CREATE TABLE IF NOT EXISTS COURSES

(

CCODE VARCHAR(10),

CNAME VARCHAR(50) NOT NULL ,

COURSEFEE NUMERIC(6 ,2) NOT NULL ,

DURATION NUMERIC(3) NOT NULL ,

PREREQ VARCHAR(100),

CONSTRAINT ccode\_pk PRIMARY KEY (CCODE)

);

CREATE TABLE IF NOT EXISTS BATCHES

(

BATCHCODE VARCHAR(12) CHECK ( BATCHCODE REGEXP ('^[A-Z]{2,6}[0-9]{6}$')),

CCODE VARCHAR(10),

STDATE DATE ,

ENDDATE DATE ,

TIMINGS VARCHAR(20),

CONSTRAINT ccode\_fk FOREIGN KEY (CCODE) REFERENCES COURSES(CCODE),

CONSTRAINT bcode\_pk PRIMARY KEY (BATCHCODE),

CONSTRAINT endtime\_check CHECK ( STDATE < ENDDATE )

);

CREATE TABLE IF NOT EXISTS STUDENTS

(

ADMNO NUMERIC(5),

BATCHCODE VARCHAR(12),

ROLLNO NUMERIC(3) ,

FULLNAME VARCHAR(50),

FATHERNAME VARCHAR(50),

EMAIL VARCHAR(50) UNIQUE,

PHONENO VARCHAR(20) UNIQUE CHECK ( PHONENO REGEXP ('(\+[0-9]{2}-[0-9]{10})|[0-9]{10}')) ,

DJ DATE,

CONSTRAINT bcode\_fk FOREIGN KEY (BATCHCODE) REFERENCES BATCHES(BATCHCODE),

CONSTRAINT admno\_pk PRIMARY KEY (ADMNO)

);

CREATE TABLE IF NOT EXISTS PAYMENTS

(

RCPTNO NUMERIC(5),

ADMNO NUMERIC(5),

AMOUNT NUMERIC(6,2),

PAYDATE DATE,

REMARKS VARCHAR(200),

CONSTRAINT rcptno\_pk PRIMARY KEY (RCPTNO),

CONSTRAINT admno\_fk FOREIGN KEY (ADMNO) REFERENCES STUDENTS(ADMNO)

);

#COURSES

INSERT INTO COURSES VALUES

('ORACLE11G','ORACLE DATABASE 11G',2500,40,'COMPUTER KONWLEDGE'),

('JAVASE6.0','JAVA SE 6.0',2500,40,'C LANGUAGE'),

('DOTNET3.5','MICROSOFT.NET 3.5',3750,80,'C LANGUAGE & SQL');

#BATCHES

INSERT INTO BATCHES VALUES

('ORA130508','ORACLE11G','08-05-13','17-06-08','4:30 TO 6:00 PM'),

('DOTNET130508','DOTNET3.5','08-05-13','08-06-26','7:00 TO 9:00 AM'),

('ORA270608','ORACLE11G','08-06-27',NULL,'5:00 TO 6:00 PM'),

('JS270608','JAVASE6.0','08-06-27',NULL,'6:00 TO 7:00 PM');

#STUDENTS

INSERT INTO STUDENTS VALUES

(1,'ORA130508',1,'MICHEAL JORDON', 'TIM JORDON','MJORDON@YAHOO.COM', '9873737334','08-05-11'),

(2,'ORA130508',2,'TIM SLIM', 'TIM KEN','TIM@YAHOO.COM', '9833334334','08-05-11'),

(3,'DOTNET130508',1,'HUNTER JASON', 'HUNTER BOB','JHUNTER@YAHOO.COM', '34344343','08-05-11'),

(4,'JS270608',1,'JAMES GOODWILL', 'JAMES ROBERTS','JAMES@YAHOO.COM', '9989898998','08-06-26'),

(5,'JS270608',2,'KENNY PETERSON', 'KENNY JACOB','KPERERSON@GMAIL.COM', '9983373333','08-06-27'),

(6,'ORA270608',1,'GLEN JHONSON', 'GLEN HENDRICK','GLEN@GMAIL.COM', '9898398985','08-06-28'),

(7,'ORA270608',2,'BATES KATHY', 'BATES ROBERTS','KATHY@YMAIL.COM', '234423232','08-06-30');

#PAYMENTS

INSERT INTO PAYMENTS VALUES

(1,1,300,'08-05-11','REG. FEE'),

(2,2,2500,'08-05-11','TOTAL FEE'),

(3,3,1000,'08-05-11','REG. FEE'),

(4,3,2750,'08-05-12',NULL),

(5,4,300,'08-06-26','REG. FEE'),

(6,5,300,'08-06-27','REG. FEE'),

(7,4,1700,'08-06-27',NULL),

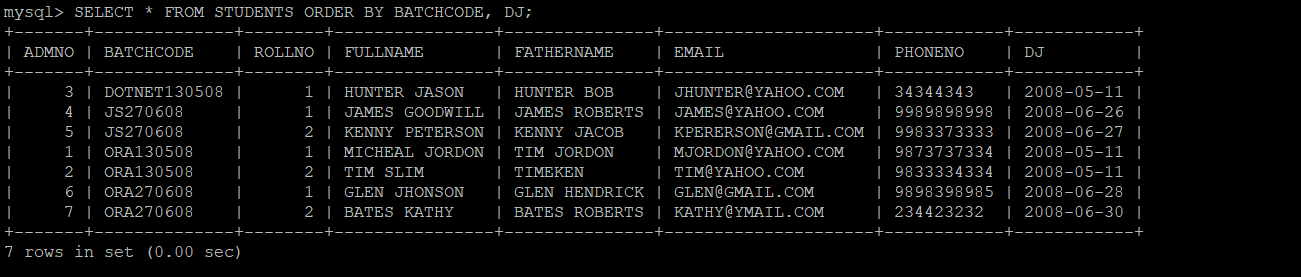
(8,5,1700,'08-06-29',NULL),

(9,6,2500,'08-06-28','CHEQUE NO:3434343 SBI DWK'),

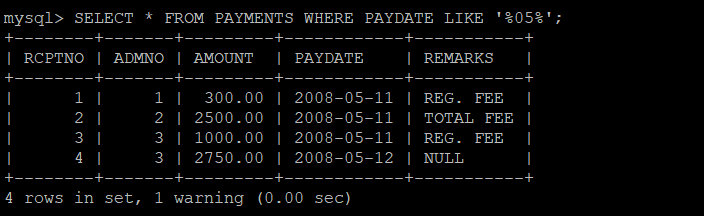
(10,7,2500,'08-06-30',NULL);

**Simple queries**

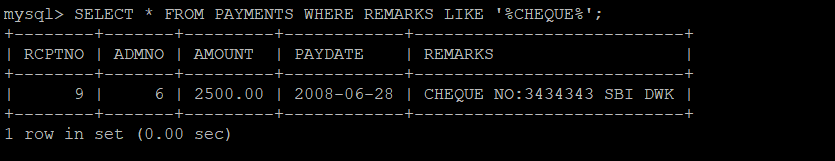
1)SELECT \* FROM STUDENTS ORDER BY BATCHCODE, DJ;



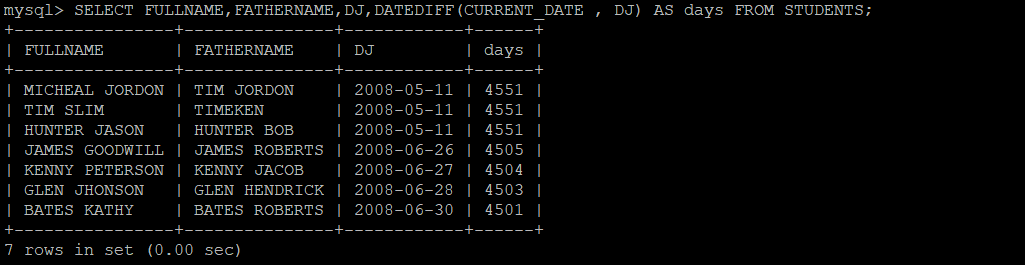
2)SELECT \* FROM PAYMENTS WHERE PAYDATE LIKE '%05%';



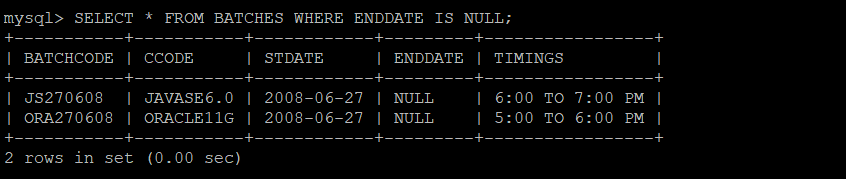
3) SELECT \* FROM PAYMENTS WHERE REMARKS LIKE '%CHEQUE%';



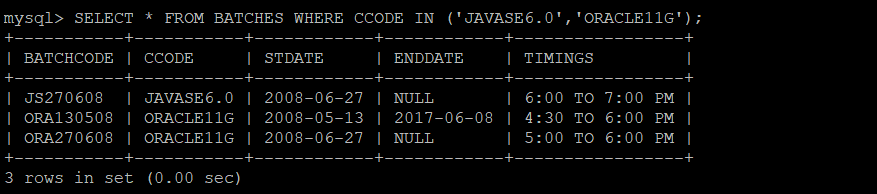
4)SELECT FULLNAME,FATHERNAME,DJ,DATEDIFF(CURRENT\_DATE , DJ) AS days FROM STUDENTS;



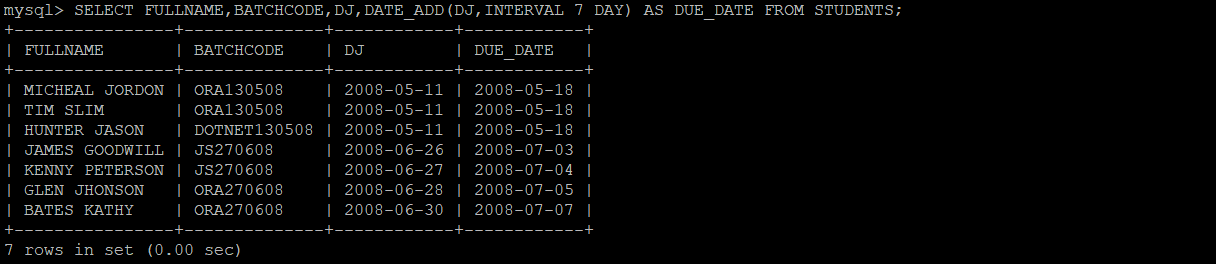
5) SELECT \* FROM BATCHES WHERE ENDDATE IS NULL;



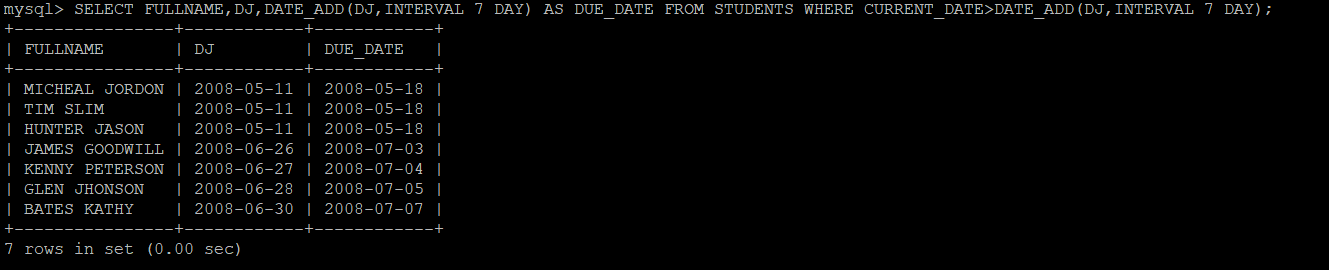
6)SELECT \* FROM BATCHES WHERE CCODE IN ('JAVASE6.0','ORACLE11G');



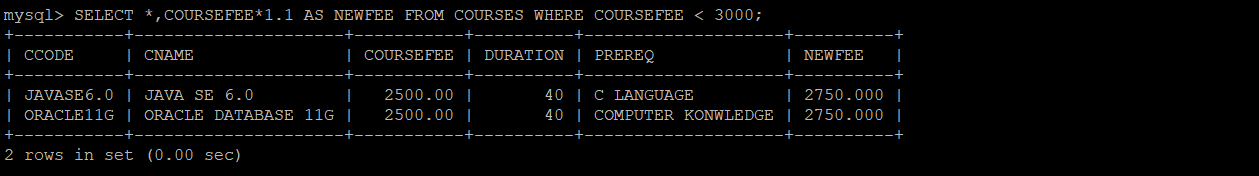
7) SELECT FULLNAME,BATCHCODE,DJ,DATE\_ADD(DJ,INTERVAL 7 DAY) AS DUE\_DATE FROM STUDENTS;



8)SELECT FULLNAME,DJ,DATE\_ADD(DJ,INTERVAL 7 DAY) AS DUE\_DATE FROM STUDENTS WHERE CURRENT\_DATE>DATE\_ADD(DJ,INTERVAL 7 DAY);



9)SELECT \*,COURSEFEE\*1.1 AS NEWFEE FROM COURSES WHERE COURSEFEE < 3000;



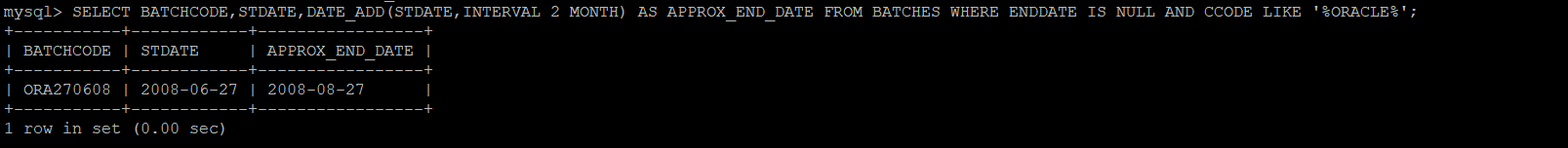
10)SELECT FULLNAME FROM STUDENTS WHERE FULLNAME LIKE '%S%' AND FATHERNAME LIKE '%P%';



11) SELECT \* FROM BATCHES WHERE DATEDIFF(STDATE,CURRENT\_DATE) > 45;



12) SELECT BATCHCODE,STDATE,DATE\_ADD(STDATE,INTERVAL 2 MONTH) AS APPROX\_END\_DATE FROM BATCHES WHERE ENDDATE IS NULL AND CCODE LIKE '%ORACLE%';



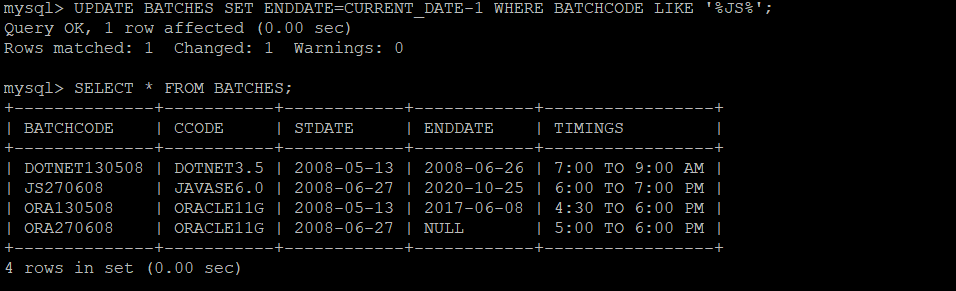
13) SELECT \*,DATEDIFF( DATE\_ADD(STDATE,INTERVAL 2 MONTH) ,ENDDATE) AS DAYS FROM BATCHES WHERE ENDDATE IS NOT NULL AND CCODE LIKE '%JAVA%';



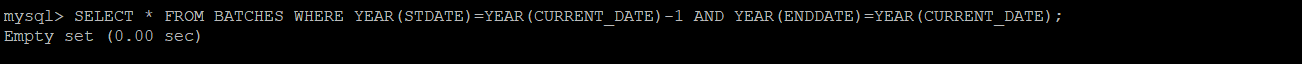
14)INSERT INTO COURSES VALUE ('JAVAEEWEB','JAVA EE(WEB APPLICATIONS)',3000,40,'JAVA LANG AND SQL');



15) UPDATE BATCHES SET ENDDATE=CURRENT\_DATE-1 WHERE BATCHCODE LIKE '%JS%';

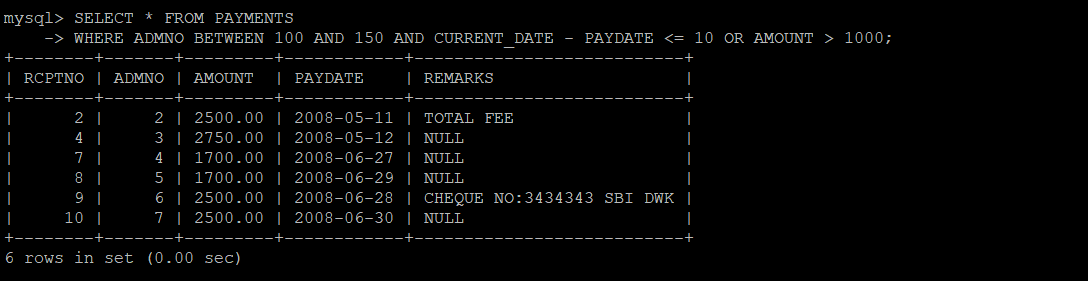


16)SELECT \* FROM BATCHES WHERE YEAR(STDATE)=YEAR(CURRENT\_DATE)-1 AND YEAR(ENDDATE)=YEAR(CURRENT\_DATE);

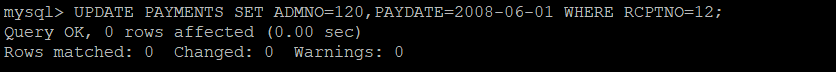


17) SELECT \* FROM PAYMENTS

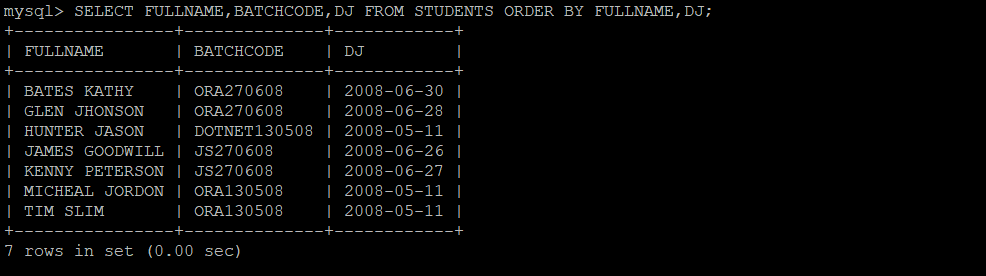
WHERE ADMNO BETWEEN 100 AND 150 AND CURRENT\_DATE - PAYDATE <= 10 OR AMOUNT > 1000;



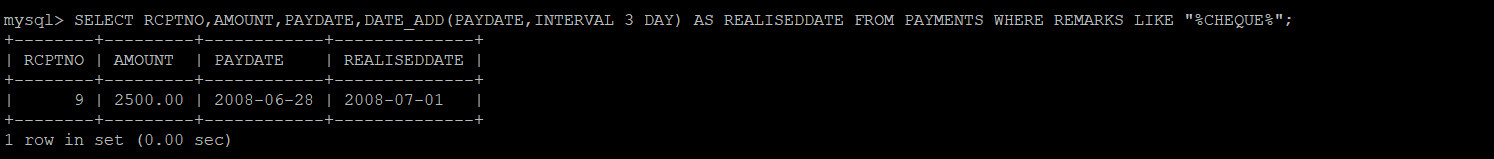
18) UPDATE PAYMENTS SET ADMNO=120,PAYDATE=2008-06-01 WHERE RCPTNO=12;



19)SELECT FULLNAME,BATCHCODE,DJ FROM STUDENTS ORDER BY FULLNAME,DJ;

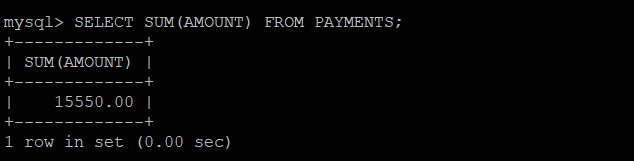


20) SELECT RCPTNO,AMOUNT,PAYDATE,DATE\_ADD(PAYDATE,INTERVAL 3 DAY) AS REALISEDDATE FROM PAYMENTS WHERE REMARKS LIKE "%CHEQUE%";

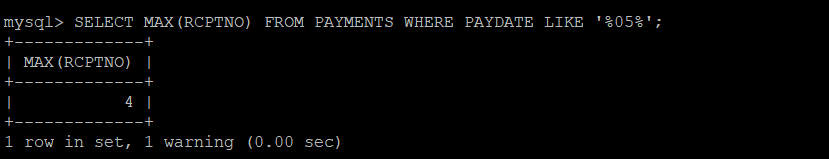


**GROUPING**

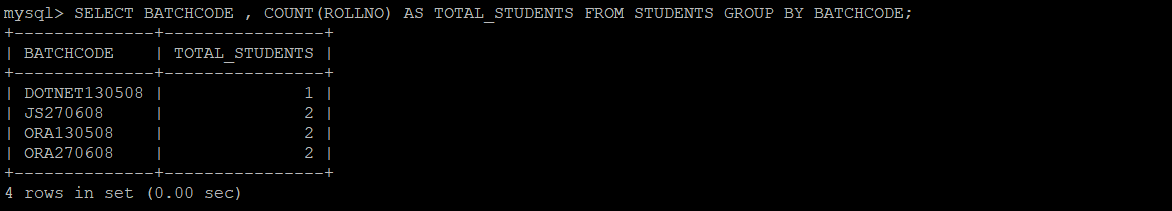
1) SELECT SUM(AMOUNT) FROM PAYMENTS;



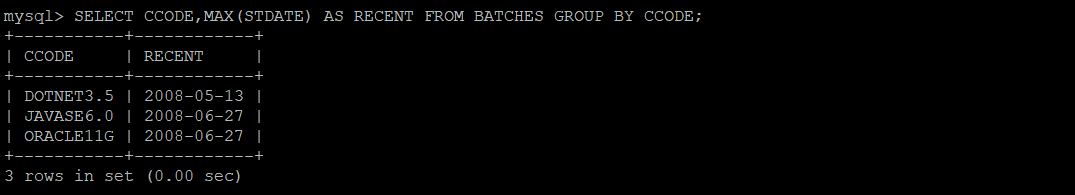
2) SELECT MAX(RCPTNO) FROM PAYMENTS WHERE PAYDATE LIKE '%05%';



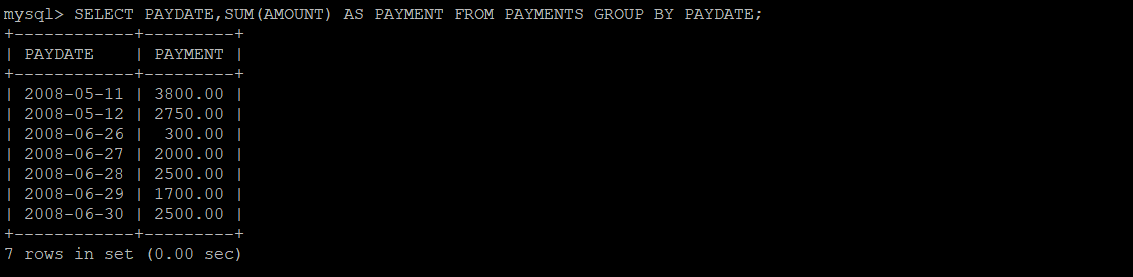
3) SELECT BATCHCODE , COUNT(ROLLNO) AS TOTAL\_STUDENTS FROM STUDENTS GROUP BY BATCHCODE;



4) SELECT CCODE,MAX(STDATE) AS RECENT FROM BATCHES GROUP BY CCODE;



5) SELECT PAYDATE,SUM(AMOUNT) AS PAYMENT FROM PAYMENTS GROUP BY PAYDATE;



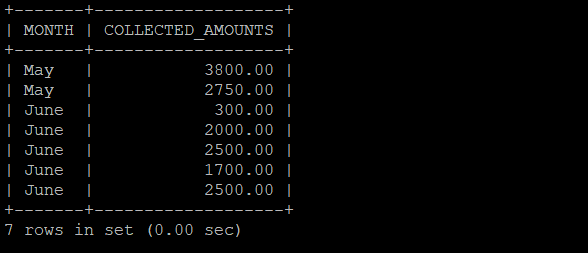
6)SELECT CCODE, COUNT(\*) NOBATCHES FROM BATCHES WHERE YEAR(STDATE) = YEAR(CURRENT\_DATE) GROUP BY CCODE;



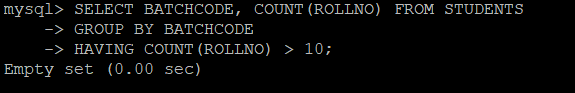
7)SELECT MONTHNAME(PAYDATE) AS MONTH,SUM(AMOUNT) AS COLLECTED\_AMOUNTS

FROM PAYMENTS

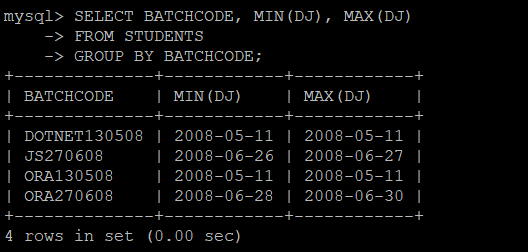
GROUP BY PAYDATE;



8)SELECT BATCHCODE, COUNT(ROLLNO) FROM STUDENTS GROUP BY BATCHCODE HAVING COUNT(ROLLNO) > 10;



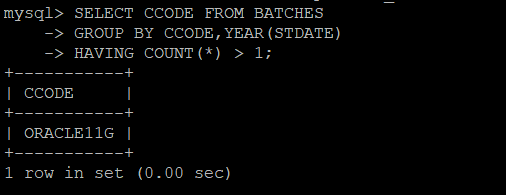
9)SELECT BATCHCODE, MIN(DJ), MAX(DJ) FROM STUDENTS GROUP BY BATCHCODE;



10)SELECT CCODE FROM BATCHES

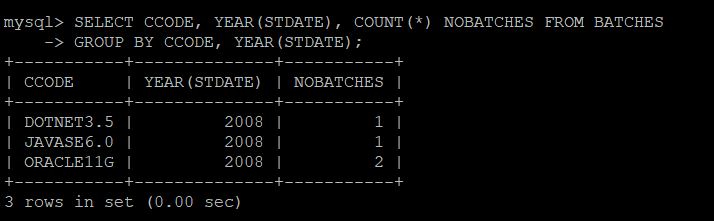
GROUP BY CCODE,YEAR(STDATE)

HAVING COUNT(\*) > 1;



11)SELECT CCODE, YEAR(STDATE), COUNT(\*) NOBATCHES FROM BATCHES

GROUP BY CCODE, YEAR(STDATE);



12)SELECT CCODE, YEAR(STDATE), COUNT(\*) NOBATCHES FROM BATCHES

GROUP BY ROLLUP(CCODE, YEAR(STDATE));

SELECT CCODE, YEAR(STDATE), COUNT(\*) NOBATCHES FROM BATCHES

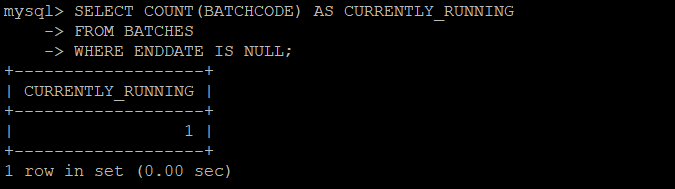
GROUP BY CUBE(CCODE, YEAR(STDATE))

ORDER BY CCODE,YEAR(STDATE);

13)SELECT COUNT(BATCHCODE) AS CURRENTLY\_RUNNING

FROM BATCHES

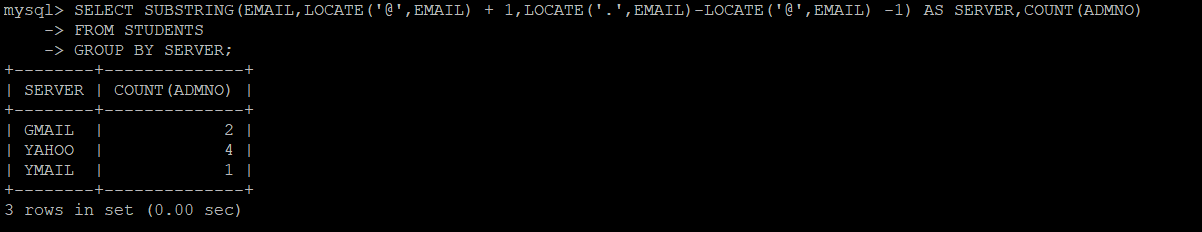
WHERE ENDDATE IS NULL;



14)SELECT SUBSTRING(EMAIL,LOCATE('@',EMAIL) + 1,LOCATE('.',EMAIL)-LOCATE('@',EMAIL) -1) AS SERVER,COUNT(ADMNO)

FROM STUDENTS

GROUP BY SERVER;



15)SELECT S.BATCHCODE,COUNT(S.ADMNO) AS STUDENTS

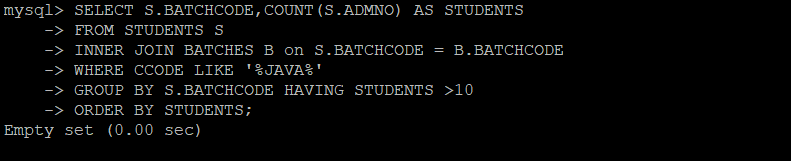
FROM STUDENTS S

INNER JOIN BATCHES B on S.BATCHCODE = B.BATCHCODE

WHERE CCODE LIKE '%JAVA%'

GROUP BY S.BATCHCODE HAVING STUDENTS >10

ORDER BY STUDENTS;

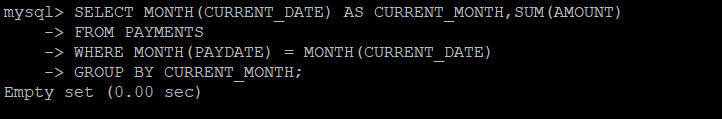


16)SELECT MONTH(CURRENT\_DATE) AS CURRENT\_MONTH,SUM(AMOUNT)

FROM PAYMENTS

WHERE MONTH(PAYDATE) = MONTH(CURRENT\_DATE)

GROUP BY CURRENT\_MONTH;

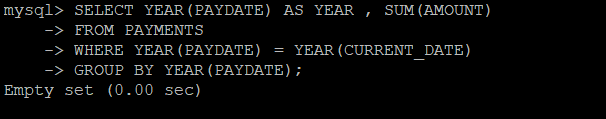


17)SELECT YEAR(PAYDATE) AS YEAR , SUM(AMOUNT)

FROM PAYMENTS

WHERE YEAR(PAYDATE) = YEAR(CURRENT\_DATE)

GROUP BY YEAR(PAYDATE);



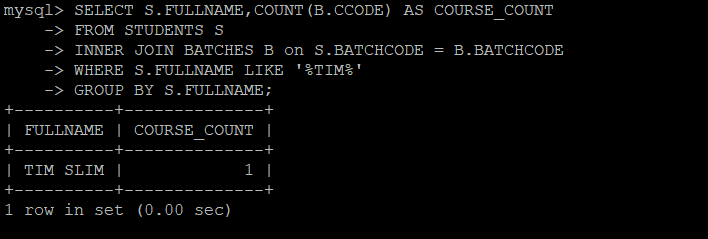
18)SELECT S.FULLNAME,COUNT(B.CCODE) AS COURSE\_COUNT

FROM STUDENTS S

INNER JOIN BATCHES B on S.BATCHCODE = B.BATCHCODE

WHERE S.FULLNAME LIKE '%TIM%'

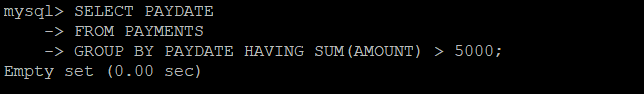
GROUP BY S.FULLNAME;



19)SELECT PAYDATE

FROM PAYMENTS

GROUP BY PAYDATE HAVING SUM(AMOUNT) > 5000;

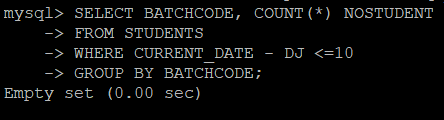


20)SELECT BATCHCODE, COUNT(\*) NOSTUDENT

FROM STUDENTS

WHERE CURRENT\_DATE - DJ <=10

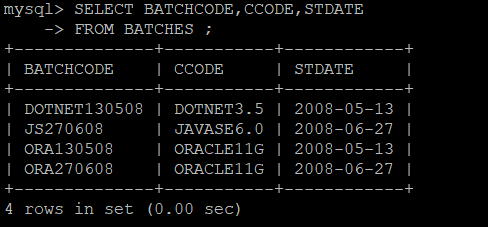
GROUP BY BATCHCODE;



**JOINING**

1)SELECT BATCHCODE,CCODE,STDATE

FROM BATCHES ;

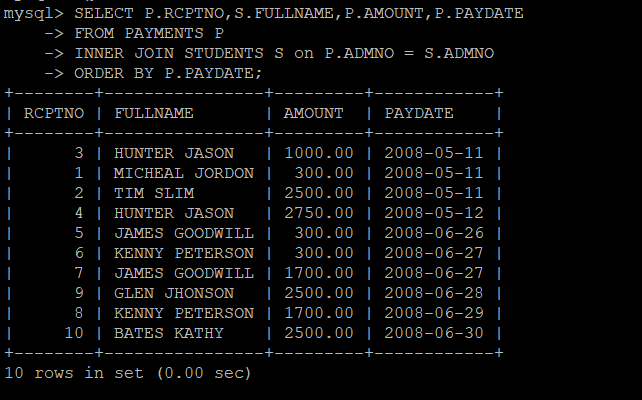


2)SELECT P.RCPTNO,S.FULLNAME,P.AMOUNT,P.PAYDATE

FROM PAYMENTS P

INNER JOIN STUDENTS S on P.ADMNO = S.ADMNO

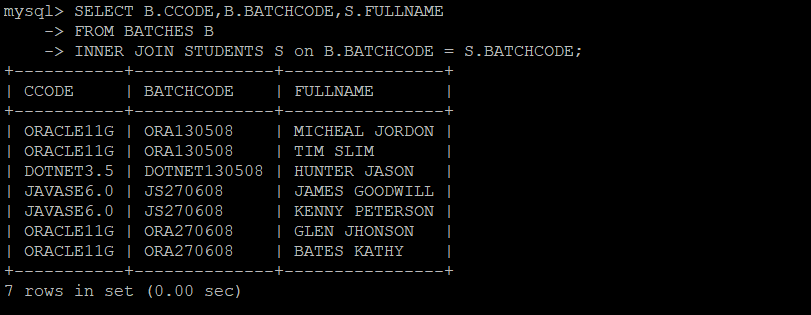
ORDER BY P.PAYDATE;



3)SELECT B.CCODE,B.BATCHCODE,S.FULLNAME

FROM BATCHES B

INNER JOIN STUDENTS S on B.BATCHCODE = S.BATCHCODE;

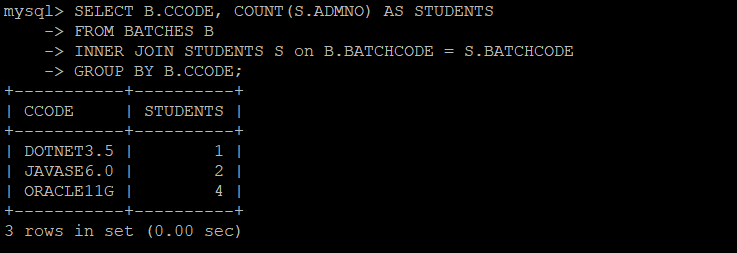


4)SELECT B.CCODE, COUNT(S.ADMNO) AS STUDENTS

FROM BATCHES B

INNER JOIN STUDENTS S on B.BATCHCODE = S.BATCHCODE

GROUP BY B.CCODE;



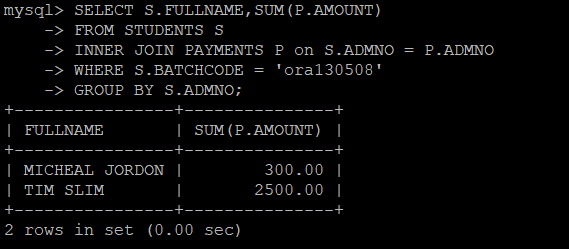
5)SELECT S.FULLNAME,SUM(P.AMOUNT)

FROM STUDENTS S

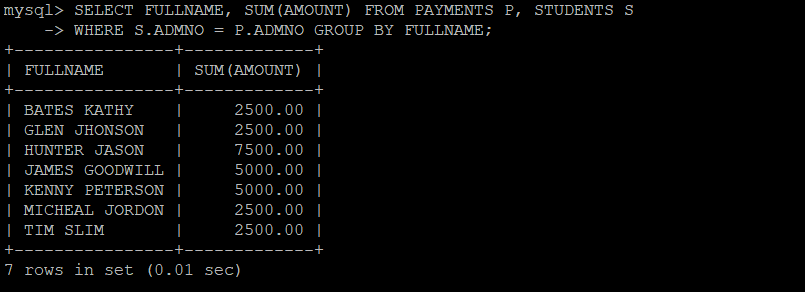
INNER JOIN PAYMENTS P on S.ADMNO = P.ADMNO

WHERE S.BATCHCODE = 'ora130508'

GROUP BY S.ADMNO;



6)SELECT FULLNAME, SUM(AMOUNT) FROM PAYMENTS P, STUDENTS S WHERE S.ADMNO = P.ADMNO GROUP BY FULLNAME;

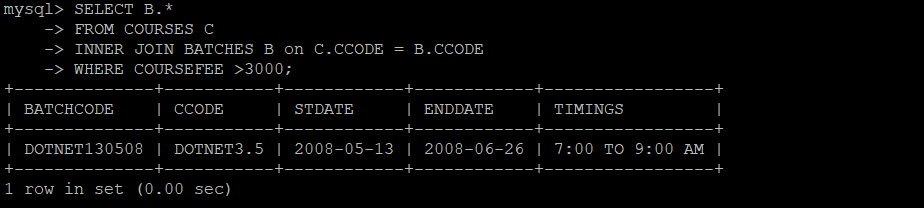


7)SELECT B.\*

FROM COURSES C

INNER JOIN BATCHES B on C.CCODE = B.CCODE

WHERE COURSEFEE >3000;

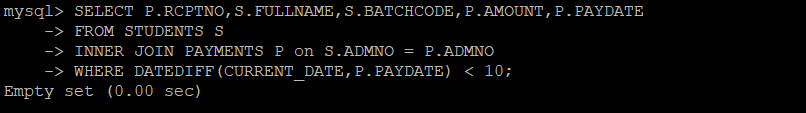


8)SELECT P.RCPTNO,S.FULLNAME,S.BATCHCODE,P.AMOUNT,P.PAYDATE

FROM STUDENTS S

INNER JOIN PAYMENTS P on S.ADMNO = P.ADMNO

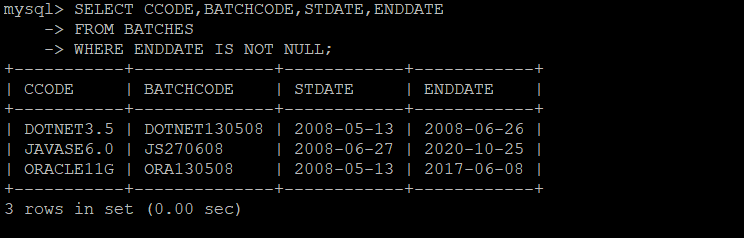
WHERE DATEDIFF(CURRENT\_DATE,P.PAYDATE) < 10;



9)SELECT CCODE,BATCHCODE,STDATE,ENDDATE

FROM BATCHES

WHERE ENDDATE IS NOT NULL;

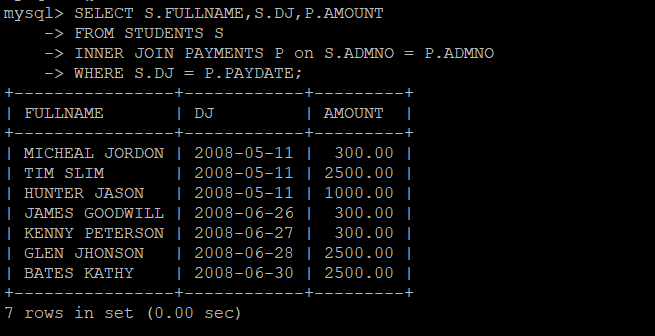


10)SELECT S.FULLNAME,S.DJ,P.AMOUNT

FROM STUDENTS S

INNER JOIN PAYMENTS P on S.ADMNO = P.ADMNO

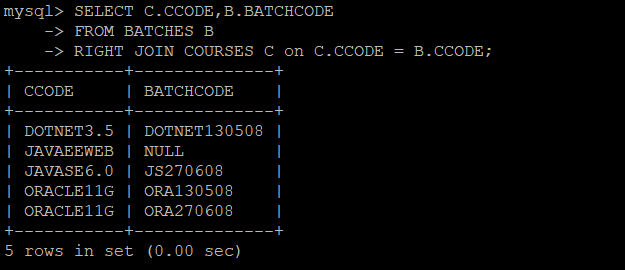
WHERE S.DJ = P.PAYDATE;



11)SELECT C.CCODE,B.BATCHCODE

FROM BATCHES B

RIGHT JOIN COURSES C on C.CCODE = B.CCODE;

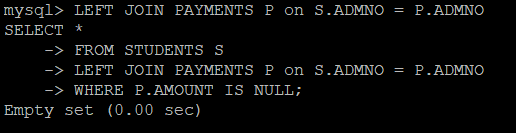


12)SELECT \*

FROM STUDENTS S

LEFT JOIN PAYMENTS P on S.ADMNO = P.ADMNO

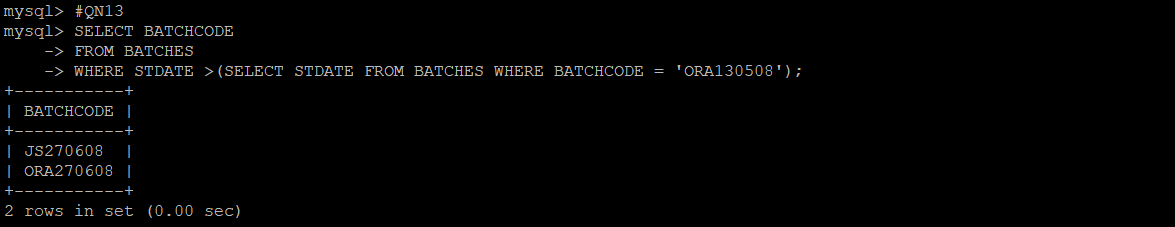
WHERE P.AMOUNT IS NULL;



13)SELECT BATCHCODE

FROM BATCHES

WHERE STDATE >(SELECT STDATE FROM BATCHES WHERE BATCHCODE = 'ORA130508');



14)SELECT FULLNAME,BATCHCODE

FROM STUDENTS S

INNER JOIN PAYMENTS P on S.ADMNO = P.ADMNO

GROUP BY P.ADMNO HAVING (P.ADMNO,SUM(AMOUNT)) IN

(SELECT P2.ADMNO,AMOUNT

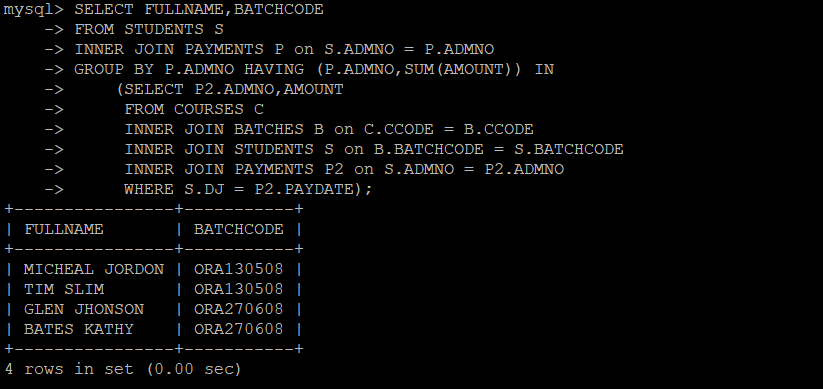
FROM COURSES C

INNER JOIN BATCHES B on C.CCODE = B.CCODE

INNER JOIN STUDENTS S on B.BATCHCODE = S.BATCHCODE

INNER JOIN PAYMENTS P2 on S.ADMNO = P2.ADMNO

WHERE S.DJ = P2.PAYDATE);



15)SELECT \*

FROM STUDENTS S

INNER JOIN BATCHES B2 on S.BATCHCODE = B2.BATCHCODE

INNER JOIN COURSES C2 on B2.CCODE = C2.CCODE

GROUP BY ADMNO HAVING (ADMNO,SUM(COURSEFEE)) NOT IN

(SELECT P2.ADMNO,SUM(AMOUNT)

FROM COURSES C

INNER JOIN BATCHES B on C.CCODE = B.CCODE

INNER JOIN STUDENTS S on B.BATCHCODE = S.BATCHCODE

INNER JOIN PAYMENTS P2 on S.ADMNO = P2.ADMNO

GROUP BY P2.ADMNO);

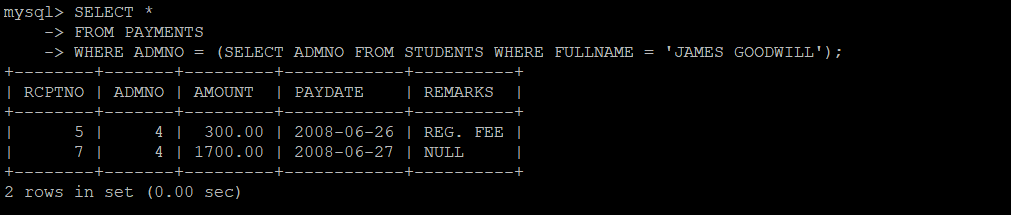


**SUBQUERIES**

1)SELECT \*

FROM PAYMENTS

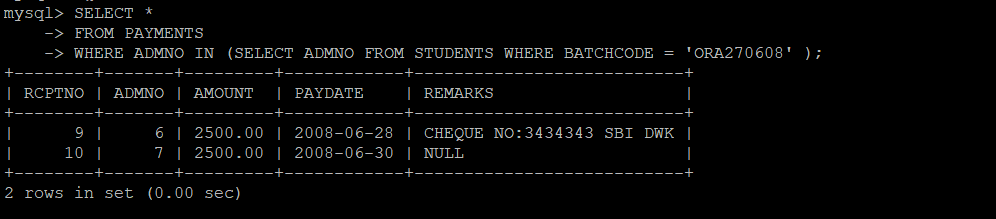
WHERE ADMNO = (SELECT ADMNO FROM STUDENTS WHERE FULLNAME = 'JAMES GOODWILL');



2)SELECT \*

FROM PAYMENTS

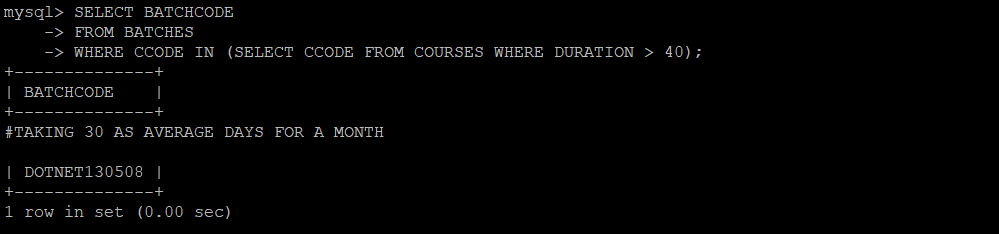
WHERE ADMNO IN (SELECT ADMNO FROM STUDENTS WHERE BATCHCODE = 'ORA270608' );



3)SELECT BATCHCODE

FROM BATCHES

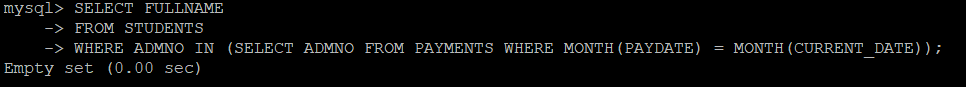
WHERE CCODE IN (SELECT CCODE FROM COURSES WHERE DURATION > 40);



4)SELECT FULLNAME

FROM STUDENTS

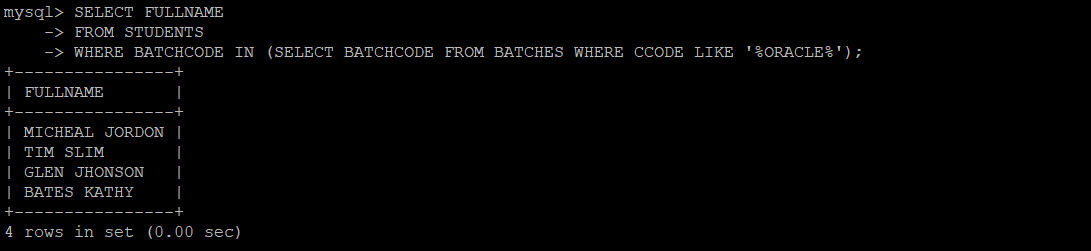
WHERE ADMNO IN (SELECT ADMNO FROM PAYMENTS WHERE MONTH(PAYDATE) = MONTH(CURRENT\_DATE));



5)SELECT FULLNAME

FROM STUDENTS

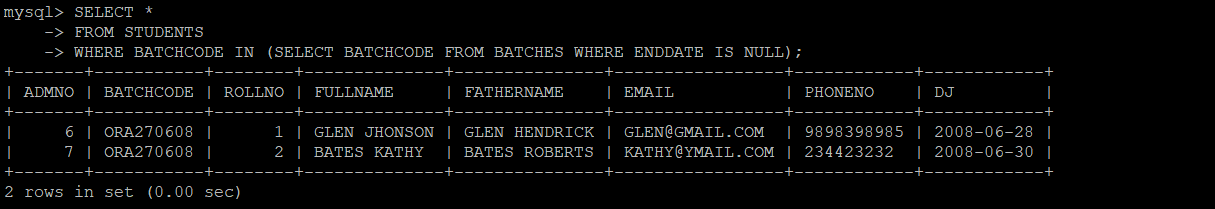
WHERE BATCHCODE IN (SELECT BATCHCODE FROM BATCHES WHERE CCODE LIKE '%ORACLE%');



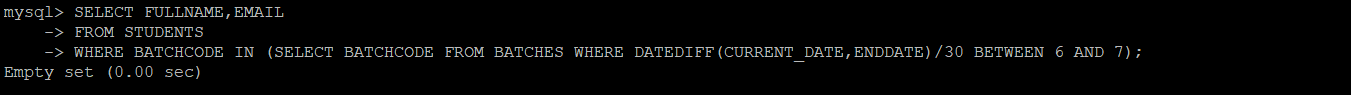
6)SELECT \*

FROM STUDENTS

WHERE BATCHCODE IN (SELECT BATCHCODE FROM BATCHES WHERE ENDDATE IS NULL);



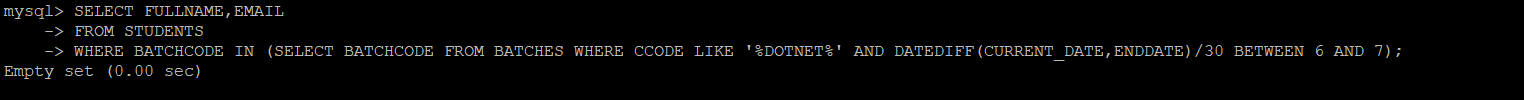
7)SELECT FULLNAME, EMAIL FROM STUDENTS WHERE BATCHCODE IN (SELECT BATCHCODE FROM BATCHES WHERE MONTHS\_BETWEEN(SYSDATE,ENDDATE) > 6);



8)SELECT FULLNAME,EMAIL

FROM STUDENTS

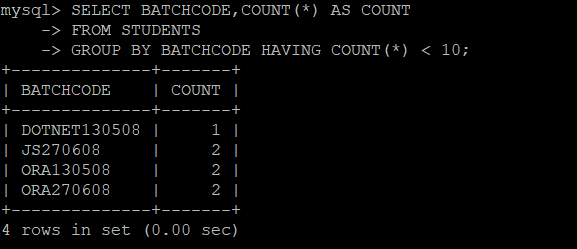
WHERE BATCHCODE IN (SELECT BATCHCODE FROM BATCHES WHERE CCODE LIKE '%DOTNET%' AND DATEDIFF(CURRENT\_DATE,ENDDATE)/30 BETWEEN 6 AND 7);



9)SELECT BATCHCODE,COUNT(\*) AS COUNT

FROM STUDENTS

GROUP BY BATCHCODE HAVING COUNT(\*) < 10;



10 )SELECT P.RCPTNO,S.FULLNAME,S.BATCHCODE,P.AMOUNT,P.PAYDATE

FROM STUDENTS S

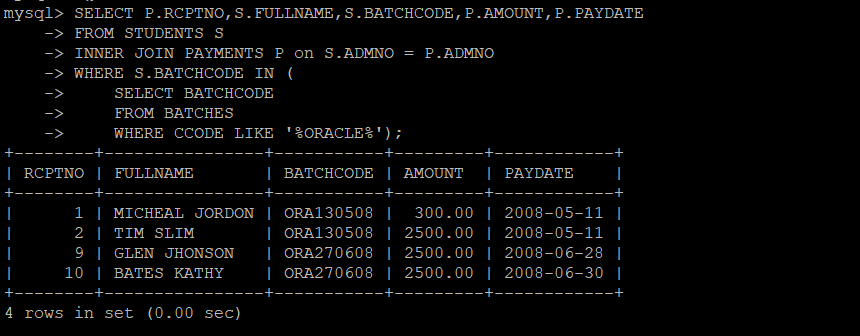
INNER JOIN PAYMENTS P on S.ADMNO = P.ADMNO

WHERE S.BATCHCODE IN (

SELECT BATCHCODE

FROM BATCHES

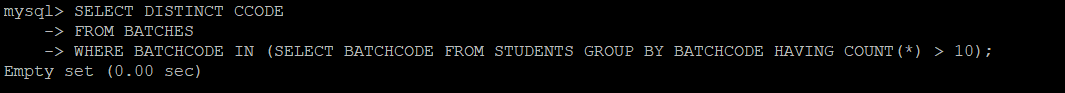
WHERE CCODE LIKE '%ORACLE%');



11)SELECT DISTINCT CCODE

FROM BATCHES

WHERE BATCHCODE IN (SELECT BATCHCODE FROM STUDENTS GROUP BY BATCHCODE HAVING COUNT(\*) > 10);

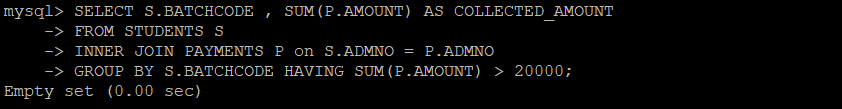


12)SELECT S.BATCHCODE , SUM(P.AMOUNT) AS COLLECTED\_AMOUNT

FROM STUDENTS S

INNER JOIN PAYMENTS P on S.ADMNO = P.ADMNO

GROUP BY S.BATCHCODE HAVING SUM(P.AMOUNT) > 20000;



13)SELECT FULLNAME

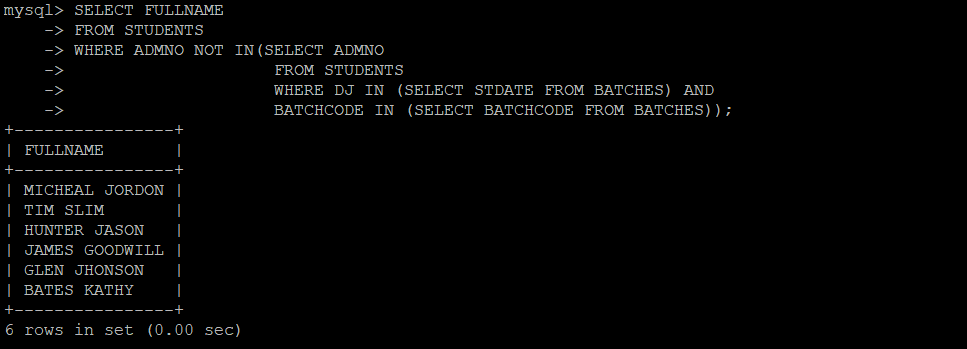
FROM STUDENTS

WHERE ADMNO NOT IN(SELECT ADMNO

FROM STUDENTS

WHERE DJ IN (SELECT STDATE FROM BATCHES) AND

BATCHCODE IN (SELECT BATCHCODE FROM BATCHES));



14)SELECT BATCHCODE

FROM BATCHES

WHERE BATCHCODE IN

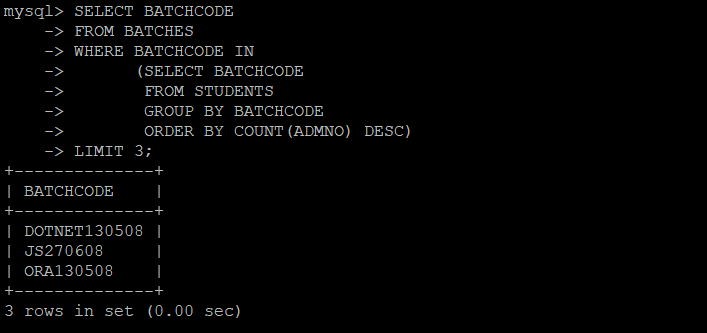
(SELECT BATCHCODE

FROM STUDENTS

GROUP BY BATCHCODE

ORDER BY COUNT(ADMNO) DESC)

LIMIT 3;



15)

UPDATE PAYMENTS P SET AMOUNT = ( SELECT COURSEFEE FROM COURSES C, BATCHES B, STUDENTS S WHERE C.CCODE = B.CCODE AND B.BATCHCODE = S.BATCHCODE AND S.ADMNO = P.ADMNO); WHERE RCPTNO = 2;

