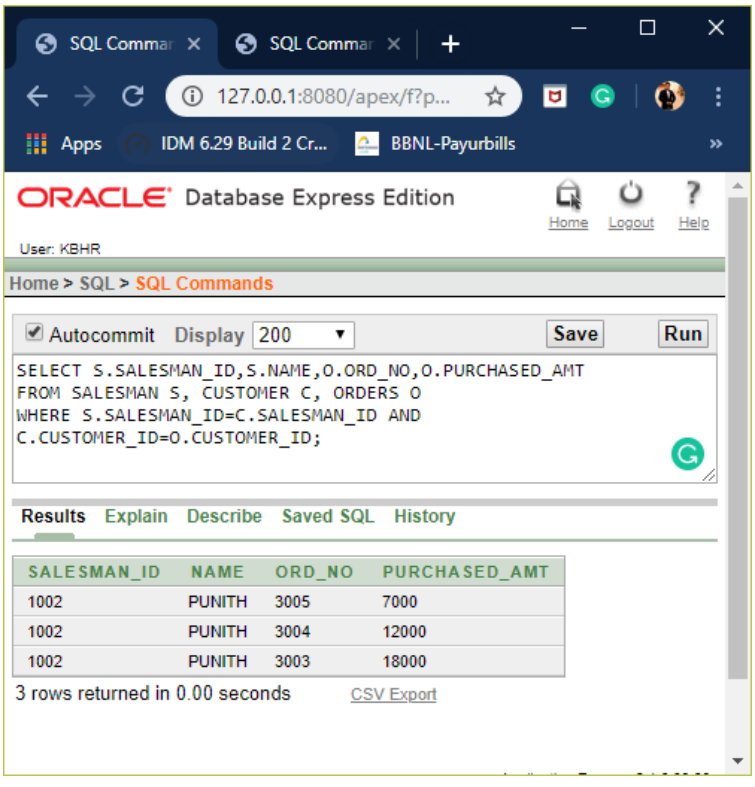


5. Demonstrate the DELETE operation by removing salesman with id 1001. All his orders must also be deleted.

```
DELETE FROM SALESMAN WHERE SALESMAN_ID=1001;
```




```
SELECT S.SALESMAN_ID,S.NAME,O.ORD_NO,O.PURCHASED_AMT
FROM SALESMAN S, CUSTOMER C, ORDERS O
WHERE S.SALESMAN_ID=C.SALESMAN_ID AND
C.CUSTOMER_ID=O.CUSTOMER_ID;
```



THE END
