复习

存储中文Unicode编码 utf8

把**脚本**文件使用**客户端**提交给**服务器**

列类型

数值类型

TINYINT/SMALLINT/INT/BIGINT/

FLOAT/DOUBLE/DECIMAL/BOOL

日期时间类型

DATE/TIME/DATETIME

字符串类型

VARCHAR/CHAR/TEXT

列约束

PRIMARY KEY

UNIQUE

1.列约束

(1)主键约束——PRIMARY KEY

(2)唯一约束——UNIQUE

(3)非空约束——NOT NULL

声明了非空约束的列上不能插入NULL

(4)默认值约束——DEFAULT

可以使用DEFAULT关键字声明默认值，有两种方式可以应用默认值

INSERT INTO xz\_laptop\_family VALUES(40,'苹果',DEFAULT);

INSERT INTO xz\_laptop\_family(fid,fname) VALUES(50,'华硕');

(5)检查约束——CHECK

检查约束可以对插入的数据进行检验

CREATE TABLE student(

age TINYINT CHECK(age>=18 AND age<=60 )

);

MySQL不支持检查约束，会降低数据的插入速度。

(6)外键约束——FOREIGN KEY

声明了外键约束的列，取值必须在另一个表的主键列上出现过，列类型要保持一致，取值可以是NULL

FOREIGN KEY(familyId)

REFERENCES xz\_laptop\_family(fid)

2.MySQL中的自增列

AUTO\_INCREMENT: 自动增长，假入一个列声明了自增列，无需手动赋值，直接设置为NULL，会获取当前的最大值，然后加1插入。

注意：

自增列允许手动赋值

只适用于整数型的主键列上

练习：创建脚本文件02\_tedu.sql，创建数据库tedu，设置编码。

创建部门表dept (did、dname)

10 研发部 20 市场部 30运营部 40 测试部

创建员工表emp(eid、ename、sex、birthday生日、salary工资、deptId所属部门编号)

插入15条记录，有一个员工没有确定部门，有一个部门没有员工。

3.简单查询

(1)查询特定的列

示例：查询所有员工的姓名、工资、生日

SELECT ename,salary,birthday FROM emp;

练习：查询所有员工的编号，姓名，性别，所属部门编号

SELECT eid,ename,sex,deptId FROM emp;

(2)查询所有的列

SELECT \* FROM emp;

SELECT eid,ename,sex,birthday,salary,deptId FROM emp;

(3)给列起别名

示例：查询所有员工的姓名和工资，使用汉字的别名显示

SELECT ename AS 姓名,salary AS 工资 FROM emp;

练习：查询所有员工的编号，姓名，性别，生日，用中文别名

SELECT eid AS 编号,ename AS 姓名,sex AS 性别,birthday AS 生日 FROM emp;

练习：查询所有员工的编号，姓名，使用1个英文字母别名

SELECT eid a,ename b FROM emp;

|  |
| --- |
| 注意：AS关键字是可以省略的，保留空格。 |

(4)只显示不同的记录/合并相同的记录

示例：查询出员工都在哪些部门

SELECT DISTINCT deptId FROM emp;

练习：查询出都有哪些性别的员工

SELECT DISTINCT sex FROM emp;

(5)在查询时执行计算

示例：计算2\*33/47-8+10

SELECT 2\*33/47-8+10;

练习：查询出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500元，年终奖5000元，查询所有员工的姓名及其年薪，要给列起别名

SELECT ename AS 姓名,(salary+500)\*12+5000 AS 年薪 FROM emp;

(6)查询结果集的排序

示例：查询所有的部门，结果按照部门编号升序排序

SELECT \* FROM dept ORDER BY did ASC; #ascendant

示例：查询所有的部门，结果按照部门编号降序排序

SELECT \* FROM dept ORDER BY did DESC; #descendant

练习：查询所有员工信息，结果按照工资由大到小排序

SELECT \* FROM emp ORDER BY salary DESC;

练习：查询所有员工，结果按照年龄由大到小排序

SELECT \* FROM emp ORDER BY birthday ASC;

练习：查询所有员工，结果按照年龄由小到大排序

SELECT \* FROM emp ORDER BY birthday DESC;

练习：查询所有员工，结果按照姓名的升序排序

SELECT \* FROM emp ORDER BY ename;

练习：查询所有员工，结果按照工资降序排序，如果工资相同，按照姓名排序。

SELECT \* FROM emp ORDER BY salary DESC,ename;

练习：查询所有员工，结果按照性别排序，如果性别相同，按照工资的升序排序。

SELECT \* FROM emp ORDER BY sex,salary;

|  |
| --- |
| ORDER BY 可以按照**数值、日期/时间、字符串**来排序  默认按照ASC升序排序 |

(7)条件查询

示例：查询出编号为5的员工所有信息

SELECT \* FROM emp WHERE eid=5;

练习：查询出姓名为king的员工的编号，工资，生日。

SELECT eid,salary,birthday FROM emp WHERE ename='king';

练习：查询出20号部门下所有员工信息

SELECT \* FROM emp WHERE deptId=20;

练习：查询出男员工的所有信息

SELECT \* FROM emp WHERE sex=1;

练习：查询出工资大于等于5000的员工所有信息

SELECT \* FROM emp WHERE salary>=5000;

|  |
| --- |
| >= <= > < = !=(不等于) |

练习：查询出1991-1-1后出生的员工所有信息

SELECT \* FROM emp WHERE birthday>'1991-1-1';

练习：查询出不在10号部门的员工所有信息

SELECT \* FROM emp WHERE deptId!=10;

练习：查询出没有明确部门的员工所有信息

SELECT \* FROM emp WHERE deptId IS NULL;

练习：查询出有明确部门的员工所有信息

SELECT \* FROM emp WHERE deptId IS NOT NULL;

练习：查询出工资大于6000的女员工所有信息

SELECT \* FROM emp WHERE salary>6000 AND sex=0;

练习：查询出工资在5000~7000之间的员工所有信息

SELECT \* FROM emp WHERE salary>=5000 AND salary<=7000;

SELECT \* FROM emp WHERE salary **BETWEEN** 5000 **AND** 7000;

练习：查询出工资为5000以下，7000以上的员工所有信息。

SELECT \* FROM emp WHERE salary<5000 OR salary>7000;

SELECT \* FROM emp WHERE salary NOT BETWEEN 5000 AND 7000;

练习：查询出1990之前和1993以后出生的员工所有信息

SELECT \* FROM emp WHERE birthday<'1990-1-1' OR birthday>'1993-12-31';

练习：查询出1993年出生的员工所有信息

SELECT \* FROM emp WHERE birthday>='1993-1-1' AND birthday<='1993-12-31';

练习：查询出10号部门和30号部门员工所有信息

SELECT \* FROM emp WHERE deptId=10 OR deptId=30;

SELECT \* FROM emp WHERE deptId IN(10,30);

练习：查询出不在10号和30号部门员工所有信息

SELECT \* FROM emp WHERE deptId NOT IN(10,30);

|  |
| --- |
| IS NULL / IS NOT NULL  AND / OR  BETWEEN...AND.../NOT BETWEEN...AND...  IN() / NOT IN() |

注意：删除、更改、查询都可以结合条件查询。

(8)模糊条件条件

示例：查询出姓名含有字母e的员工所有信息

SELECT \* FROM emp WHERE ename LIKE '%e%';

练习：查询出姓名中以e结尾的员工所有信息

SELECT \* FROM emp WHERE ename LIKE '%e';

练习：查询出姓名中倒数第2个字符为e的员工所有信息

SELECT \* FROM emp WHERE ename LIKE '%e\_';

|  |
| --- |
| SQL中提供了两个模糊查询的字符  % 可以匹配任意多个字符 >=0  \_ 可以匹配任意一个字符 =1  注意：以上两个匹配不能和=使用，必须使用LIKE关键字 |

(9)分页查询

假如查询的结果集中有太多的数据，一次显示不完，可以分页显示。

需要有两个条件：当前的页码、每页的数据量

SELECT \* FROM emp LIMIT start,count;

start: 是一个数字，从结果集中的哪1条开始读取，第1条称为0.

count: 是一个数字，最多读取的行数

|  |
| --- |
| 每页开始的算法  start=(页码-1)\*count |

假设每一页显示5条记录

第1页：SELECT \* FROM emp LIMIT 0,5;

第2页：SELECT \* FROM emp LIMIT 5,5;

第3页：SELECT \* FROM emp LIMIT 10,5;

第4页：SELECT \* FROM emp LIMIT 15,5;

假设每一页显示6条记录

第1页：SELECT \* FROM emp LIMIT 0,6;

第2页：SELECT \* FROM emp LIMIT 6,6;

课后任务：

(1)复习今天内容，删除代码，保留注释重新编写代码

(2)练习：

查询出工资为8000以上的女员工的姓名、性别、生日，结果集按照工资降序排序，取前3个人。

(3)查看“学子商城”表结构图