**任课教师：张涵翠、霍戌文**

**《数据库系统实训》**

**（2020-2021学年第2学期）**

**实**

**验**

**报**

**告**

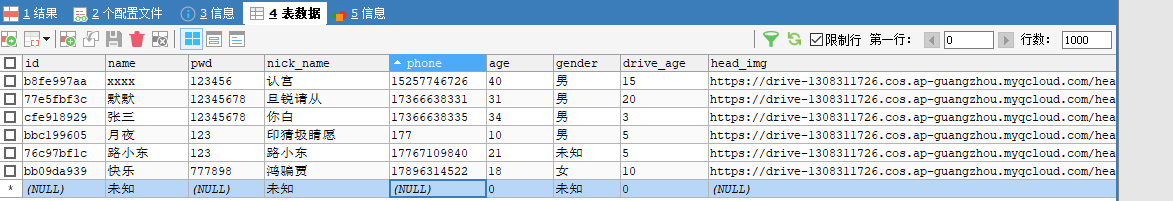
**学号：2019329621004**

**姓名：梅雨欣**

**班级：计算机科学与技术19（1）班**

**实验一：单表查询**

**本章使用用户user表进行操作：**



1. **第四章---检索数据**

**1.1 检索单个列**

**SQL语句描述：SELECT id FROM USER**

**查询功能描述：查询所有用户的id**

**执行结果截图：**

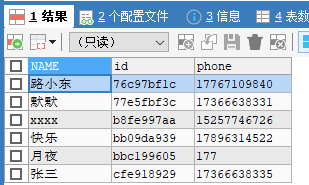


**1.2 检索多个列**

**SQL语句描述：SELECT NAME,id,phone FROM USER**

**查询功能描述：查询所有用户的名字、id和电话号码**

**执行结果截图：**

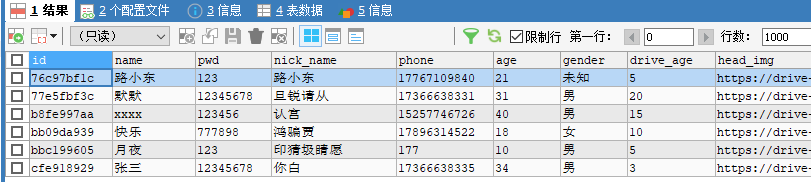


**1.3 检索所有列**

**SQL语句描述：SELECT \* FROM USER**

**查询功能描述：查询所有用户的信息**

**执行结果截图：**

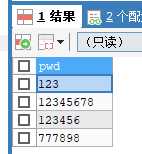


**1.4 检索不同行**

**SQL语句描述：SELECT DISTINCT pwd FROM USER '**

**查询功能描述：查询所有用户的不重复密码**

**执行结果截图：**



**1.5 限制结果，使用limit子句**

**SQL语句描述：SELECT DISTINCT phone FROM USER LIMIT 1,2**

**查询功能描述：查询所有用户的不重复电话号码，返回第1行开始的2行**

**执行结果截图：**

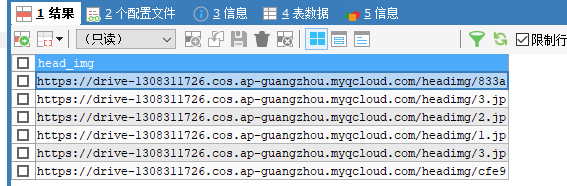


**1.6使用完全限定的表名**

**SQL语句描述：SELECT user.head\_img FROM USER2**

**查询功能描述：查询user表中所有用户的头像**

**执行结果截图：**



1. **第五单---排序数据**

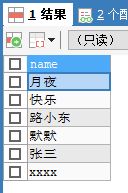
**2.1 单列排序**

**SQL语句描述：**

|  |
| --- |
| **SELECT NAME**  **FROM USER**  **ORDER BY age** |

**查询功能描述：查询所有用户中的用户姓名，并按照年龄升序排序**

**执行结果截图：**



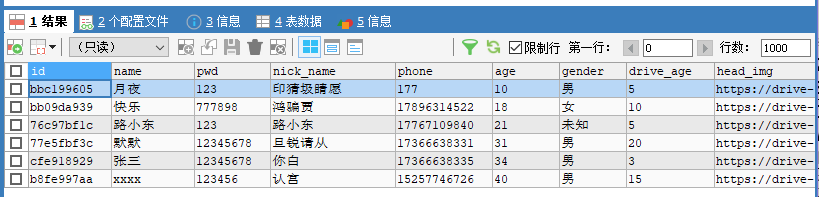
**2.2 按多个列排序**

**SQL语句描述：**

|  |
| --- |
| **SELECT \***  **FROM USER**  **ORDER BY age, drive\_age** |

**查询功能描述：查询所有用户中的用户信息，并按照年龄、驾龄升序排序**

**执行结果截图：**



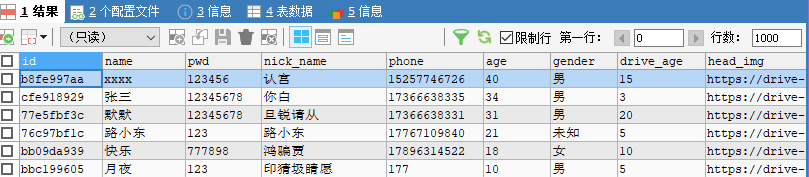
**2.3 指定方向排序**

**SQL语句描述：**

|  |
| --- |
| **SELECT \***  **FROM USER**  **ORDER BY age DESC, drive\_age** |

**查询功能描述：查询所有用户的用户信息，并先按照年龄降序排序，再按照驾龄升序排序**

**执行结果截图：**



**2.4 最大值**

**SQL语句描述：**

|  |
| --- |
| **SELECT age**  **FROM USER**  **ORDER BY drive\_age LIMIT 1** |

**查询功能描述：查询所有用户中驾龄最大者的年龄**

**执行结果截图：**



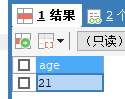
**2.5 最小值**

**SQL语句描述：**

|  |
| --- |
| **SELECT age**  **FROM USER**  **ORDER BY drive\_age DESC LIMIT 1** |

**查询功能描述：查询所有用户中驾龄最小者的年龄**

**执行结果截图：**



1. **第六章---过滤数据**

**3.1 使用where子句**

**SQL语句描述：**

|  |
| --- |
| **SELECT head\_img**  **FROM USER**  **WHERE phone='17767109840'** |

**查询功能描述：查询所有用户中号码为‘17767109840’的用户头像地址**

**执行结果截图：**



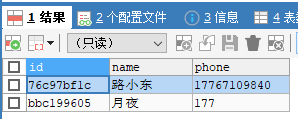
**3.2 检查单个值**

**SQL语句描述：**

|  |
| --- |
| **SELECT id,NAME,phone**  **FROM USER**  **WHERE pwd='123'** |

**查询功能描述：查询所有用户中密码为‘123’的id、姓名和电话号码**

**执行结果截图：**



**3.3 不匹配检查**

**SQL语句描述：**

|  |
| --- |
| **SELECT id,NAME,phone**  **FROM USER**  **WHERE pwd<>'123'** |

**查询功能描述：查询所有用户中密码不是‘123’的id、姓名和电话号码**

**执行结果截图：**



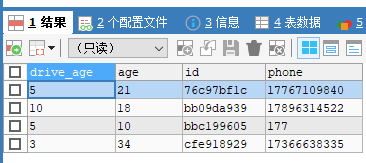
**3.4 范围值检查**

**SQL语句描述：**

|  |
| --- |
| **SELECT drive\_age, age, id, phone**  **FROM USER**  **WHERE drive\_age BETWEEN 3 AND 10** |

**查询功能描述：查询所有用户中驾龄大于3年且小于10年的用户驾龄、年龄、id和电话号码**

**执行结果截图：**



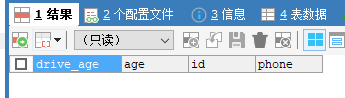
**3.5 空值检查**

**SQL语句描述：**

|  |
| --- |
| **SELECT drive\_age, age, id, phone**  **FROM USER**  **WHERE head\_img IS NULL** |

**查询功能描述：查询所有用户中头像为空的用户驾龄、年龄、id和电话号码**

**执行结果截图：**



1. **第七章---数据过滤**

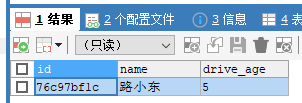
**4.1 组合where子句**

**SQL语句描述：**

|  |
| --- |
| **SELECT id,NAME,drive\_age**  **FROM USER**  **WHERE age = 21 AND drive\_age >= 3** |

**查询功能描述：查询所有用户中年龄为21岁且驾龄大于3年的用户id、姓名和驾龄**

**执行结果截图：**



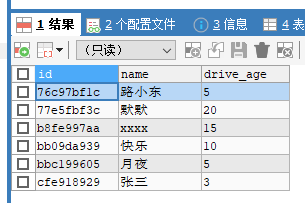
**4.2 OR操作符**

**SQL语句描述：**

|  |
| --- |
| **SELECT \***  **FROM USER**  **WHERE pwd='123'** |

**查询功能描述：查询所有用户中年龄为21岁或驾龄大于3年的用户id、姓名和驾龄**

**执行结果截图：**



**4.3 比较次序**

**SQL语句描述：**

|  |
| --- |
| **SELECT id,NAME,drive\_age**  **FROM USER**  **WHERE age > 20 OR(drive\_age >= 3 AND pwd <> '1234')** |

**查询功能描述：查询所有用户中驾龄大于3年并且密码不为1234的用户，或年龄大于20岁的用户id、姓名和驾龄**

**执行结果截图：**



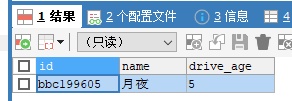
**4.4 IN操作符**

**SQL语句描述：**

|  |
| --- |
| **SELECT id,NAME,drive\_age**  **FROM USER**  **WHERE age IN (5,10)** |

**查询功能描述：查询所有用户中年龄在5岁和10岁之间的用户id、姓名和驾龄**

**执行结果截图：**



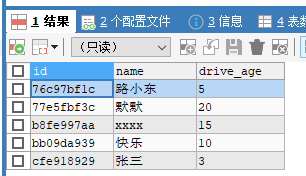
**4.5 NOT操作符**

**SQL语句描述：**

|  |
| --- |
| **SELECT id,NAME,drive\_age**  **FROM USER**  **WHERE age NOT IN (5,10)** |

**查询功能描述：查询所有用户中年龄不在5岁和10岁之间的用户id、姓名和驾龄**

**执行结果截图：**



1. **第八章---通配符**

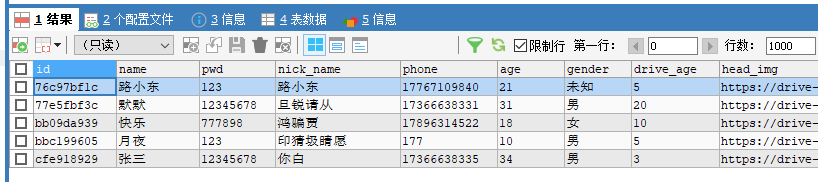
**5.1 LIKE通配符**

**SQL语句描述：**

|  |
| --- |
| **SELECT \***  **FROM USER**  **WHERE phone LIKE '17%'** |

**查询功能描述：查询所有用户电话号码以‘17’开头的用户信息**

**执行结果截图：**



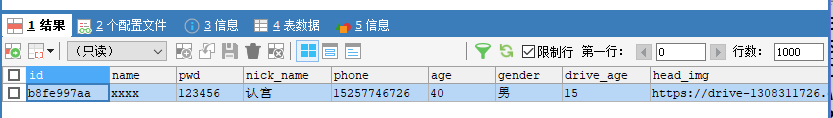
**5.2 下划线通配符**

**SQL语句描述：**

|  |
| --- |
| **SELECT \***  **FROM USER**  **WHERE NAME LIKE 'xxx\_'** |

**查询功能描述：查询所有用户中名字以‘xxx’开头的四字符用户信息**

**执行结果截图：**



1. **第九章---正则表达式**

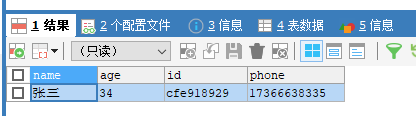
**6.1 基本字符匹配**

**SQL语句描述：**

|  |
| --- |
| **SELECT NAME, age, id, phone**  **FROM USER**  **WHERE NAME REGEXP '张三'** |

**查询功能描述：查询所有用户中姓名叫“张三”的用户姓名、年龄、id和电话号码**

**执行结果截图：**



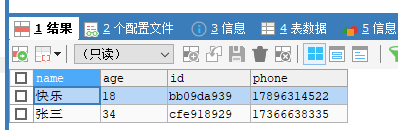
**6.2 进行OR匹配**

**SQL语句描述：**

|  |
| --- |
| **SELECT NAME, age, id, phone**  **FROM USER**  **WHERE NAME REGEXP '张三|快乐'** |

**查询功能描述：查询所有用户中姓名叫“张三”或者“快乐”的用户姓名、年龄、id和电话号码**

**执行结果截图：**



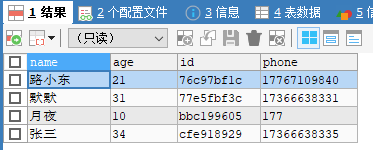
**6.3 匹配几个字符之一**

**SQL语句描述：**

|  |
| --- |
| **SELECT NAME, age, id, phone**  **FROM USER**  **WHERE pwd REGEXP '[^5]23'** |

**查询功能描述：查询所有用户中密码能匹配[^5]23的用户姓名、年龄、id和电话号码**

**执行结果截图：**



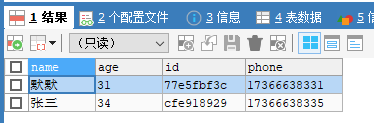
**6.4 匹配范围**

**SQL语句描述：**

|  |
| --- |
| **SELECT NAME, age, id, phone**  **FROM USER**  **WHERE pwd REGEXP '[1-5]23456'** |

**查询功能描述：查询所有用户中密码能匹配[1-5]23456的用户姓名、年龄、id和电话号码**

**执行结果截图：**



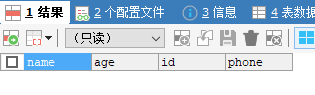
**6.5 匹配特殊字符**

**SQL语句描述：**

|  |
| --- |
| **SELECT NAME, age, id, phone**  **FROM USER**  **WHERE pwd REGEXP '\\.'** |

**查询功能描述：查询所有用户中密码能匹配\.的用户姓名、年龄、id和电话号码**

**执行结果截图：**



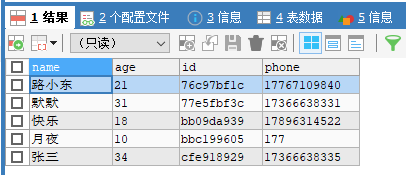
**6.6 匹配字符类**

**SQL语句描述：**

|  |
| --- |
| **SELECT NAME, age, id, phone**  **FROM USER**  **WHERE pwd REGEXP '[:digit:]'** |

**查询功能描述：查询所有用户中密码能匹配数字的用户姓名、年龄、id和电话号码**

**执行结果截图：**



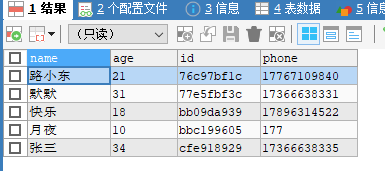
**6.7 匹配多个实例**

**SQL语句描述：**

|  |
| --- |
| **SELECT NAME, age, id, phone**  **FROM USER**  **WHERE pwd REGEXP '[[:digit:]]{3}'** |

**查询功能描述：查询所有用户中密码能匹配3个连续数字的用户姓名、年龄、id和电话号码**

**执行结果截图：**



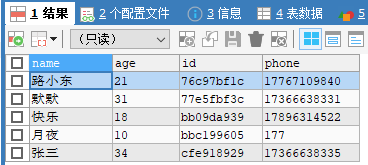
**6.8 定位符**

**SQL语句描述：**

|  |
| --- |
| **SELECT NAME, age, id, phone**  **FROM USER**  **WHERE pwd REGEXP '^[0-9\\.]'** |

**查询功能描述：查询所有用户中密码是匹配以数或小数点开始的数的用户姓名、年龄、id和电话号码**

**执行结果截图：**



1. **第十章---创建计算字段**

**7.1 拼接字段**

**SQL语句描述：**

|  |
| --- |
| **SELECT CONCAT(NAME,' (',id,')')**  **FROM USER**  **ORDER BY id** |

**查询功能描述：查询所有用户的姓名和id并拼接，按照id升序排序**

**执行结果截图：**



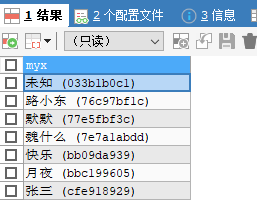
**7.2 使用别名**

**SQL语句描述：**

|  |
| --- |
| **SELECT CONCAT(NAME,' (',id,')') AS myx**  **FROM USER**  **ORDER BY id** |

**查询功能描述：查询所有用户中姓名和id并拼接取名为myx，并按照id升序排序**

**执行结果截图：**



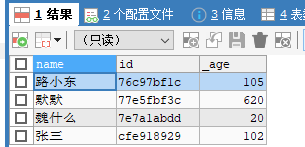
**3.3 执行算术计算**

**SQL语句描述：**

|  |
| --- |
| **SELECT NAME,id,age\*drive\_age AS \_age**  **FROM USER**  **WHERE age > 18** |

**查询功能描述：查询所有用户中年龄大于18岁的用户姓名和id,\_age为age\*drive\_age**

**执行结果截图：**



1. **第十一章---数据处理函数**

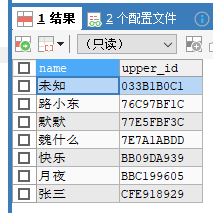
**8.1 文本处理函数**

**SQL语句描述：**

|  |
| --- |
| **SELECT NAME,UPPER(id) AS upper\_id**  **FROM USER**  **ORDER BY id** |

**查询功能描述：查询所有用户的姓名、大写id，并按照id升序排序**

**执行结果截图：**



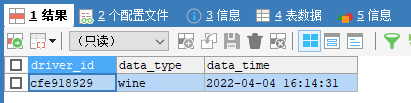
**8.2 日期和时间处理函数**

**SQL语句描述：**

|  |
| --- |
| **SELECT driver\_id, data\_type,data\_time**  **FROM data\_user**  **WHERE data\_time='2022-04-04 16:14:31'** |

**查询功能描述：查询所有用户健康数据中时间2022-04-04 16:14:31的检测用户id、类型、时间**

**执行结果截图：**



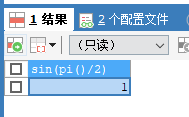
**8.3 数值处理函数**

**SQL语句描述：**

|  |
| --- |
| **SELECT SIN(PI()/2)** |

**查询功能描述：求出sin（π/2）的值**

**执行结果截图：**



1. **第十二章---汇总数据（聚集函数）**

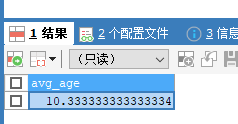
**9.1 聚集函数**

**SQL语句描述：**

|  |
| --- |
| **SELECT AVG(age) AS avg\_age**  **FROM USER**  **WHERE LENGTH(pwd) = 3** |

**查询功能描述：查询所有用户中密码长度为3的年龄平均数**

**执行结果截图：**



**9.2 聚集不同值**

**SQL语句描述：**

|  |
| --- |
| **SELECT AVG(DISTINCT drive\_age) AS AVG**  **FROM USER**  **WHERE LENGTH(pwd) = 3** |

**查询功能描述：查询所有用户中密码长度为3的不同驾龄的平均数**

**执行结果截图：**



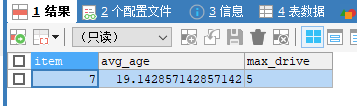
**9.3 组合聚集函数**

**SQL语句描述：**

|  |
| --- |
| **SELECT COUNT(\*) AS item,**  **AVG(age) AS avg\_age,**  **MAX(drive\_age) AS max\_drive**  **FROM USER** |

**查询功能描述：查询所有用户数目、平均年龄、最大驾龄**

**执行结果截图：**



1. **第十三章---分组数据**

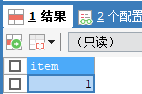
**10.1 数据分组**

**SQL语句描述：**

|  |
| --- |
| **SELECT COUNT(\*) AS item**  **FROM USER**  **WHERE id = '76c97bf1c'** |

**查询功能描述：查询id=‘76c97bf1c’的用户数目**

**执行结果截图：**



**10.2 创建分组**

**SQL语句描述：**

|  |
| --- |
| **SELECT id, COUNT(\*) AS item**  **FROM USER**  **GROUP BY pwd** |

**查询功能描述：查询所有用户按密码分类的id和数目**

**执行结果截图：**



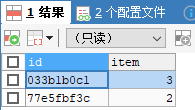
**10.3 过滤分组**

**SQL语句描述：**

|  |
| --- |
| **SELECT id, COUNT(\*) AS item**  **FROM USER**  **GROUP BY pwd**  **HAVING COUNT(\*)>1** |

**查询功能描述：查询所有用户按密码分类的、数目大于2的id和数目**

**执行结果截图：**



**10.4 分组和排序**

**SQL语句描述：**

|  |
| --- |
| **SELECT id, COUNT(\*) AS item**  **FROM USER**  **GROUP BY pwd**  **HAVING COUNT(\*)>1**  **ORDER BY item** |

**查询功能描述：查询所有用户按密码分类的、数目大于2的id和数目，并按照item数目升序排序**

**执行结果截图：**

