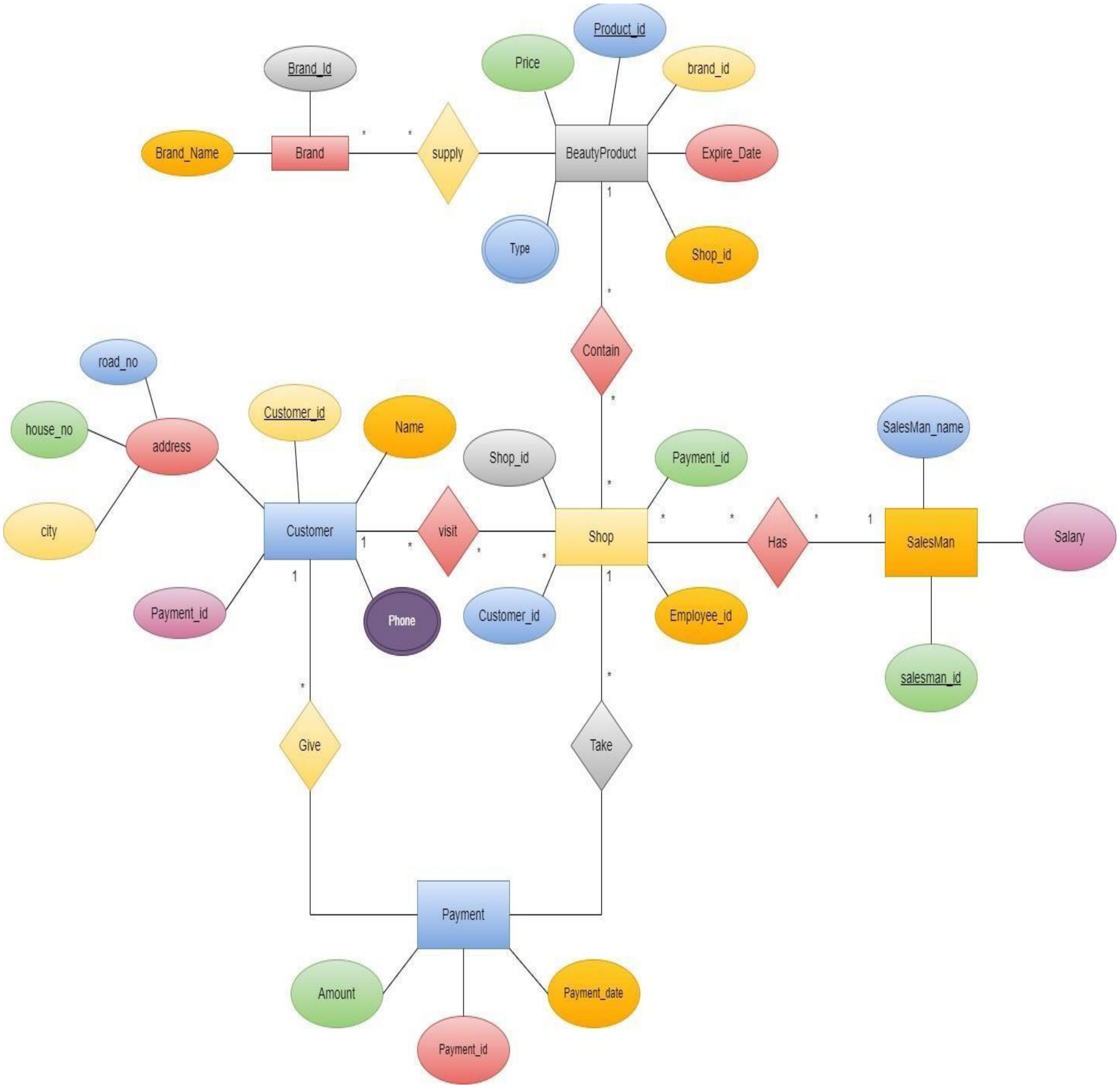
Scenario

Alice is the owner of a beauty product shop. She is looking for a way to better manage her shop. She decides to invest in an online beauty product shop management system. The system allows Alice to create and manage product listings, process payments, track inventory, and communicate with customers. Alice is able to easily upload product images, descriptions, and pricing information. She can also custom product categories, manage promotional discounts, and track customer orders. The system has a customer database. This will store all of the customer information such as name, email, phone number and purchase history. alice will also be able to keep track of any customer loyalty discounts that she offers. Next, Alice will set up an automated email system. This will allow her to send automated emails to customers whenever she has new products, promotions or special offers. It will also help her keep in touch with her customers and let them know when she is running any special events. Alice uses the system to create a customer loyalty program, which offers rewards for frequent customers. She also sets up automated shipping and billing systems, so customers can receive their products quickly and securely. The software can also be used to generate customer loyalty programs and promotions, as well as track customer details and feedback. In addition, the software can help streamline the purchase process, allowing customers to make purchases online, view product details, and track order status. It can also help with managing product returns and exchanges, as well as generate invoices and receipts Alice also takes advantage of the system's analytics features, which provide her with valuable insights into customer behavior. She's able to use this information to optimize her marketing campaigns and better serve her customers. Alice sets up the system and enters her employee information, including their names, contact information, job titles, and hours worked. She is able to set up the system to automatically calculate each employee's wages and send out paychecks on the designated day. The system also allows Alice to easily track her employees' hours and make sure that they are not overworking themselves. She can also set up the system to alert her when an employee reaches a certain number of hours, so she can remind them to take a break or rotate them out. Alice is pleased with the system and finds it extremely helpful in managing her store's staff. She can now focus more on customer service and store operations, knowing that her employees' information and payroll are taken care of.. Alice is now able to manage her online beauty product shop more effectively and efficiently. She is able to provide her customers with a better shopping experience and increase sales.

ER Diagram


Brand_Id
Product_id
Price
brand_id
Brand_Name
Brand
*
*
supply
BeautyProduct
Expire_Date
1
Type
Shop_id
*
Contain
*
road_no
Customer_id
Name
house_no
address
Shop_id
Payment_id
SalesMan_name
*
city
Customer
1
visit
*
*
*
Shop
*
*
Has
*
1
SalesMan
Salary
1
1
Payment_id
Phone
Customer_id
Employee_id
salesman_id
*
*
Give
Take
Payment
Amount
Payment_date
Payment_id

Normalization:

Relation Name: Customer--- 1------ visit ------*----Shop

UNF: customer_id , name, phone, payment_id, city, house_no, road_no, shop_id, employee_id.

1NF: phone is a multivalued attribute here.

1st : phone, customer_id , shop_id , name, payment_id, city, house_no, road_no, employee_id.

2NF: name, phone, payment_id, city, house_no, road_no are partially depended because they depend on customer_id but not shop_id.

1st : customer_id , phone , name, payment_id, city, house_no, road_no.

2nd: shop_id (PK), customer_id (FK), employee_id, payment_id.

3NF: city and road_no depends on house_no which is a non key attribute.

1st : customer_id , phone , name, payment_id.

2nd: shop_id , customer_id(FK) , employee_id, payment_id.

3rd: customer_id , house_no , city, road_no.

Relation Name: Brand ……*……..supply…….*………. Beauty Product

UNF: brand_id , brand_name, product_id , expired_date, type, shop_id, price.

1NF: type is a multivalued attribute here.

1st : type , brand_id , brand_name, product_id , expired_date, shop_id, price.

2NF: brand_name is partially depended because it depends on brand_id but not product_id.

1st : brand_id , brand_name.

2nd: product_id , type , brand_id(FK), expired_date, shop_id, price.

3NF: No transitive dependencies found.

Relation Name: Shop …..*……contain……..*……….Beauty Product

UNF: shop_id , customer_id, employee_id, payment_id, product_id , brand_id, expired_date, type, price.

1NF: type is a multivalued attribute here.

1st : shop_id , type , customer_id, employee_id, payment_id, product_id , brand_id, expired_date, price.

2NF: brand_id, expired_date, type, price are partially depended because they depend on product_id but not shop_id.

1st : product_id , type , brand_id, expired_date, shop_id (FK), price.

2nd: shop_id , customer_id, employee_id, payment_id.

3NF: No transitive dependencies found.

Relation Name: Customer……1……give….*……. Payment

UNF: customer_id , name, phone, city, house_no, road_no, payment_id, payment_date, amount.

1NF: phone is a multivalued attribute here.

1st : phone , customer_id , name, city, house_no, road_no, payment_id , payment_date, amount.

2NF: payment_date and amount are partially depended because they depend on payment_id but not customer_id.

1st : customer_id , phone ,name, city, house_no, road_no, payment_id(FK).

2nd: payment_id , amount, payment_date.

3NF: city and road_no depends on house_no which is a non key attribute.

1st : customer_id , phone , name, payment_id .

2nd: payment_id , amount, payment_date(FK).

3rd: customer_id , house_no , city, road_no.

Relation Name: Shop…….1…..take…..*…… Payment

UNF: shop_id , customer_id, employee_id, payment_id, payment_date, amount.

1NF: no multivalued attribute found.

2NF: payment_date and amount are partially depended because they depend on payment_id but not shop_id.

1st : shop_id, customer_id, employee_id, payment_id(FK) .

2nd: payment_id , amount, payment_date.

3NF: No transitive dependencies found.

Relation Name: Shop……1….. has…..*……. salesman

UNF: salesman_id , salary, salesman_name, shop_id , customer_id, employee_id, payment_id,

1NF: no multivalued attribute found.

1st: salesman_id , salary, salesman_name shop_id , customer_id, employee_id, payment_id.

2NF: salesman_name, salary are partially depended because they depend on salesman_id but not Shop_id.

1st : salesman_id ,shop_id(FK) salary, salesman_name.

2nd: employe_id, shop_id, employee_name, salesman_id

3NF: No transitive dependencies found.

Final Tables:

1st: customer_id (PK), phone (PK), name, payment_id (FK).

2nd: shop_id (PK), customer_id (FK), employee_id (FK), payment_id (FK).

3rd: customer_id (PK), house_no (PK), city, road_no.

4th: brand_id (PK), brand_name.

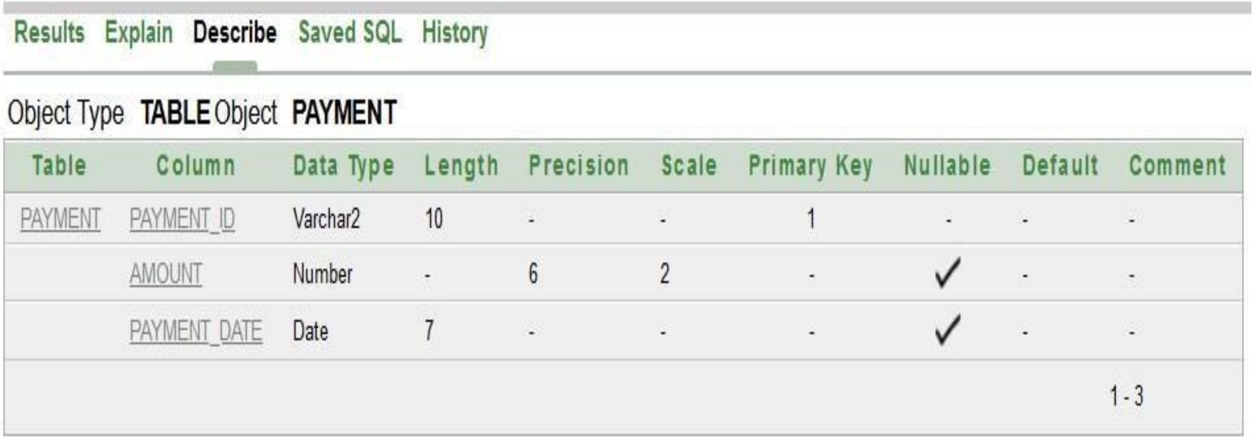
5th: product_id (PK), type (PK), brand_id (FK), expired_date, shop_id (FK), price. 6th: payment_id (PK), amount, payment_date.

7th: salesman_id (PK), salary, salesman_name.

Table Creation:

TABLE: PAYMENT

```
CREATE TABLE PAYMENT
(
   PAYMENT_ID VARCHAR2(10),
  AMOUNT DECIMAL(6,2),
  PAYMENT_DATE DATE,
 PRIMARY KEY(PAYMENT_ID)
);
 DESC PAYMENT;
```



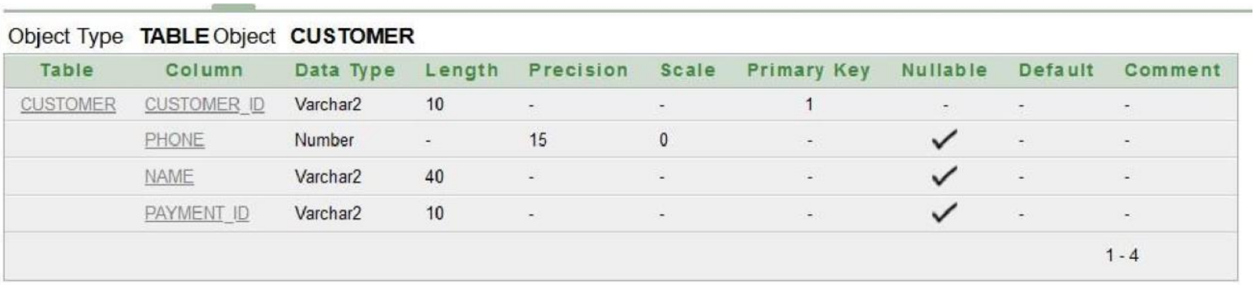
Object Type **TABLE** Object **PAYMENT**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PAYMENT	PAYMENT_ID	Varchar2	10	-	-	1	-	-	-
	AMOUNT	Number	-	6	2	-	✓	-	-
	PAYMENT_DATE	Date	7	-	-	-	✓	-	-

1 - 3

TABLE: CUSTOMER

```
CREATE TABLE CUSTOMER
(
   CUSTOMER_ID VARCHAR (10),
   PHONE NUMBER (15),
   NAME VARCHAR2(40),
   PAYMENT_ID VARCHAR2(10),
   PRIMARY KEY(CUSTOMER_ID)
);
DESC CUSTOMER;
```



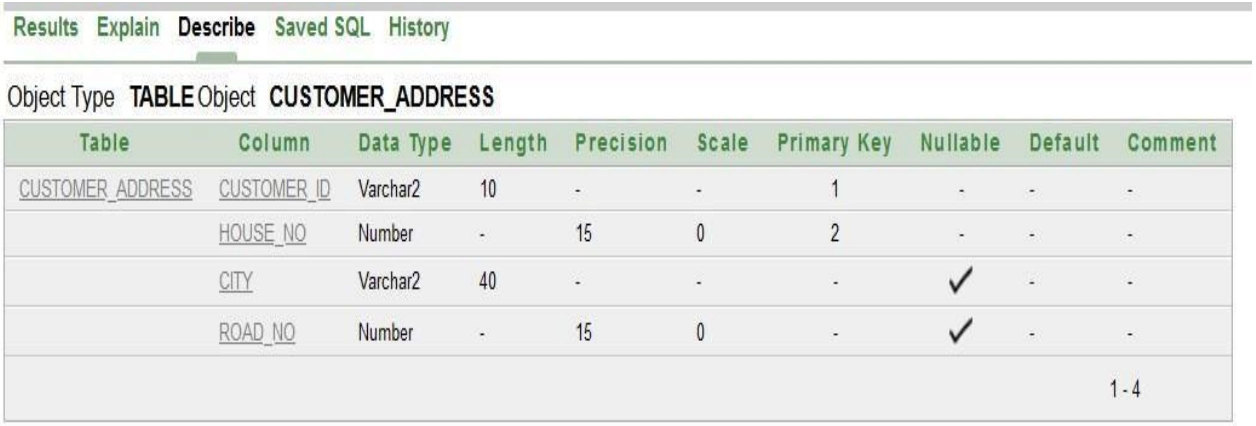
Object Type **TABLE** Object **CUSTOMER**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CUSTOMER	CUSTOMER_ID	Varchar2	10	-	-	1	-	-	-
	PHONE	Number	-	15	0	-	✓	-	-
	NAME	Varchar2	40	-	-	-	✓	-	-
	PAYMENT_ID	Varchar2	10	-	-	-	✓	-	-

1 - 4

TABLE: CUSTOMER_ADDRESS

```
CREATE TABLE CUSTOMER_ADDRESS
(
   CUSTOMER_ID VARCHAR2(10),
   HOUSE_NO NUMBER (15),
   CITY VARCHAR2(40),
   ROAD_NO NUMBER (15),
   PRIMARY KEY (CUSTOMER_ID, HOUSE_NO)
);
DESC CUSTOMER_ADDRESS;
```



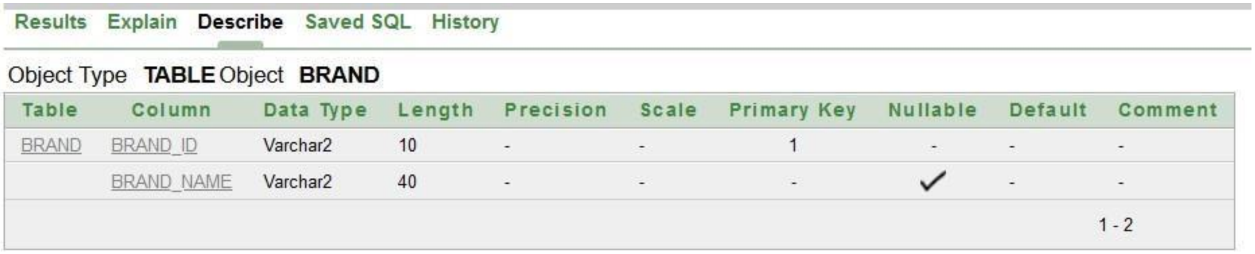
Object Type **TABLE** Object **CUSTOMER_ADDRESS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CUSTOMER_ADDRESS	CUSTOMER_ID	Varchar2	10	-	-	1	-	-	-
	HOUSE_NO	Number	-	15	0	2	-	-	-
	CITY	Varchar2	40	-	-	-	✓	-	-
	ROAD_NO	Number	-	15	0	-	✓	-	-

1 - 4

TABLE: BRAND

```
CREATE TABLE BRAND

(

   BRAND_ID VARCHAR2(10) ,

   BRAND_NAME VARCHAR2(40),

   PRIMARY KEY(BRAND_ID)

);

DESC BRAND;
```



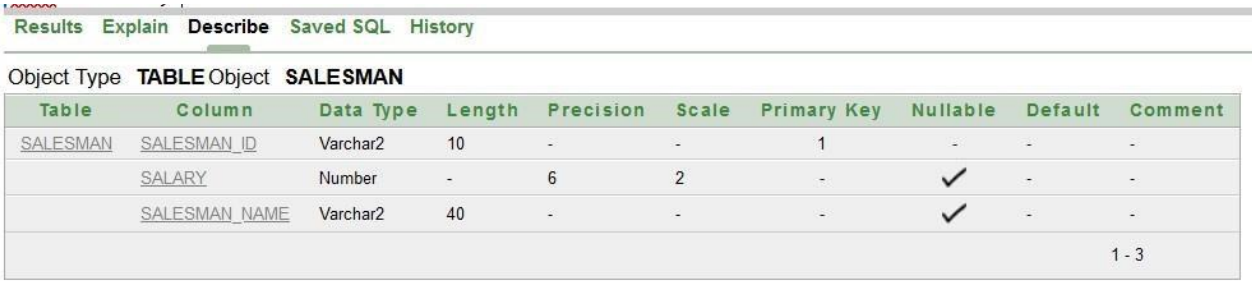
Object Type **TABLE** Object **BRAND**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
BRAND	BRAND_ID	Varchar2	10	-	-	1	-	-	-
	BRAND_NAME	Varchar2	40	-	-	-	✓	-	-

1 - 2

TABLE: SALESMAN

```
CREATE TABLE SALESMAN
(
   SALESMAN_ID VARCHAR2(10),
   SALARY DECIMAL (6,2),
   SALESMAN_NAME VARCHAR2(40),
   PRIMARY KEY(SALESMAN_ID)
);

DESC SALESMAN;
```



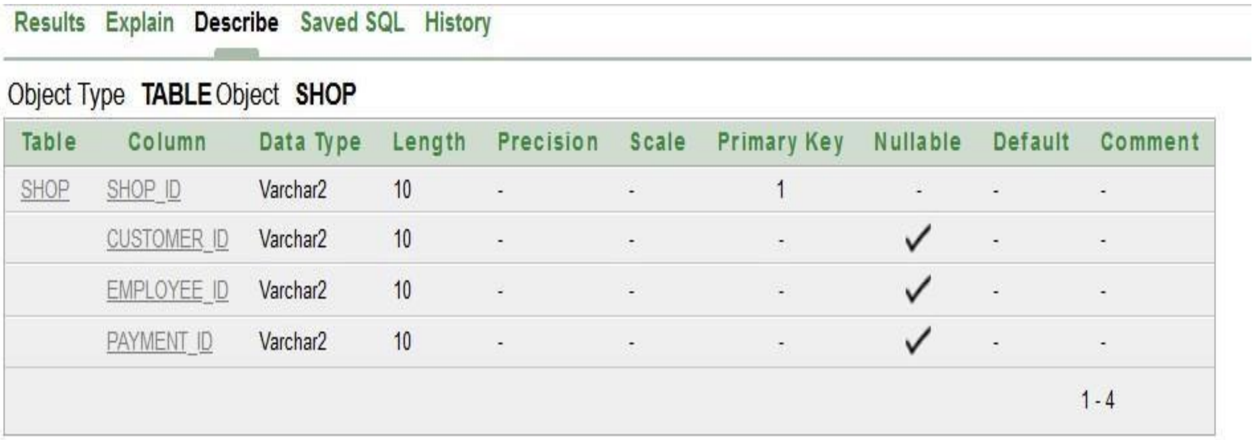
Object Type **TABLE** Object **SALESMAN**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
SALESMAN	SALESMAN_ID	Varchar2	10	-	-	1	-	-	-
	SALARY	Number	-	6	2	-	✓	-	-
	SALESMAN_NAME	Varchar2	40	-	-	-	✓	-	-

1 - 3

TABLE :SHOP

```
CREATE TABLE SHOP
(
   SHOP_ID VARCHAR2(10),
   CUSTOMER_ID VARCHAR2(10),
   EMPLOYEE_ID VARCHAR2(10),
   PAYMENT_ID VARCHAR2(10),
   PRIMARY KEY(SHOP_ID)
   )
DESC SHOP;
```



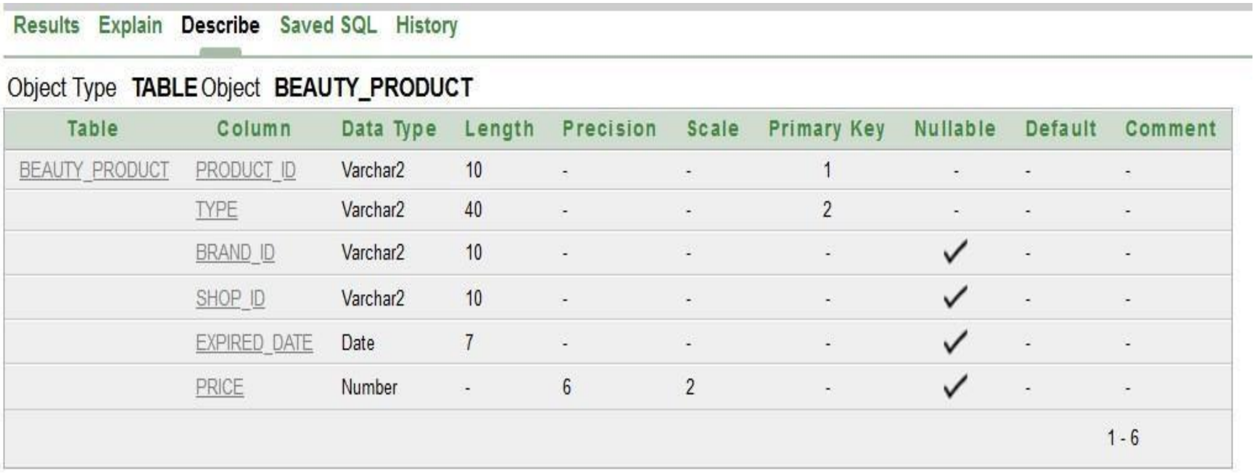
Object Type **TABLE** Object **SHOP**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
SHOP	SHOP_ID	Varchar2	10	-	-	1	-	-	-
	CUSTOMER_ID	Varchar2	10	-	-	-	✓	-	-
	EMPLOYEE_ID	Varchar2	10	-	-	-	✓	-	-
	PAYMENT_ID	Varchar2	10	-	-	-	✓	-	-

1 - 4

TABLE : BEAUTY_PRODUCT

```
CREATE TABLE BEAUTY_PRODUCT
(
   PRODUCT_ID VARCHAR2(10),
   TYPE VARCHAR2(40),
   BRAND_ID VARCHAR2(10),
   SHOP_ID VARCHAR2(10),
   EXPIRED_DATE DATE,
   PRICE DECIMAL (6,2),
   PRIMARY KEY (PRODUCT_ID, TYPE)
);
DESC BEAUTY_PRODUCT;
```



Object Type **TABLE** Object **BEAUTY_PRODUCT**

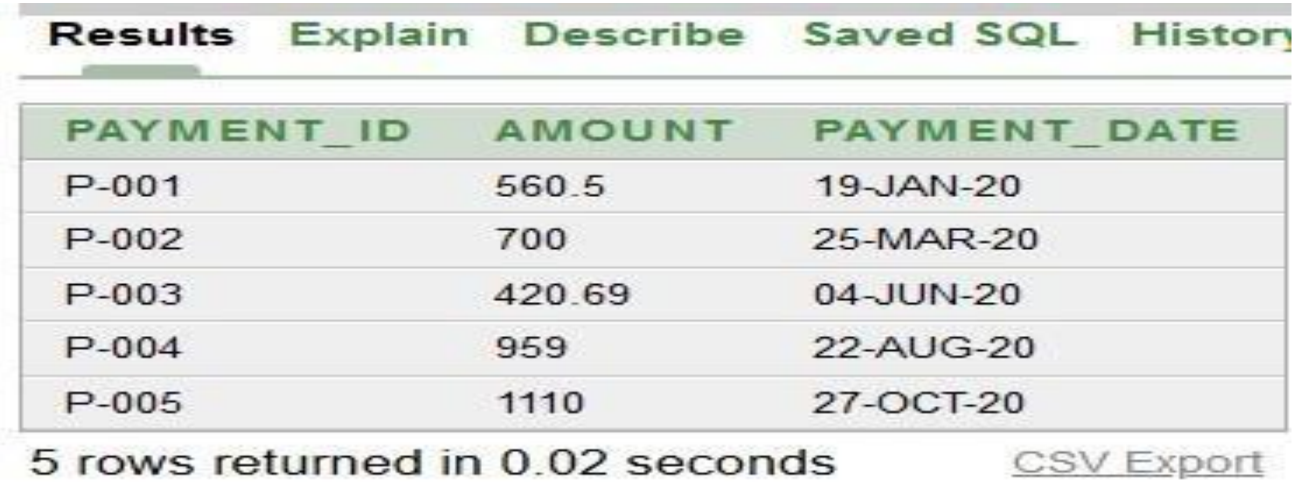
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
BEAUTY_PRODUCT	PRODUCT_ID	Varchar2	10	-	-	1	-	-	-
	TYPE	Varchar2	40	-	-	2	-	-	-
	BRAND_ID	Varchar2	10	-	-	-	✓	-	-
	SHOP_ID	Varchar2	10	-	-	-	✓	-	-
	EXPIRED_DATE	Date	7	-	-	-	✓	-	-
	PRICE	Number	-	6	2	-	✓	-	-

1 - 6

Data Insertion:

DATA INSERTION IN PAYMENT TABLE

```
INSERT INTO PAYMENT VALUES ('P-001', 560.50, TO_DATE ('19-1-2020', 'DD-MM-YYYY'));
INSERT INTO PAYMENT VALUES ('P-002', 700.00, TO_DATE ('25-3-2020', 'DD-MM-YYYY'));
INSERT INTO PAYMENT VALUES ('P-003', 420.69, TO_DATE ('4-6-2020', 'DD-MM-YYYY'));
INSERT INTO PAYMENT VALUES ('P-004', 959.00, TO_DATE ('22-8-2020', 'DD-MM-YYYY'));
INSERT INTO PAYMENT VALUES ('P-005', 1110.00, TO_DATE ('27-10-2020', 'DD-MM-YYYY'));

SELECT * FROM PAYMENT;
```

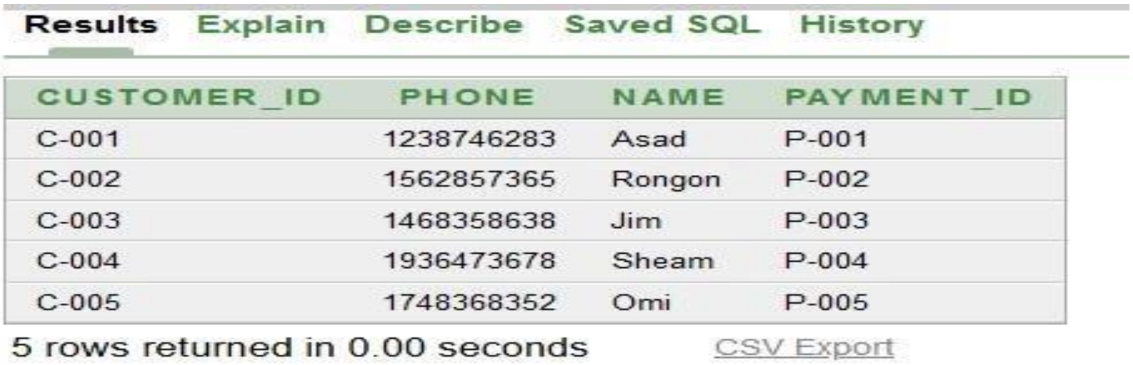


PAYMENT_ID	AMOUNT	PAYMENT_DATE
P-001	560.5	19-JAN-20
P-002	700	25-MAR-20
P-003	420.69	04-JUN-20
P-004	959	22-AUG-20
P-005	1110	27-OCT-20

5 rows returned in 0.02 seconds

DATA INSERTION IN CUSTOMER TABLE

```
INSERT INTO CUSTOMER VALUES ('C-001', 01238746283, 'Asad', 'P-001');
INSERT INTO CUSTOMER VALUES ('C-002', 01562857365, 'Rongon', 'P-002');
INSERT INTO CUSTOMER VALUES ('C-003', 01468358638, 'Jim', 'P-003');
INSERT INTO CUSTOMER VALUES ('C-004', 01936473678, 'Sheam', 'P-004'); INSERT
INTO CUSTOMER VALUES ('C-005', 01748368352, 'Omi', 'P-005');
SELECT * FROM CUSTOMER;
```



CUSTOMER_ID	PHONE	NAME	PAYMENT_ID
C-001	1238746283	Asad	P-001
C-002	1562857365	Rongon	P-002
C-003	1468358638	Jim	P-003
C-004	1936473678	Sheam	P-004
C-005	1748368352	Omi	P-005

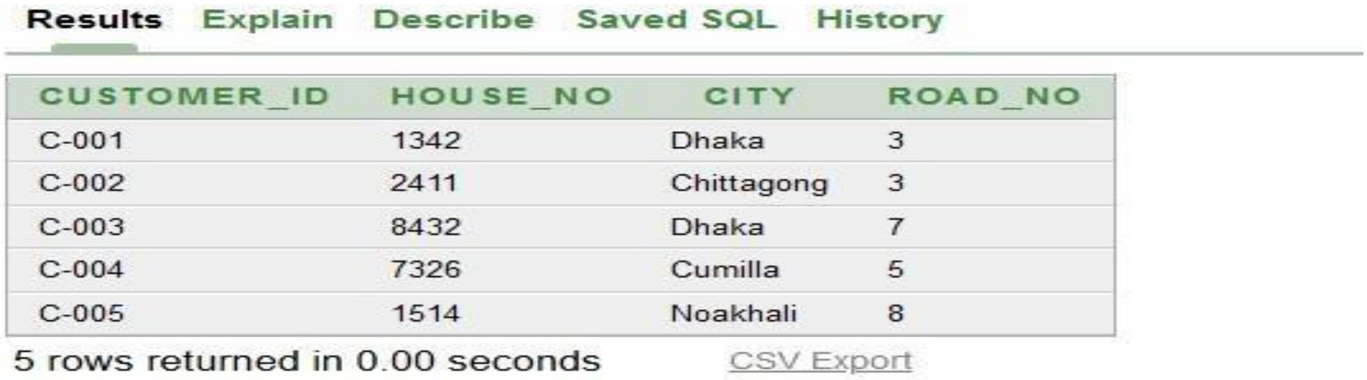
5 rows returned in 0.00 seconds

DATA INSERTION IN CUSTOMER_ADDRESS TABLE

```
INSERT INTO CUSTOMER_ADDRESS VALUES ('C-001', 1342, 'Dhaka', 3);
INSERT INTO CUSTOMER_ADDRESS VALUES ('C-002', 2411, 'Chittagong', 3);
INSERT INTO CUSTOMER_ADDRESS VALUES ('C-003', 8432, 'Dhaka', 7);
INSERT INTO CUSTOMER_ADDRESS VALUES ('C-004', 7326, 'Cumilla', 5);
```

INSERT INTO CUSTOMER_ADDRESS VALUES ('C-005', 1514, 'Noakhali', 8);

SELECT * FROM CUSTOMER_ADDRESS;



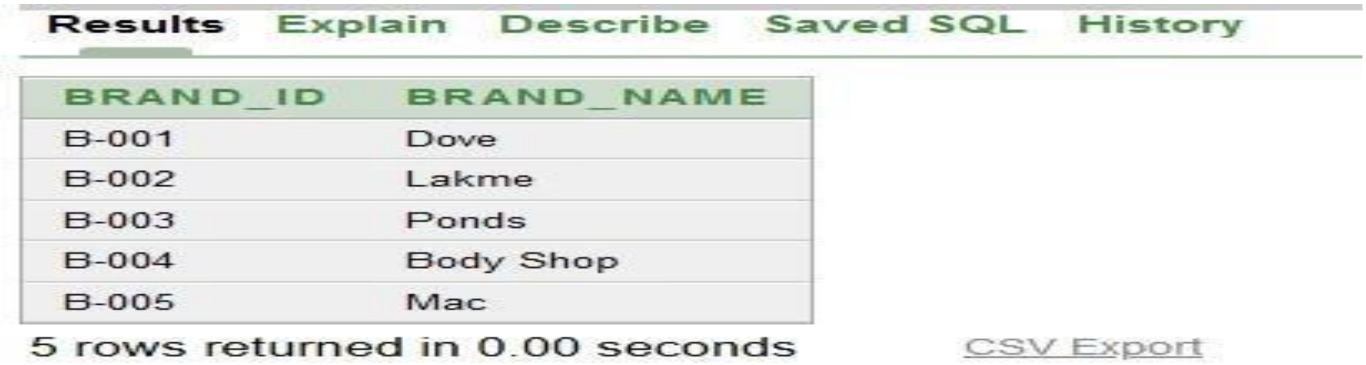
CUSTOMER_ID	HOUSE_NO	CITY	ROAD_NO
C-001	1342	Dhaka	3
C-002	2411	Chittagong	3
C-003	8432	Dhaka	7
C-004	7326	Cumilla	5
C-005	1514	Noakhali	8

```
5 rows returned in 0.00 seconds
```

DATA INSERTION IN BRAND TABLE

INSERT INTO BRAND VALUES ('B-001', 'Dove');
INSERT INTO BRAND VALUES ('B-002', 'Lakme');
INSERT INTO BRAND VALUES ('B-003', 'Ponds');
INSERT INTO BRAND VALUES ('B-004', 'Body Shop');
INSERT INTO BRAND VALUES ('B-005', 'Mac');

SELECT * FROM BRAND;



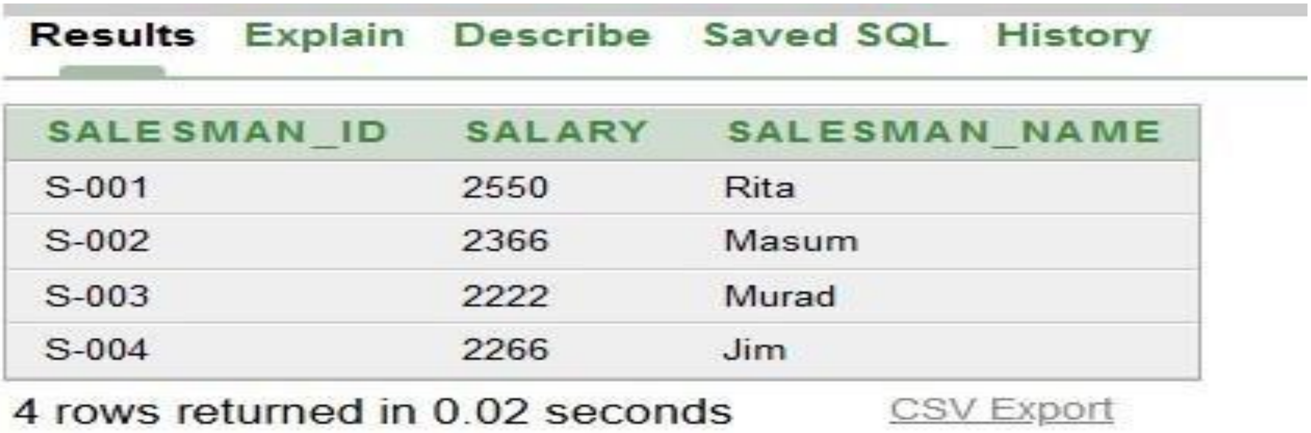
BRAND_ID	BRAND_NAME
B-001	Dove
B-002	Lakme
B-003	Ponds
B-004	Body Shop
B-005	Mac

```
5 rows returned in 0.00 seconds
```

DATA INSERTION IN SALESMAN TABLE

INSERT INTO SALESMAN VALUES ('S-001', 2550, 'Rita');
INSERT INTO SALESMAN VALUES ('S-002', 2366, 'Masum');
INSERT INTO SALESMAN VALUES ('S-003', 2222, 'Murad');
INSERT INTO SALESMAN VALUES ('S-004', 2266, 'Jim');

SELECT * FROM SALESMAN;



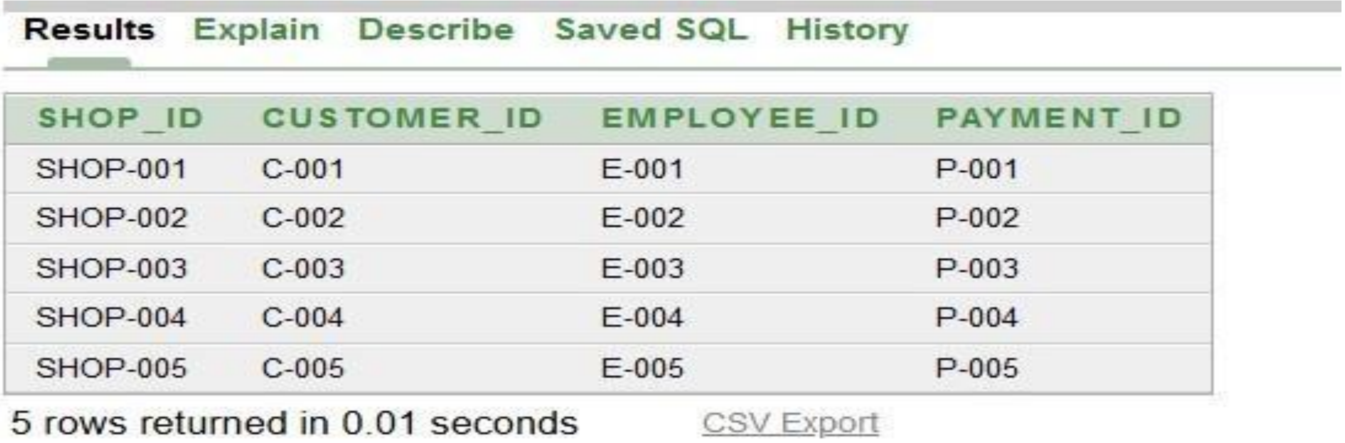
SALESMAN_ID	SALARY	SALESMAN_NAME
S-001	2550	Rita
S-002	2366	Masum
S-003	2222	Murad
S-004	2266	Jim

4 rows returned in 0.02 seconds

DATA INSERTION IN SHOP TABLE

```
INSERT INTO SHOP VALUES ('SHOP-001', 'C-001', 'E-001', 'P-001');
INSERT INTO SHOP VALUES ('SHOP-002', 'C-002', 'E-002', 'P-002');
INSERT INTO SHOP VALUES ('SHOP-003', 'C-003', 'E-003', 'P-003'); INSERT
INTO SHOP VALUES ('SHOP-004', 'C-004', 'E-004', 'P-004');
INSERT INTO SHOP VALUES ('SHOP-005', 'C-005', 'E-005', 'P-005');
```

```
SELECT * FROM SHOP;
```



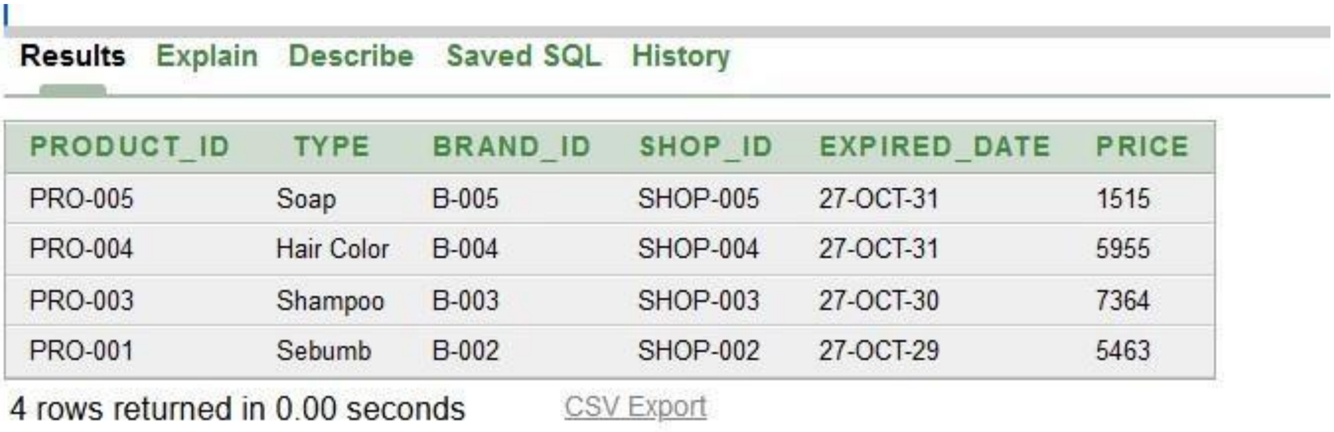
SHOP_ID	CUSTOMER_ID	EMPLOYEE_ID	PAYMENT_ID
SHOP-001	C-001	E-001	P-001
SHOP-002	C-002	E-002	P-002
SHOP-003	C-003	E-003	P-003
SHOP-004	C-004	E-004	P-004
SHOP-005	C-005	E-005	P-005

5 rows returned in 0.01 seconds

DATA INSERTION IN BEAUTY_PRODUCT TABLE

```
INSERT INTO BEAUTY_PRODUCT VALUES ('PRO-005', 'Soap', 'B-005', 'SHOP-005', TO_DATE ('2710-2031',
'DDMM-YYYY'), 1515);
INSERT INTO BEAUTY_PRODUCT VALUES ('PRO-004', 'Hair Color','B-004','SHOP-004', TO_DATE ('27-102031',
'DD-MM-YYYY'), 5955);
INSERT INTO BEAUTY_PRODUCT VALUES ('PRO-003', 'Shampoo', 'B-003', 'SHOP-003', TO_DATE ('27-102030',
'DD-MM-YYYY'), 7364);
INSERT INTO BEAUTY_PRODUCT VALUES ('PRO-001','Sebumb', 'B-002', 'SHOP-002', TO_DATE ('2710-2029',
'DD-MM-YYYY'),5463);
```

```
SELECT * FROM BEAUTY_PRODUCT;
```



PRODUCT_ID	TYPE	BRAND_ID	SHOP_ID	EXPIRED_DATE	PRICE
PRO-005	Soap	B-005	SHOP-005	27-OCT-31	1515
PRO-004	Hair Color	B-004	SHOP-004	27-OCT-31	5955
PRO-003	Shampoo	B-003	SHOP-003	27-OCT-30	7364
PRO-001	Sebumb	B-002	SHOP-002	27-OCT-29	5463

4 rows returned in 0.00 seconds

CONSTRAINT:

1. ALTER TABLE PAYMENT ADD CONSTRAINT PRIMARY KEY(PAYMENT_ID) REFERANCE SHOP (PRODUCT);

2. ALTER TABLE CUSTOMER ADD CONSTRAINT PRIMARY KEY(CUSTOMER_ID) REFERANCE SHOP (PRODUCT_ID);

3. ALTER TABLE CUSTOMER_ADDRESS ADD CONSTRAINT PRIMARY KEY (CUSTOMER_ID, HOUSE_NO) REFERANCE SHOP (PRODUCT_ID);

4. ALTER TABLE BRAND ADD CONSTRAINT PRIMARY KEY(BRAND_ID) REFERANCE BRAND (BRAND_NAME);

5.ALTER TABLE SALESMAN ADD CONSTRAINT PRIMARY KEY(SALESMAN_ID) REFERANCE CUSTOMER (CUSTOMER_ID);

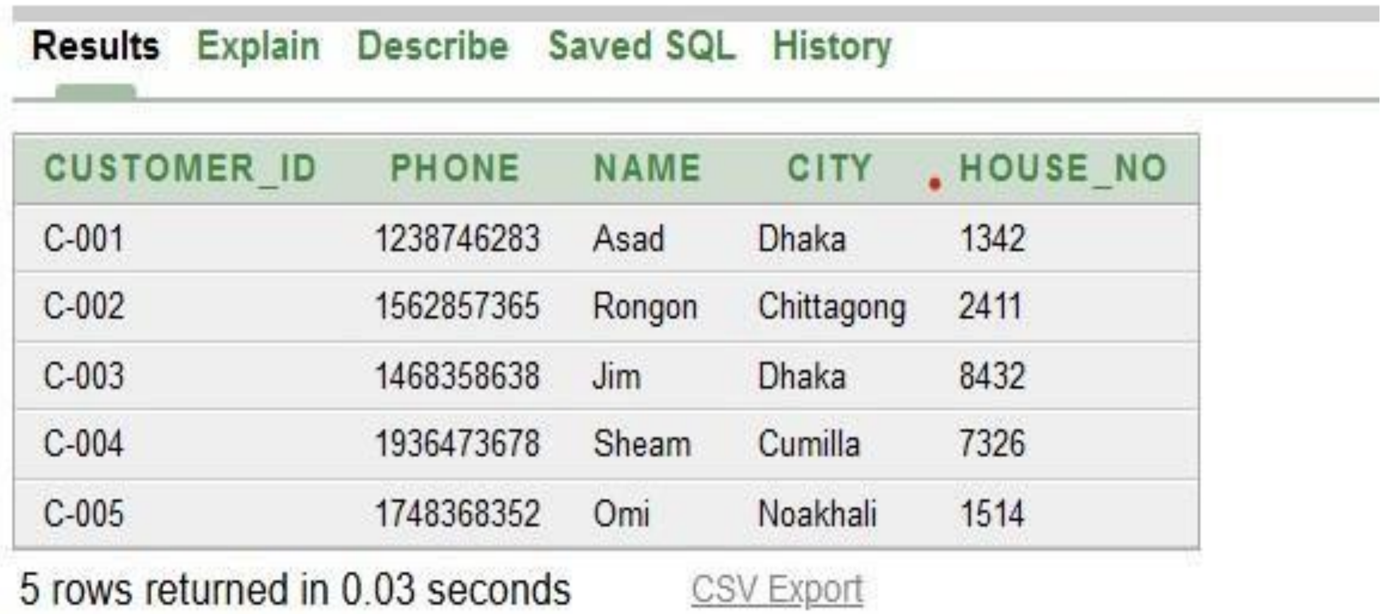
6.ALTER TABLE SHOP ADD CONSTRAINT PRIMARY KEY(SHOP_ID) REFERANCE SHOP (SHOP_ID);

7. ALTER TABLE BEAUTY_PRODUCT ADD CONSTRAINT PRIMARY KEY (PRODUCT_ID, TYPE) REFERANCE BRAND (BRAND_ID);

JOINING TABLES

1. **Write a query which will show the customer id, phone, name, city, house number.**

```
SELECT
CUSTOMER.CUSTOMER_ID, CUSTOMER.PHONE, CUSTOMER.NAME, CUSTOME
R_ADDRESS.CITY, CUSTOMER_ADDRESS.HOUSE_NO FROM
CUSTOMER, CUSTOMER_ADDRESS WHERE
CUSTOMER.CUSTOMER_ID=CUSTOMER_ADDRESS.CUSTOMER_ID;
```

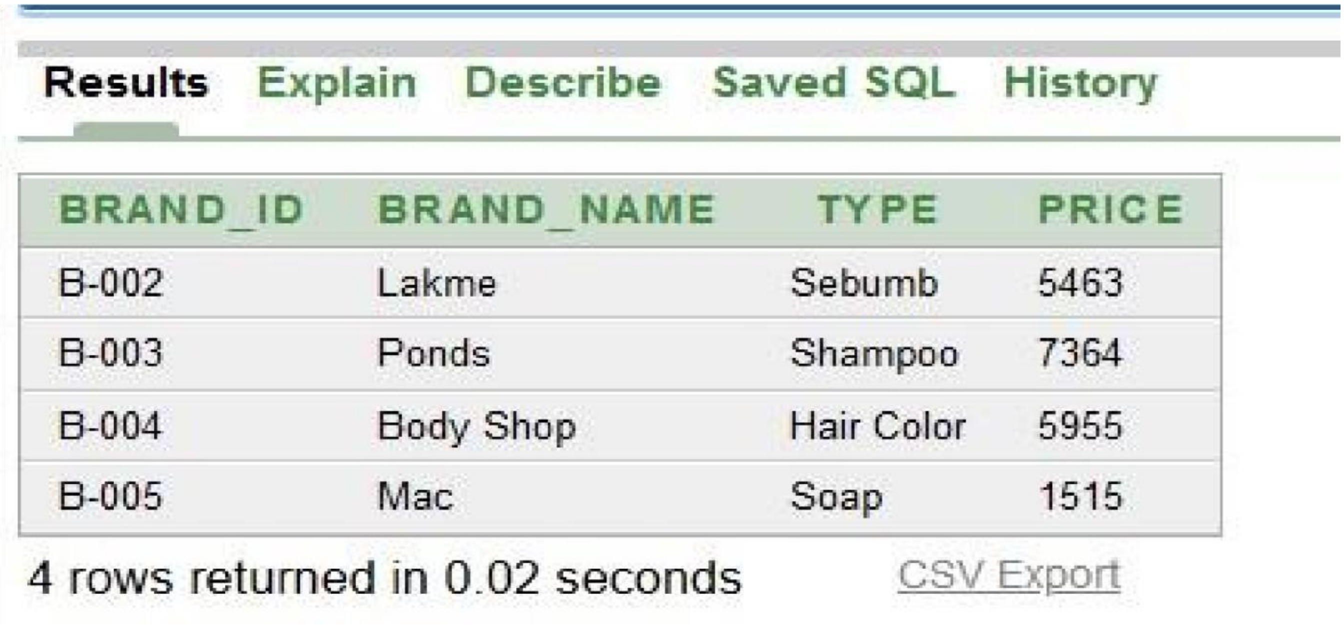


CUSTOMER_ID	PHONE	NAME	CITY	HOUSE_NO
C-001	1238746283	Asad	Dhaka	1342
C-002	1562857365	Rongon	Chittagong	2411
C-003	1468358638	Jim	Dhaka	8432
C-004	1936473678	Sheam	Cumilla	7326
C-005	1748368352	Omi	Noakhali	1514

5 rows returned in 0.03 seconds

2. Write a query which will show brand id, brand name, type, price.

```
SELECT BRAND.BRAND_ID, BRAND.BRAND_NAME, BEAUTY_PRODUCT.TYPE,
BEAUTY_PRODUCT.PRICE
FROM BRAND, BEAUTY_PRODUCT
WHERE BRAND.BRAND_ID = BEAUTY_PRODUCT.BRAND_ID;
```

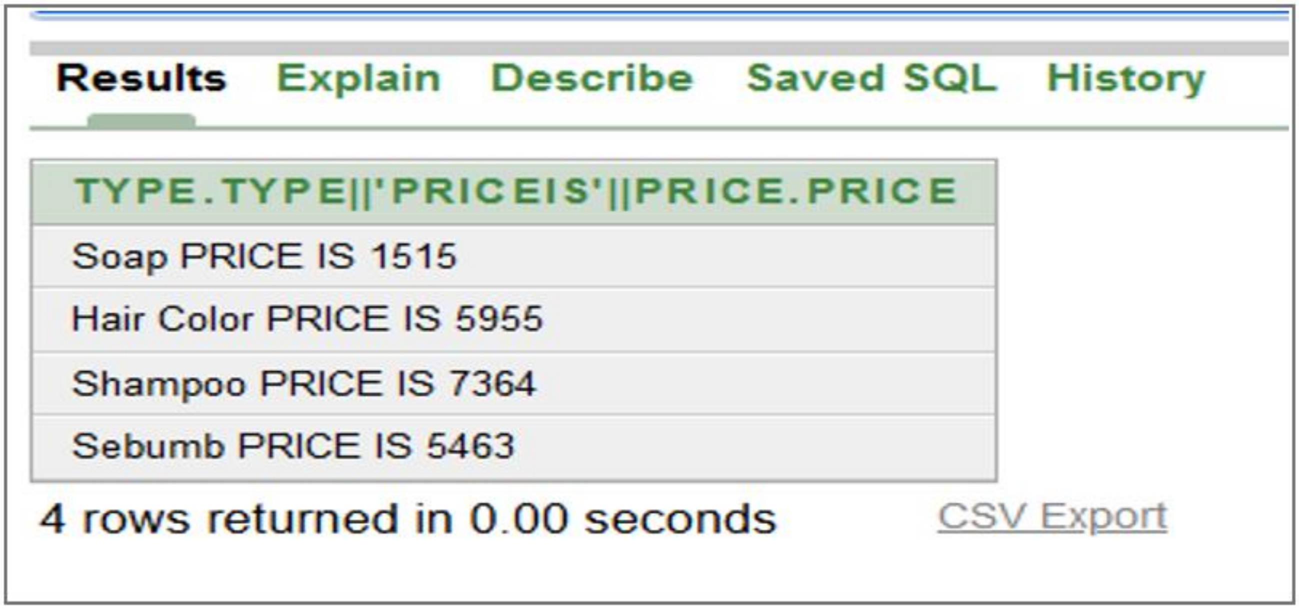


BRAND_ID	BRAND_NAME	TYPE	PRICE
B-002	Lakme	Sebumb	5463
B-003	Ponds	Shampoo	7364
B-004	Body Shop	Hair Color	5955
B-005	Mac	Soap	1515

4 rows returned in 0.02 seconds

3. **Write a query which will show product price.**

```
SELECT TYPE.TYPE || ' PRICE IS ' || PRICE.PRICE FROM BEAUTY_PRODUCT TYPE, BEAUTY_PRODUCT PRICE WHERE
TYPE.SHOP_ID=PRICE.SHOP_ID;
```



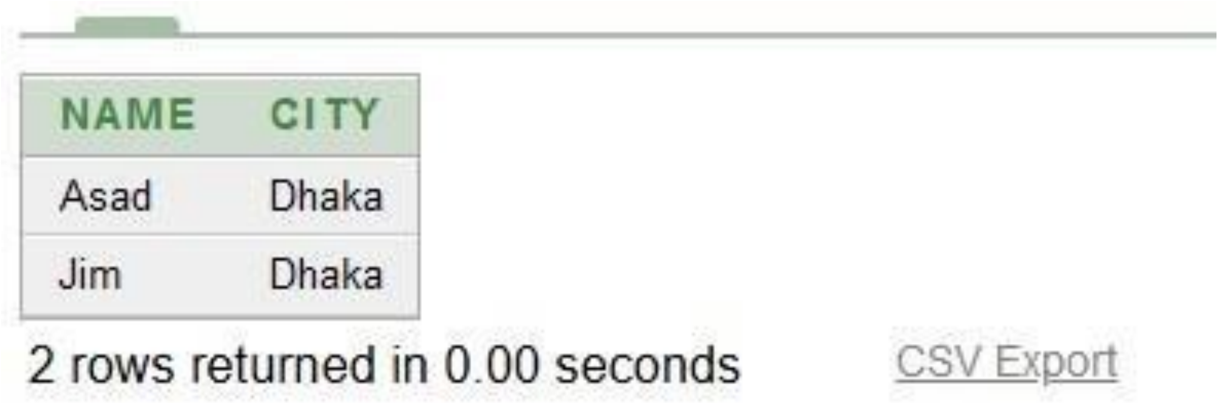
TYPE.TYPE||'PRICEIS'||PRICE.PRICE
Soap PRICE IS 1515
Hair Color PRICE IS 5955
Shampoo PRICE IS 7364
Sebumb PRICE IS 5463

4 rows returned in 0.00 seconds

SUB QUERY

1. Select all the customers who live in Dhaka.

```
SELECT   CUSTOMER.NAME, CUSTOMER_ADDRESS.CITY from
CUSTOMER, CUSTOMER_ADDRESS
WHERE CUSTOMER.CUSTOMER_ID = CUSTOMER_ADDRESS.CUSTOMER_ID AND
CUSTOMER.CUSTOMER_ID = ANY (SELECT CUSTOMER_ID
                        FROM CUSTOMER_ADDRESS
                        WHERE CITY = 'Dhaka');
```

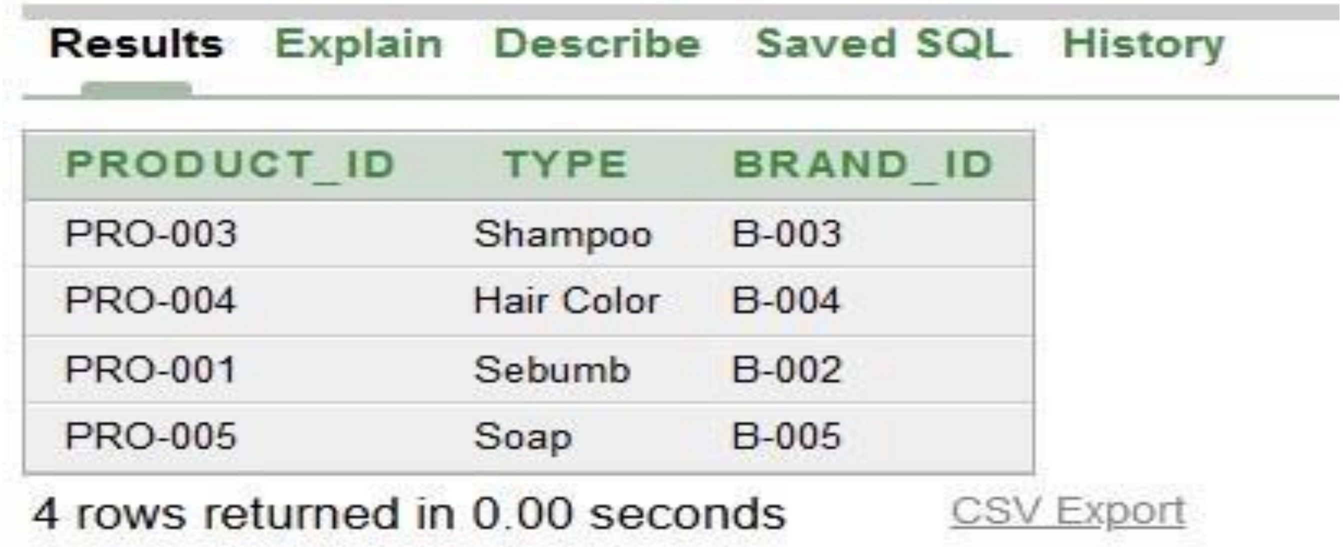


NAME	CITY
Asad	Dhaka
Jim	Dhaka

2 rows returned in 0.00 seconds

2.Select product id,type,brand id which price more than any payment amount.

```
SELECT PRODUCT_ID, TYPE, BRAND_ID
FROM BEAUTY_PRODUCT
WHERE PRICE > ANY (SELECT AMOUNT FROM PAYMENT);
```

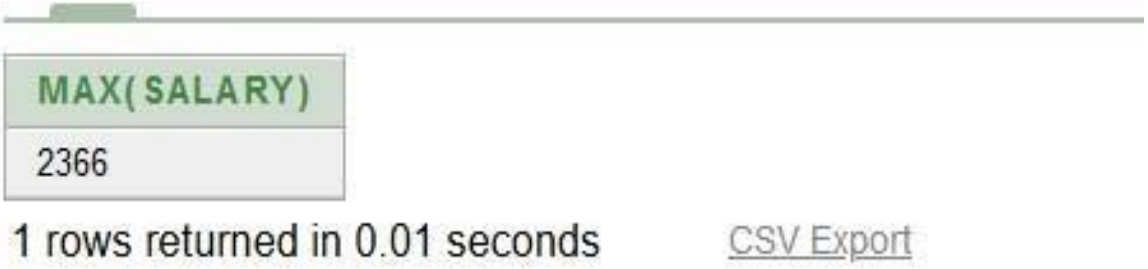


PRODUCT_ID	TYPE	BRAND_ID
PRO-003	Shampoo	B-003
PRO-004	Hair Color	B-004
PRO-001	Sebumb	B-002
PRO-005	Soap	B-005

4 rows returned in 0.00 seconds

3.Display the first maximum salary from salesman.

```
SELECT MAX(SALARY)
FROM SALASMAN
WHERE SALARY < (SELECT MAX(SALARY)
                FROM SALASMAN);
```



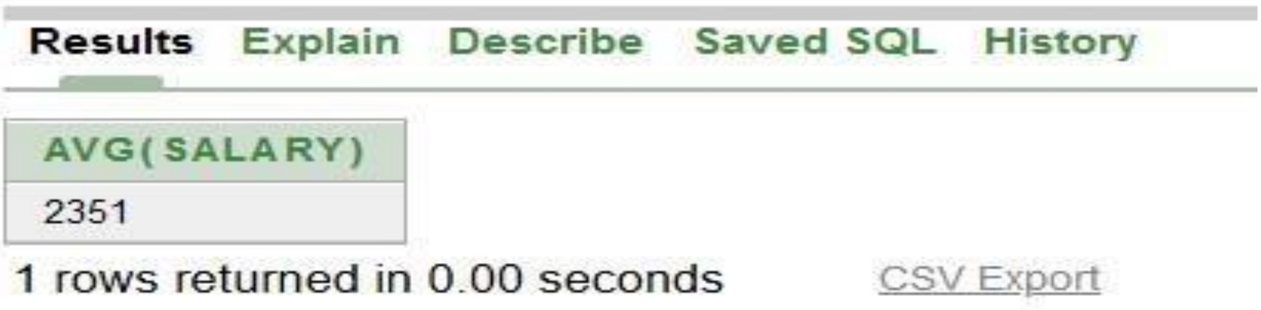
MAX(SALARY)
2366

1 rows returned in 0.01 seconds

Group Function

1. Write a query to find the average salary of Salesman.

```
SELECT AVG(SALARY) FROM SALESMAN;
```

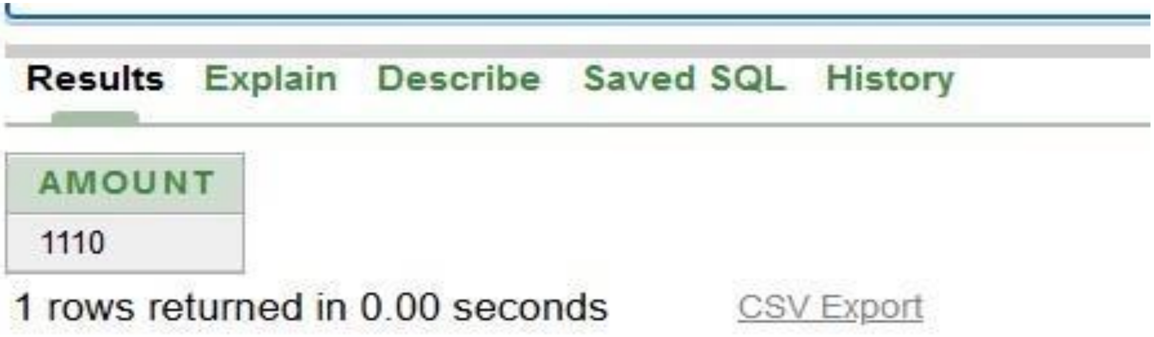


AVG(SALARY)
2351

1 rows returned in 0.00 seconds

2. Write a query to find the maximum payment.

```
SELECT MAX(AMOUNT) AS AMOUNT FROM PAYMENT;
```



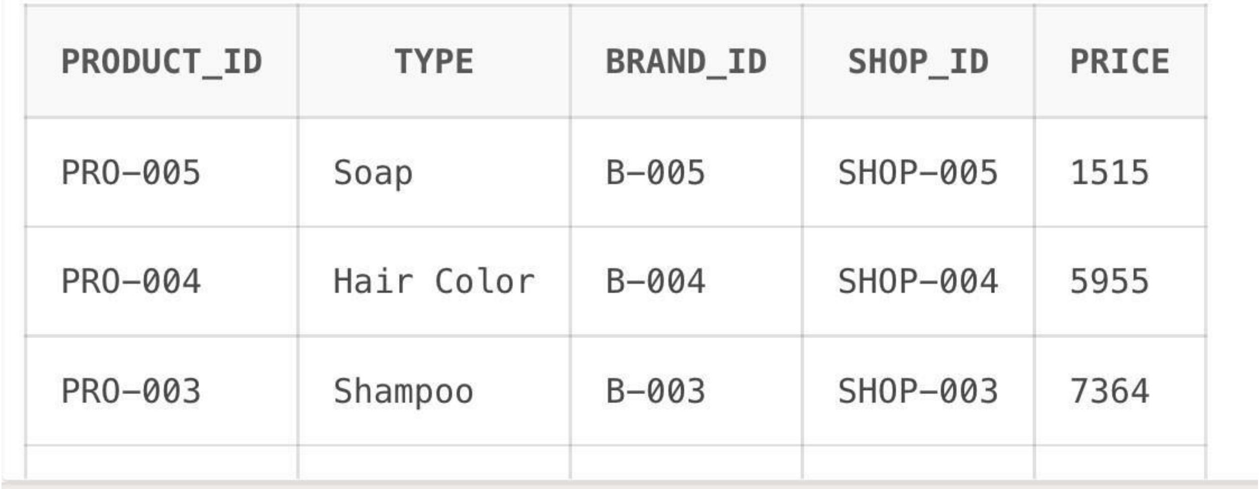
AMOUNT
1110

1 rows returned in 0.00 seconds

VIEW

1. Create a view named special beauty product where breauty product price is more than 1000

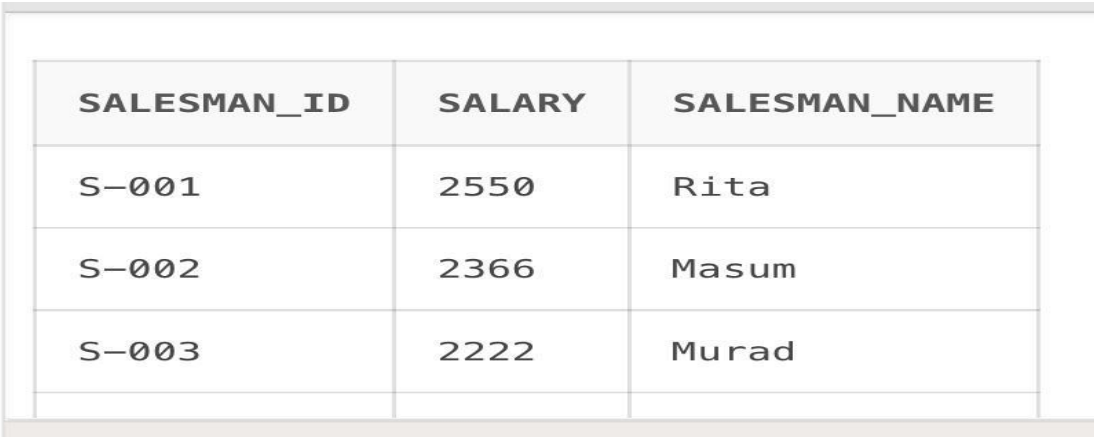
CREATE VIEW SPECIAL_BEAUTY_PRODUCT AS SELECT
PRODUCT_ID, TYPE, BRAND_ID, SHOP_ID, PRICE FROM BEAUTY_PRODUCT
WHERE PRICE>1000;



PRODUCT_ID	TYPE	BRAND_ID	SHOP_ID	PRICE
PRO-005	Soap	B-005	SHOP-005	1515
PRO-004	Hair Color	B-004	SHOP-004	5955
PRO-003	Shampoo	B-003	SHOP-003	7364

2. Create a view named poor salesman who get salary less than 3000.

CREATE VIEW POOR_SALESMAN AS SELECT
SALESMAN_ID, SALARY, SALESMAN_NAME FROM SALESMAN WHERE
3000>SALARY;

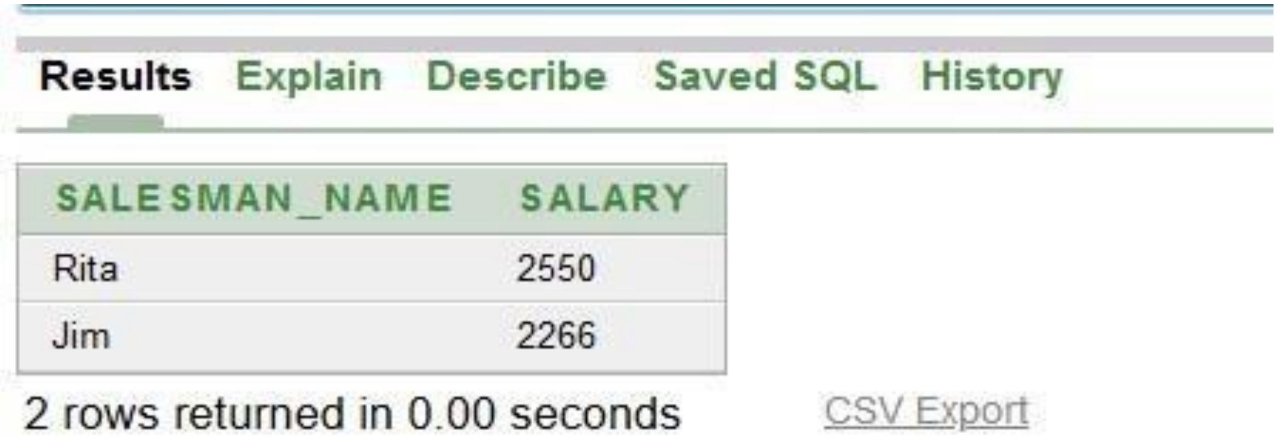


SALESMAN_ID	SALARY	SALESMAN_NAME
S-001	2550	Rita
S-002	2366	Masum
S-003	2222	Murad

OTHERS

1.Write a query to display the salesman name and salary whose name start with R or J.

SELECT SALESMAN_NAME, SALARY FROM SALESMAN WHERE SALESMAN_NAME LIKE ('R%') OR SALESMAN_NAME LIKE ('J%');

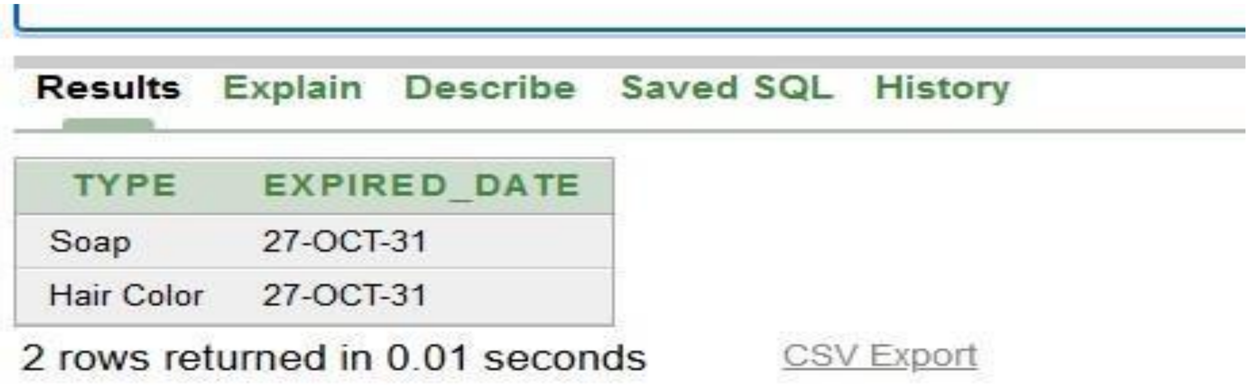


SALESMAN_NAME	SALARY
Rita	2550
Jim	2266

2 rows returned in 0.00 seconds

2.Write a query to find the product name and expired date which price more than 1000 and expired date 27-oct-31.

SELECT TYPE, EXPIRED_DATE FROM BEAUTY_PRODUCT WHERE PRICE>1000 AND EXPIRED_DATE='27-OCT-31';



TYPE	EXPIRED_DATE
Soap	27-OCT-31
Hair Color	27-OCT-31

2 rows returned in 0.01 seconds