database design:

1. Create a DATABASE: SCHOOL

TABLES

Create the following three tables with same names and data types as provided below:

**CourseMaster**

|  |  |  |
| --- | --- | --- |
| ***Column Name*** | ***Data Type*** | ***Remarks*** |
| CID | Integer | Primary Key |
| CourseName | Varchar(40) | NOT NULL |
| Category | Char(1) | NULL, Basic/Medium/Advanced |
| Fee | Smallmoney | NOT NULL; Fee can’t be negative |

**StudentMaster**

|  |  |  |
| --- | --- | --- |
| ***Column Name*** | ***Data Type*** | ***Remarks*** |
| SID | TinyInt | Primary Key |
| StudentName | Varchar(40) | NOT NULL |
| Origin | Char(1) | NOT NULL, Local/Foreign |
| Type | Char(1) | NOT NULL, UnderGraduate/Graduate |

**EnrollmentMaster**

|  |  |  |
| --- | --- | --- |
| ***Column Name*** | ***Data Type*** | ***Remarks*** |
| CID | Integer | NOT NULL Foreign Key |
| SID | Tinyint | NOT NULL Foreign Key |
| DOE | DateTime | NOT NULL |
| FWF (Fee Waiver Flag) | Bit | NOT NULL |
| Grade | Char(1) | O/A/B/C |

Using the above table layouts as schema, write T-SQL Statements for the following requirements:

1. List the course wise total no. of Students enrolled. Provide the information only for students of foreign origin and only if the total exceeds 10.

SELECT CourseName, COUNT(\*) AS noofenrolls

FROM CourseMaster as cm

JOIN EnrollmentMaster as em

ON cm.CID = em.CID

JOIN StudentMaster as sm

ON sm.SID = em.sid

WHERE Origin = 'f'

GROUP BY CourseName

HAVING count(\*) >10

1. List the names of the Students who have not enrolled for Java course.

SELECT NAME

FROM StudentMaster

WHERE SID in (

SELECT DISTINCT SID

FROM EnrollmentMaster

WHERE CID not in (

SELECT DISTINCT CID

FROM CourseMaster

WHERE CourseName = 'JAVA'

)

)

1. List the name of the advanced course where the enrollment by foreign students is the highest.

with cte

as (

SELECT CourseName,COUNT(\*) AS CNT

FROM CourseMaster AS CM

JOIN EnrollmentMaster AS EM

ON CM.CID=EM.CID

JOIN StudentMaster AS SM

ON SM.SID=EM.SID

WHERE Category='A' AND ORIGIN = 'F'

GROUP BY CourseName

)

select CourseName

from cte

where CNT = (select max(cnt) from cte)

1. List the names of the students who have enrolled for at least one basic course in the current month.

select DISTINCT Name AS studentname

from EnrollmentMaster as em

join StudentMaster as sm

on em.sid=sm.SID

join CourseMaster as cm

on cm.CID = em.cid

WHERE Category = 'b' AND DATEDIFF(MM,DOE,GETDATE())=0

1. List the names of the Undergraduate, local students who have got a “C” grade in any basic course.

SELECT Name

FROM EnrollmentMaster as em

join StudentMaster as sm

on sm.SID=em.sid

join CourseMaster as cm

on cm.CID=em.cid

WHERE Type = 'U' and Origin = 'L' and Grade = 'B'

1. List the names of the courses for which no student has enrolled in the month of SEPT 2022.

SELECT CM.CID , CM.COURSENAME

FROM COURSEMASTER AS CM

LEFT JOIN ( SELECT DISTINCT CM.CID , COURSENAME

FROM ENROLLMENTMASTER AS EM

JOIN STUDENTMASTER AS SM

ON SM.SID=EM.SID

JOIN COURSEMASTER AS CM

ON CM.CID=EM.CID

WHERE DATENAME(MM,DOE) = 'SEPTEMBER' AND DATEPART(YY,DOE) = 2022

) AS NEWT

ON NEWT.CID = CM.CID

WHERE NEWT.CID IS NULL

1. List name, Number of Enrollments and Popularity for all Courses. Popularity has to be displayed as “High” if number of enrollments is higher than 5, “Medium” if greater than or equal to 3 and less than 5, and “Low” if the no. Is less than 3.

SELECT [COURSENAME] , COUNT(\*) AS NOOFENR,

CASE

WHEN COUNT(\*) >5 THEN 'HIGH'

WHEN COUNT(\*) BETWEEN 3 AND 5 THEN 'MEDIUM'

ELSE 'LOW'

END AS POPULARITY

FROM COURSEMASTER AS CM JOIN ENROLLMENTMASTER AS EM ON CM.CID=EM.CID

GROUP BY COURSENAME

1. List the most recent enrollment details with information on Student Name, Course name and age of enrollment in days.

select [Name] as Sname , [CourseName] , DOE ,datediff(dd,doe,getdate()) as AgeofEnr

from EnrollmentMaster as em

join CourseMaster as cm

on em.cid=cm.cid

join StudentMaster as sm

on sm.SID=em.sid

where datediff(dd,doe,getdate())<=5

1. List the names of the Local students who have enrolled for exactly 3 basic courses.

SELECT SM.NAME , COUNT(\*) AS ENROLLMENT

FROM STUDENTMASTER AS SM

JOIN ENROLLMENTMASTER AS EM

ON SM.SID=EM.SID

JOIN COURSEMASTER AS CM

ON CM.CID=EM.CID

WHERE ORIGIN = 'L' AND CATEGORY = 'B'

GROUP BY SM.NAME

HAVING COUNT(\*) = 2

1. List the names of the Courses enrolled by all (every) students.

SELECT DISTINCT CourseName

FROM CourseMaster AS CM

LEFT JOIN ENROLLMENTMASTER AS EM

ON CM.CID = EM.CID

WHERE EM.CID IS NOT NULL

1. For those enrollments for which fee have been waived, provide the names of students who have got ‘O’ grade.

with cte

as

(select distinct sid

from enrollmentmaster

where fwf = 1 and grade = 'o'

)

select Name , SID

from StudentMaster

where SID in (select sid from cte)

-- or

SELECT NAME,

SID

FROM STUDENTMASTER

WHERE SID IN

(SELECT DISTINCT SID

FROM ENROLLMENTMASTER

WHERE FWF = 1 AND GRADE = 'O'

)

1. List the names of the foreign, undergraduate students who have got grade ‘C’ in any basic course.

SELECT sm.SID , sm.Name as [Student Name]

from StudentMaster as sm

join EnrollmentMaster as em

on sm.SID=em.sid

join CourseMaster as cm

on cm.CID=em.cid

where Origin='F' and Type = 'U' and grade = 'c' and Category = 'B'

1. List the course name, total no. of enrollments in the current month.

SELECT COURSENAME , NOOFENR

FROM COURSEMASTER AS CM

JOIN

(SELECT CID, COUNT(\*) AS NOOFENR

FROM ENROLLMENTMASTER

WHERE DATEDIFF(YY,DOE,GETDATE()) = 1

GROUP BY CID) AS K

ON K.CID=CM.CID

order by CourseName