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)

);

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');

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');

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');

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);

Q1)

SELECT \* FROM STUDENTS ORDER BY BATCHCODE ,DJ;

Q2)

SELECT \* FROM PAYMENTS WHERE MONTH(PAYDATE) = 05 AND YEAR(PAYDATE) = 2008;

Q3)

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

Q4)

SELECT FULLNAME,FATHERNAME,DJ,DATEDIFF(CURRENT\_DATE,DJ) AS NO\_OF\_DAYS\_OF\_JOINING

FROM STUDENTS;

Q5)

SELECT BATCHCODE FROM BATCHES WHERE ENDDATE IS NOT NULL;

Q6)

SELECT BATCHCODE FROM BATCHES WHERE CCODE LIKE '%JAVASE%' OR '%ORACLE%';

Q7)

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

Q8)

SELECT \* FROM STUDENTS WHERE DATE\_ADD(DJ,INTERVAL 7 DAY) < CURRENT\_DATE;

Q9)

SELECT \* , (COURSEFEE\*11)/10 AS INCREASED\_FEE FROM COURSES WHERE COURSEFEE < 3000;

Q10)

SELECT FULLNAME FROM STUDENTS WHERE FULLNAME LIKE '%s%' AND FATHERNAME LIKE '%p%';

Q11)

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

Q12)

SELECT BATCHCODE,STDATE,DATE\_ADD(STDATE ,INTERVAL 2 MONTH) AS APPROXIMATE\_ENDING

FROM BATCHES WHERE ENDDATE IS NULL;

Q13)

SELECT DATEDIFF(ENDDATE,(DATE\_ADD(STDATE,INTERVAL 2 MONTH ))) AS DIFFERENCE\_BTW\_ESTIMATED\_AND\_ACTUAL FROM BATCHES

WHERE ENDDATE IS NOT NULL;

Q14)

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

Q15)

UPDATE BATCHES SET ENDDATE = DATE\_ADD(CURRENT\_DATE,INTERVAL -1 DAY )

WHERE BATCHCODE = 'JS130508';

Q16)

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

Q17)

SELECT \* FROM PAYMENTS WHERE AMOUNT > 1000 OR (ADMNO BETWEEN 100 AND 150 AND DATEDIFF(PAYDATE ,CURRENT\_DATE) < 10);

Q18)

UPDATE PAYMENTS SET ADMNO = 120 AND PAYDATE = '08-06-01' WHERE RCPTNO = 12;

Q19)

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

Q20

#SHOULD DO

#QROUPING

Q1)

SELECT SUM(AMOUNT) AS TOTAL\_PAID\_AMOUNT FROM PAYMENTS GROUP BY ADMNO;

Q2)

#TODO

Q3)

SELECT BATCHCODE,COUNT(ADMNO) FROM STUDENTS GROUP BY BATCHCODE;

Q4)

SELECT CCODE , MAX(STDATE) AS RECENTLY\_STARTED FROM BATCHES

GROUP BY CCODE ;

Q5)

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

Q6)

SELECT CCODE ,SUM(\*) AS NUMBER\_OF\_BATCHES FROM BATCHES GROUP BY CCODE;

Q7)

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

FROM PAYMENTS GROUP BY PAYDATE;

Q8)

SELECT B.BATCHCODE FROM BATCHES B INNER JOIN STUDENTS S on B.BATCHCODE = S.BATCHCODE

GROUP BY B.BATCHCODE HAVING COUNT(\*) > 10;

Q9)

SELECT BATCHCODE,MIN(DJ) AS FIRST\_ADMISSION,MAX(DJ) AS LAST\_ADMISSION

FROM STUDENTS GROUP BY BATCHCODE;

Q10)

SELECT CCODE FROM BATCHES GROUP BY CCODE HAVING COUNT(BATCHCODE) > 1;

#TODO

Q11)

#YEARWISE

SELECT YEAR(STDATE) AS BATCH\_YEAR , COUNT(BATCHCODE)FROM BATCHES

GROUP BY YEAR(STDATE);

#COURSEWISE

SELECT CCODE,COUNT(BATCHCODE) FROM BATCHES GROUP BY CCODE;

Q12)

#TODO

Q13)

SELECT COUNT(BATCHCODE) AS CURRENTLY\_RUNNING FROM BATCHES

WHERE ENDDATE IS NULL;

Q14)

SELECT SUBSTRING(EMAIL,LOCATE('@',EMAIL) + 1,LOCATE('.',EMAIL)-LOCATE('@',EMAIL) -1) AS SERVER,COUNT(ADMNO) FROM STUDENTS GROUP BY SERVER;

Q15)

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;

Q16)

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

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

GROUP BY CURRENT\_MONTH;

Q17)

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

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

GROUP BY YEAR(PAYDATE);

Q18)

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;

Q19)

SELECT PAYDATE FROM PAYMENTS GROUP BY PAYDATE HAVING SUM(AMOUNT) > 5000;

Q20)

SELECT BATCHCODE FROM STUDENTS WHERE DATEDIFF(CURRENT\_DATE,DJ) < 10

GROUP BY DJ HAVING COUNT(\*) > 5;

#JOINING

Q1)

SELECT BATCHCODE,CCODE,STDATE FROM BATCHES ;

Q2)

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;

Q3)

SELECT B.CCODE,B.BATCHCODE,S.FULLNAME FROM BATCHES B

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

Q4)

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

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

GROUP BY B.CCODE;

Q5)

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;

Q6)

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;

Q7)

SELECT B.\* FROM COURSES C

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

WHERE COURSEFEE >3000;

Q8)

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;

Q9)

SELECT CCODE,BATCHCODE,STDATE,ENDDATE

FROM BATCHES WHERE ENDDATE IS NOT NULL;

Q10)

SELECT S.FULLNAME,S.DJ,P.AMOUNT FROM STUDENTS S

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

WHERE S.DJ = P.PAYDATE;

Q11

SELECT C.CCODE,B.BATCHCODE FROM BATCHES B

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

Q12)

SELECT \* FROM STUDENTS S

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

WHERE P.AMOUNT IS NULL;

Q13)

SELECT BATCHCODE FROM BATCHES

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

Q14)

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);

Q15)

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

Q1)

SELECT \* FROM PAYMENTS

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

Q2)

SELECT \* FROM PAYMENTS

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

Q3)

SELECT BATCHCODE FROM BATCHES

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

Q4)

SELECT FULLNAME FROM STUDENTS

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

Q5)

SELECT FULLNAME FROM STUDENTS

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

Q6)

SELECT \* FROM STUDENTS

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

Q7)

SELECT FULLNAME,EMAIL FROM STUDENTS

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

#HERE DATEDIFF(MONTH,CURRENT\_DATE,ENDDATE) IS NOT WORKING

#TAKING 30 AS AVERAGE DAYS FOR A MONTH

Q8)

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);

Q9)

SELECT BATCHCODE,COUNT(\*) AS COUNT

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

Q10)

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%');

Q11)

SELECT DISTINCT CCODE

FROM BATCHES

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

Q12)

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;

Q13)

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));

Q14)

SELECT BATCHCODE

FROM BATCHES

WHERE BATCHCODE IN

(SELECT BATCHCODE

FROM STUDENTS

GROUP BY BATCHCODE

ORDER BY COUNT(ADMNO) DESC)

LIMIT 3;

Q15)

UPDATE PAYMENTS

SET AMOUNT =

(SELECT SUM(COURSEFEE)

FROM COURSES C

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

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

WHERE ADMNO IN (SELECT ADMNO FROM PAYMENTS WHERE RCPTNO = '200')

GROUP BY ADMNO)

WHERE RCPTNO = 200;

OUTPUTS:-





































