

-- 1.

```
select dept_name, max(salary) from instructor
group by dept_name;
```

-- 2.

```
select distinct(ID) from takes
where (course_id, sec_id, semester) in
(select course_id, sec_id, semester from teaches
where ID in (select ID from instructor
where name = 'Katz'));
```

-- 3

```
select distinct c.course_id, title
from course as c, section as s, time_slot as t
where c.course_id = s.course_id
and s.time_slot_id = t.time_slot_id
and t.end_hr >= 12
and c.dept_name = 'Comp. Sci.';
```

-- 4

```
select course_id, title from course
where course_id in (select prereq_id from course
inner join prereq on
course.course_id = prereq.course_id
where course.title = 'Robotics');
```

-- 5

```
select ID, name
from instructor
where salary = (select max(salary) from instructor);
```

-- 6.

```
select course_id, sec_id, count(course_id) as student_num from takes
where semester = 'Spring'
and year = 2017
group by
course_id;
```

-- 7.

```
select course_id, sec_id,
(select count(ID)
from takes as t
where t.course_id = s.course_id
and t.sec_id = s.sec_id
and t.semester = s.semester
and t.year = s.year) as students_num
from section as s
where semester = 'Spring' and year = 2017;
```

-- 8.

```
select instructor.ID, name
from instructor, teaches
where instructor.ID = teaches.ID
group by teaches.ID
having count(distinct course_id) >= 3;
```

-- 9

```
select ID, name from student
where ID in (select ID from takes
where grade like 'A%'
group by ID
having count(ID) >= all(select count(ID)
```

```
from takes where grade like 'A%'
group by ID));
```

```
-- 10
```

```
select ID, name
from student as s
where s.dept_name = 'History'
and ID not in
(select ID
from takes as t, course as c
where t.course_id = c.course_id
and c.dept_name = 'Music');
```

```
-- 11
```

```
select ID, name
from instructor
where ID not in ( select t1.ID
from teaches as t1, takes as t2
where t1.sec_id = t2.sec_id
and t1.course_id = t2.course_id
and t1.semester = t2.semester
and grade = 'A'
and t1.year = t2.year);
```

```
-- 12
```

```
select instructor.ID, name
from instructor
where exists (
select teaches.ID from teaches, takes
```

```
where teaches.ID = instructor.ID
and teaches.sec_id = takes.sec_id
and teaches.course_id = takes.course_id
and teaches.semester = takes.semester
and teaches.year = takes.year
and grade is not null)
and not exists (
select teaches.id
from teacher, takes
where teaches.ID = instructor.ID
and teaches.course_id = takes.course_id
and teaches.sec_id = takes.sec_id
and teaches.semester = takes.semester
and teaches.year = takes.year
and grade = 'A');
```

-- 13

```
select takes.ID, student.name, takes.course_id from takes
natural join student where
takes.id = student.id
group by takes.ID, takes.course_id
having count(*) > 1;
```

-- 14

```
select ID
from (select ID, course_id
from takes
group by ID, course_id
having count(*) >= 2) as m
```

group by ID

having count(course_id) >= 3;

-- 15

select ID, name

from instructor as i

where not exists

(select * from course as c

where c.dept_name = i.dept_name

and not exists (

select *

from teaches as t

where t.id = i.id

and t.course_id = c.course_id)

);

-- 2.1

insert into course values ('CS-001', 'Weekly Seminar', 'Comp. Sci.', 2);

-- 2.2

insert into section values ('CS-001', 1, 'Spring', 2022, null, null, null);

-- 2.3

insert into takes

select ID, 'CS-001', 1, 'Spring', 2022, null

from student where dept_name = 'Comp. Sci.';

-- 2.4

delete from takes

where ID = 12345 and course_id = 'CS-001' and sec_id = 1

```
and year = 2022 and semester = 'Spring';
```

```
-- 2.5
```

```
delete from course
```

```
where course_d = 'CS-001'
```

```
/* cascade delete has been set up before. Therefore, the associated enrollments and sections  
have been removed as a result of the association */
```

```
/*
```

```
3rd Question
```

```
create table person
```

```
(
```

```
driver_id varchar(20),
```

```
name varchar(20) not null,
```

```
address varchar(100),
```

```
primary key (driver_id);
```

```
)
```

```
create table car
```

```
(
```

```
license_plate varchar(10),
```

```
model varchar(20) not null,
```

```
year numeric(4, 0) check (year > 0),
```

```
primary key (license_plate);
```

```
)
```

```
create table owns
```

```
(
```

```
driver_id varchar(10),
```

```
license_plate varchar(10) not null,
```

```
primary key (driver_id, license_plate),
```

```
foreign key (driver_id) references person (driver_id),
```

```
on delete cascade,
```

```
on update cascade,  
foreign key (license_plate) references car(license_plate),  
on delete cascade  
on update cascade,  
);
```

```
create table accident  
(  
report_number numeric(5,0) not null,  
date date not null,  
location varchar (100),  
primary key (report_number);  
);
```

```
create table participated  
(  
report_number numeric(4, 0),  
license_plate varchar(10),  
driver_id varchar(20),  
damage_amount numeric(4, 0) check (damage_amount >= 0),  
primary key (report_number, license_plate),  
foreign key (license_plate) references car(license_plate),  
on delete cascade,  
foreign key (driver_id) references person(driver_id),  
on delete cascade,  
foreign key (report_number) references accident(report_number)  
on delete cascade  
);  
  
*/
```

