1.列约束

(1)唯一约束 —— unique

声明了唯一约束的列不允许出现重复的值，允许插入null，甚至多个null；可能会影响数据的排序

练习：给商品表laptop的标题列添加唯一约束，并插入数据测试

(2)默认值约束 —— default

可以使用default关键字来设置默认值，具体应用方式有两种

insert into family values(40, default); #通过default调用当前列的默认值

insert into family(fid) values(50); #给指定的列提供了值，没有出现的列自动调用默认值

练习：给商品表laptop的价格列添加默认值为3000，插入数据应用默认值

(3)检查约束 —— check

也称为自定义约束

create table student(

score tinyint check(score>=0 and score<=100)

);

mysql不支持检查约束，会影响数据的插入速度

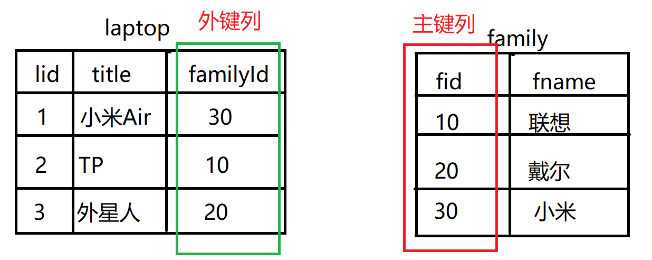
后期用JS来完成验证

(4)外键约束

声明了外键约束的列，插入的值必须得在另一个表的主键列中出现过才行；外键列要和对应的主键列类型要保持一致

foreign key(外键列) references 另一个表(主键列)

|  |
| --- |
| 外键约束使用目的是为了让两个表之间产生关联 |



2.自增列

auto\_increment 自动增长，如果设置了自增列，在插入数据的时候，只需要赋值为null，就会获取当前的最大值然后加1插入

注意事项：

自增列只能添加在整数形式的主键列

允许手动赋值

练习：编写脚本文件01\_tedu.sql，先丢弃再创建数据库tedu，设置编码utf8，进入数据库创建保存部门数据的表dept，包含编号did(主键、自增)，部门名称dname(唯一)，插入以下数据

10 研发部 20 运营部 30 市场部 40 测试部

创建保存员工数据的表emp，包含编号eid（主键，自增），姓名ename(非空约束)，性别sex(默认值约束)，生日birthday，工资salary，所属部门编号deptId(外键约束)

3.简单查询

(1)查询特定的列

示例：查询出所有员工的编号和姓名

select eid,ename from emp;

练习：查询出所有员工的姓名、性别、生日、工资

select ename,sex,birthday,salary from emp;

(2)查询所有的列

select eid,ename,sex,birthday,salary,deptId from emp;

select \* from emp;

(3)给列起别名

示例：查询出所有员工的编号和姓名，使用汉字别名

select eid as 编号, ename as 姓名 from emp;

练习：查询出所有员工的姓名、生日、工资，使用汉字别名

Select ename 姓名,birthday 生日,salary 工资 from emp;

练习：查询出所有员工的姓名和性别，使用一个字母作为别名

select ename a,sex b from emp;

(4)显示不同的记录

示例：查询出都有哪些性别的员工

select distinct sex from emp;

练习：查询出员工都分布在哪些部门

select distinct deptId from emp;

(5)查询时执行计算

select 2+3+4+5;

练习：查询所有员工的姓名及其年薪

select ename,salary\*12 from emp;

练习：假设每个员工工资增加1000，年终奖20000，查询出所有员工的性别及其年薪，使用别名

select sex a,(salary+1000)\*12+20000 b from emp;

(6)查询的结果排序

示例：查询出所有的部门，结果按照编号升序排列

Select \* from dept order by did asc; #ascendant 升序的

示例：查询出所有的部门，结果按照编号降序排列

Select \* from dept order by did desc;

describe 描述

descendant 降序的

练习：查询出所有的员工，结果按照工资的降序排列

select \* from emp order by salary desc;

练习：查询出所有的员工，结果按照年龄从大到小排列

select \* from emp order by birthday asc;

练习：查询出所有的员工，结果按照姓名升序排列

select \* from emp order by ename;

|  |
| --- |
| 字符串排序，按照字符的Unicode码排列  不加排序规则，默认是按照升序排列 |

练习：查询出所有的员工，按照工资降序排列，如果工资相同按照姓名排列

select \* from emp order by salary desc,ename;

练习：查询出所有的员工，按照性别排列，如果性别相同按照年龄从大到小排列

Select \* from emp order by sex,birthday;

(7)条件查询

示例：查询出编号为5的员工

select \* from emp where eid=5;

练习：查询出姓名为king的员工

Select \* from emp where ename='king';

练习：查询出20号部门的员工有哪些

select \* from emp where deptId=20;

练习：查询出所有的女员工有哪些

select \* from emp where sex=0;

练习：查询出工资在8000以上的员工有哪些

select \* from emp where salary>8000;

|  |
| --- |
| > < >= <= = !=(不等于) |

练习：查询出不在20号部门的员工有哪些

select \* from emp where deptId!=20;

练习：查询出没有明确部门的员工有哪些

select \* from emp where deptId is null;

练习：查询出有明确部门的员工有哪些

select \* from emp where deptId is not null;

练习：查询出工资在5000~8000之间的员工有哪些

select \* from emp where salary>=5000 and salary<=8000;

select \* from emp where salary>=5000 && salary<=8000;

|  |
| --- |
| and(&&) 并且  or(||) 或者 |

练习：查询出工资在5000以下或者8000以上的员工有哪些

select \* from emp where salary<5000 or salary>8000;

select \* from emp where salary<5000 || salary>8000;

练习：查询出1993年出生的员工有哪些

select \* from emp where birthday>='1993-1-1' and birthday<='1993-12-31';

练习：查询出20号部门或者30号部门的员工有哪些

select \* from emp where deptId=20 or deptId=30;

select \* from emp where deptId in(20,30);

练习：查询出不在20号部门并且也不在30号部门的员工有哪些

select \* from emp where deptId!=20 and deptId!=30;

select \* from emp where deptId not in(20,30);

(8)模糊条件查询

示例：查询出姓名中含有字母a的员工有哪些

select \* from emp where ename like '%a%';

练习：查询出姓名中以a结尾的员工有哪些

select \* from emp where ename like '%a';

练习：查询出姓名中倒数第2个字符是a的员工有哪些

select \* from emp where ename like '%a\_';

% 匹配任意个字符 >=0

\_ 匹配任意一个字符 =1

以上两个匹配符必须结合like关键字使用

课后任务

(1)复习今天内容，整理思维导图

(2)练习

删除代码保留注释部分，重新编写查询语句

查询出工资在8000以上的男员工的姓名、生日、工资，结果按照工资的降序排列，使用别名显示。

查看学子商城数据库设计

(3)预习JS第1天

http://www.codece.com/archives/187