

**American International University- Bangladesh**

**CSC 4181: Advanced Database Management Systems**

**Midterm Report**

**Spring 22-23**

**Project Title:** Hands-for-Hands

**Section:** A

|  |  |
| --- | --- |
| Student Name | Student Id |
| Jim, Jamin Rahman | 20-42732-1 |
| Partha Malakar | 20-42908-1 |
| Sadir Ahmed Zidan | 20-43263-1 |
| Fabiha Tasnim | 20-43426-1 |

Table of Contents

|  |  |  |
| --- | --- | --- |
| System Summary | :................................................................................ | 03 |
| ERD Diagram | :................................................................................ | 04 |
| Class Diagram | :................................................................................ | 05 |
| Use Case Diagram | :................................................................................ | 06 |
| Activity Diagram | :................................................................................ | 07-10 |
| Schema Diagram | :................................................................................ | 11 |
| Creating Tables | :................................................................................ | 12-19 |
| Tables with Data | :................................................................................ | 19-28 |
| Task | :................................................................................ | 29-33 |
| User Interface | :................................................................................ | 34-35 |

**System Summary:**

Throughout daily life, people occasionally desire various things to satisfy their ad hoc needs. These things might be a camera, a bicycle, a book, or anything else. A person decides to purchase the item even though it won't be useful in the long run since the necessity is only temporary. As a result, it qualifies as trash. Consequently, the person has the option of borrowing from others who already own such products rather than making a purchase.

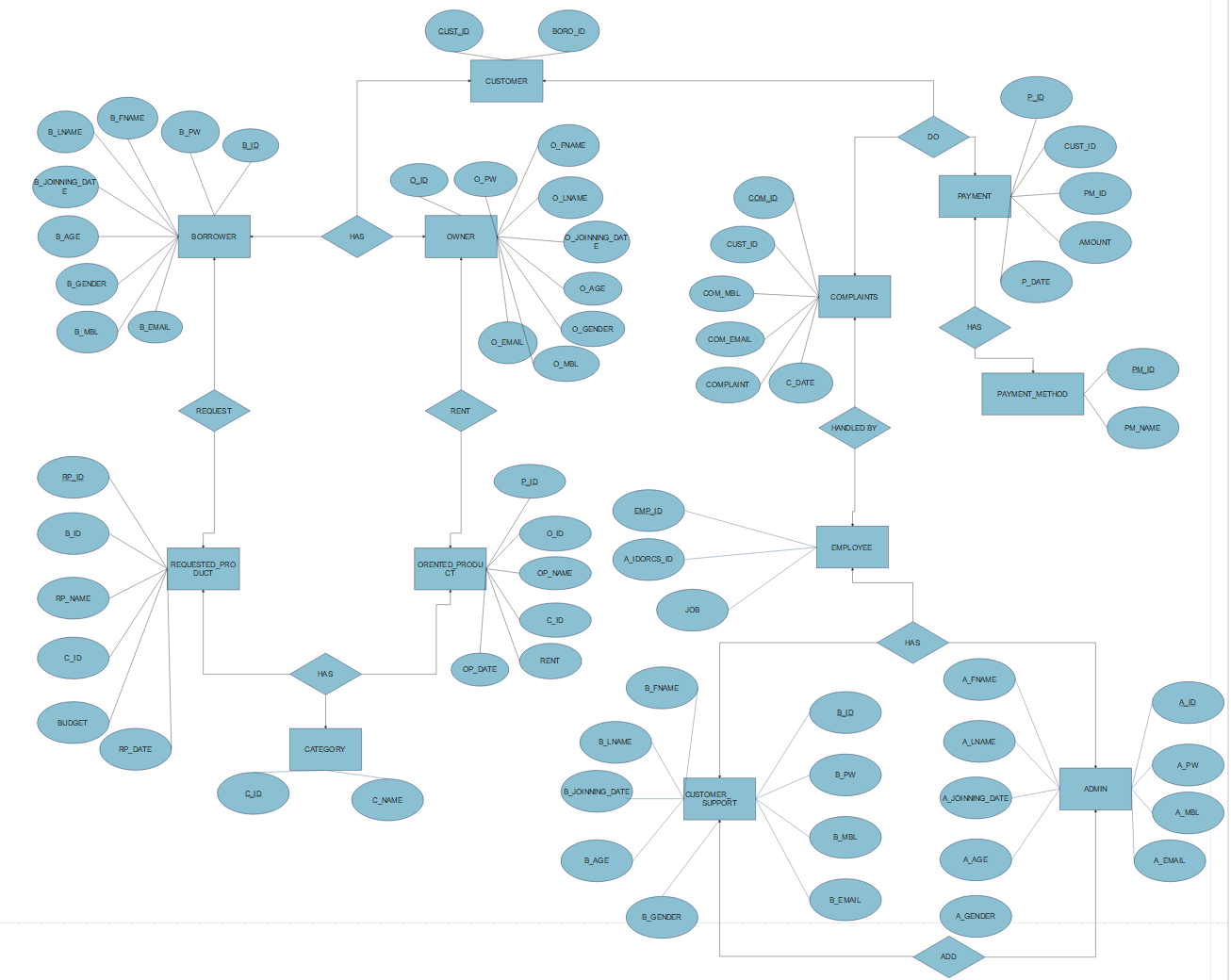
There are four types of user categories –

1. Admin
2. Customer Support
3. Owner
4. Borrower

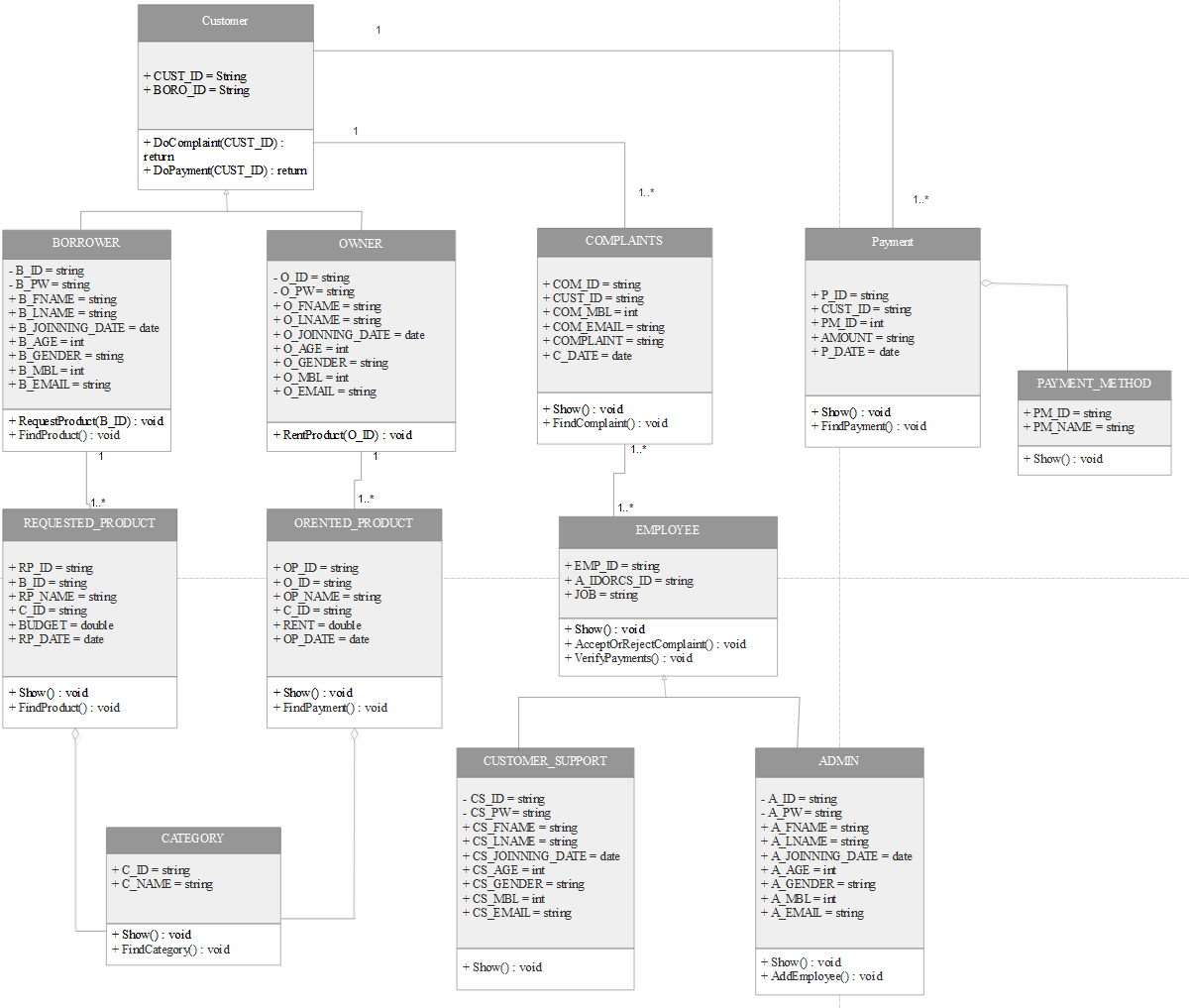
For this system, the administrative and customer support organizations are crucial. They are in charge of properly maintaining the platform.

Like any contemporary system, anyone may set up an account, log in, and utilize the many features. There is also the opportunity to change your account details and password. Moreover, users might have a distinctive username. Navigating the user interface is simple.

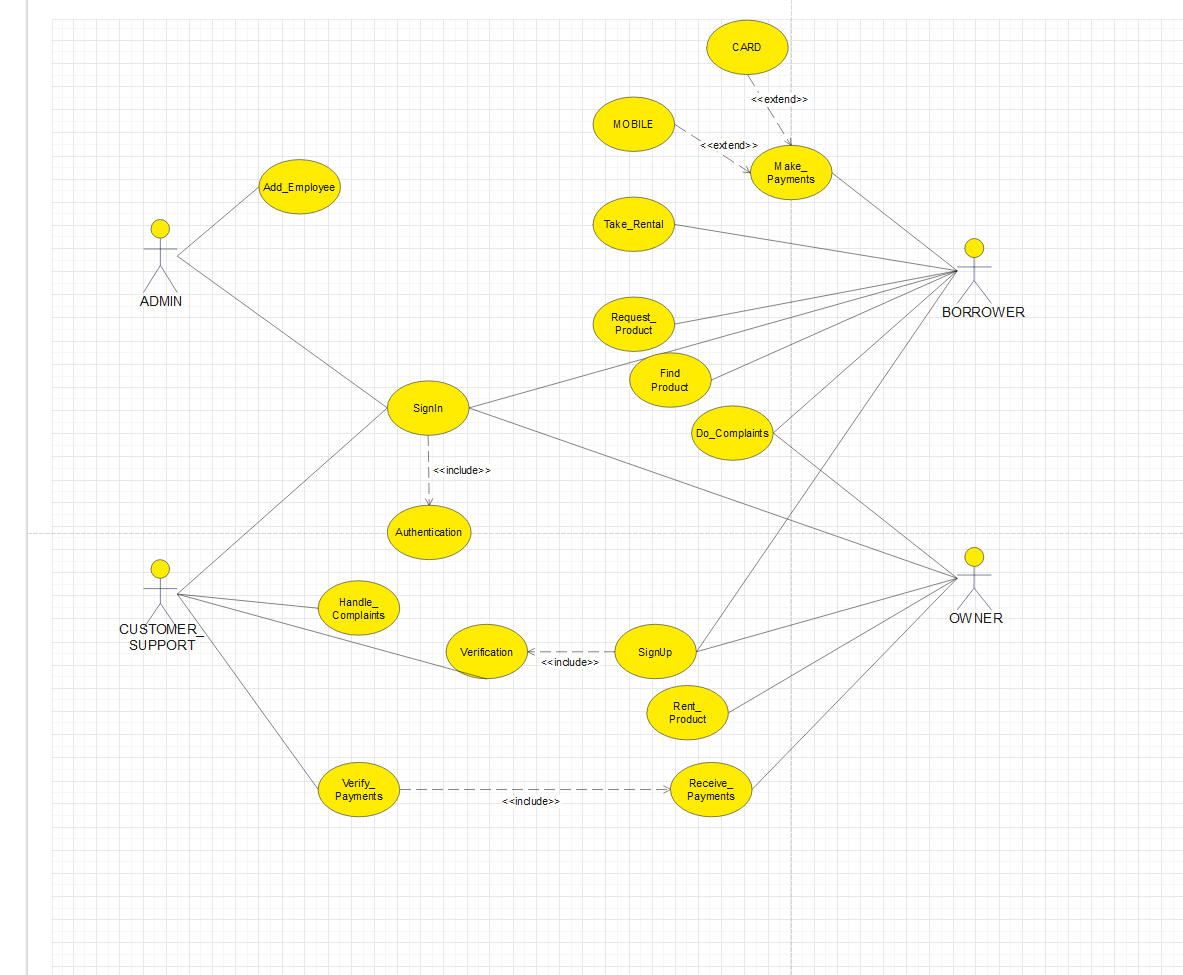
**ERD Diagram:**

****

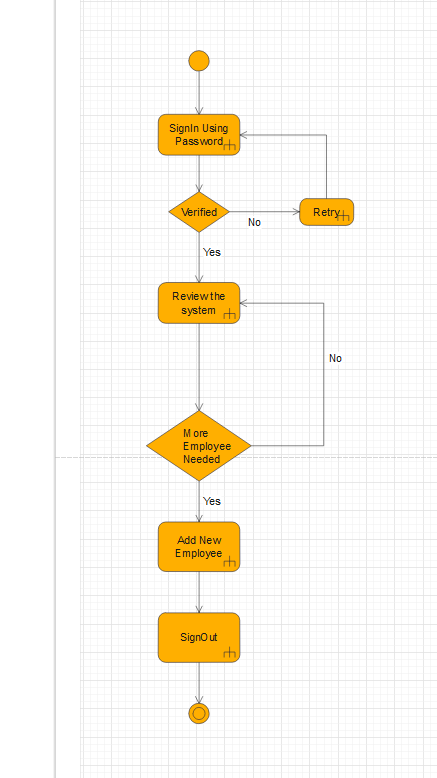
**Class Diagram:**

****

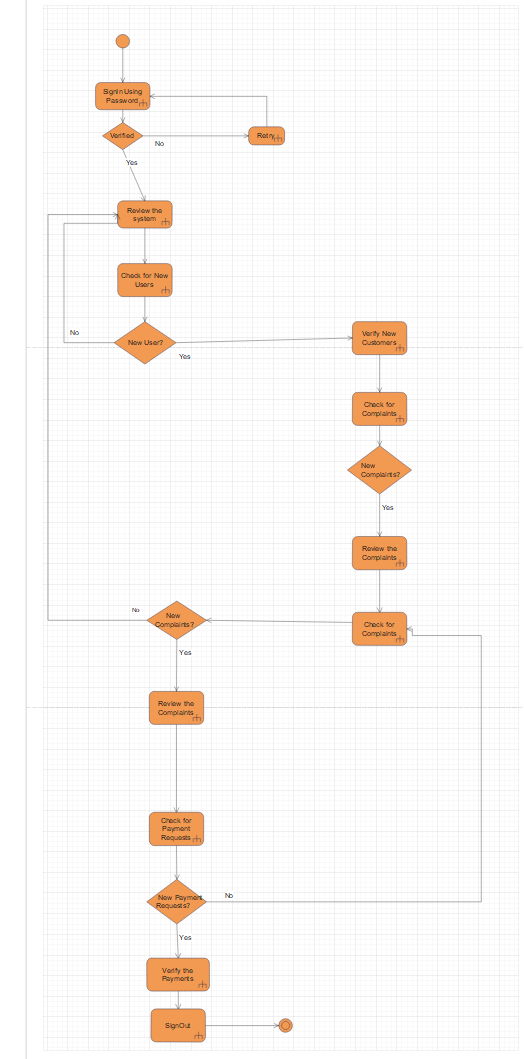
**Use Case Diagram:**

****

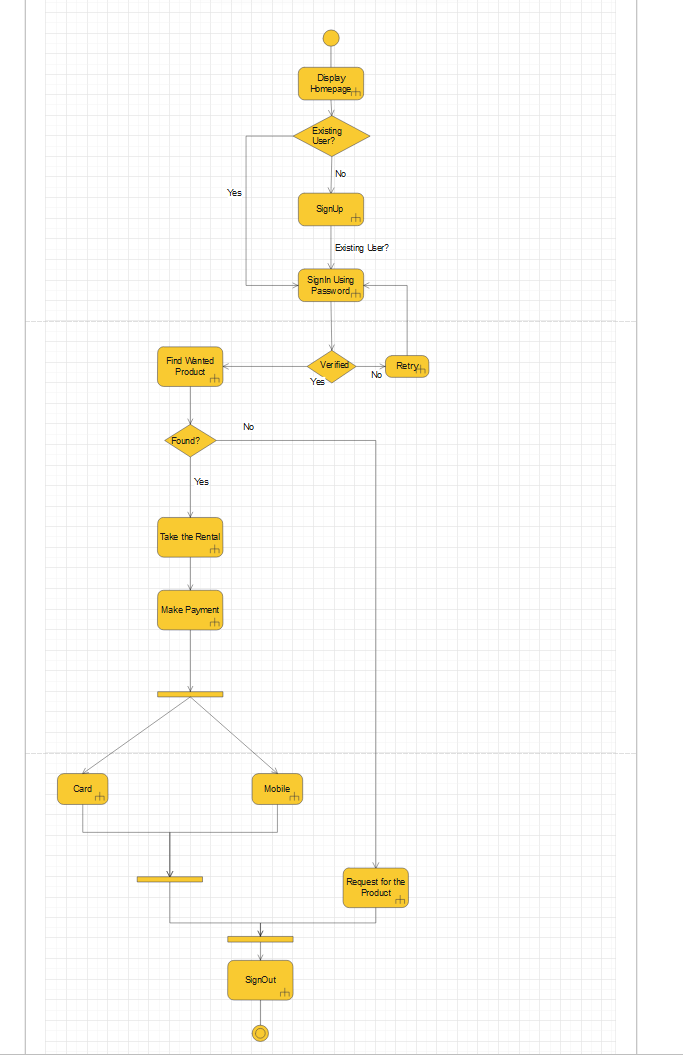
**Activity Diagram: (Admin)**

****

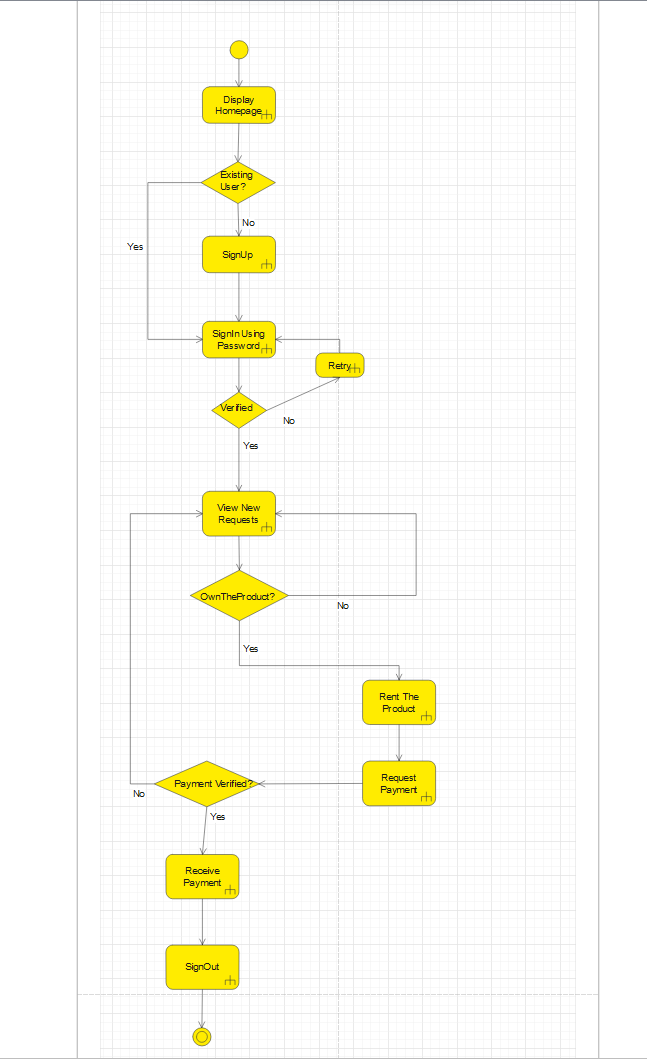
**Activity Diagram: (Customer Support)**

****

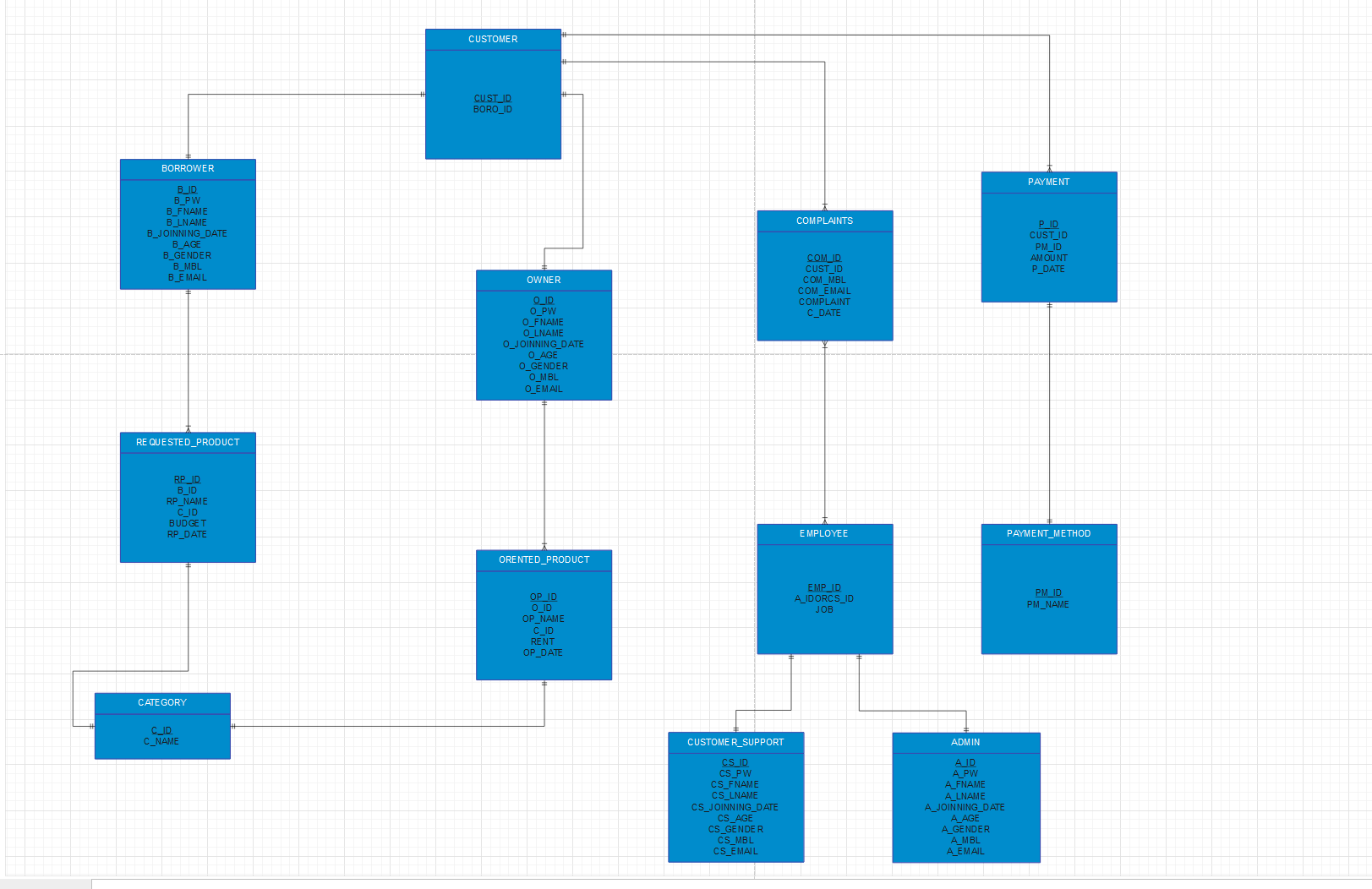
**Activity Diagram: (Borrower)**

****

**Activity Diagram: (Owner)**

****

**Schema Diagram:**

****

**Create Tables:**

**OWNER**

CREATE TABLE OWNER(

O\_ID varchar(30) not null,

O\_PW varchar(20) not null,

O\_FNAME varchar(40) not null,

O\_LNAME varchar(40) not null,

O\_JOINNING\_DATE date not null,

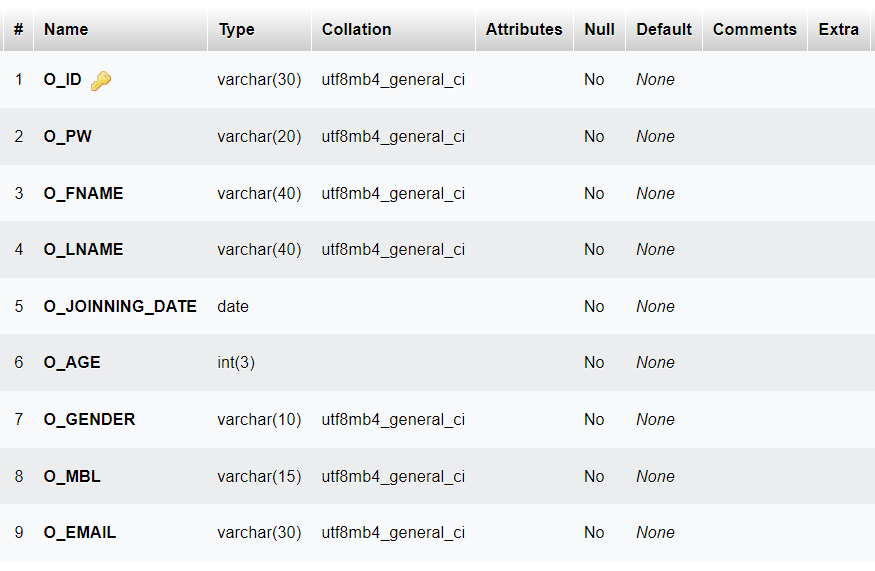
O\_AGE number(3) not null,

O\_GENDER varchar(10) not null,

O\_MBL varchar(15) not null,

O\_EMAIL varchar(30) not null,

PRIMARY KEY (O\_ID ) );



**ADMIN**

CREATE TABLE ADMIN(

A\_ID varchar(30) not null,

A\_PW varchar(20) not null,

A\_FNAME varchar(40) not null,

A\_LNAME varchar(40) not null,

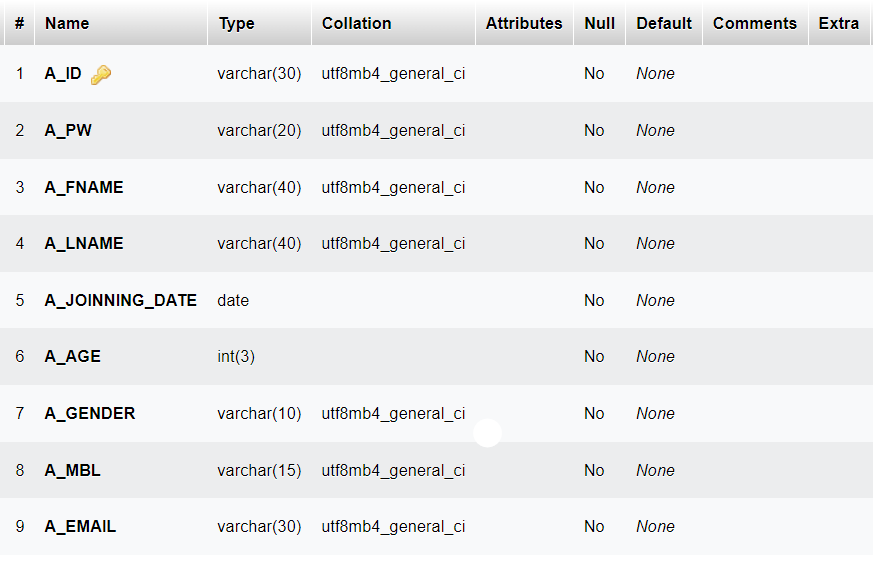
A\_JOINNING\_DATE date not null,

A\_AGE number(3) not null,

A\_GENDER varchar(10) not null,

A\_MBL varchar(15) not null,

A\_EMAIL varchar(30) not null, PRIMARY KEY (A\_ID ) );



**CUSTOMER\_SUPPORT**

CREATE TABLE CUSTOMER\_SUPPORT(

CS\_ID varchar(30) not null,

CS\_PW varchar(20) not null,

CS\_FNAME varchar(40) not null,

CS\_LNAME varchar(40) not null,

CS\_JOINNING\_DATE date not null,

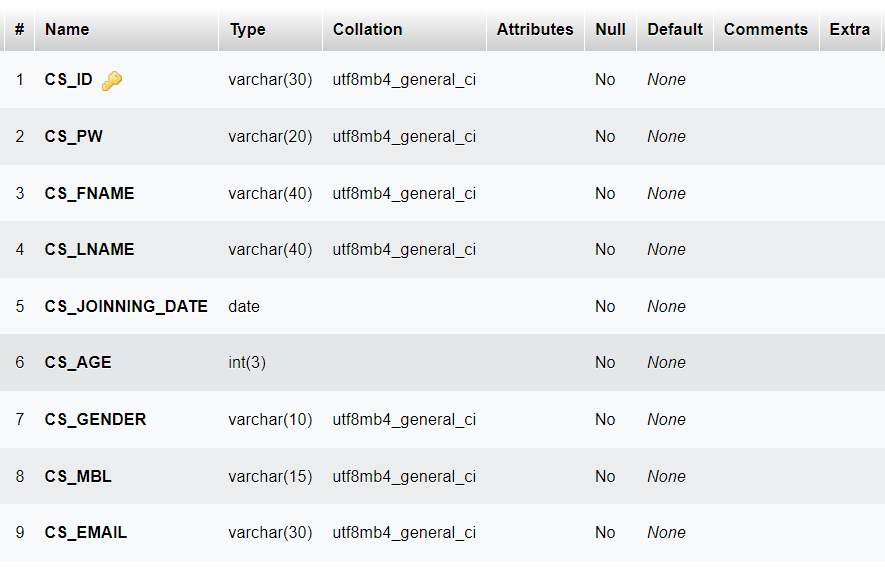
CS\_AGE number(3) not null,

CS\_GENDER varchar(10) not null,

CS\_MBL varchar(15) not null,

CS\_EMAIL varchar(30) not null,

PRIMARY KEY (CS\_ID ) );



**BORROWER**

CREATE TABLE BORROWER(

B\_ID varchar(30) not null,

B\_PW varchar(20) not null,

B\_FNAME varchar(40) not null,

B\_LNAME varchar(40) not null,

B\_JOINNING\_DATE date not null,

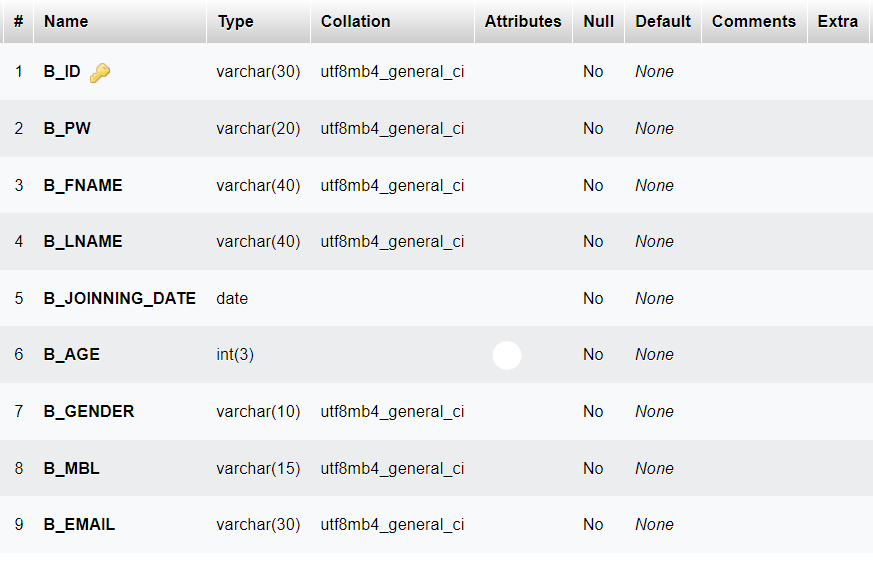
B\_AGE number(3) not null,

B\_GENDER varchar(10) not null,

B\_MBL varchar(15) not null,

B\_EMAIL varchar(30) not null,

PRIMARY KEY (B\_ID ) );



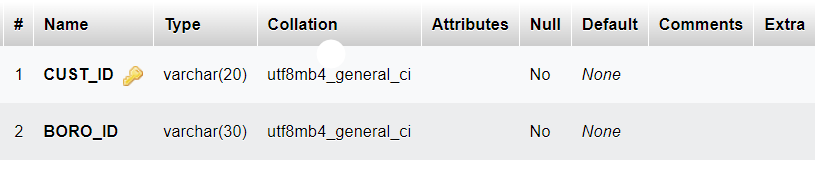
**CUSTOMER**

CREATE TABLE CUSTOMER(

CUST\_ID varchar(20) not null,

BORO\_ID varchar(30) not null,

PRIMARY KEY (CUST\_ID ) );



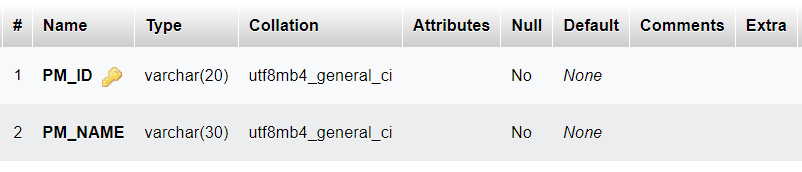
**PAYMENT\_METHOD**

CREATE TABLE PAYMENT\_METHOD(

PM\_ID varchar(20) not null,

PM\_NAME varchar(30) not null,

PRIMARY KEY (PM\_ID ) );



**EMPLOYEE**

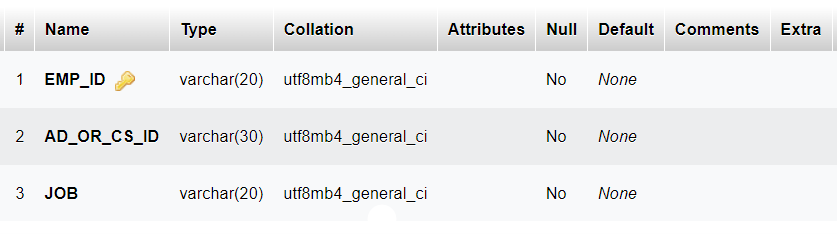
CREATE TABLE EMPLOYEE(

EMP\_ID varchar(20) not null,

AD\_OR\_CS\_ID varchar(30) not null,

JOB varchar(20) not null,

PRIMARY KEY (EMP\_ID ) );



**COMPLAINTS**

CREATE TABLE COMPLAINTS(

COM\_ID varchar(20) not null,

CUST\_ID varchar(20) not null,

COM\_MBL varchar(15) not null,

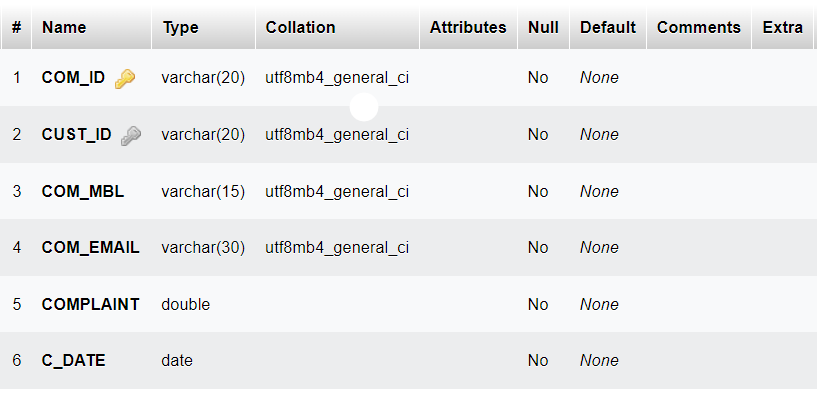
COM\_EMAIL varchar(30) not null,

COMPLAINT long not null,

C\_DATE date not null,

PRIMARY KEY (COM\_ID ),

FOREIGN KEY(CUST\_ID) REFERENCES CUSTOMER(CUST\_ID) );



**PAYMENT**

CREATE TABLE PAYMENT(

P\_ID varchar(20) not null,

CUST\_ID varchar(20) not null,

PM\_ID varchar(20) not null,

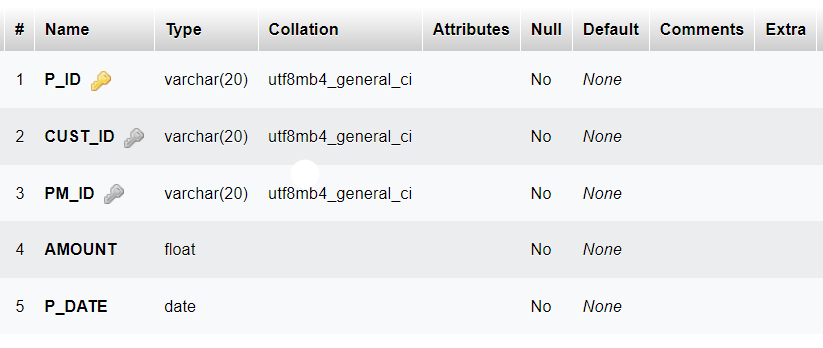
AMOUNT float(10) not null,

P\_DATE date not null,

PRIMARY KEY (P\_ID ),

FOREIGN KEY(CUST\_ID) REFERENCES CUSTOMER(CUST\_ID),

FOREIGN KEY(PM\_ID) REFERENCES PAYMENT\_METHOD(PM\_ID) );



**REQUESTED\_PRODUCT**

CREATE TABLE REQUESTED\_PRODUCT(

RP\_ID varchar(20) not null,

B\_ID varchar(30) not null,

RP\_NAME varchar(50) not null,

C\_ID varchar(20) not null,

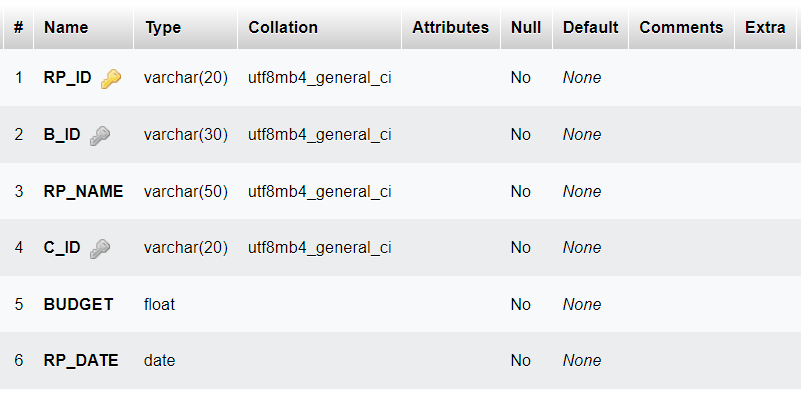
BUDGET float(10) not null,

RP\_DATE date not null,

PRIMARY KEY (RP\_ID),

FOREIGN KEY(B\_ID) REFERENCES BORROWER(B\_ID),

FOREIGN KEY(C\_ID) REFERENCES CATEGORY(C\_ID) );



**ORENTED\_PRODUCT**

CREATE TABLE ORENTED\_PRODUCT(

OP\_ID varchar(20) not null,

O\_ID varchar(30) not null,

OP\_NAME varchar(50) not null,

C\_ID varchar(20) not null,

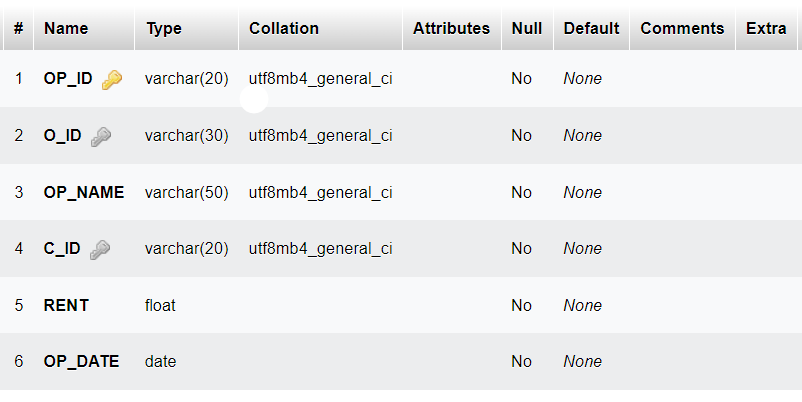
RENT float(10) not null,

OP\_DATE date not null,

PRIMARY KEY (OP\_ID ),

FOREIGN KEY(O\_ID ) REFERENCES OWNER(O\_ID ),

FOREIGN KEY(C\_ID) REFERENCES CATEGORY(C\_ID) );



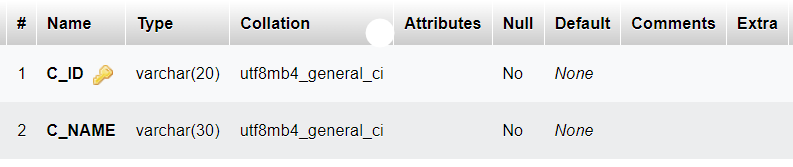
**CATEGORY**

CREATE TABLE CATEGORY(

C\_ID varchar(20) not null,

C\_NAME varchar(30) not null,

PRIMARY KEY (C\_ID ) );



**Insert Into Table:**

**OWNER**

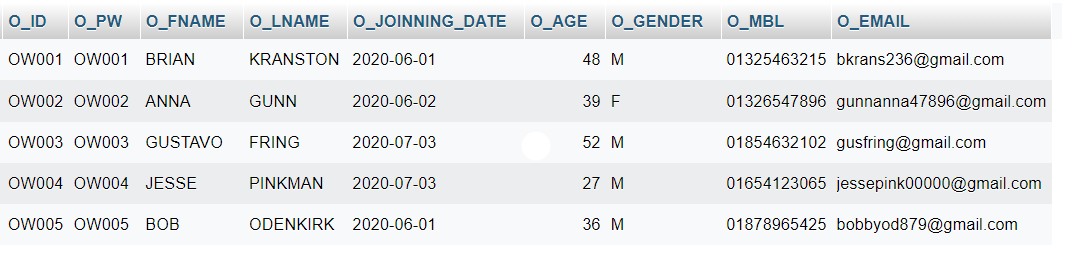
INSERT INTO OWNER(O\_ID, O\_PW, O\_FNAME, O\_LNAME, O\_JOINNING\_DATE, O\_AGE, O\_GENDER, O\_MBL, O\_EMAIL) VALUES('OW001','OW001','BRIAN','KRANSTON', '2020/06/01', 48, 'M', '01325463215', 'bkrans236@gmail.com');

INSERT INTO OWNER(O\_ID, O\_PW, O\_FNAME, O\_LNAME, O\_JOINNING\_DATE, O\_AGE, O\_GENDER, O\_MBL, O\_EMAIL) VALUES('OW002','OW002','ANNA','GUNN', '2020/06/02', 39, 'F', '01326547896', 'gunnanna47896@gmail.com');

INSERT INTO OWNER(O\_ID, O\_PW, O\_FNAME, O\_LNAME, O\_JOINNING\_DATE, O\_AGE, O\_GENDER, O\_MBL, O\_EMAIL) VALUES('OW003','OW003','GUSTAVO','FRING', '2020/07/03', 52, 'M', '01854632102', '[gusfring@gmail.com](mailto:gusfring@gmail.com)');

INSERT INTO OWNER(O\_ID, O\_PW, O\_FNAME, O\_LNAME, O\_JOINNING\_DATE, O\_AGE, O\_GENDER, O\_MBL, O\_EMAIL) VALUES('OW004','OW004','JESSE','PINKMAN', '2020/07/03', 27, 'M', '01654123065', 'jessepink00000@gmail.com');

INSERT INTO OWNER(O\_ID, O\_PW, O\_FNAME, O\_LNAME, O\_JOINNING\_DATE, O\_AGE, O\_GENDER, O\_MBL, O\_EMAIL) VALUES('OW005','OW005','BOB','ODENKIRK', '2020/06/01', 36, 'M', '01878965425', '[bobbyod879@gmail.com](mailto:bobbyod879@gmail.com)');



**ADMIN**

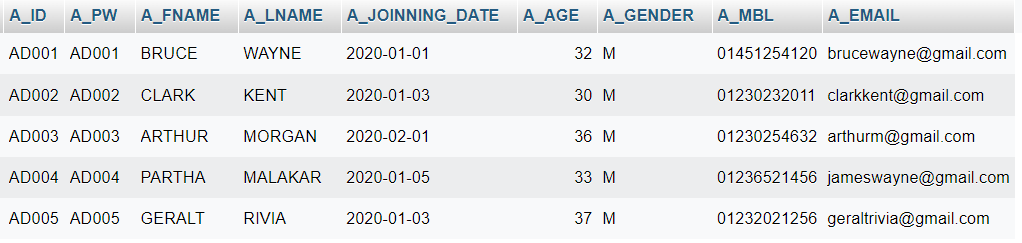
INSERT INTO ADMIN (A\_ID, A\_PW, A\_FNAME, A\_LNAME, A\_JOINNING\_DATE, A\_AGE, A\_GENDER, A\_MBL, A\_EMAIL) VALUES('AD001','AD001','BRUCE','WAYNE', '2020/01/01', 32, 'M', '01451254120', 'brucewayne@gmail.com');

INSERT INTO ADMIN (A\_ID, A\_PW, A\_FNAME, A\_LNAME, A\_JOINNING\_DATE, A\_AGE, A\_GENDER, A\_MBL, A\_EMAIL) VALUES('AD002','AD002','CLARK','KENT', '2020/01/03', 30, 'M', '01230232011', 'clarkkent@gmail.com');

INSERT INTO ADMIN (A\_ID, A\_PW, A\_FNAME, A\_LNAME, A\_JOINNING\_DATE, A\_AGE, A\_GENDER, A\_MBL, A\_EMAIL) VALUES('AD003','AD003','ARTHUR','MORGAN', '2020/02/01', 36, 'M', '01230254632', 'arthurm@gmail.com');

INSERT INTO ADMIN (A\_ID, A\_PW, A\_FNAME, A\_LNAME, A\_JOINNING\_DATE, A\_AGE, A\_GENDER, A\_MBL, A\_EMAIL) VALUES('AD004','AD004','JAMES','WAYNE', '2020/01/05', 33, 'M', '01236521456', 'jameswayne@gmail.com');

INSERT INTO ADMIN (A\_ID, A\_PW, A\_FNAME, A\_LNAME, A\_JOINNING\_DATE, A\_AGE, A\_GENDER, A\_MBL, A\_EMAIL) VALUES('AD005','AD005','GERALT','RIVIA','2020/01/03', 37, 'M', '01232021256', '[geraltrivia@gmail.com](mailto:geraltrivia@gmail.com)');



**CUSTOMER\_SUPPORT**

INSERT INTO CUSTOMER\_SUPPORT(CS\_ID, CS\_PW, CS\_FNAME, CS\_LNAME, CS\_JOINNING\_DATE, CS\_AGE, CS\_GENDER, CS\_MBL, CS\_EMAIL) VALUES('CS001','CS001','ALFRED','PENNYWORTH', '2020/04/01', 25, 'M', '01230232549', 'alfredpenny@gmail.com');

INSERT INTO CUSTOMER\_SUPPORT(CS\_ID, CS\_PW, CS\_FNAME, CS\_LNAME, CS\_JOINNING\_DATE, CS\_AGE, CS\_GENDER, CS\_MBL, CS\_EMAIL) VALUES('CS002','CS002','JACK','SPRING', '2021/01/01', 24, 'M', '01232125478', 'jacks@gmail.com');

INSERT INTO CUSTOMER\_SUPPORT(CS\_ID, CS\_PW, CS\_FNAME, CS\_LNAME, CS\_JOINNING\_DATE, CS\_AGE, CS\_GENDER, CS\_MBL, CS\_EMAIL) VALUES('CS003','CS003','JOHN','HALO', '2021/03/04', 26, 'M', '01232125469', 'johnhalo002@gmail.com');

INSERT INTO CUSTOMER\_SUPPORT(CS\_ID, CS\_PW, CS\_FNAME, CS\_LNAME, CS\_JOINNING\_DATE, CS\_AGE, CS\_GENDER, CS\_MBL, CS\_EMAIL) VALUES('CS004','CS004','EDWARD','KENWAY', '2021/03/04', 26, 'M', '02312547896', '[kenwayed23@gmail.com](mailto:kenwayed23@gmail.com)');

INSERT INTO CUSTOMER\_SUPPORT(CS\_ID, CS\_PW, CS\_FNAME, CS\_LNAME, CS\_JOINNING\_DATE, CS\_AGE, CS\_GENDER, CS\_MBL, CS\_EMAIL) VALUES('CS005','CS005','HOLY LANG','PENNYWORTH', '2022/01/01', 24, 'F', '01546321203', '[langholy456@yahoo.com](mailto:langholy456@yahoo.com)');



**BORROWER**

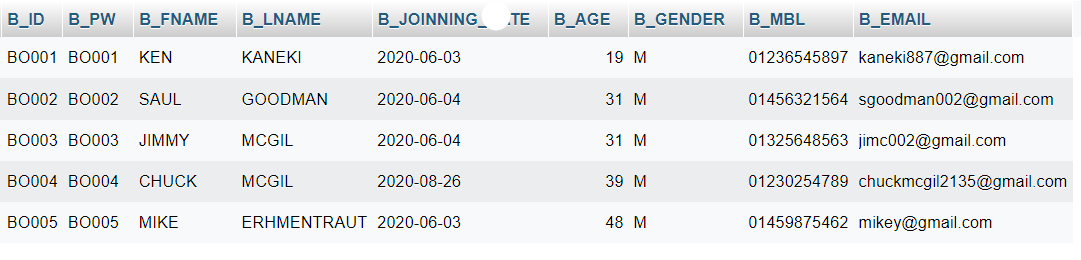
INSERT INTO BORROWER(B\_ID, B\_PW, B\_FNAME, B\_LNAME, B\_JOINNING\_DATE, B\_AGE, B\_GENDER, B\_MBL, B\_EMAIL) VALUES('BO001','BO001','KEN','KANEKI', '2020/06/03', 19, 'M', '01236545897', 'kaneki887@gmail.com');

INSERT INTO BORROWER(B\_ID, B\_PW, B\_FNAME, B\_LNAME, B\_JOINNING\_DATE, B\_AGE, B\_GENDER, B\_MBL, B\_EMAIL) VALUES('BO002','BO002','SAUL','GOODMAN', '2020/06/04', 31, 'M', '01456321564', 'sgoodman002@gmail.com');

INSERT INTO BORROWER(B\_ID, B\_PW, B\_FNAME, B\_LNAME, B\_JOINNING\_DATE, B\_AGE, B\_GENDER, B\_MBL, B\_EMAIL) VALUES('BO003','BO003','JIMMY','MCGIL','2020/06/04', 31, 'M', '01325648563', 'jimc002@gmail.com');

INSERT INTO BORROWER(B\_ID, B\_PW, B\_FNAME, B\_LNAME, B\_JOINNING\_DATE, B\_AGE, B\_GENDER, B\_MBL, B\_EMAIL) VALUES('BO004','BO004','CHUCK','MCGIL','2020/08/26', 39, 'M', '01230254789', 'chuckmcgil2135@gmail.com');

INSERT INTO BORROWER(B\_ID, B\_PW, B\_FNAME, B\_LNAME, B\_JOINNING\_DATE, B\_AGE, B\_GENDER, B\_MBL, B\_EMAIL) VALUES('BO005','BO005','MIKE','ERHMENTRAUT', '2022020/06/03',48, 'M', '01459875462', '[mikey@gmail.com](mailto:mikey@gmail.com)');



**CUSTOMER**

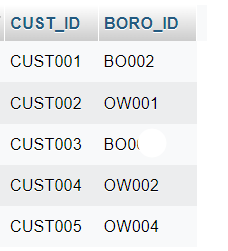
INSERT INTO CUSTOMER(CUST\_ID, BORO\_ID) VALUES('CUST001','BO002');

INSERT INTO CUSTOMER(CUST\_ID, BORO\_ID) VALUES('CUST002','OW001');

INSERT INTO CUSTOMER(CUST\_ID, BORO\_ID) VALUES('CUST003','BO005');

INSERT INTO CUSTOMER(CUST\_ID, BORO\_ID) VALUES('CUST004','OW002');

INSERT INTO CUSTOMER(CUST\_ID, BORO\_ID) VALUES('CUST005','OW004');



**PAYMENT\_METHOD**

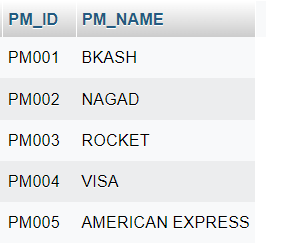
INSERT INTO PAYMENT\_METHOD(PM\_ID, PM\_NAME) VALUES('PM001','BKASH');

INSERT INTO PAYMENT\_METHOD(PM\_ID, PM\_NAME) VALUES('PM002','NAGAD');

INSERT INTO PAYMENT\_METHOD(PM\_ID, PM\_NAME) VALUES('PM003','ROCKET');

INSERT INTO PAYMENT\_METHOD(PM\_ID, PM\_NAME) VALUES('PM004','VISA');

INSERT INTO PAYMENT\_METHOD(PM\_ID, PM\_NAME) VALUES('PM005','AMERICAN EXPRESS');



**EMPLOYEE**

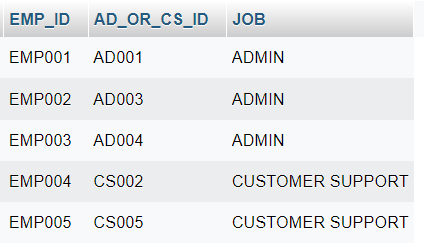
INSERT INTO EMPLOYEE(EMP\_ID, AD\_OR\_CS\_ID, JOB) VALUES('EMP001','AD001','ADMIN');

INSERT INTO EMPLOYEE(EMP\_ID, AD\_OR\_CS\_ID, JOB) VALUES('EMP002','AD003','ADMIN');

INSERT INTO EMPLOYEE(EMP\_ID, AD\_OR\_CS\_ID, JOB) VALUES('EMP003','AD004','ADMIN');

INSERT INTO EMPLOYEE(EMP\_ID, AD\_OR\_CS\_ID, JOB) VALUES('EMP004','CS002','CUSTOMER SUPPORT');

INSERT INTO EMPLOYEE(EMP\_ID, AD\_OR\_CS\_ID, JOB) VALUES('EMP005','CS005','CUSTOMER SUPPORT');



**COMPLAINTS**

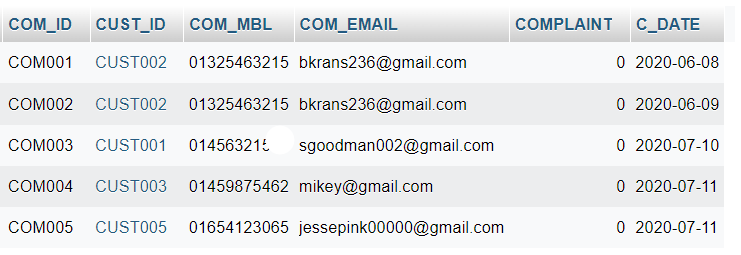
INSERT INTO COMPLAINTS(COM\_ID, CUST\_ID, COM\_MBL, COM\_EMAIL, COMPLAINT, C\_DATE) VALUES('COM001', 'CUST002', '01325463215', 'bkrans236@gmail.com', 'I did not get the requested product.','2020-06-08');

INSERT INTO COMPLAINTS(COM\_ID, CUST\_ID, COM\_MBL, COM\_EMAIL, COMPLAINT, C\_DATE) VALUES('COM002', 'CUST002', '01325463215', 'bkrans236@gmail.com', 'The customer support was not able to help.', '2020-06-09');

INSERT INTO COMPLAINTS(COM\_ID, CUST\_ID, COM\_MBL, COM\_EMAIL, COMPLAINT, C\_DATE) VALUES('COM003', 'CUST001', '01456321564', 'sgoodman002@gmail.com', 'I was charged more than the shown amount.','2020-07-10');

INSERT INTO COMPLAINTS(COM\_ID, CUST\_ID, COM\_MBL, COM\_EMAIL, COMPLAINT, C\_DATE) VALUES('COM004', 'CUST003', '01459875462', 'mikey@gmail.com', 'I received a faulty product.', '2020-07-11');

INSERT INTO COMPLAINTS(COM\_ID, CUST\_ID, COM\_MBL, COM\_EMAIL, COMPLAINT, C\_DATE) VALUES('COM005', 'CUST005', '01654123065', 'jessepink00000@gmail.com', 'I was charged more than the shown amount.', '2020-07-11');



**PAYMENT**

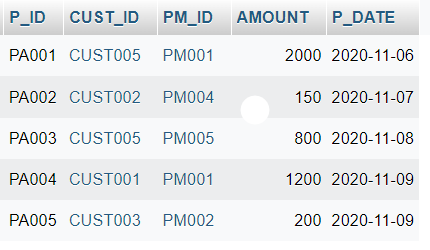
INSERT INTO PAYMENT(P\_ID, CUST\_ID, PM\_ID, AMOUNT, P\_DATE) VALUES('PA001', 'CUST005', 'PM001', 2000, '2020-11-06');

INSERT INTO PAYMENT(P\_ID, CUST\_ID, PM\_ID, AMOUNT, P\_DATE) VALUES('PA002', 'CUST002', 'PM004', 150, '2020-11-07');

INSERT INTO PAYMENT(P\_ID, CUST\_ID, PM\_ID, AMOUNT, P\_DATE) VALUES('PA003', 'CUST005', 'PM005', 800, '2020-11-08');

INSERT INTO PAYMENT(P\_ID, CUST\_ID, PM\_ID, AMOUNT, P\_DATE) VALUES('PA004', 'CUST001', 'PM001', 1200, '2020-11-09');

INSERT INTO PAYMENT(P\_ID, CUST\_ID, PM\_ID, AMOUNT, P\_DATE) VALUES('PA005', 'CUST003', 'PM002', 200, '2020-11-09');



**REQUESTED\_PRODUCT**

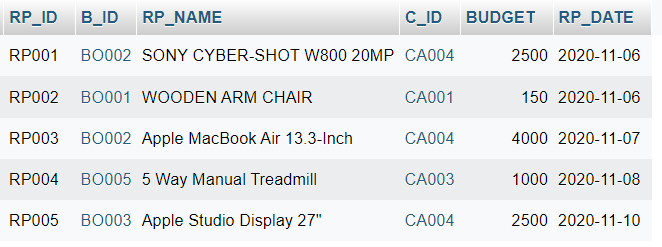
INSERT INTO REQUESTED\_PRODUCT(RP\_ID, B\_ID, RP\_NAME, C\_ID, BUDGET, RP\_DATE) VALUES('RP001', 'BO002', 'SONY CYBER-SHOT W800 20MP', 'CA004', 2500, '2020-11-06');

INSERT INTO REQUESTED\_PRODUCT(RP\_ID, B\_ID, RP\_NAME, C\_ID, BUDGET, RP\_DATE) VALUES('RP002', 'BO001', 'WOODEN ARM CHAIR', 'CA001', 150, '2020-11-06');

INSERT INTO REQUESTED\_PRODUCT(RP\_ID, B\_ID, RP\_NAME, C\_ID, BUDGET, RP\_DATE) VALUES('RP003', 'BO002', 'Apple MacBook Air 13.3-Inch', 'CA004', 4000, '2020-11-07');

INSERT INTO REQUESTED\_PRODUCT(RP\_ID, B\_ID, RP\_NAME, C\_ID, BUDGET, RP\_DATE) VALUES('RP004', 'BO005', '5 Way Manual Treadmill', 'CA003', 1000, '2020-11-08');

INSERT INTO REQUESTED\_PRODUCT(RP\_ID, B\_ID, RP\_NAME, C\_ID, BUDGET, RP\_DATE) VALUES('RP005', 'BO003', 'Apple Studio Display 27"', 'CA004', 2500, '2020-11-10');



**ORENTED\_PRODUCT**

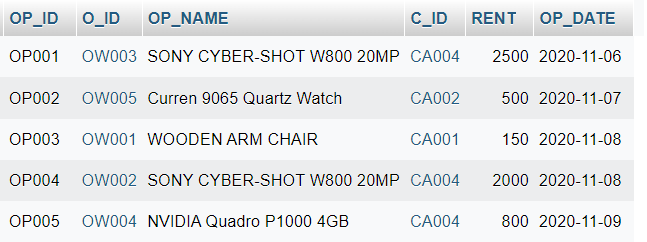
INSERT INTO ORENTED\_PRODUCT (OP\_ID, O\_ID, OP\_NAME, C\_ID, RENT, OP\_DATE) VALUES('OP001', 'OW003', 'SONY CYBER-SHOT W800 20MP', 'CA004', 2500, '2020-11-06');

INSERT INTO ORENTED\_PRODUCT (OP\_ID, O\_ID, OP\_NAME, C\_ID, RENT, OP\_DATE) VALUES('OP002', 'OW005', 'Curren 9065 Quartz Watch ', 'CA002', 500, '2020-11-07');

INSERT INTO ORENTED\_PRODUCT (OP\_ID, O\_ID, OP\_NAME, C\_ID, RENT, OP\_DATE) VALUES('OP003', 'OW001', 'WOODEN ARM CHAIR', 'CA001', 150, '2020-11-08');

INSERT INTO ORENTED\_PRODUCT (OP\_ID, O\_ID, OP\_NAME, C\_ID, RENT, OP\_DATE) VALUES('OP004', 'OW002', 'SONY CYBER-SHOT W800 20MP', 'CA004', 2000,'2020-11-08');

INSERT INTO ORENTED\_PRODUCT (OP\_ID, O\_ID, OP\_NAME, C\_ID, RENT, OP\_DATE) VALUES('OP005', 'OW004', 'NVIDIA Quadro P1000 4GB', 'CA004', 800, '2020-11-09');



**CATEGORY**

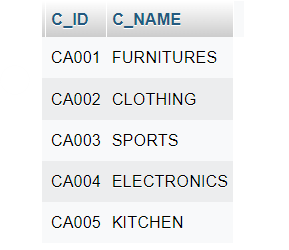
INSERT INTO CATEGORY(C\_ID, C\_NAME) VALUES('CA001','FURNITURES');

INSERT INTO CATEGORY(C\_ID, C\_NAME) VALUES('CA002','CLOTHING');

INSERT INTO CATEGORY(C\_ID, C\_NAME) VALUES('CA003','SPORTS');

INSERT INTO CATEGORY(C\_ID, C\_NAME) VALUES('CA004','ELECTRONICS');

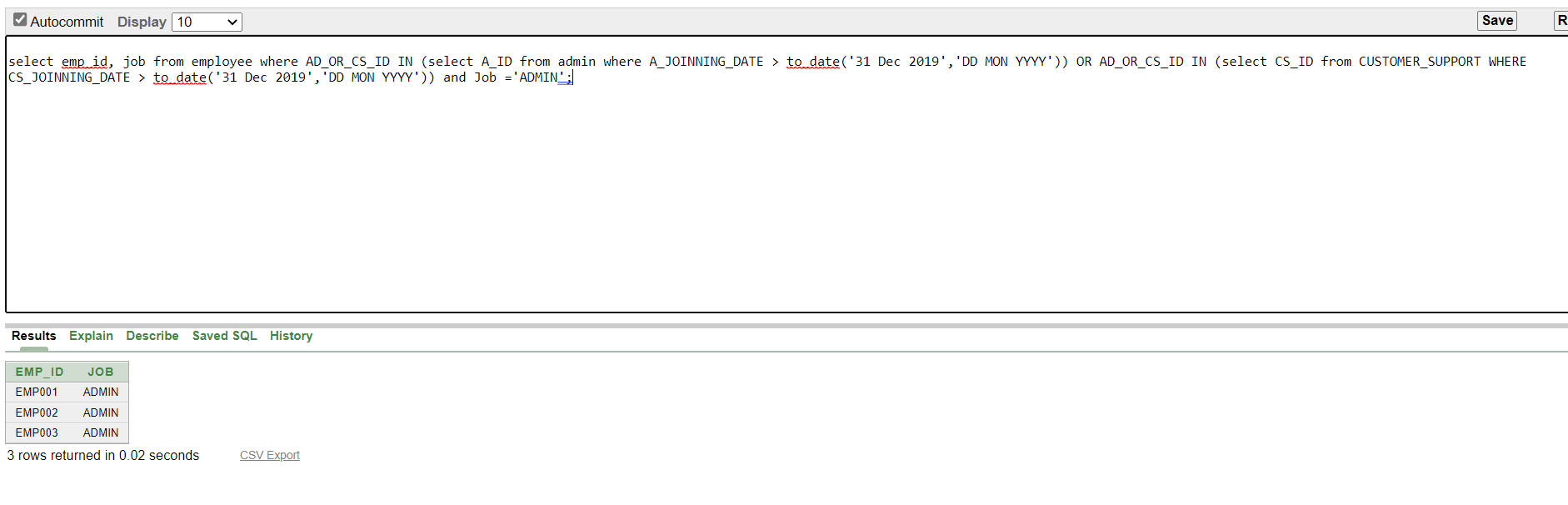
INSERT INTO CATEGORY(C\_ID, C\_NAME) VALUES('CA005','KITCHEN');



**Tasks:**

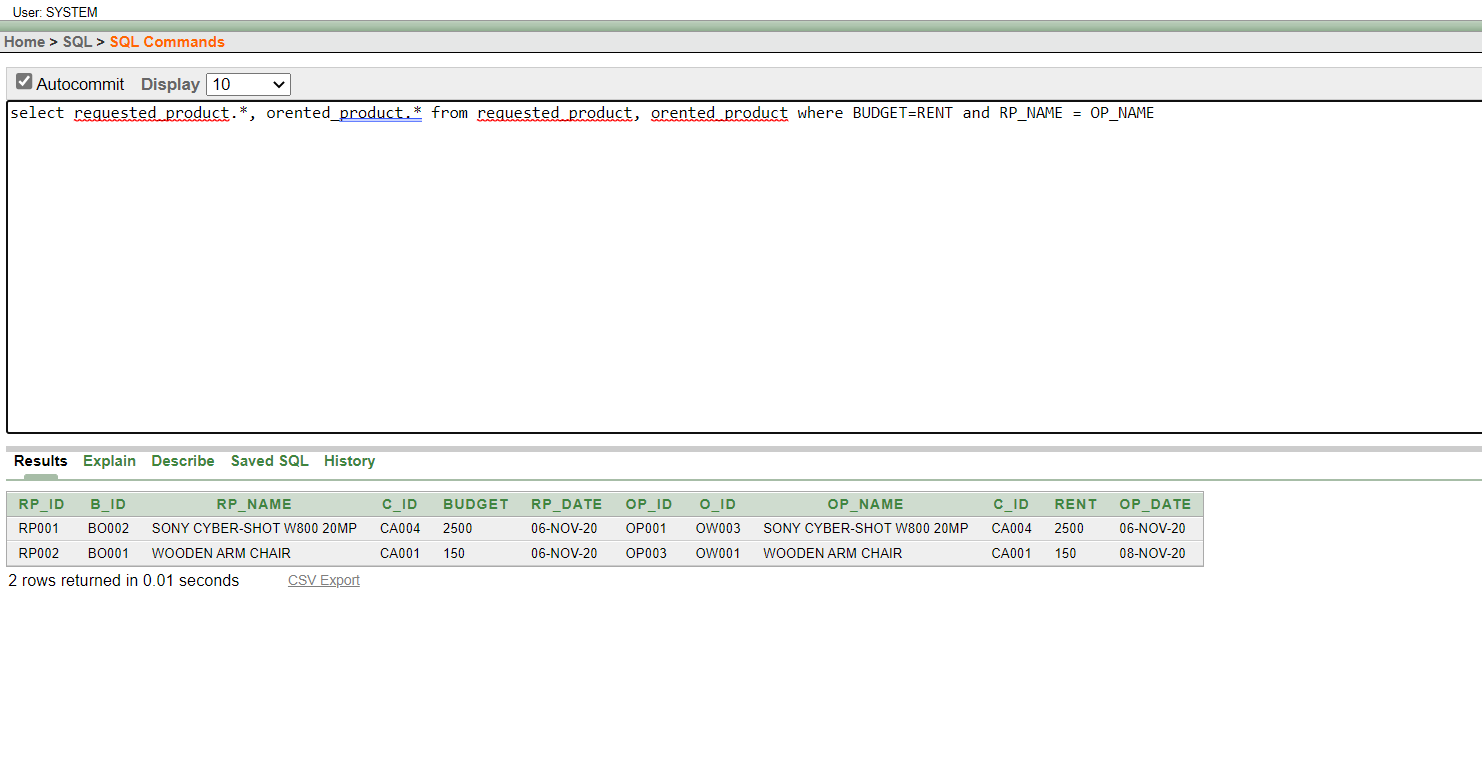
1. Find employees who joined after year 2019 and employee type is admin.

=select emp\_id, job from employee where AD\_OR\_CS\_ID IN (select A\_ID from admin where A\_JOINNING\_DATE > to\_date('31 Dec 2019','DD MON YYYY')) OR AD\_OR\_CS\_ID IN (select CS\_ID from CUSTOMER\_SUPPORT WHERE CS\_JOINNING\_DATE > to\_date('31 Dec 2019','DD MON YYYY')) and Job ='ADMIN';



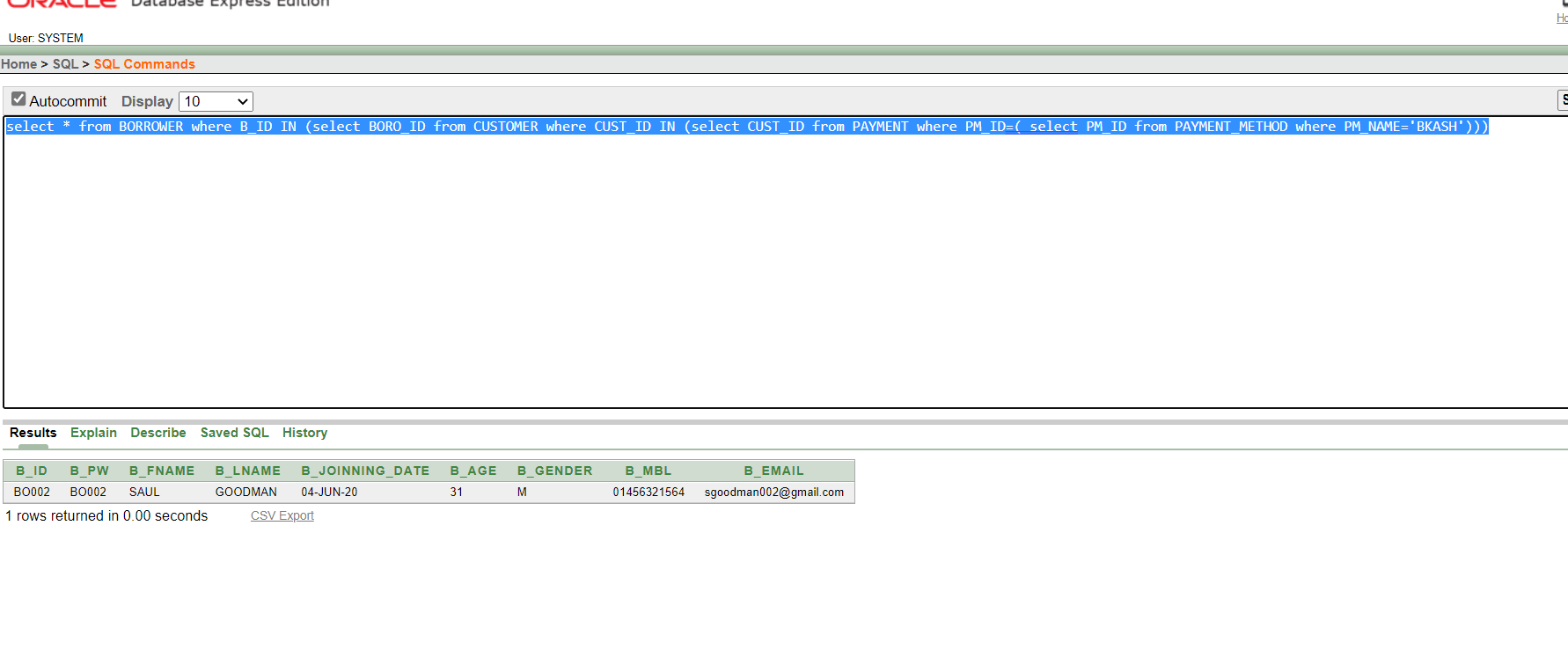
1. Find the requested and rented products which have same budget.

=select requested\_product.\*, orented\_product.\* from requested\_product, orented\_product where BUDGET=RENT and RP\_NAME = OP\_NAME;

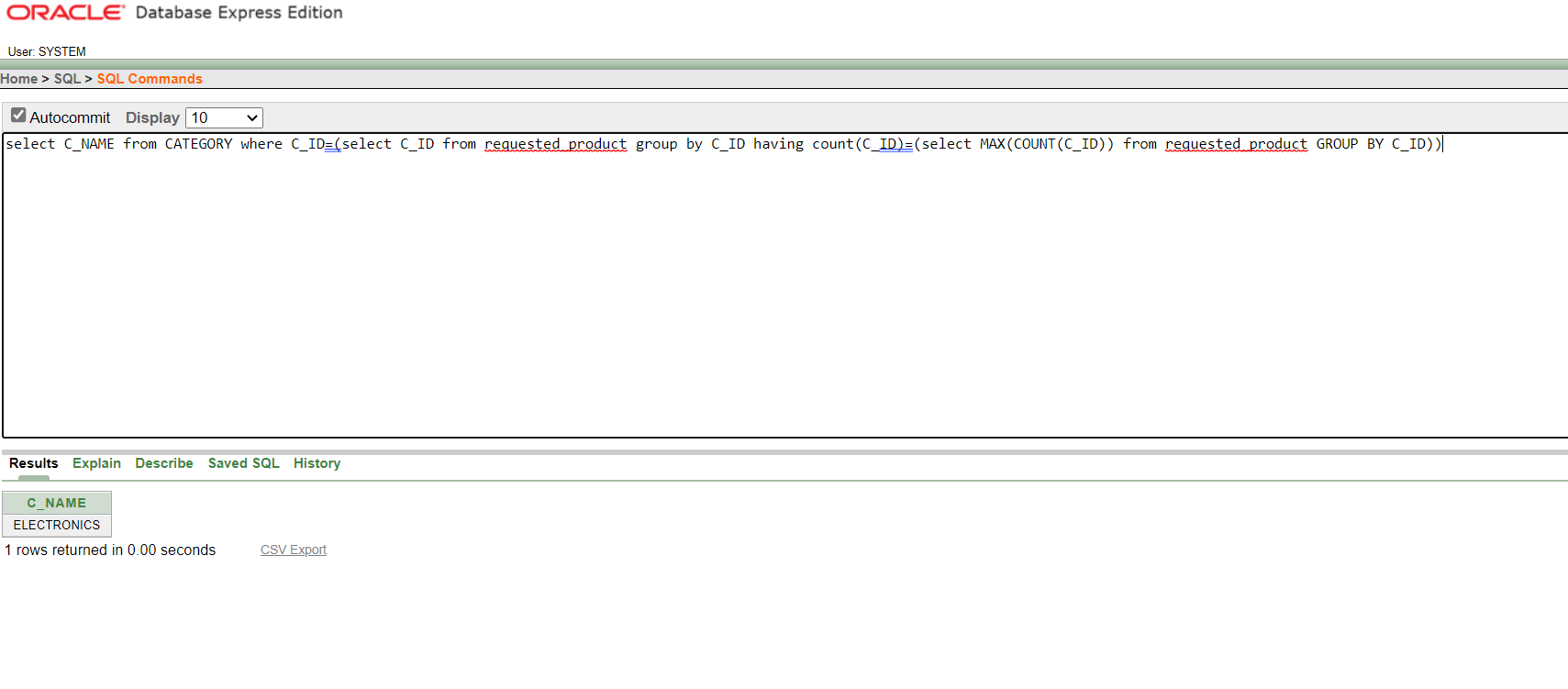


1. Find the borrowers who paid with BKASH.

= select \* from BORROWER where B\_ID IN (select BORO\_ID from CUSTOMER where CUST\_ID IN (select CUST\_ID from PAYMENT where PM\_ID=( select PM\_ID from PAYMENT\_METHOD where PM\_NAME='BKASH')))

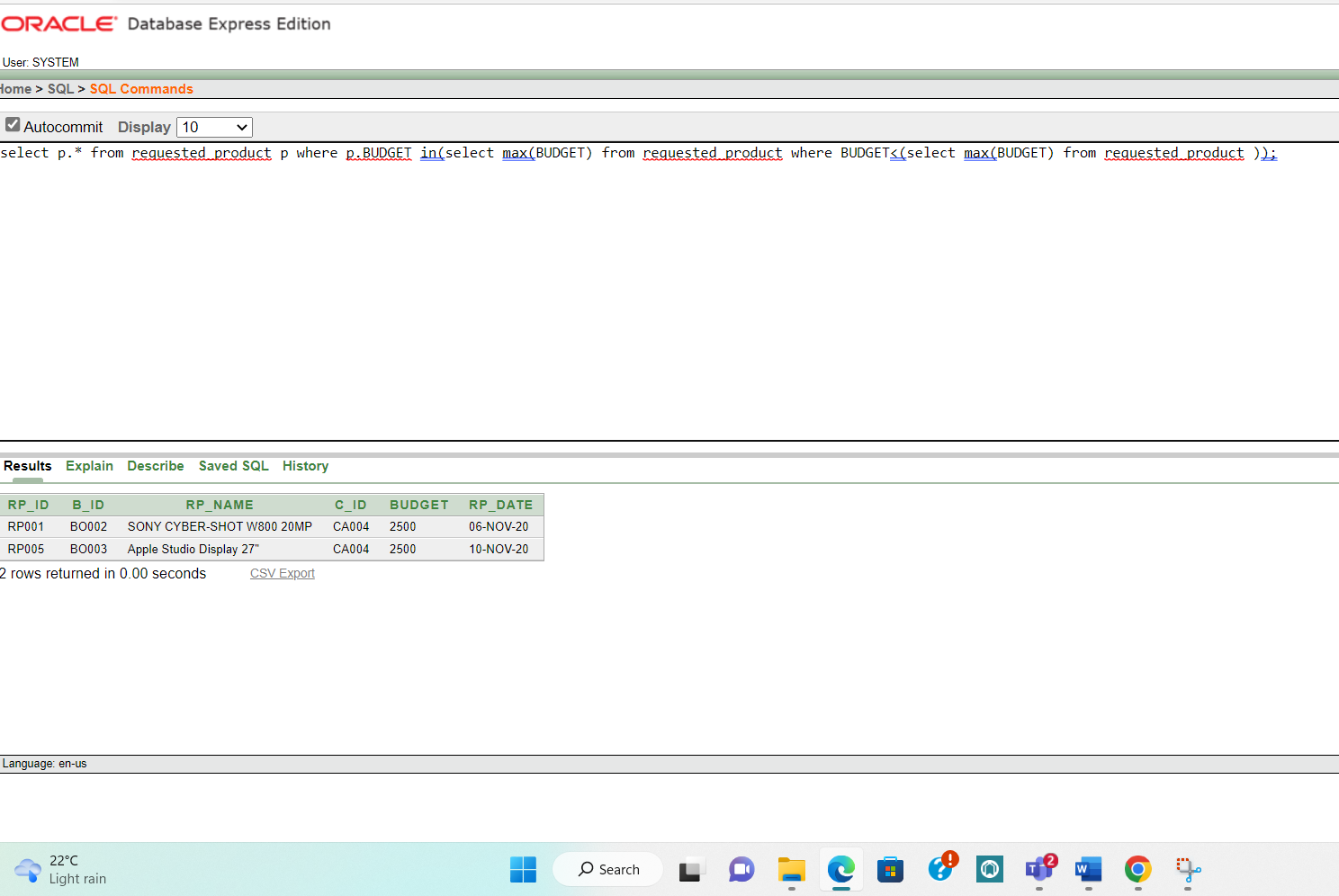


1. Find the product category that are requested most.

=select C\_NAME from CATEGORY where C\_ID=(select C\_ID from requested\_product group by C\_ID having count(C\_ID)=(select MAX(COUNT(C\_ID)) from requested\_product GROUP BY C\_ID))

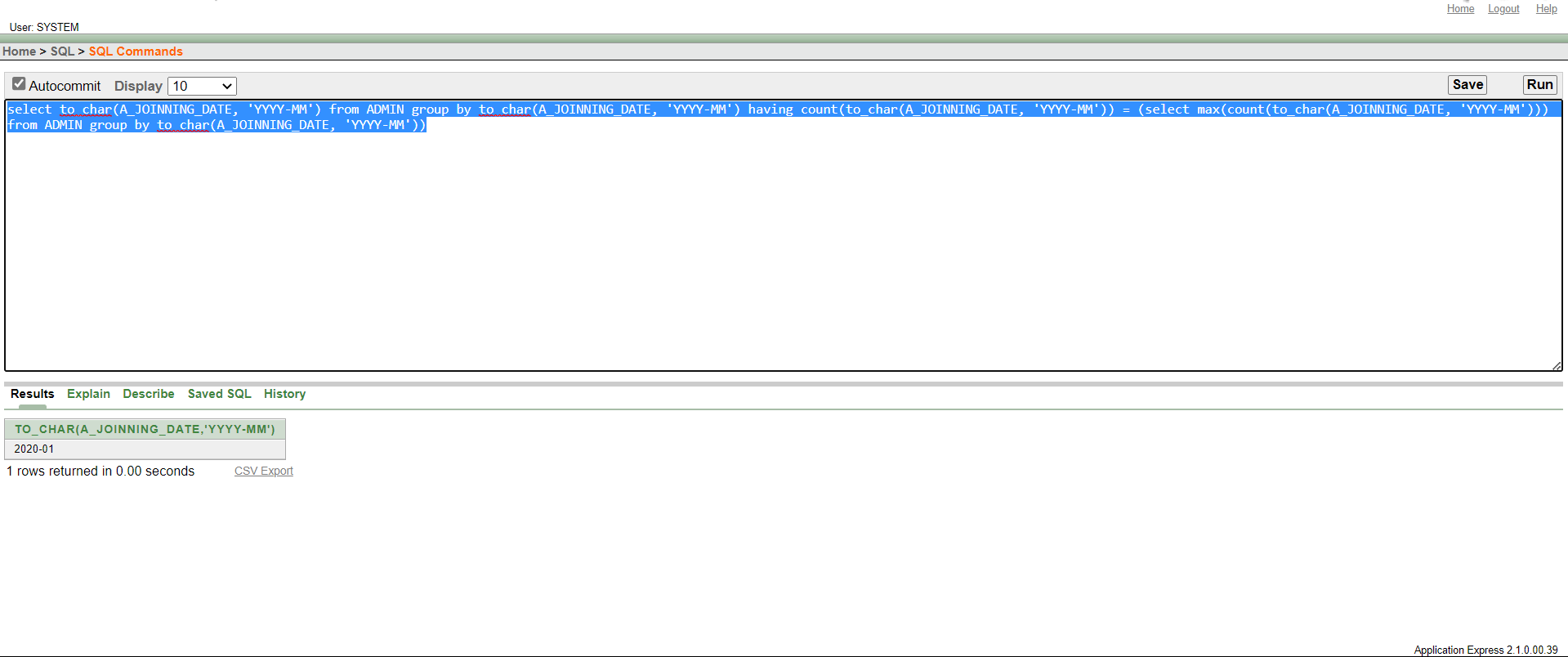
1. Find out the product of second highest price.

=select p.\* from requested\_product p where p.BUDGET in(select max(BUDGET) from requested\_product where BUDGET<(select max(BUDGET) from requested\_product ));



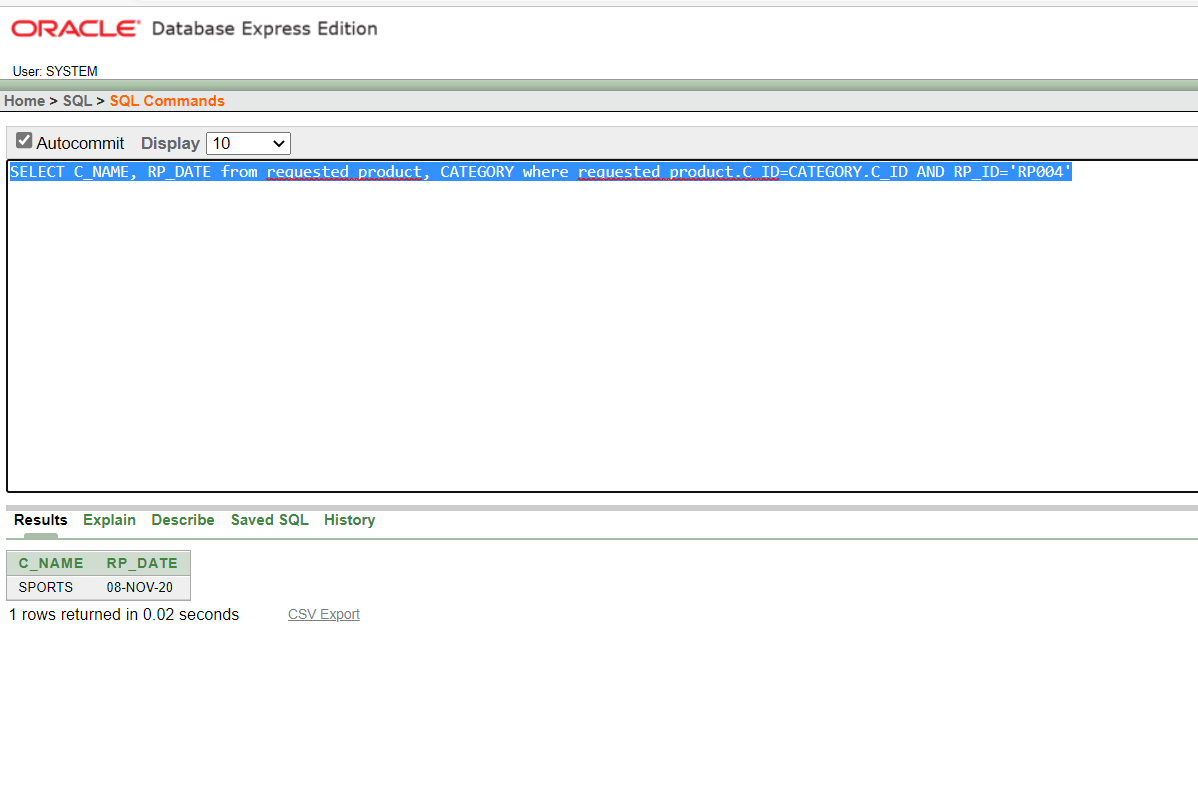
1. Find out the month in which maximum number of admin joined.

= select to\_char(A\_JOINNING\_DATE, 'YYYY-MM') from ADMIN group by to\_char(A\_JOINNING\_DATE, 'YYYY-MM') having count(to\_char(A\_JOINNING\_DATE, 'YYYY-MM')) = (select max(count(to\_char(A\_JOINNING\_DATE, 'YYYY-MM'))) from ADMIN group by to\_char(A\_JOINNING\_DATE, 'YYYY-MM'))



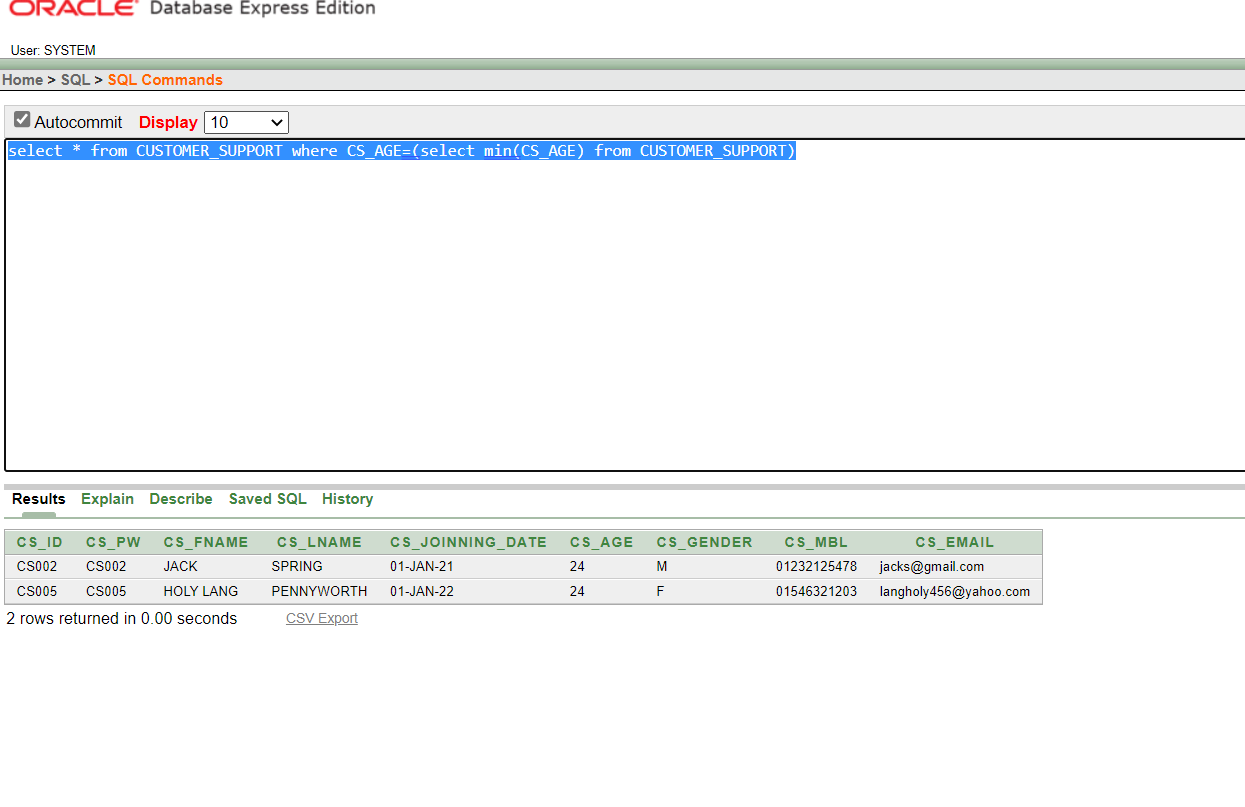
1. Find out the category and the requested date of requested product 4.

= select C\_NAME, RP\_DATE from requested\_product, CATEGORY where requested\_product.C\_ID=CATEGORY.C\_ID AND RP\_ID='RP004'



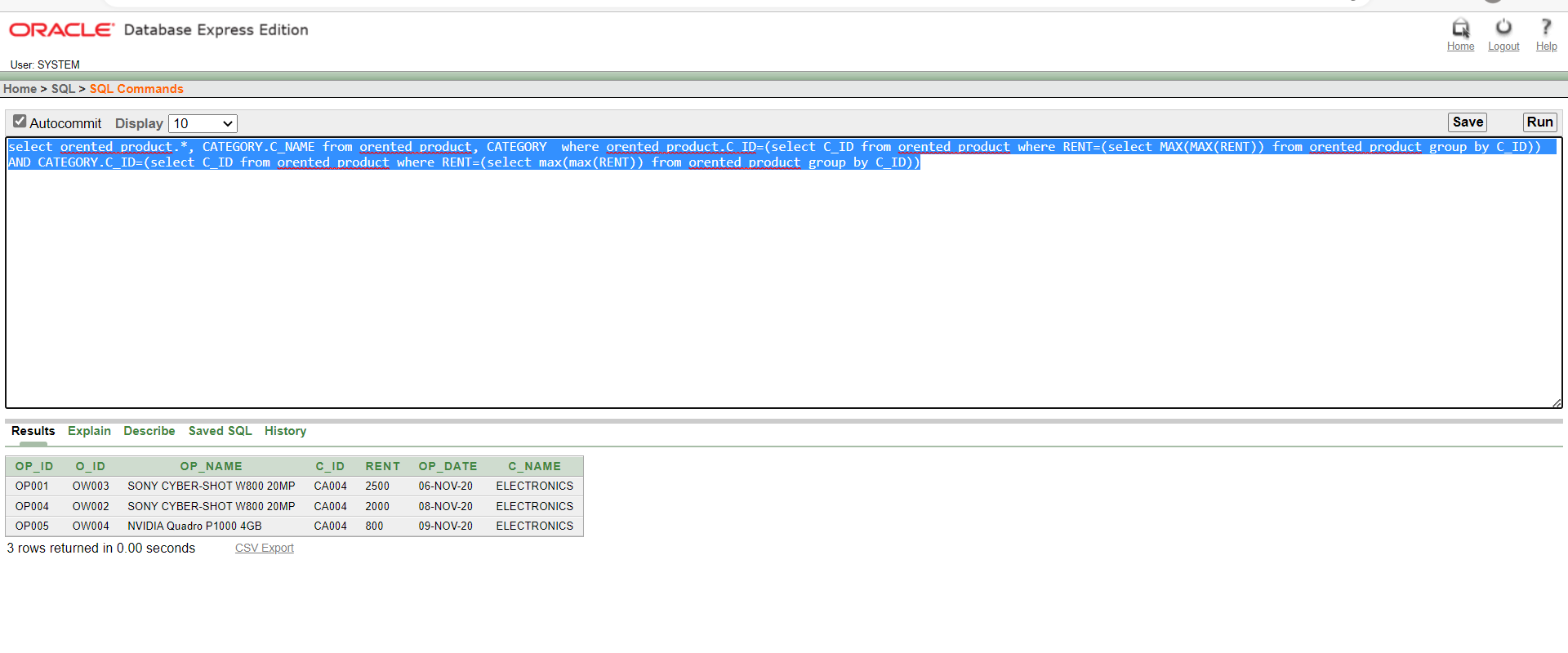
1. Find out the joinning date of most junior customer support executive.

= select \* from CUSTOMER\_SUPPORT where CS\_AGE=(select min(CS\_AGE) from CUSTOMER\_SUPPORT)



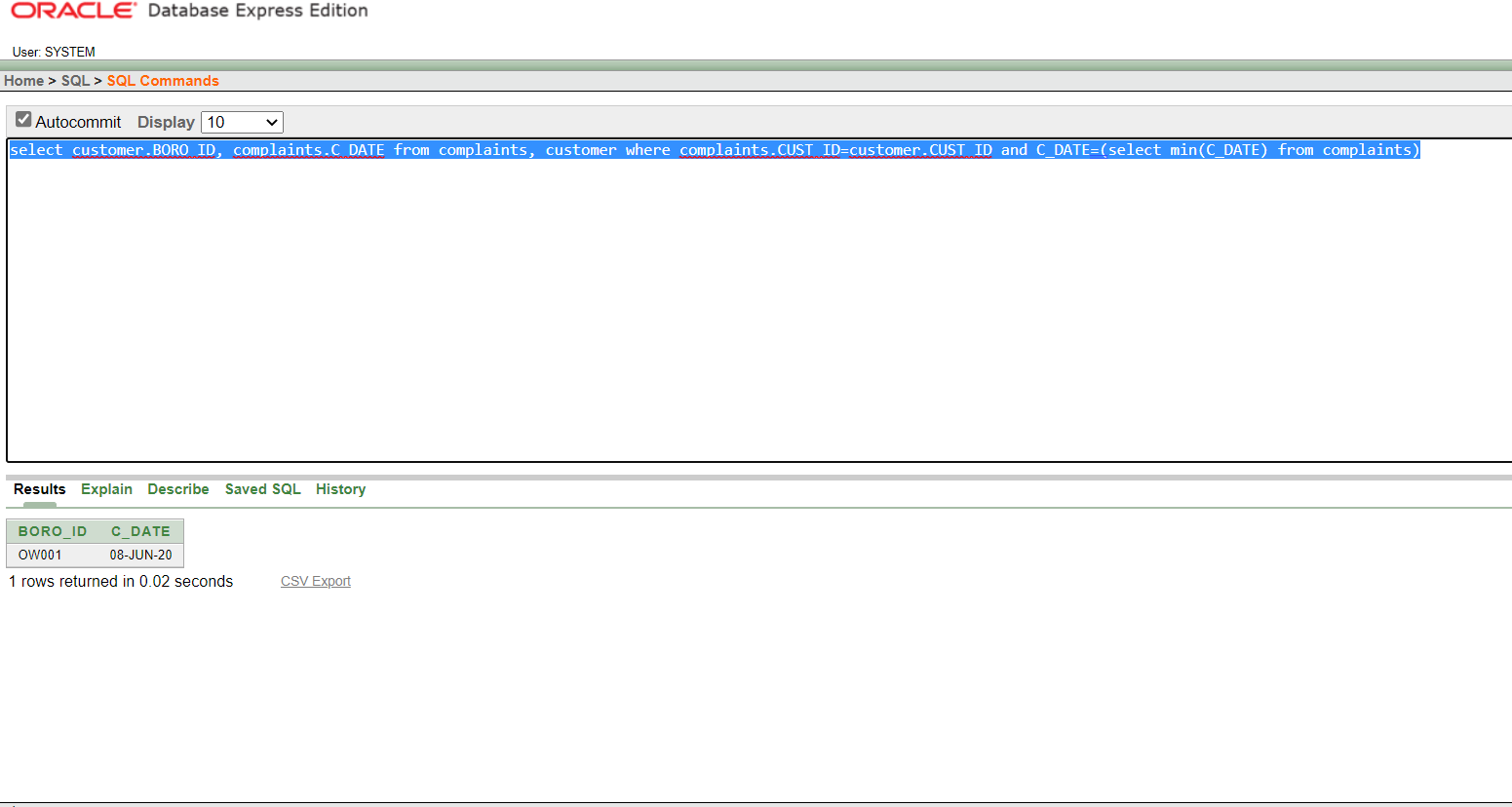
1. Find out the rented product and its category with the highest price.

=select orented\_product.\*, CATEGORY.C\_NAME from orented\_product, CATEGORY where orented\_product.C\_ID=(select C\_ID from orented\_product where RENT=(select MAX(MAX(RENT)) from orented\_product group by C\_ID)) AND CATEGORY.C\_ID=(select C\_ID from orented\_product where RENT=(select max(max(RENT)) from orented\_product group by C\_ID))



1. Find out the customer with oldest complaint.

= select customer.BORO\_ID, complaints.C\_DATE from complaints, customer where complaints.CUST\_ID=customer.CUST\_ID and C\_DATE=(select min(C\_DATE) from complaints)



**User Interface**

