

刘老师 09 年开始做 IT 的

12 年出国做了 2 年半 2014 年 7 月份 -- 至今

8000 学生左右 腾讯、阿里、米哈游....

从事软件测试 -- 软件开发

知识点集中一些 (1、2 周时间上完一点) 专挑企业面试/企业工作的内容上 (开发)

Mysql、linux(centos) == 基础课

数据库学习目的:

开发和企业的 DBA (数据库管理员: 4-5 个项目 1 个 DBA) 进行对接, 项目组 (每个项目都是项目经理自负盈亏, 项目进度也不确定 基本上公司员工 6 个月左右调薪 1000 左右 经历 3000 左右)

DBA: 设计数据库表结构 -- java -- jdbc mybatis 等

数据库产品:

Mysql(电商、中小型项目)、oracle (甲骨文)、db2、内存数据库 (股票交易)

oracle 公司 -- 把 mysql 收购了, 把 sun 的 java 也收购了

mysql 的作者又做了一款开源的数据库 mariadb (女儿的名字)

后期：linux 上或开源的软件项目都会用到 mariadb

开源：android 源代码免费共享 商业化：不公开源代码并且收费

经验：几乎所有的数据库 sql 语句都差不多

数据库学习：

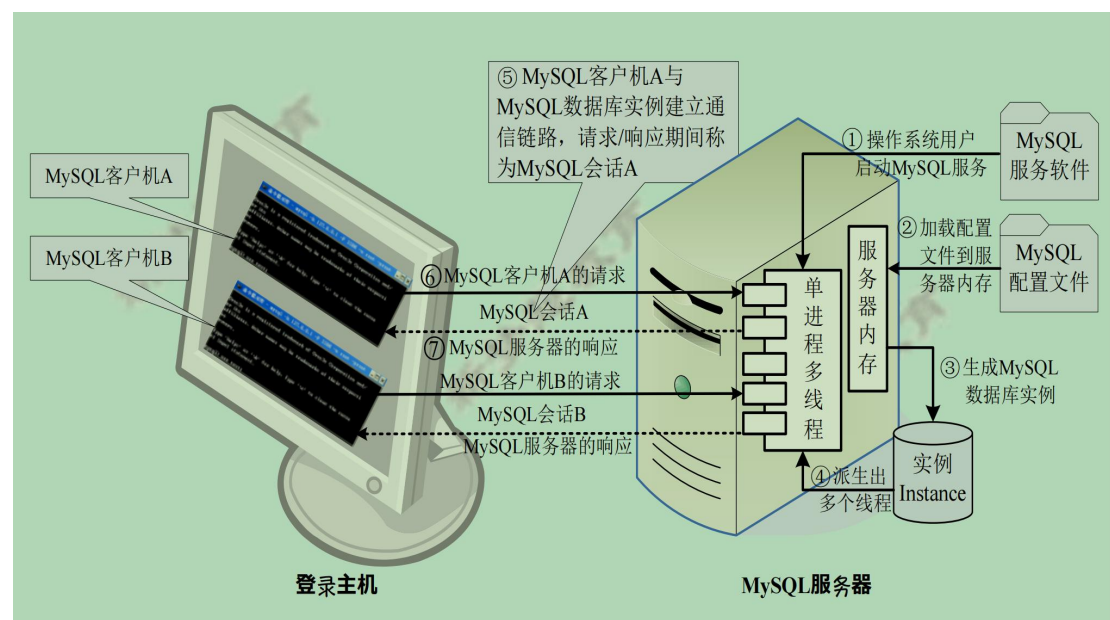
数据库作用：

存储大量数据，方便检索和访问

保持数据信息的一致、完整

共享和安全

可以进行汇总分析 -- 得出更有价值的数



行业里，一般把项目分为两类；

b/s : browser/server 浏览器/服务器模式 网站 淘宝网、腾讯网

c/s: client/server 客户端/服务器模式 手机 QQ pc 版 QQ 等

mysql 数据库 是 c/s 架构的项目:

Server: 安装了 mysql57\80 等于安装了 mysql 服务器

Client: navicat / sqlyog(小海豚)



mysql 的配置文件: my.ini (一般采用默认配置即可, 如需配置, DBA 负责的)

mysql 自带的数据库:

安装好 mysql 之后，系统会自动包含 4 个数据库：

information_schema:信息数据库，显示当前数据库的相关信息（一般不用）

Mysql:这个是 mysql 的核心数据库，主要负责存储数据库的用户、权限设置、关键字等 mysql 自己需要使用的控制和管理信息。不可以删除；在该数据库中的 **user** 表中存放了用户的用户名和密码等相关信息

	Host	User	Select_priv	Insert_priv	Update_priv	Delete_priv	Create_priv	Drop_priv
	localhost	root	Y	Y	Y	Y	Y	Y
	localhost	mysql.session	N	N	N	N	N	N
	localhost	mysql.sys	N	N	N	N	N	N
I	localhost	test	N	N	N	N	N	N
	localhost	testa	N	N	N	N	N	N
	%	root	Y	Y	Y	Y	Y	Y

FLUSH PRIVILEGES; # 刷新数据库权限

user 表中每一行代表一个 mysql 服务器的账号

如果 host 字段 为 % ，表示任意 IP（任意可联通网络的电脑）都能登录该账号。

可联通网络的电脑： 如何判定两台电脑网络是互通的：

Ping 对方 IP 地址 ping 192.168.2.185

Mysql 常用的 sql 语句操作：

mysql 服务器中，创建数据库（真正存放数据的地方）

create database **java_demo**; # 逻辑上有一个数据库

创建一个表格：

```

4
5 create table student_info( -- 表名
6     sid int, -- 字段名 字段类型
7     sname varchar(20),
8     sex char(1)
9 )DEFAULT charset='utf8'; -- 字段的值支持中文

```

表格插入数据

```

insert into student_info values (1, '王红', '女'); -- 插入数据

select * from student_info; -- 查询语句

```

数据完整性:

实体完整性: 在有业务要求的情况下, 数据应该是独一无二的, 不能重复, 如: 学号、订单编号

在 **mysql** 的 **sql** 语句中, 通过约束来实现:

唯一约束、主键约束、标识列

主键约束: 主键特征 给字段设置的、值的必须是唯一的且非空的, 一般把编号等核心数据设置成主键

```

create table stu_01(
    sid varchar(10) PRIMARY KEY, -- 主键约束
    sname varchar(10),
    address varchar(20)
-)default charset='utf8';

```

唯一约束: 设置后, 字段的值必须是唯一的。

```

create table stu_01(
    sid varchar(10) ,
    sname varchar(10) UNIQUE , -- 唯一约束
    address varchar(20)
-)default charset='utf8';

```

标识列: 可以不用赋值且自动增长的列 (开发的时候, 不用去传入标识列字段)

```

create table stu_01(
  sid int PRIMARY key auto_increment, -- 标示列
  sname varchar(10),
  address varchar(20)
)default charset='utf8';

```

域完整性：值域完整性，录入信息的时候，值不能发生问题

限制数据类型（字段设置类型）、检查约束、外键约束、默认值、非空约束

默认值约束：如果不输入内容，则使用默认内容

```

create table stu_01(
  sid varchar(10),
  sname varchar(10),
  address varchar(20) DEFAULT '地址不详' -- 默认值约束
)default charset='utf8';

```

非空约束：字段必须输入内容

```

create table stu_01(
  sid varchar(10),
  sname varchar(10) not null, -- 非空约束
  address varchar(20)
)default charset='utf8';

```

检查约束（mysql 可以写，但是 5.8 以下的版本不生效）

```

0 create table stu_01(
1   sid varchar(10),
2   sname varchar(10),
3   age int, -- 10--20
4   address varchar(20),
5   CHECK(age>=10 and age<=20), -- 检查约束
6   check(length(sname)>=1 and length(sname)<=4) -- 检查约束
7 )default charset='utf8';
8

```

引用完整性：通过外建约束实现；多个表的数据在引用的时候要一一对应

外键约束要求：首先必须是主从表（至少 2 个表）、主表的主键作为从表的外键

```
create table stu_01(
  sid varchar(10) PRIMARY key,
  sname varchar(10) ,
  address varchar(20)
)default charset='utf8';

create table score(
  course varchar(10),
  sid varchar(10),
  score int,
  CONSTRAINT fk_01 FOREIGN key(sid) REFERENCES stu_01(sid) -- 外键约束 fk_01约束名称
)default charset='utf8';

insert into score values('数学','0010024',80);
```

SQL 语句分为四大类:

DDL 语句: Data Define language 数据定义语句

Create drop alter truncate

DML 语句: 数据操作语句

Insert delete update select

DCL 语句: 数据控制语句

Grant on 授权 revoke from 回收

TCL 语句: 事务控制语句

Commit 提交 rollback 回滚

DDL 语句

create database 数据库名;

show create database 数据库名;


```

1 show create database java_demo; -- 查看数据库信息
2 drop DATABASE aa_db; -- 删除数据库
3

```

mysql 数据类型:

整型	tinyint	1个字节
	smallint	2个字节
	mediumint	3个字节
	int	4个字节
	bigint	8个字节

类型	字节数	范围 (有符号)	范围 (无符号)
tinyint	1字节	(-128, 127)	(0, 255)
smallint	2字节	(-32768, 32767)	(0, 65535)
mediumint	3字节	(-8388608, 8388607)	(0, 16777215)
int	4字节	(-2147483648, 2147483647)	(0, 4294967295)
bigint	8字节	(-9223372036854775808, 9223372036854775807)	(0, 18446744073709551615)

```

create table A(
  IID1 TINYINT, -- 有符号的短整型
  IID2 TINYINT UNSIGNED -- 无符号的短整型
);

INSERT into A VALUES(126, 258 );
select * from A ;

```

小数类型:

小数类型	decimal	
	float	4个字节
	double	8个字节

类型	字节数	负数的取值范围	非负数的取值范围
float	4	-3.402823466E+38到-1.175494351E-38	0和1.175494351E-38到3.402823466E+38
double	8	-1.7976931348623157E+308到-2.2250738585072014E-308	0和2.2250738585072014E-308到1.7976931348623157E+308

decimal(length, precision)用于表示精度确定 (小数点后数字的位数确定) 的小数类型, **length** 决定了该小数的最大位数, **precision** 用于设置精度 (小数点后数字的位数) 。

```
create table B(  
  score01 FLOAT,  
  score DECIMAL(4,1) -- -999.9 --- 999.9  
)  
  
INSERT into B values( 10.5,1000 );
```

字符串类型:

字符类型	char	定长字符串
	varchar	变长字符串
	text	tinytext, text, mediumtext, longtext

Char(10):固定长度的字符串类型 “hello” 字段还是占 10 个宽度

Varchar(10):可变长度的字符串类型 “hello” 字段占 5 个宽度

Text:大文本类型

目录 5.7/5.8 版本超出上限就报错

Use 数据库名; 使用数据库

建表格的语法:

Create table 表名 (字段名 1, 数据类型 [约束条件],

...

[其他约束条件]

....) 其他选项 (例如存储引擎、字符集等选项) ;

```

create table stu_info(
    sid int , -- 字段名 (列名)
    sname varchar(6),
    birth datetime -- 日期时间类型
) DEFAULT CHARSET='utf8';

insert into stu_info values (1,'小红','2003-03-09 10:00:56');
select * from stu_info;
desc stu_info; -- 显示表结构
show create table stu_info; -- 查看表的创建信息
drop table stu_info ; -- 删除表格
alter table stu_info add address varchar(10) DEFAULT '湖南长沙'; -- 表格中增加新字段
alter table stu_info drop address; -- 删除字段
alter table stu_info change birth birthday datetime; -- 修改字段名称
alter table stu_info MODIFY birthday varchar(100) ; -- 修改字段的数据类型
rename table stu_info to stu_i; -- 修改表名

```

DML 语句:

insert into 表名 [(字段列表)] values (值列表)

```

insert into stu_i(sid,sname) values (1,'小红');
INSERT into stu_i values (1,'小红','2003-03-09'),(2,'小黑','2003-03-09'),(3,'小白','2003-03-11');

```