**Payment Table**

* **Create Query:**

**Create table PAYMENT with attributes CUSTOMER\_ID (PK), CNAME for customer name, ADVANCE for customer paid advance amount and REMAININGS for remaining amount.**

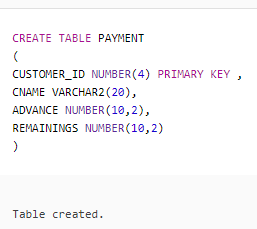
CREATE TABLE PAYMENT

(CUSTOMER\_ID NUMBER(4) PRIMARY KEY ,

CNAME VARCHAR2(20),

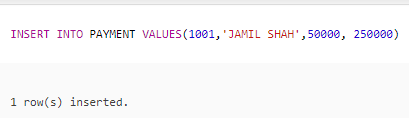
ADVANCE NUMBER(10,2),

REMAININGS NUMBER(10,2));

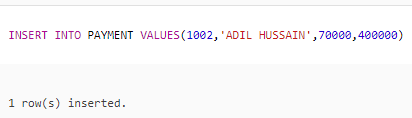


* **Insert Queries:**

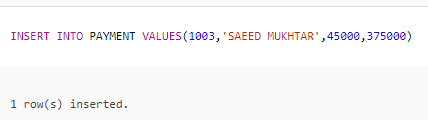
INSERT INTO PAYMENT VALUES(1001,'JAMIL SHAH',50000, 250000);



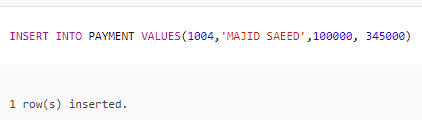
INSERT INTO PAYMENT VALUES(1002,'ADIL HUSSAIN',70000,400000);



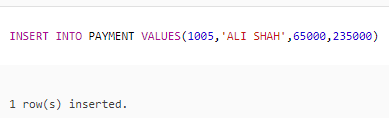
INSERT INTO PAYMENT VALUES(1003,'SAEED MUKHTAR',45000,375000);

****

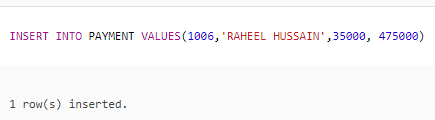
INSERT INTO PAYMENT VALUES(1004,'MAJID SAEED',100000, 345000);



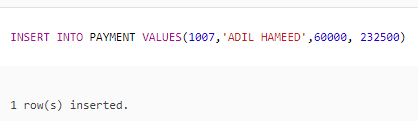
INSERT INTO PAYMENT VALUES(1005,'ALI SHAH',65000,235000);



INSERT INTO PAYMENT VALUES(1006,'RAHEEL HUSSAIN',35000, 475000);



INSERT INTO PAYMENT VALUES(1007,'ADIL HAMEED',60000, 232500);



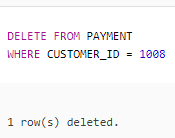
* **Delete Queries:**



**Delete the record of row where CUSTOMER\_ID is 1008.**

DELETE FROM PAYMENT

WHERE CUSTOMER\_NO = 1008;

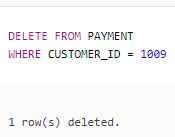




**Delete the record of row where CUSTOMER\_ID is 1009.**

DELETE FROM PAYMENT

WHERE CUSTOMER\_NO = 1009;

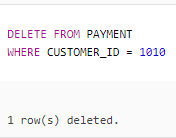




**Delete the record of row where CUSTOMER\_ID is 1010.**

DELETE FROM PAYMENT

WHERE CUSTOMER\_NO = 1010;

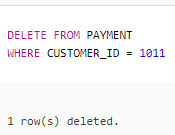




**Delete the record of row where CUSTOMER\_ID is 1011.**

DELETE FROM PAYMENT

WHERE CUSTOMER\_NO = 1011;

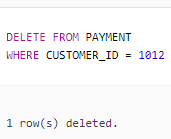




**Delete the record of row where CUSTOMER\_ID is 1012.**

DELETE FROM PAYMENT

WHERE CUSTOMER\_NO = 1012;



* **Update Queries:**

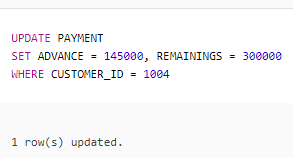


**Update the payment record of a customer having CUSTOMER\_ID 1004. The ADVANCE is 145000 and REMAININGS 300000.**

UPDATE PAYMENT

SET ADVANCE = 145000, REMAININGS = 300000

WHERE CUSTOMER\_ID = 1004;



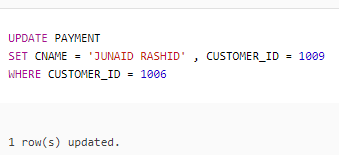


**Update the customer name “JUNAID RASHID” and customer id 1009 where customer id is 1006.**

UPDATE PAYMENT

SET CNAME = 'JUNAID RASHID', CUSTOMER\_ID = 1009

WHERE CUSTOMER\_ID = 1006;

****

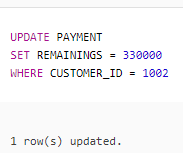


**Change the remaining from 400000 to 330000 of a customer having customer id 1002.**

UPDATE PAYMENT

SET REMAININGS = 330000

WHERE CUSTOMER\_ID = 1002;

****

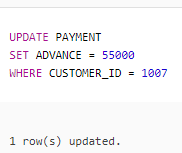


**Update the advance from 60000 to 55000 of a customer having customer id 1007.**

UPDATE PAYMENT

SET ADVANCE = 55000

WHERE CUSTOMER\_ID = 1007;

****

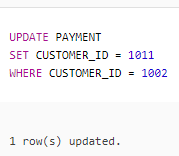


**Change the customer id as 1011 where customer id is 1002.**

UPDATE PAYMENT

SET CUSTOMER\_ID = 1011

WHERE CUSTOMER\_ID = 1002;



* **Other Queries:**

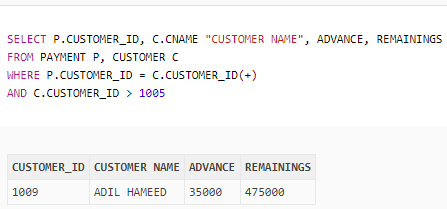
**Display customer id, customer name as “CUSTOMER NAME”, advance and remainings of all those customers who have id greater than 1005. Apply outer join with customer id of customer table.**

SELECT P.CUSTOMER\_ID, C.CNAME “CUSTOMER NAME”, ADVANCE, REMAININGS

FROM PAYMENT P, CUSTOMER C

WHERE P.CUSTOMER\_ID = C.CUSTOMER\_ID (+)

AND C.CUSTOMER\_ID > 1005;

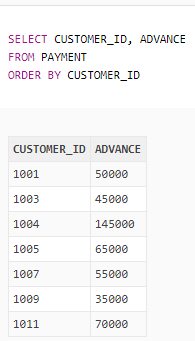
****

**Give customer id and advance in ascending order of customer id.**

SELECT CUSTOMER\_ID, ADVANCE

FROM PAYMENT

ORDER BY CUSTOMER\_ID;

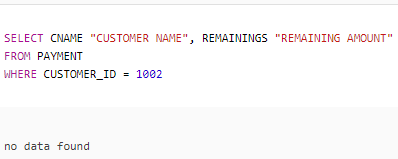
****

**Give customer name as “CUSTOMER NAME”, remainings as “REMAINING AMOUNT” of all those customers who have id 1002.**

SELECT CNAME “CUSTOMER NAME”, REMAININGS “REMAINING AMOUNT”

FROM PAYMENT

WHERE CUSTOMER\_ID = 1002;

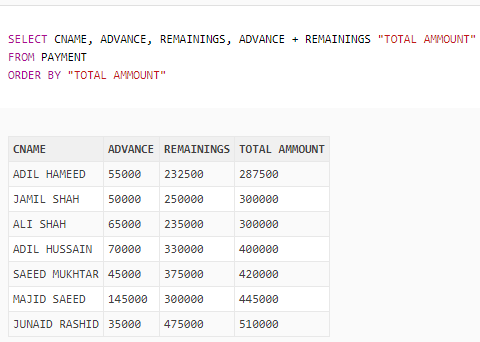
****

**Give customer name, advance, remainings and sum of advance and remainings as “TOTAL AMOUNT”. Display output in order of TOTAL AMOUNT.**

SELECT CNAME, ADVANCE, REMAININGS, ADVANCE + REMAININGS “TOTAL AMOUNT”

FROM PAYMENT

ORDER BY “TOTAL AMOUNT”;

****

**Give customer id and remainings of all those customers whose customer id is not 1002, 1003 or 1011 and have double E in their names and remainings between 300000 and 500000.**

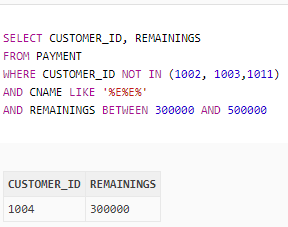
SELECT CUSTOMER\_ID, REMAINGS

FROM PAYMENT

WHERE CUSTOMER\_ID NOT IN (1002, 1003, 1011)

AND CNAME LIKE ‘%E%E%’

AND REMAININGS BETWEEN 300000 AND 500000;

****

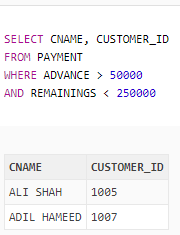
**Show customer name and customer id of all those customers who have advance greater than 50000 and remainings are less than 250000.**

SELECT CNAME, CUSTOMER\_ID

FROM PAYMENT

WHERE ADVANCE > 50000

AND REMAINGINS < 250000;

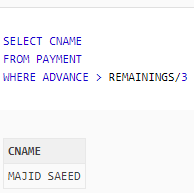
****

**Show name of all those customers whose advance is greater than one third of remainings.**

SELECT CNAME

FROM PAYMENT

WHERE ADVANCE > REMAINGS / 3;

****

**Show booking date, advance and remainings of all those customers who booked AL-JANNAT, MUGHAL-E-AZAM or LAHORE CASTLE and booking date is after 20-JAN-2018 and have 1 in their contact number.**

SELECT BOOKING\_DATE, ADVANCE, REMAININGS

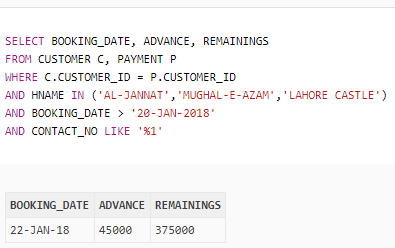
FROM CUSTOMER C, PAYMENT P

WHERE C.CUSTOMER\_ID = P.CUSTOMER\_ID

AND HNAME IN (‘AL-JANNAT’, ‘MUGHAL-E-AZAM’, ‘LAHORE CASTLE’)

AND BOOKING\_DATE > ’20-JAN-2018’

AND CONTACT\_NO LIKE ‘%1’;

****

**Give customer name, nic number and contact number of all those customers whose advance is greater than 70000 and booking date is not 19-JAN-2018.**

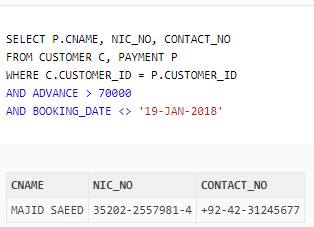
SELECT P.CNAME, NIC\_NO, CONTACT\_NO

FROM CUSTOMER C, PAYMENT P

WHERE C.CUSTOMER\_ID = P.CUSTOMER\_ID

AND ADVANCE > 70000

AND BOOKING\_DATE <> ’19-JAN-2018’;

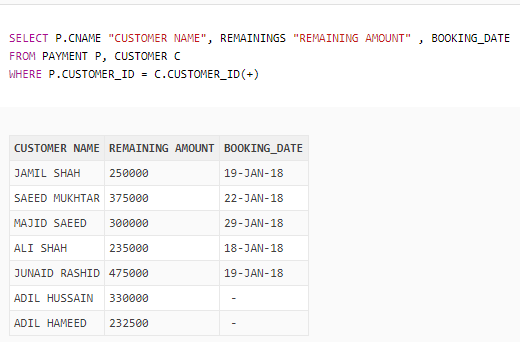
****

**Display name as “CUSTOMER NAME”, remainings as “REMAINING AMOUNT” and booking date of all customers. Apply outer join with customer id of customer table.**

SELECT P.CNAME “CUSTOMER NAME”, REMAININGS “REMAINING AMOUNT”, BOOKING\_DATE

FROM PAYMENT P, CUSTOMER C

WHERE P.CUSTOMER\_ID = C.CUSTOMER\_ID(+);

****

**Give customer id, customer name, booking date, advance and remainings of those customers whose id is greater than 1004 and booking date is before 15-JAN-2018 and advance between 50000 and 60000 and remainings is less than 3 times of advance.**

SELECT P.CUSTOMER\_ID, C.CNAME, BOOKING\_DATE, ADVANCE, REMAININGS

FROM CUSTOMER C, PAYMENT P

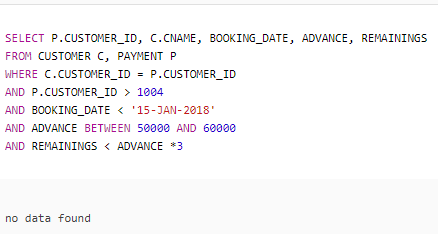
WHERE C.CUSTOMER\_ID = P.CUSTOMER\_ID

AND P.CUSTOMER\_ID > 1004

AND BOOKING\_DATE < ’15-JAN-2018’

AND ADVANCE BETWEEN 50000 AND 60000

AND REMAININGS < ADVANCE \* 3;

****

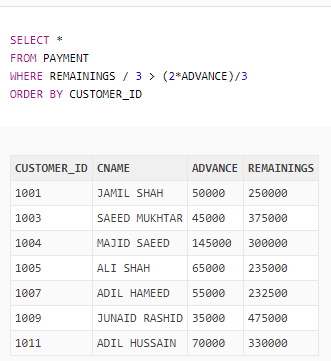
**Show all record of those customers whose one third remainings is greater than two third of advance payment. Display output in ascending order of customer id.**

SELECT \*

FROM PAYMENT

WHERE REMAININGS / 3 > (2 \* ADVANCE) / 3

OREDER BY CUSTOMER\_ID;

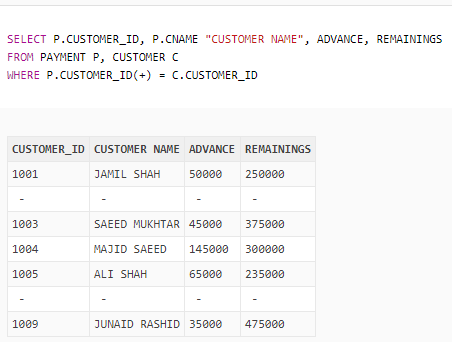
****

**Show customer id, customer name as “CUSTOMER NAME”, advance and remainings of all customers. Apply outer join with customer id of payment table.**

SELECT P.CUSTOMER\_ID, P.CNAME “CUSTOMER NAME”, ADVANCE, REMAININGS

FROM PAYMENT P, CUSTOMER C

WHERE P.CUSTOMER\_ID (+) = C.CUSTOMER\_ID;

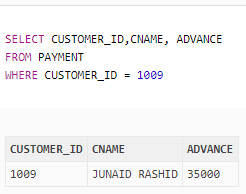
****

**Show customer id, customer name and advance of those customers who have id 1009.**

SELECT CUSTOMER\_ID, CNAME, ADVANCE

FROM PAYMENT

WHERE CUSTOMER\_ID = 1009;

****

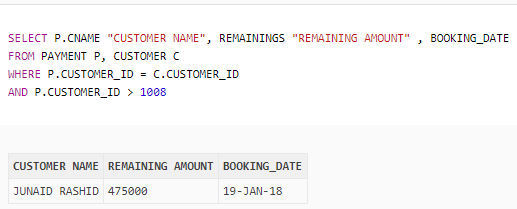
**Display name as “CUSTOMER NAME”, remainings as “REMAINING AMOUNT” and booking date of those customers whose id is greater than 1008.**

SELECT P.CNAME “CUSTOMER NAME”, REMAININGS “REMAINING AMOUNT”, BOOKING\_DATE

FROM PAYMENT P, CUSTOMER C

WHERE P.CUSTOMER\_ID = C.CUSTOMER\_ID

AND P.CUSTOMER\_ID > 1008;

****

* **Function Queries:**

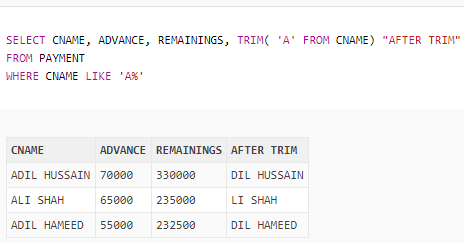


**Show customer name, advance, remainings and trim A from customer name of those customers whose name start with A.**

SELECT CNAME, ADVANCE, REMAININGS, TRIM (‘A’ FROM CNAME) “AFTER TRIM”

FROM PAYMENT

WHERE CNAME LIKE ‘A%’;

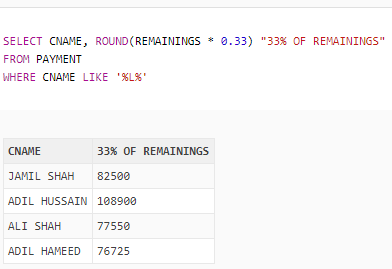


**Show customer name and 33 percent of remainings as “33% OF REMAININGS” rounded of those customers who have L in their names.**

SELECT CNAME, ROUND (REMAININGS \* 0.33) “33% OF REMAININGS”

FROM PAYMENT

WHERE CNAME LIKE ‘%L%’;

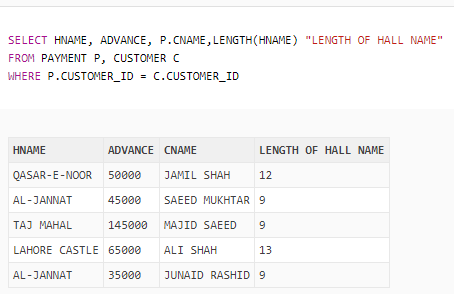


**Display hall name, advance, customer name and hall name length as “LENGTH OF HALL NAME” where customer id of PAYMENT table is equal to CUSTOMER table.**

SELECT HNAME, ADVANCE, P.CNAME, LENGTH (HNAME) “LENGTH OF HALL NAME”

FROM PAYMENT P, CUSTOMER C

WHERE P.CUSTOMER\_ID = C.CUSTOMER\_ID;



* **Sub-Queries:**

**Show name, advance and remainings of all those customers whose advance is less than the advance of those customer whose id is 1001.**

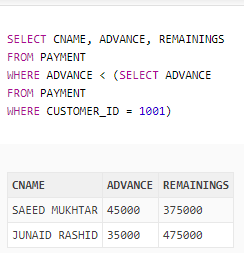
SELECT CNAME, ADVANCE, REMAININGS

FROM PAYMENT

WHERE ADVANCE < (SELECT ADVANCE

FROM PAYMENT

WHERE CUSTOMER\_ID = 1001);



**Display id, advance and remainings of all those customers whose advance is not equal to the advance of those customers whose id is 1007 and remainings less than the remainings of JAMIL SHAH.**

SELECT CUSTOMER\_ID, ADVANCE, REMAININGS

FROM PAYMENT

WHERE ADVANCE <> (SELECT ADVANCE

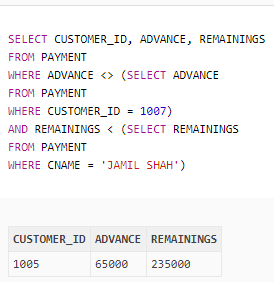
FROM PAYMENT

WHERE CUSTOMER\_ID = 1007)

AND REMAININGS < (SELECT REMAININGS

FROM PAYMENT

WHERE CNAME = ‘JAMIL SHAH’);



**Give name of all those customers whose remainings are greater than the remainings of SAEED MUKHTAR.**

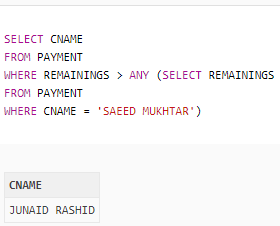
SELECT CNAME

FROM PAYMENT

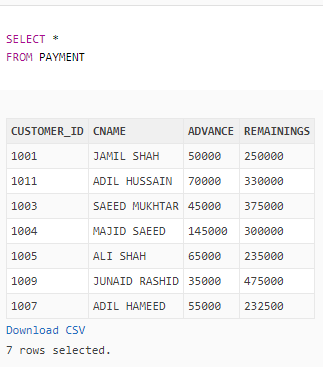
WHERE REMAININGS > ANY (SELECT REMAININGS

FROM PAYMENT

WHERE CNAME = ‘SAEED MUKHTAR’);

****

**Table**

****