Exercises on SQL language (no.2)

Exercise

STUDENT (SID, SName, City)
TEACHER (TID, TName, Depart)
COURSE (CCode, CName, TID)
EXAM (CCode, SID, Date, Grade)

⟫ For each student with average grade on exams greater than 28 and who has taken exams on at least 10 different dates, show the student ID, student name, and maximum, minimum, and average grade on exams

Exercise

STUDENT (<u>SID</u>, SName, City)

TEACHER (<u>TID</u>, TName, Depart)

COURSE (<u>CCode</u>, CName, TID)

EXAM (<u>CCode</u>, <u>SID</u>, Date, Grade)

➣ For each student with average grade on exams greater than 28 and who has taken exams on at least 10 different dates, show the student ID, student name, and maximum, minimum, and average grade on exams

SELECT S.SID, Sname, MIN(Grade), MAX(Grade), AVG(Grade)

FROM STUDENT S, EXAM E

WHERE S.SID=E.SID

GROUP BY S.SID, Sname

HAVING AVG(Grade) > 28 AND COUNT(DISTINCT Date) >=10

Exercise

MAGAZINE (<u>MID</u>, MName, Publisher)
ARTICLE (<u>AID</u>, Title, Topic, MId)

▷ Find the identifier, the name and publisher of the magazines that have published at least two articles about motorcycles

Exercise

MAGAZINE (<u>MID</u>, MName, Publisher)
ARTICLE (<u>AID</u>, Title, Topic, MId)

▷ Find the identifier, the name and publisher of the magazines that have published at least two articles about motorcycles

SELECT M.MID, MName, Publisher
FROM MAGAZINE M, ARTICLE A
WHERE M.MID=A.MID AND Topic ='Motorcycle'
GROUP BY M.MID, MName, Publisher
HAVING COUNT (*) >=2

Exercise

SAILOR (<u>SID</u>, SName, Expertise, DateofBirth)
BOAT (<u>BID</u>, BName, Color)
BOOKING (<u>SID</u>, <u>BID</u>, <u>Date</u>)

⟩ Find the codes and the names of the sailors who have performed three bookings

Exercise

SAILOR (<u>SID</u>, SName, Expertise, DateofBirth)
BOAT (<u>BID</u>, BName, Color)
BOOKING (<u>SID</u>, <u>BID</u>, <u>Date</u>)

▷ Find the codes and the names of the sailors who have performed three bookings

SELECT S.SID, SName
FROM SAILOR S, BOOKING BK
WHERE S.SID=BK.SID
GROUP BY S.SID, SName
HAVING COUNT(*) =3

Exercise

SAILOR (<u>SID</u>, SName, Expertise, DateofBirth)
BOAT (<u>BID</u>, BName, Color)
BOOKING (<u>SID</u>, <u>BID</u>, <u>Date</u>)

▷ Find the code and the name of the sailors who have booked at least two boats

Exercise

SAILOR (<u>SID</u>, SName, Expertise, DateofBirth)
BOAT (<u>BID</u>, BName, Color)
BOOKING (<u>SID</u>, <u>BID</u>, <u>Date</u>)

⊅   Find the code and the name of the sailors who have booked at least two boats

SELECT S.SID, SName
FROM SAILOR S, BOOKING BK
WHERE S.SID=BK.SID
GROUP BY S.SID, SName
HAVING COUNT(DISTINCT BID)>=2

Exercise

AIRCRAFT (<u>AID</u>, AName, MaximumRange)
PILOT (<u>PID</u>, PName, Salary)
CERTIFICATE (<u>AID</u>, <u>PID</u>)

⟫ Find the code and the name of the pilots who are qualified to fly on at least two aircrafts that can cover distances greater than 5,000 km (MaximumRange>=5,000)

Exercise

AIRCRAFT (<u>AID</u>, AName, MaximumRange)
PILOT (<u>PID</u>, PName, Salary)
CERTIFICATE (<u>AID</u>, <u>PID</u>)

▷ Find the code and the name of the pilots who are qualified to fly on at least two aircrafts that can cover distances greater than 5,000 km (MaximumRange>=5,000)

SELECT P.PID, PName
FROM AIRCRAFT A, CERTIFICATE C, PILOT P
WHERE A.AID=C.AID AND C.PID=P.PID
        AND MaximumRange >= 5,000
GROUP BY P.PID, PName
HAVING COUNT(*) >=2

Exercise

FLAT (FCode, Address, City, Surface)
LEASING-CONTRACT (LCCode, StartDate, EndDate, PersonName, MonthlyPrice, Fcode)

⟩ Find the codes and the addresses of flats in Turin for which more than 5 contracts have been signed.

Exercise

FLAT (<u>FCode</u>, Address, City, Surface)
LEASING-CONTRACT (<u>LCCode</u>, StartDate, EndDate, PersonName, MonthlyPrice, Fcode)

⟫ Find the codes and the addresses of flats in Turin for which more than 5 contracts have been signed.

SELECT F.FCode, Address
FROM FLAT F, LEASING-CONTRACT LC
WHERE City='Turin' AND LC.Fcode=F.Fcode
GROUP BY F.Fcode, Address
HAVING COUNT(*)>5

Exercise

FLAT (FCode, Address, City, Surface)
LEASING-CONTRACT (LCCode, StartDate, EndDate, PersonName, MonthlyPrice, FCode)

> For the cities in which at least 100 contracts have been signed, find the city, the maximum monthly price, the average monthly price, the maximum duration of the leasing contracts, the average duration of the leasing contracts and the total number of signed contracts.

Exercise

FLAT (FCode, Address, City, Surface)
LEASING-CONTRACT (LCCode, StartDate, EndDate, PersonName, MonthlyPrice, FCode)

> For the cities in which at least 100 contracts have been signed, find the city, the maximum monthly price, the average monthly price, the maximum duration of the leasing contracts, the average duration of the leasing contracts and the total number of signed contracts.

SELECT F.City, MAX(L.MonthlyPrice), AVG(L.MonthlyPrice),
        MAX(L.EndDate-L.StartDate), AVG(L.EndDate-L.StartDate), COUNT(*)
FROM FLAT F, LEASING-CONTRACT L
WHERE F.FCode=L.Fcode
GROUP BY F.City
HAVING COUNT (*)>=100

Exercise

FLAT (FCode, Address, City, Surface)
LEASING-CONTRACT (LCCode, StartDate, EndDate, PersonName, MonthlyPrice, FCode)

⟩ Find the names of people who have signed more than two leasing contracts for the same flat (in different periods).

Exercise

FLAT (FCode, Address, City, Surface)
LEASING-CONTRACT (LCCode, StartDate, EndDate, PersonName, MonthlyPrice, FCode)

⟩ Find the names of people who have signed more than two leasing contracts for the same flat (in different periods).

SELECT DISTINCT PersonName
FROM LEASING_CONTRACT LC
GROUP BY PersonName, FCode
HAVING COUNT(*)>2

Exercise

COURSE (<u>CourseCode</u>, CourseName, Year, Semester)
COURSE-SCHEDULE (<u>CourseCode</u>, <u>DayOfWeek</u>, <u>StartTime</u>, EndTime, Room)

⟩ Find the codes, the names and the total number of weekly hours of the third-year courses whose total number of weekly hours is greater than 10

Exercise

COURSE (<u>CourseCode</u>, CourseName, Year, Semester)
COURSE-SCHEDULE (<u>CourseCode</u>, <u>DayOfWeek</u>, <u>StartTime</u>, EndTime, Room)

> Find the codes, the names and the total number of weekly hours of the third-year courses whose total number of weekly hours is greater than 10

SELECT C.CourseCode, CourseName, SUM(EndTime-StartTime)
FROM COURSE C, COURSE-SCHEDULE CS
WHERE C.CourseCode=CS.CourseCode AND Year = 3
GROUP BY C.CourseCode, CourseName
HAVING SUM(EndTime-StartTime)>10