Creat Table and Relation

In [3]:
```
%load_ext sql
%sql postgresql://postgres:test@notebpostgres/cs425
```

```
The sql extension is already loaded. To reload it, use:
  %reload_ext sql
```

Out[3]:
```
'Connected: postgres@cs425'
```

In [305]:
```
%%sql
drop table book cascade;
drop table course cascade;
drop table student cascade;
drop table faculty cascade;
drop table enroll cascade;
drop table book_checkout cascade
```

```
Done.
Done.
Done.
Done.
Done.
Done.
```

Out[305]:
```
[]
```

In [306]:
```
%%sql

create table book(

    bookid serial,
    title text,
    price float,
    total_copies integer,
    primary key(bookid)
);

insert into book(title, price, total_copies) values ('Introduction of Al
gorithm', 84.66, 4);
insert into book(title, price, total_copies) values ('Database System Co
ncept', 74.99, 5);
insert into book(title, price, total_copies) values ('Stochastic  Calcul
us for Finance I', 41.02, 3);
insert into book(title, price, total_copies) values ('Stochastic  Calcul
us for Finance II', 55.22, 3);

create table course(

    courseid integer,
    title text,
    instructorid integer,
    textbookid integer,
    primary key(courseid)
);

insert into course(courseid, title, instructorid, textbookid) values (1,
```

```
 'Algorithm', 1, 1);
insert into course(courseid, title, instructorid, textbookid) values (2,
 'DB Organisation', 2, 2);
insert into course(courseid, title, instructorid, textbookid) values (3,
 'Advanced DB Organisation', 3, 2);
insert into course(courseid, title, instructorid, textbookid) values (4,
 'Math Finance I', 1, 3);
insert into course(courseid, title, instructorid, textbookid) values (5,
 'Math Finance II', 4, 4);

create table student(

    studentid integer,
    name text,
    gpa float,
    primary key(studentid)
);

insert into student(studentid, name, gpa) values (1, 'Tom ', 3.3);
insert into student(studentid, name, gpa) values (2, 'John', 3.8);
insert into student(studentid, name, gpa) values (3, 'Mary', 3.0);
insert into student(studentid, name, gpa) values (4, 'Kris', 3.6);
insert into student(studentid, name, gpa) values (5, 'Alex', 3.5);

create table faculty(

    facultyid integer,
    name text,
    salary integer,
    primary key(facultyid)
);

insert into faculty(facultyid, name, salary) values (1, 'James', 70000);
insert into faculty(facultyid, name, salary) values (2, 'Sarah', 60000);
insert into faculty(facultyid, name, salary) values (3, 'Jay  ', 80000);
insert into faculty(facultyid, name, salary) values (4, 'Rachel',70000);
insert into faculty(facultyid, name, salary) values (5, 'Paul ', 85000);

create table enroll(

    studentid integer,
    courseid integer,
    foreign key(courseid) references course,
    foreign key(studentid) references student
);

insert into enroll(studentid, courseid) values (1, 1);
insert into enroll(studentid, courseid) values (1, 2);
insert into enroll(studentid, courseid) values (2, 1);
insert into enroll(studentid, courseid) values (4, 3);
insert into enroll(studentid, courseid) values (4, 4);
insert into enroll(studentid, courseid) values (5, 5);

create table book_checkout(

    date date,
    bookid integer,
    studentid integer,
    foreign key(bookid) references book,
    foreign key(studentid) references student
);
```

```
insert into book_checkout(date, bookid, studentid) values ('2017-08-29',
 1, 1);
insert into book_checkout(date, bookid, studentid) values ('2017-09-02',
 4, 4);
insert into book_checkout(date, bookid, studentid) values ('2017-09-07',
 1, 4);
```

```
Done.
1 rows affected.
1 rows affected.
1 rows affected.
Done.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
Done.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
Done.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
Done.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
1 rows affected.
Done.
1 rows affected.
1 rows affected.
1 rows affected.
```

Out[306]: []

In [307]:
```
%%sql
select * from book
```

```
4 rows affected.
```

Out[307]:

bookid	title	price	total_copies
1	Introduction of Algorithm	84.66	4
2	Database System Concept	74.99	5
3	Stochastic Calculus for Finance I	41.02	3
4	Stochastic Calculus for Finance II	55.22	3

In [308]:
```
%%sql
select * from course
```

```
5 rows affected.
```

Out[308]:

courseid	title	instructorid	textbookid
1	Algorithm	1	1
2	DB Organisation	2	2
3	Advanced DB Organisation	3	2
4	Math Finance I	1	3
5	Math Finance II	4	4

In [309]:
```
%%sql
select * from student
```

```
5 rows affected.
```

Out[309]:

studentid	name	gpa
1	Tom	3.3
2	John	3.8
3	Mary	3.0
4	Kris	3.6
5	Alex	3.5

In [310]:
```
%%sql
select * from faculty
```

```
5 rows affected.
```

Out[310]:

facultyid	name	salary
1	James	70000
2	Sarah	60000
3	Jay	80000
4	Rachel	70000
5	Paul	85000

In [311]:
```
%%sql
select * from enroll
```

```
6 rows affected.
```

Out[311]:

studentid	courseid
1	1
1	2
2	1
4	3
4	4
5	5

In [312]:
```
%%sql
select * from book_checkout
```

```
3 rows affected.
```

Out[312]:

date	bookid	studentid
2017-08-29	1	1
2017-09-02	4	4
2017-09-07	1	4

SQL DDL

Part 2.1

Question 2.1.1

In [240]:
```
%%sql
alter table student add advisorid integer;
alter table student
add constraint faculty foreign key(advisorid) references faculty
    on update cascade
    on delete set NULL
```

```
Done.
Done.
```

Out[240]: []

In [241]:
```
%%sql
select * from student
```

```
5 rows affected.
```

Out[241]:

studentid	name	gpa	advisorid
1	Tom	3.3	None
2	John	3.8	None
3	Mary	3.0	None
4	Kris	3.6	None
5	Alex	3.5	None

Question 2.1.2

In [260]:
```
%%sql
alter table student
alter column gpa set not NULL;
alter table student
alter column gpa set default 3;
alter table student
```

```
add constraint gpa  check(gpa >= 0 and gpa <= 4)
```

```
Done.
Done.
Done.
```

Out[260]: []

In [261]:
```
%%sql
select * from student
```

```
5 rows affected.
```

Out[261]:

studentid	name	gpa
1	Tom	3.3
2	John	3.8
3	Mary	3.0
4	Kris	3.6
5	Alex	3.5

SQL Query

Part 2.2

Question 2.2.1

In [150]:
```
%%sql
select distinct studentid, name  from (
    student natural join(
        (select distinct studentid from book_checkout
         where (current_date - book_checkout.date > 30) )
    ) as overdue
)
```

```
2 rows affected.
```

Out[150]:

studentid	name
1	Tom
4	Kris

Question 2.2.2

In [314]:
```
%%sql
select distinct studentid, name from student
where (gpa > (select avg(gpa)
              from student) )
```

```
3 rows affected.
```

Out[314]:

studentid	name

4	Kris
2	John
5	Alex

Question 2.2.3

In [316]:
```
%%sql
select facultyid, name from faculty
where( facultyid not in (select instructorid from course) and faculty.sa
lary > 8000 )
```

```
1 rows affected.
```

Out[316]:

facultyid	name
5	Paul

Question 2.2.4

In [320]:
```
%%sql
select bookid, title from book
where bookid in (select textbookid from course
group by textbookid having count(*) >1)
```

```
1 rows affected.
```

Out[320]:

bookid	title
2	Database System Concept

Question 2.2.5

In [322]:
```
%%sql
select studentid, name from (
    book_checkout natural join book natural join student)
group by studentid, name having sum(price) > 100
```

```
1 rows affected.
```

Out[322]:

studentid	name
4	Kris

Question 2.2.6

In [326]:
```
%%sql
select distinct studentid, name from(
    student natural join (
        select studentid, bookid from book_checkout
        except all(
            select studentid, textbookid from
            (enroll natural join course) as course_complete
        )
    )as excludeed
```

```
)
```

1 rows affected.

Out[326]:

studentid	name
4	Kris

Question 2.2.7

In [335]:
```
%%sql
select studentid, name from student
where(
    studentid not in(
        select studentid from(
        student natural join(
            (select studentid, count(courseid) as course_number
            from enroll group by studentid) as work_load
            natural left outer join(
                select studentid, count(bookid) as checknum
                from book_checkout
                group by studentid
            )as status
        )
    )where (checknum is null or checknum < course_number)
    )
)
```

2 rows affected.

Out[335]:

studentid	name
3	Mary
4	Kris

Question 2.2.8

In [338]:
```
%%sql
select bookid, title from(
    book natural left outer join(
        select bookid, count(studentid) as num
        from book_checkout
        group by bookid
    ) as checked_out
)

where(
    case
    when num is null then total_copies >= 3
    else total_copies-num >=3
    end
)
```

2 rows affected.

Out[338]:

bookid	title
2	Database System Concept

3	Stochastic Calculus for Finance I

Question 2.2.9

In [340]:
```
%%sql
create temporary table tmp2 as
select courseid,price from(
    book natural join (select courseid,
                        textbookid as bookid from course)as cor
);

select courseid, title from course
where(
    courseid in (
        select courseid from tmp2
        where price = (select max(price) from tmp)
    )
)
```

```
5 rows affected.
1 rows affected.
```

Out[340]:

courseid	title
1	Algorithm

Question 2.2.10

In [346]:
```
%%sql
select distinct studentid, name from(
    enroll natural join course natural join student)
where (title like'%finance%'or title like'%Finance%')
```

```
2 rows affected.
```

Out[346]:

studentid	name
4	Kris
5	Alex

SQL Updates

Question 2.3.1

In [347]:
```
%%sql
delete from course
where course.courseid not in(select courseid from enroll group by course
id having count(*)>0)
```

```
0 rows affected.
```

Out[347]:
```
[]
```

Question 2.3.2

In [210]:
```
%%sql
insert into book(title,price,total_copies) values('Distribuition and Clo
ud Computing', 50.00,4);
```

```
1 rows affected.
```

Out[210]: []

In [212]:
```
%%sql
select * from book
```

```
5 rows affected.
```

Out[212]:

bookid	title	price	total_copies
1	Introduction of Algorithm	84.66	4
2	Database System Concept	74.99	5
3	Stochastic Calculus for Finance I	41.02	3
4	Stochastic Calculus for Finance II	55.22	3
5	Distribuition and Cloud Computing	50.0	4

Question 2.2.4

In [351]:
```
%%sql
update student set gpa = (case
     when gpa is NULL then 0
     when gpa > 4 then 4
     when gpa < 0 then 0
     end)
where(gpa is NULL or gpa>4 or gpa <0)
```

```
1 rows affected.
```

Out[351]: []

Question 2.3.4

In [218]:
```
%%sql
update faculty set salary = salary + 1000
```

```
5 rows affected.
```

Out[218]: []

Question 2.3.5

In [219]:
```
%%sql
select * from faculty
```

```
5 rows affected.
```

Out[219]:

facultyid	name	salary
1	James	71000

2	Sarah	61000
3	Jay	81000
4	Rachel	71000
5	Paul	86000