```
create table publisher
(name varchar(20),
phone integer,
address varchar(20),
constraint pkp primary key(name));

create table book
(book_id varchar(8),
title varchar(20),
pub_year integer,
publisher_name varchar(20),
constraint pkb primary key(book_id),
constraint fkb foreign key(publisher_name) references
publisher(name));

create table book_authors
(book_id varchar(8),
author_name varchar(20),
constraint pkba primary key(book_id,author_name),
constraint fkba foreign key(book_id) references
book(book_id) on delete cascade);

create table library_branch
(branch_id varchar(8),
address varchar(20),
branch_name varchar(20),
constraint pklb primary key(branch_id));

create table book_copies
(book_id varchar(8),
branch_id varchar(8),
no_of_copies integer,
constraint pkbc primary key(book_id,branch_id),
constraint fkbc foreign key(book_id) references
book(book_id) on delete cascade,
constraint fkbb foreign key(branch_id) references
library_branch(branch_id));

create table book_lending
(book_id varchar(8),
branch_id varchar(8),
card_no integer,
date_out date,
due_date date,
constraint pkbl primary key(book_id,branch_id,card_no),
constraint fkbl foreign key(book_id) references
book(book_id) on delete cascade);
```

queries

1. Retrieve details of all books in the library – id, title, name of

publisher,authors, number of copies in each branch, etc.

```
select b.book_id, b.title, b.publisher_name,
a.author_name, c.no_of_copies, l.branch_id
from book b, book_authors a, book_copies c,
library_branch l
where b.book_id=a.book_id
and b.book_id=c.book_id
and l.branch_id=c.branch_id;
```

2. Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun 2017.

```
select card_no
from book_lending
where date_out between '01-jan-2017' and '01-jun-2017'
group by card_no
having count(*)>3;
```

3. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation.

```
select * from book;
select * from book_copies;

delete from book
where book_id='03';
select * from book;
select * from book_copies;
```

4. Partition the BOOK table based on year of publication. Demonstrate its working with a simple query.

```
create table bookpart
(partition by range (pub_year)
partition p1 values less than(2018),
partition p2 values less than (maxvalue)
as select * from book);

select table_name,partition_name from
user_tab_partitions;
select * from bookpart partition (p1);
select * from bookpart partition (p2);
```

5. Create a view of all books and its number of copies that are currently available in the Library.

```
create view bc as select b.book_id,c.title,b.branch_id,
(b.no_of_copies-(select count(*) from book_lending where
b.book_id=book_id and b.branch_id=branch_id)) as
no_copy
from book_copies b,book c
where b.book_id=c.book_id;
```