## **昨天内容回顾**

|  |
| --- |
| 1、数据库概念  2、mysql结构  3、安装mysql服务端  4、安装mysql客户端软件  5、数据库操作sql语句  6、数据库表操作sql语句  7、添加记录sql语句 |

## **今天内容介绍**

|  |
| --- |
| 1、DML  （1）修改  \* update 表 set 字段=值,字段=值 where 条件  （2）删除（delete和truncate）  \* delete from 表 where 条件  \* truncate 表  \* delete from和truncate区别  2、mysql简单查询  （1）查询语法  \* SELECT 字段 FROM 表 WHERE 条件 GROUP BY 字段 HAVING ..ORDER BY ..LIMIT  （2）查询表所有记录  \* SELECT 字段,字段.. FROM 表  （3）简单where条件查询  \* SELECT 字段,字段.. FROM 表 WHERE 条件  （4）别名  \* SELECT sid AS s,sname AS name from stu u  （5）着重号（飘）  3、mysql外键约束（多表）  （1）多表三种关系和建表原则  （2）外键约束使用  （3）删除外键数据  4、mysql关联查询（多表查询）  （1）笛卡尔积  （2）关联条件语法  （3）内连接  （4）左外连接  （5）右外连接  （6）FULL join（全连接）  5、mysql条件查询  （1）where条件查询  （2）like模糊查询  （3）order by排序  （4）查询区间范围  （5）limit分页查询  （6）聚合函数 |

## **DML-修改操作**

|  |  |
| --- | --- |
| |  | | --- | | CREATE TABLE `stu` (   `sid` int(11) NOT NULL AUTO\_INCREMENT,   `sname` varchar(100) NOT NULL,   `age` int(11) DEFAULT NULL,   `address` varchar(100) DEFAULT NULL,   PRIMARY KEY (`sid`) ) |   **1、修改操作sql语句语法**  （1）update 表名称 set 字段名称=值, 字段名称=值… where 条件  2、练习  # 修改张无忌的地址是嵩山  UPDATE stu SET address='嵩山' WHERE sname='张无忌'  # 修改周芷若的年龄50和地址峨眉  UPDATE stu SET age=50,address='峨眉' WHERE sname='周芷若'  # 修改所有人的年龄+10  UPDATE stu SET age=age+10 |

## **DML-删除操作**

|  |
| --- |
| 1、删除操作sql语句语法  （1）delete from 表名称 where 条件  # 删除sid=2的记录  DELETE FROM stu WHERE sid=2  2、删除表所有记录 delete from 和 truncate方式  # 删除表所有记录  DELETE FROM stu  # 把整张表先删除，再创建和之前相同结构表  TRUNCATE stu |

## **Mysql简单查询操作**

|  |
| --- |
| 1、mysql查询sql语句语法  （1）SELECT 字段1,字段2… FROM 表名称1,表名称2…  WHERE 条件  GROUP BY 字段 HAVING 筛选条件  ORDER BY 字段 ASC | DESC  LIMIT  2、查询表里面所有记录  # 查询stu表所有记录  SELECT sid,sname,age,address FROM stu  SELECT \* FROM stu  # 查询stu表所有sid和sname值  SELECT sid,sname FROM stu  3、别名  # 别名操作  SELECT sid AS sid\_id,sname AS sname\_name FROM stu s  4、飘符号（反引号）  CREATE TABLE t1 (  id INT,  `name` VARCHAR(100)  )  5、简单where条件  # 查询名字是谢逊信息  SELECT \* FROM stu s WHERE s.sname='谢逊'  # 查询sid大于2，并且年龄大于30的学生信息  SELECT \* FROM stu WHERE sid>2 AND age>30 |

## **多表关系和建表原则**

|  |
| --- |
| **1、多表之间有哪些关系**  **（1）一对多关系**  \* 部门和员工  \* 一个部门有很多员工，一个员工属于一个部门  \* 部门是一，员工是多  **（2）多对多关系**  \* 学生和课程  \* 一个学生可以选择多门课程，一门课程可以被多个学生选择  **（3）一对一关系**  \* 人和身份证号  \* 一个人只有一个身份证号，一个身份证号对应一个人  **2、一对多关系建表**    **（2）在多的那一方创建字段，作为外键，关联一的那一方主键**  **3、多对多建表**    **4、一对一建表**  **（1）主键对应建表**    **5、以一对多为例创建表，使用外键**  ### 创建部门表  CREATE TABLE dept (  did INT PRIMARY KEY,  dname VARCHAR(100)  )  ### 创建员工表  CREATE TABLE emp (  eid INT PRIMARY KEY,  ename VARCHAR(100),  age INT,  edid INT,  FOREIGN KEY(edid) REFERENCES dept(did)  )  # 声明外键  **6、外键主要作用：**  **（1）用于数据关联**  **（2）外键保证数据完整性和一致性**  **\* 添加员工时候，添加部门id，这个部门id必须是部门表id**  **\* 删除部门时候，如果部门下面有关联员工，部门不能删除的**  **7、实际创建表时候，一般外键不声明出来，把作为关联使用就可以了** |

## **Mysql多表关联查询操作**

|  |
| --- |
| **一、mysql关联查询分类**  1、笛卡尔积（了解）      2、内连接查询    （1）获取两张表有关联数据  3、外连接-左外连接    （1）左边表所有数据，右边表关联数据  4、外连接-右外连接    （1）右边表所有数据，左边表关联数据  **二、编写sql语句实现mysql多表关联查询**  **1、笛卡尔积**  # 笛卡尔积  SELECT \* FROM dept,emp  2、内连接查询    # 内连接查询  # select 字段 from 表1 inner join 表2 on 关联条件  SELECT \* FROM dept INNER JOIN emp ON dept.did=emp.edid  SELECT \* FROM dept,emp WHERE dept.did=emp.edid  3、外连接-左外连接    # 外连接-左外连接  # select 字段 from 表1 left outer join 表2 on 关联条件  SELECT \* FROM dept LEFT OUTER JOIN emp ON dept.did=emp.edid  4、外连接-右外连接    # 外连接-右外连接  # select 字段 from 表1 right outer join 表2 on 关联条件  SELECT \* FROM dept RIGHT OUTER JOIN emp ON dept.did=emp.edid  5、A表独有数据    # A表独有  SELECT \* FROM dept LEFT JOIN emp ON dept.did=emp.edid  WHERE emp.edid IS NULL  6、B表独有数据    # B表独有数据  SELECT \* FROM dept RIGHT JOIN emp ON dept.did=emp.edid  WHERE dept.did IS NULL  7、AB全有（类似于全连接效果）    # AB全有  SELECT \* FROM dept LEFT JOIN emp ON dept.did=emp.edid  UNION  SELECT \* FROM dept RIGHT JOIN emp ON dept.did=emp.edid  8、A独有 + B独有    # A独有 + B独有  SELECT \* FROM dept LEFT JOIN emp ON dept.did=emp.edid WHERE emp.edid IS NULL  UNION  SELECT \* FROM dept RIGHT JOIN emp ON dept.did=emp.edid WHERE dept.did IS NULL |

## **Mysql查询操作（1）**

|  |
| --- |
| 1、去重操作 DISTINCT  SELECT DISTINCT ename FROM emp  2、where条件使用  # 查询年龄大于20岁  SELECT \* FROM emp WHERE age>20  3、模糊查询  # 查询所有姓张人员信息  # %代表通配符，匹配任意内容  SELECT \* FROM emp WHERE ename LIKE '张%'  # 查询姓名中最后一个字 亮结尾  SELECT \* FROM emp WHERE ename LIKE '%亮'  # 查询姓名包含张  SELECT \* FROM emp WHERE ename LIKE '%张%'  # 查询姓名里面第一个字母是m，m后面包含三个字母  SELECT \* FROM emp WHERE ename LIKE 'm\_\_\_'  4、排序操作  （1）升序和降序  # 查询emp表所有记录，根据eid进行排序  SELECT \* FROM emp ORDER BY eid ASC  SELECT \* FROM emp ORDER BY eid DESC  5、查询区间范围  # 查询区间范围  # 查询emp表年龄 30-50范围  SELECT \* FROM emp WHERE age>=30 AND age<=50  # BETWEEN 开始值 AND 结束值  SELECT \* FROM emp WHERE age BETWEEN 30 AND 50  # 查询年龄是 20 40 60  SELECT \* FROM emp WHERE age IN(20,40,60) |

## **Mysql查询操作（2）**

|  |
| --- |
| 1、mysql分页查询  （1）mysql实现分页，使用关键字 limit，limit关键字只能使用在mysql数据库中，不是标准sql  2、limit用法  （1）limit后面有两个参数  第一个参数 查询数据开始位置，从0开始  第二个参数 每页显示多少条记录  （2）查询数据开始位置计算公式  \* **(当前页-1)\*每页显示记录数**  3、编写分页语句  # 分页语句  SELECT \* FROM emp LIMIT 0,3  SELECT \* FROM emp LIMIT 3,3  4、mysql针对特定功能，写好一些函数，直接使用这些函数实现特定功能，这些函数成为聚合（集）函数  （1）count() ：统计  ## 查询表有多少条记录  SELECT COUNT(\*) AS num FROM emp  ## 查询年龄大于40人员数量  SELECT COUNT(\*) FROM emp WHERE age>40  （2）sum()：求和  # sum()：求和  SELECT SUM(age) FROM emp  （3）avg(): 计算平均数  # avg(): 计算平均数  SELECT AVG(age) FROM emp  ## 把平均数值小数点后面位数  SELECT CAST(AVG(age) AS DECIMAL(10,2)) AS avgnum FROM emp  （4）max(): 获取最大值  （5）min(): 获取最小值  SELECT MAX(age) FROM emp  SELECT MIN(age) FROM emp |

## **Mysql查询操作（3）**

|  |
| --- |
| 1、分组查询  （1）group by 分组字段  2、编写分组sql  # 查询每个部门里面有多少个员工  # 标准sql  SELECT COUNT(\*) AS num,edid AS deptid FROM emp GROUP BY edid |