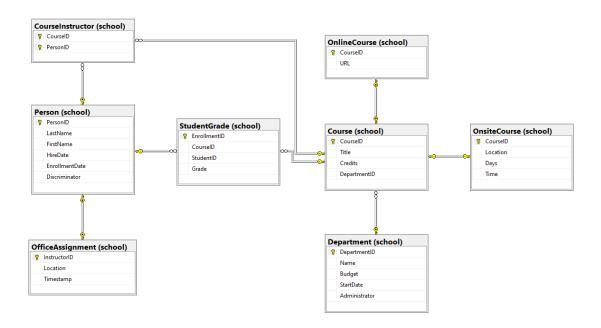Esercizio SQL

Prendere in considerazione il database School rappresentato dal seguente diagramma

e scrivere le istruzioni SQL necessarie per estrarre le seguenti informazioni:

1. il voto medio complessivo di tutti gli esami
```
select title, avg(grade) as votomedio
from school.StudentGrade inner join school.course
on school.StudentGrade.CourseID = school.course.CourseID
where grade is not null
group by title
```

	title	votomedio
1	Calculus	2.000000
2	Chemistry	3.500000
3	Composition	3.200000
4	Literature	3.500000

2. gli studenti che hanno una media dei loro esami inferiore alla media complessiva degli esami
```
select  StudentID, avg(grade) as mediaesame from
school.StudentGrade
group by StudentID
having avg(grade) > (select avg(votomedio)
from (select title, avg(grade) as votomedio
from school.StudentGrade inner join school.course
on school.StudentGrade.CourseID = school.course.CourseID
where grade is not null
group by title) as avg)
```

	StudentID	mediaesame
1	2	3.750000
2	3	3.500000
3	7	3.750000
4	9	3.500000

3. lo studente con la media più alta

```
select studentId, avg(grade) as media
from school.StudentGrade
group by studentId
having avg(grade) = (select max(media)
from (select studentId, avg(grade) as media
from school.StudentGrade
group by studentId) as medpiualto)
```

	studentId	media
1	13	4.000000
2	20	4.000000
3	24	4.000000
4	29	4.000000

4. il corso che ha fatto più esami

```
select title, count(school.StudentGrade.courseID) as totexams
from school.StudentGrade inner join school.Course
on school.StudentGrade.CourseID = school.Course.CourseID
group by title
having count(school.StudentGrade.courseID) = (select
max(totexams)
from (select title, count(school.StudentGrade.courseID) as
totexams
from school.StudentGrade inner join school.Course
on school.StudentGrade.CourseID = school.Course.CourseID
group by title) as max)
```

	title	totexams
1	Microeconomics	8

5. i docenti del corso che ha fatto più esami

```
select PersonID, count(title) as exams
from school.CourseInstructor inner join school.Course
on school.CourseInstructor.CourseID = school.Course.CourseID
inner join school.StudentGrade
on school.Course.CourseID = school.StudentGrade.CourseID
group by PersonID
having count(title) = (select max(exams)
from (select PersonID, count(title) as exams
from school.CourseInstructor inner join school.Course
on school.CourseInstructor.CourseID = school.Course.CourseID
inner join school.StudentGrade
on school.Course.CourseID = school.StudentGrade.CourseID
group by PersonID) as tot)
```

	PersonID	exams
1	18	8

6. i corsi che iniziano con la A
```
select * from school.Course
where title like 'A%'
```
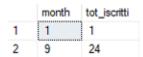| CourseID	Title	Credits	DepartmentID

7. i corsi che si tengono il lunedì
```
select * from school.Course inner join school.OnsiteCourse
on school.Course.CourseID = school.OnsiteCourse.CourseID
where days like 'M%'
```
	CourseID	Title	Credits	DepartmentID	CourseID	Location	Days	Time
1	1045	Calculus	4	7	1045	121 Smith	MWHF	1900-01-01 15:30:00
2	1050	Chemistry	4	1	1050	123 Smith	MTWH	1900-01-01 11:30:00
3	2042	Literature	4	2	2042	225 Adams	MTWH	1900-01-01 11:00:00

8. per ogni mese dell'anno, il numero di studenti che si sono iscritti in quel mese
```
SELECT MONTH(school.Person.EnrollmentDate) [month],
     count(school.Person.LastName) tot_iscritti
FROM school.Person
WHERE school.Person.EnrollmentDate IS NOT NULL
GROUP BY MONTH(school.Person.EnrollmentDate)
ORDER BY [month];
```
	month	tot_iscritti
1	1	1
2	9	24

9. gli studenti che si sono iscritti di lunedì
   ?