Assignment 1: Design a database schema for a library system, including tables, fields, and constraints like NOT NULL, UNIQUE, and CHECK. Include primary and foreign keys to establish relationships between tables.

```
create table book
(
name varchar2(30),
author varchar2(30),
bookid number(10) primary key check(length(booid)>2)
sectionno number(10),
constraints ref foreign key(sectionno) references section(sectionno)
);

create table section
(
sectionno number(10) primary key,
sectionname varchar2(30) not null unique,
);
```

Assignment 2: GitHub questions.

1. Select the names of all the products in the store.

```
select name from products;
```

2. Select the names and the prices of all the products in the store.

```
select name,price from products;
```

3. Select the name of the products with a price less than or equal to $200.

```
select name from products where price<=200;
```

4. Select all the products with a price between $60 and $120.

```
select * from products where price>=60 and price<=120.
```

5. Select the name and price in cents (i.e., the price must be multiplied by 100)

```
select name,price*100 as priceincents from products;
```

6. Compute the average price of all the products.

```
select avg(price) as avgprice from products;
```

7. Compute the average price of all products with manufacturer code equal to 2.

```
select avg(price) as avgprice from products where manufacturer=2;
```

8. Compute the number of products with a price larger than or equal to $180.

```
select count(price) from products where price>=180;
```

9. Select the name and price of all products with a price larger than or equal to $180, and sort first by price (in descending order), and then by name (in ascending order).

```
select name,price from products where price>=180
order by price desc
select name,price from products where price>=180
order by name;
```

10. Select all the data from the products, including all the data for each product's manufacturer.

```
select * from products
full outer join
on
products.code=manufactureres.code;
```

11. Select the product name, price, and manufacturer name of all the products.

```
select products.name,products.price,manufacturers from products
inner join manufacturers
on products.manufacturer=manufacturers.code
```

12. Select the average price of each manufacturer's products, showing only the manufacturer's code.

```
select products.manufacturer,avg(products.price) as avgprice from
products
group by products.manufacturer;
```

13. Select the average price of each manufacturer's products, showing the manufacturer's name.

select manufacturers.name,avgtable.avgprice from (select products.manufacturer,avg(products.price) as avgprice from products
group by products.manufacturer) avgtable
inner join manufacurers
on manufacturers.code=avgtable.manufacturer;

14. Select the names of manufacturer whose products have an average price larger than or equal to $150.

select avgname.name (select manufacturers.name,avgtable.avgprice from (select products.manufacturer,avg(products.price) as avgprice from products
group by products.manufacturer) avgtable
inner join manufacurers
on manufacturers.code=avgtable.manufacturer) avgname
where avgname.avgprice>=150;

15. Select the name and price of the cheapest product.

select products.name,products.price from products where price=(select min(price) from products);

16. Select the name of each manufacturer along with the name and price of its most expensive product.

select manufacturers.name as "manufacturer's name", maxcodetable."product's name", maxcodetable.maxprice as "price" from (select products.name as "product's name",maxtable.maxprice,products.manufacturer from products inner join (select products.manufacturer, max(products.price) as maxprice from products group by products.manufacturer) maxtable
on products.manufacturer=maxtable.manufacturer and products.price=maxtable.price) maxcodetable
inner join manufacturers on manufacturers.code=maxcodetable.manufacturer

17. Add a new product: Loudspeakers, $70, manufacturer 2.

insert into products values(11,'Loudspeakers',70,2);

18. Update the name of product 8 to "Laser Printer".

update products name='Laser Printer' where code=8;

19.Apply a 10% discount to all products.

```
update products set price=price*0.90;
```

20. Apply a 10% discount to all products with a price larger than or equal to $120.

```
update products set price=price*0.90 where price>=120;
```