**每日作业卷**

**javaweb第20天mysql**

传智播客.黑马程序员

# 关卡1

## 训练案例1

### 训练描述

常用聚合函数使用说明

* count：统计指定列不为NULL的记录行数；
* sum：计算指定列的数值和，如果指定列类型不是数值类型，那么计算结果为0；
* max：计算指定列的最大值，如果指定列是字符串类型，那么使用字符串排序运算；
* min：计算指定列的最小值，如果指定列是字符串类型，那么使用字符串排序运算；
* avg：计算指定列的平均值，如果指定列类型不是数值类型，那么计算结果为0；

### 操作步骤描述

1. 基于products表的数据,进行聚合函数查询.
2. 商品汇总,总价等查询。

#1 查询商品的总条数

#2 查询价格大于200商品的总条数

#3 查询分类为'c001'的所有商品的总和

#4 查询分类为'c002'所有商品的平均价格

#5 查询商品的最大价格和最小价格

## 训练案例2

### 训练描述

能完成sqlyog实现数据库备份和恢复

### 操作步骤描述

1:使用sqlyog连接数据库

2:操作备份和恢复数据库

## 训练案例3

### 训练描述

使用group by 完成商品分类查询

分组查询是指使用group by字句对查询信息进行分组。

* 格式：

SELECT 字段1,字段2… FROM 表名 GROUP BY分组字段 HAVING 分组条件;

分组操作中的having子语句，是用于在分组后对数据进行过滤的，作用类似于where条件。

* having与where的区别:
  + having是在分组后对数据进行过滤.

where是在分组前对数据进行过滤

* + having后面可以使用分组函数(统计函数)

where后面不可以使用分组函数。

### 操作步骤描述

#1 统计各个分类商品的个数

#2 统计各个分类商品的个数,且只显示个数大于1的信息

## 训练案例4

### 训练描述

实现表字段的主键约束,自动增长,非空约束,唯一性约束,外键约束.

### 操作步骤描述

1:创建表的时候,添加主键

2:创建表完成之后添加主键

3:唯一性约束unique

4:非空约束 not null

5:外键约束: foreign key

## 训练案例

### 训练描述

多表设计原则和sql实现

一对多表设计

多对多表设计

外键约束使用说明

### 操作步骤描述

1:完成1:n表设计,外键约束命令的使用

2:完成n:n表设计,中间表设计原则以及联合主键的使用.

3:多表设计中添加外键.

# 关卡2

## 训练案例1

### 训练描述

1:设计员工表emp和部门表dept

要求: emp表的基本字段 empno int 主键 自动自增长,ename 字符类型 salary 数值类型 bonus 奖金数值型……deptno int 员工在的部门 (外键)

dept表的基本字段 deptno int 主键 自动增长 , dname 部门名称 字符类型 dlocation 地理位置 字符类型

2: 设计表结构 在emp表中设计deptno外键.录入相关数据.

### 操作步骤描述

1:完成emp和dept 1:n设计

2:录入初始化数据

## 训练案例2

### 训练描述

1:完成学员student 和 老师 teacher 表和课程表的设计

2:多对多设计原则,引入中间表.

### 操作步骤描述

1: 完成学员和老师,课程以及中间表设计

2: 使用sql脚本完成中间表设计以及联合主键,外键的引入.

3: 录入相关数据.

# 关卡3

## 训练案例1

### 训练描述

1:基于设计和学员表,老师表,课程表以及中间表的数据,完成相关查询

### 操作步骤描述

-- 1、查询平均成绩大于70分的同学的学号和平均成绩

-- 2、查询所有同学的学号、姓名、选课数、总成绩

-- 3、查询学过赵云老师所教课的同学的学号、姓名

-- 4、查询没学过关羽老师课的同学的学号、姓名

-- 5、查询没有学三门课以上的同学的学号、姓名

-- 6、查询各科成绩最高和最低的分

## 训练案例2

### 训练描述

1:基于设计和员工表和部门表的数据完成1:n复杂查询

### 操作步骤描述

1．列出至少有4个员工的所有部门。

2．列出薪金比“SMITH”多的所有员工。

3．列出所有员工的姓名及其直接上级的姓名。

4．列出受雇日期早于其直接上级的所有员工。

5．列出部门名称和这些部门的员工信息，同时列出那些没有员工的部门。

6．列出所有“CLERK”（办事员）的姓名及其部门名称。

7．列出最低薪金大于1500的各种工作。