

Step1 创建表格,插入表格

```
mysql> create TABLE user (  
  -> id INT AUTO_INCREMENT PRIMARY KEY,  
  -> name VARCHAR(100) NOT NULL,  
  -> sex VARCHAR(10) NOT NULL,  
  -> age INT NOT NULL,  
  -> phone VARCHAR(15) NOT NULL  
  -> );
```

Query OK, 0 rows affected (0.04 sec)

```
mysql> INSERT INTO user (name, sex, age, phone) VALUES  
  -> ('John Doe', 'Male', 25, '123-456-7890'),  
  -> ('Jane Smith', 'Female', 31, '987-654-3210'),  
  -> ('Bob Johnson', 'Male', 22, '555-123-4567');
```

Query OK, 3 rows affected (0.02 sec)

Records: 3 Duplicates: 0 Warnings: 0

```
mysql> SELECT * FROM user;  ⇒ 显示表格全貌
```

id	name	sex	age	phone
1	John Doe	Male	25	123-456-7890
2	Jane Smith	Female	31	987-654-3210
3	Bob Johnson	Male	22	555-123-4567

3 rows in set (0.00 sec)

Step2. 查询年龄在20-30之间的用户

```
mysql> SELECT * FROM user  
  -> WHERE age BETWEEN 20 AND 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 (0.00 sec)

Step3. 插入信息

```
mysql> INSERT INTO user (name, sex, age, phone) VALUES  
-> ('Chloris Zhou', 'Female', 19, '511-722-1234');  
Query OK, 1 row affected (0.02 sec)
```

```
mysql> INSERT INTO user (name, sex, age, phone) VALUES  
-> ('Alice Zhou', 'Female', 24, '555-001-2345'),  
-> ('Brian Zhou', 'Male', 27, '555-002-3456'),  
-> ('Cathy Zhou', 'Female', 22, '555-003-4567'),  
-> ('David Zhou', 'Male', 30, '555-004-5678');  
Query OK, 4 rows affected (0.02 sec)  
Records: 4 Duplicates: 0 Warnings: 0
```

```
mysql> select * from user;
```

id	name	sex	age	phone
1	John Doe	Male	25	123-456-7890
2	Jane Smith	Female	31	987-654-3210
3	Bob Johnson	Male	22	555-123-4567
4	Chloris Zhou	Female	19	511-722-1234
5	Alice Zhou	Female	24	555-001-2345
6	Brian Zhou	Male	27	555-002-3456
7	Cathy Zhou	Female	22	555-003-4567
8	David Zhou	Male	30	555-004-5678

8 rows in set (0.00 sec)

Step4. 查询, 并按年龄输出

```
mysql> select * from user  
-> where age between 20 and 30 and name like '%Zhou%';  
-> order by age DESC;
```

id	name	sex	age	phone
8	David Zhou	Male	30	555-004-5678
6	Brian Zhou	Male	27	555-002-3456
5	Alice Zhou	Female	24	555-001-2345
7	Cathy Zhou	Female	22	555-003-4567

4 rows in set (0.02 sec)

Step 5 计算平均年龄

```
mysql> select AVG(age) AS average_age from user;
+-----+
| average_age |
+-----+
|      25.0000 |
+-----+
1 row in set (0.02 sec)
```

Step 6. 新建 team, score 两张表

```
mysql> CREATE TABLE team (
->     id INT AUTO_INCREMENT PRIMARY KEY,
->     teamName VARCHAR(100) NOT NULL
-> );
Query OK, 0 rows affected (0.05 sec)

mysql> CREATE TABLE score (
->     id INT AUTO_INCREMENT PRIMARY KEY,
->     teamid INT,
->     userid INT,
->     score INT,
->     FOREIGN KEY (teamid) REFERENCES team(id),
->     FOREIGN KEY (userid) REFERENCES user(id)
-> );
Query OK, 0 rows affected (0.10 sec)
```

Step 7. 插入信息后, 查询 "ECNU" 中年龄小于 20 的用户.

```
mysql> -- 向 team 表插入记录
mysql> INSERT INTO team (teamName) VALUES
-> ('ECNU'),
-> ('ABC'),
-> ('XYZ'),
-> ('DEF'),
-> ('GHI'),
-> ('JKL'),
-> ('MNO'),
-> ('ECNU'); -- 再次插入 "ECNU"
Query OK, 8 rows affected (0.01 sec)
Records: 8 Duplicates: 0 Warnings: 0

mysql> -- 向 score 表插入记录
mysql> INSERT INTO score (teamid, userid, score) VALUES
-> (1, 1, FLOOR(RAND() * 101)), -- teamid 为 ECNU 的 ID
-> (2, 2, FLOOR(RAND() * 101)), -- teamid 为 ABC 的 ID
-> (3, 3, FLOOR(RAND() * 101)), -- teamid 为 XYZ 的 ID
-> (1, 4, FLOOR(RAND() * 101)), -- teamid 为 ECNU 的 ID
-> (4, 5, FLOOR(RAND() * 101)), -- teamid 为 DEF 的 ID
-> (5, 6, FLOOR(RAND() * 101)), -- teamid 为 GHI 的 ID
-> (6, 7, FLOOR(RAND() * 101)), -- teamid 为 JKL 的 ID
-> (7, 8, FLOOR(RAND() * 101)); -- teamid 为 MNO 的 ID
Query OK, 8 rows affected (0.02 sec)
Records: 8 Duplicates: 0 Warnings: 0
```

```
mysql> SELECT u.* FROM user u
      -> JOIN score s ON u.id = s.userid
      -> JOIN team t ON s.teamid = t.id -- 根据实际的 team ID
      -> WHERE t.teamName = 'ECNU' AND u.age < 20;
+----+-----+-----+-----+-----+
| id | name       | sex   | age | phone       |
+----+-----+-----+-----+-----+
|  4 | Chloris Zhou | Female |  19 | 511-722-1234 |
+----+-----+-----+-----+-----+
1 row in set (0.01 sec)
```

Step 8. 计算ECNU队伍总分.

```
mysql> SELECT SUM(COALESCE(s.score, 0)) AS total_score
      -> FROM score s
      -> JOIN team t ON s.teamid = t.id
      -> WHERE t.teamName = 'ECNU';
+-----+
| total_score |
+-----+
|           85 |
+-----+
1 row in set (0.01 sec)
```

Step 9. 删除user中个人信息记录

不能直接删除.

```
mysql> DELETE FROM user
      -> WHERE name = 'Chloris Zhou';
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key
constraint fails ('mydatabase`.`score`, CONSTRAINT `score_ibfk_2` FOREIGN
KEY (`userid`) REFERENCES `user` (`id`))
mysql> DELETE FROM score WHERE userid IN (SELECT id FROM user WHERE name
      = 'Chloris Zhou');
Query OK, 1 row affected (0.01 sec)

mysql> DELETE FROM user WHERE name = 'Chloris Zhou';
Query OK, 1 row affected (0.01 sec)
```