1.创建 user 表

```
mysql> create table user(
    -> id int,
    -> name varchar(20),
    -> sex varchar(6),
    -> age int,
    -> phone varchar(20)
    -> );
Query OK, 0 rows affected (0.03 sec)
```

2.插入原始数据

```
mysql> insert into user value
    -> (1,'John Doe', 'Male', 25, '123-456-7890'),
    -> (2,'Jane Smith', 'Female', 31, '987-654-3210'),
    -> (3,'Bob Johnson', 'Male', 22, '555-123-4567');
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0
```

```
mysql> select * from user
    -> where age >= 20 and age <= 30;
+------+-------------+------+------+--------------+
| id   | name        | sex  | age  | phone        |
+------+-------------+------+------+--------------+
|    1 | John Doe    | Male |   25 | 123-456-7890 |
|    3 | Bob Johnson | Male |   22 | 555-123-4567 |
+------+-------------+------+------+--------------+
2 rows in set (0.01 sec)
```

3. 查询 user 表中所有年龄在 20-30 范围内的用户

4. 添加自己的个人信息，并添加几条和你姓名同姓的虚拟信息

```
mysql> insert into user value
    -> (4, 'wang hongwei', 'Male',20,'18221507609'),
    -> (5, 'wang hong', 'Female',25,'4864209'),
    -> (6, 'wang tain', 'Male',36,'116579'),
    -> (7, 'wang baoqiang', 'Female',29,'182609');
Query OK, 4 rows affected (0.00 sec)
Records: 4  Duplicates: 0  Warnings: 0
```

5. 查询 user 表中年龄在 20-30 范围内,名字包含“你的姓氏”的用户,并按照年龄从大到小排序输出

```
mysql> select * from user
    -> where name like 'wang%'
    -> and age <= 30 and age >= 20
    -> order by age desc;
+------+--------------+--------+------+-------------+
| id   | name         | sex    | age  | phone       |
+------+--------------+--------+------+-------------+
|    7 | wang baoqiang | Female |   29 | 182609      |
|    5 | wang hong     | Female |   25 | 4864209     |
|    4 | wang hongwei  | Male   |   20 | 18221507609 |
+------+--------------+--------+------+-------------+
3 rows in set (0.00 sec)
```

6. 计算 user 表中所有用户的平均年龄

```
mysql> select avg(age) as average_age from user;
+-------------+
| average_age |
+-------------+
|     26.8571 |
+-------------+
1 row in set (0.01 sec)
```

7.创建 team 表格

```
mysql> create table team(
    -> id int primary key auto_increment,
    -> teamName varchar(50) not null
    -> );
Query OK, 0 rows affected (0.02 sec)
```

8.创建 score 表格

```
mysql> CREATE TABLE score (
    ->     id INT PRIMARY KEY AUTO_INCREMENT,
    ->     teamid INT,
    ->     userid INT,
    ->     score INT,
    ->     FOREIGN KEY (teamid) REFERENCES team(id) ON DELETE CASCADE,
    ->     FOREIGN KEY (userid) REFERENCES user(id) ON DELETE CASCADE
    -> );
Query OK, 0 rows affected (0.03 sec)
```

9.

```
mysql> INSERT INTO team (teamName) VALUES
    -> ('ECNU'),
    -> ('SHU'),
    -> ('FDU');
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0
```

```
mysql> INSERT INTO score (teamid, userid, score) VALUES
    ->     (1, 1, 85),
    ->     (1, 2, 65),
    ->     (2, 3, 30),
    ->     (2, 6, 79),
    ->     (3, 4, 70),
    ->     (3, 5, 0);
Query OK, 6 rows affected (0.00 sec)
Records: 6  Duplicates: 0  Warnings: 0
```

```
mysql> select sum(coalesce(score,0)) as total_score
    -> from score
    -> join team on score.teamid = teamid
    -> where team.teamName = 'ECNU';
+-------------+
| total_score |
+-------------+
|         329 |
+-------------+
1 row in set (0.00 sec)
```

10.

```
mysql> delete from user where id = 4;
Query OK, 1 row affected (0.00 sec)
```