MySQL 数据库

1）MySQL AB公司的产品，瑞典

08年，Sun 09年，Oracle

2）特点

效率高：执行速度快

开源的

简单：安装和使用

3）安装

4）启动MySQL的两种方式

第一种：右键此电脑 --> 管理

第二种：打开com的window控制台（数据服务启动方式）

net stop mysql 停止服务

net start mysql 启动服务

查看数据库版本的函数

查看mysql的脚本

-------->select version(); ---mysql客户端

mysql --version ---windows客户端

mysql - V ---windows客户端

MySQL的语法规范

1）不区分大小写，通常关键字大写，表名和字段名小写

2）通常用；结尾，表示语句结束,也可以使用\g

select \* from studenttab\g

3）可以分行写或缩进

4）注释

# 注释内容

-- 注释内容

/\* 注释内容

注释内容

\*/

SQL

SQL(Structured Query Language):结构化的查询语言，和数据库产品无关,通用的查询语言

SQL 分类：

DQL(Data Query Language):数据的查询语言

SELECT语句

DML(Data Manipulate Language):数据的操纵语言

INSERT，UPDATE,DELETE

DDL(Data Define Language):数据的定义语言

CREATE,ALTER,DROP

TCL(Transaction Control Language):事务的控制语言

COMMIT,ROLLBACK

select查询

/\*

语法：

SELECT

显示列表

FROM

表名

显示列表，可以是字段、常量、表达式、函数

FROM不是必须有的，当显示列表中没有表中字段时，省略from

\*/

起别名

/\*

使用as关键字；as可以省略

如果别名中包含特殊符号，使用""或者''

\*/

去重

SELECT

DISTINCT department\_id

FROM

employees;

+

/\*

java中，运算符，连接符

mysql 运算符，没有连接符功能

如果是字符串，先转换成数值，然后计算；

如果不能转换成数值，转换成0，然后计算。

NULL做加法，返回NULL

\*/

IFNULL(expr1,expr2)

/\*

如果第一个表达式为null，那么返回第二个表达式的值，

否则返回第一个表达式的值

\*/

CONCAT(str1,str2,...)函数

# 把last\_name和first\_name连接在一起，字段别名 姓名

过滤语句 WHERE

/\*

语法：

SELECT

显示列表

FROM

表名

WHERE

条件

运算符

1)比较运算符：> >= < <= = != <>

2)逻辑运算符：AND OR NOT

3)其他运算符:

LIKE

BETWEEN AND

in

is null is not null

\*/

LIKE:模糊查询

/\*

通配符：% 表示任意字符

\_ 表示一个字符

\ 表示转义字符

\*/

BETWEEN AND

In

is null is not null

ORDER BY 排序

语法：

SELECT

查询列表

FROM

表名

[where ]

order by

排序的列表 升序/降序

1.升序 ASC 降序 DESC

2.默认排序规则：升序

3.排序的列表：

字段（一个字段排序，也可以按多个字段排序）

表达式：

别名

4.order by 是SELECT子句中的最后一个子句（limit除外）

LENGTH(str) --获取字符串长度(字节数)的函数

NOW() --获取系统时间 包含日期和时间

STR\_TO\_DATE(str,format) --转换函数(字符串转date)

/\*

%Y 2021

%y 21

%m 03

%c 3

%d 1

%H 24记时法

%h 12

%i 分钟

%s 秒

\*/

DATE\_FORMAT(date,format) --把日期类型转换成字符串(date转字符串)

# 多行函数(分组、聚合、统计)

#SUM()

#AVG()

#MAX()

#MIN()

#COUNT()

分组查询

# 1)格式

/\*SELECT

列表

FROM

表

[WHERE]

GROUP BY

字段列表

[HAVING]

[ORDER BY]

\*/

# WHERE:在分组之前数据的筛选

# HAVING:在分组之后筛选，包含分组函数

1.内连接：等值、非等值、自连接

2.外连接：左、右、满外连接(满外连接，MySQL不支持)

3.交叉连接：(笛卡尔集)

语法：

SELECT

查询列表

FROM

表名1 别名

type ----inner\LEFT OUTER JOIN\RIGHT OUTER JOIN连接类型

join 表名2 别名

on ----连接条件

[WHERE] 筛选条件

[GROUP BY]

[HAVING]

[ORDER BY]

[LIMIT]

1.等值连接查寻

SELECT

查询列表

FROM

表1，表2...

WHERE

连接条件(至少n-1个有效的连接条件)

AND(OR)

筛选条件

[GROUP BY]

[HAVING]

[ORDER BY]

熟悉练习

#子查询

/\*

在select语句中或者其子句中出现select语句，把select语句叫子查询

子查询和主查询

按select出现的位置：

SELECT 字段1,... (select)

FROM (select)

WHERE和HAVING (select)

按子查询和主查询的执行顺序分类：

标准子查询：先执行子查询，把结果用到主查询中

相关子查询：先执行主查询，把结构用到子查询中

\*/

熟悉练习

分页查询 ★

/\*

语法：

SELECT 列表 7

FROM 表 1

type ...

JOIN 2

ON 3

WHERE 4

GROUP BY 5

HAVING 6

ORDER BY 8

LIMIT [offset] size 9

offset:表中数据的索引，从0开始

size：表示一页显示几条数据

\*/

insert into 表名 (字段名,......) values(值,......)

insert中可以有select语句

UPDATE 语法格式：

UPDATE 表名

SET 字段名 = 值,......

WHERE 筛选条件

update中可以有select语句

删除 DELETE TRUNCATE:删除表中所有数据

#DELETE语法：

delete from 表名 where 筛选条件

#TRUNCATE语法：

TRUNCATE table 表名

#创建和删除数据库

CREATE DATABASE if not EXISTS db1;

DROP DATABASE if EXISTS db1;

#删除表

DROP TABLE if EXISTS stu\_tb1;

创建表：

CREATE TABLE 表名(

字段 数据类型[长度],

......

)

修改表：

修改字段名

修改字段的数据类型长度

添加字段

删除字段

修改表名

删除表：

DROP TABLE 表名;

#创建表

CREATE TABLE stu\_tb(

stuid int,

stuname varchar(50),

email VARCHAR(50)

)

#修改字段名 CHANGE COLUMN ---列的重命名、列类型的变更以及列位置的移动

ALTER table stu\_tb CHANGE COLUMN stuname sname VARCHAR(50);

DESC stu\_tb;

#修改字段的数据类型长度 MODIFY COLUMN ---列类型的变更以及列位置的移动，但不能给列重命名

ALTER table stu\_tb MODIFY COLUMN sname VARCHAR(100);

DESC stu\_tb;

#添加字段

ALTER table stu\_tb ADD COLUMN phone VARCHAR(11);

DESC stu\_tb;

#删除字段

ALTER TABLE stu\_tb DROP COLUMN email;

DESC stu\_tb;

#修改表名

ALTER TABLE stu\_tb RENAME TO stu\_tb1;

创建表可以有select语句（表的复制）

简答题：

字符串：

比较小的字符串:char(定长) varchar(可变长度)

char

c char(10); ---10表示字符串最大的长度

sex char ---()可以省略，默认值为1

'你好' 分配10个长度

浪费空间

效率高

varchar(10); ---10表示字符串最大的长度

---()必须要写

'你好' 分配2个长度

节省空间

效率低

时间类型：

date time year

datetime: 8个字节 1000 ———— 9999

timestamp: 4个字节 1970 ———— 2038

约束（简答题）

定义字段值的取值范围,保证数据的正确性和准确性

1)几种:6种

not null约束：表示字段值必须填,不可以为null

primary key:主键约束,表示记录的唯一性,值不能为空,一般不代表业务

unique：唯一性约束,字段值可以为空

chek：检查约束,salary > 800 (mysql不支持)

foreign key：外键约束,外键是用来关联两张表关系的字段

添加约束

create table emp(

empid int primary key,

ename varchar(50) not null unique,

age int default 23,

seat int unique,

deptid int ,

constraint fk\_emp\_dept foreign key (deptid) references dept(deptid)

);

1.事务：

一个或一组sql语句组成一个执行单元，这个执行单元要么全部执行，要么全部不执行。

转账

张三 1000

李四 1000

update 表 set 张三的余额=500 where name='张三'

意外

update 表 set 李四的余额=1500 where name='李四'

2.事务的特性：ACID

原子性（Atomicity）：原子性是指事务是一个不可分割的工作单位,

事务中的操作要么都发生，要么都不发生。

一致性（Consistency）：一个事务执行会使数据从一个一致状态切换到另外一个一致状态

隔离性（Isolation）：事务的隔离性是指一个事务的执行不能被其他事务干扰，

即一个事务内部的操作及使用的数据对并发的其他事务是隔离的，

并发执行的各个事务之间不能互相干扰。

持久性（Durability）：持久性是指一个事务一旦被提交，它对数据库中数据的改变就是永久性的，

接下来的其他操作和数据库故障不应该对其有任何影响

隔离性越高，效率越低！

#2.定义视图

CREATE VIEW 视图名称

AS

SELECT 语句

#3.为什么定义视图

# 1)简化多表连接查询的复杂度

# 2)屏蔽字段

所有的试图都可以进行修改是错的

.哪些视图不可以修改(DML)

#1)视图中包含distinct,having,group by分组函数,union,union all

#2)常量视图

#3)SELECT后面包含SELECT

#4)通过不可以修改的视图创建出来的视图

#案例1：定义一个存储过程p\_emp,向employees表中插入数据

delimiter $

CREATE PROCEDURE p\_emp()

BEGIN

INSERT INTO employees(last\_name,salary) VALUES('admin',20000);

END $

CALL p\_emp();

#案例3：定义函数f\_fun3,输入两个整数,返回和

delimiter $

CREATE FUNCTION f\_fun3(a INT,b INT) RETURNS INT

BEGIN

DECLARE sum int;

SET sum = a+b;

RETURN sum;

END $

SELECT f\_fun3(1,2);

存储过程和函数都有返回值是错的