LAMP vs LNMP互联网架构：

Linux 🡪 Apache 🡪 MySQL 🡪 PHP/Perl/Python

Linux 🡪 Nginx 🡪 MySQL 🡪 PHP

关系型数据库遵循ACID规则：

1. Atomicity原子性：事务里的所有操作要么全部做完，要么都不做。其中任一操作失败则整个事务需要回滚。
2. Consistency一致性：事务的运行不会改变数据库原本的一致性约束。
3. Isolation独立性：并发的事务之间不会户型影响。
4. Durability持久性：事务提交后会永久的保留在数据库上，即使出现宕机也不会丢失。

事务的语句：

开始事务： BEGIN TRANSACTION

提交事务： COMMIT TRANSACTION

回滚事务： ROLLBACK TRANSACTION

DTO: Data Transfer Object

DAO: Data Access Object

MySQL简介

1. 开放源码、免费的关系型数据库、中小企业最爱；
2. MySQL数据库隶属于Oracle公司；
3. 优化好可以处理百万、千万级数据量；
4. 支持InnoDB存储引擎，所以可以支持事务处理，支持行级锁；
5. 使用主流、通用、规范的SQL语言。

MySQL install on Windows

<http://www.mysql.com/>

Download 🡪 Community 🡪 Community Server 🡪 mysql-installer-community-5.6.40.1.msi

安装路径：C:\Program Files\MYSQL\MySQL Server 5.6

存储路径：C:\ProgramData\MYSQL

包含文件夹：

bin：命令、程序

data：存储数据、配置文件

share：

docs：帮助文档

include：插件、函数库

lib：库文件

MySQL链接到数据库

**MySQL通过Shell连接到数据库**

control panel 🡪 system 🡪 advanced 🡪 system variables:

MYSQL\_HOME C:\Program Files\MySQL\MySQL Server 5.6

PATH ;%MYSQL\_HOME%\bin\

start 🡪 run 🡪 cmd 🡪 run as administrator(right clik)

mysql - -version

mysql –h localhost –u root –p

mysql - - host=localhost - - user=root - - password

show databases;

mysql - - help

quit

**MySQL通过MySQL Workbench连接到数据库**

MySQL Workbench 8.0

add new database

**MySQL通过Navicat for MySQL连接到数据库**

商业化软件，收费，功能强大与MySQL Workbench。

<http://www.navicat.com.cn/>

下载 🡪 navicat for mysql 🡪免费使用14天

add new connection

使用MySQL数据库和表

**数据库操作：**

创建数据库：create database *mudidb*;

数据库名称64字符内，不可以是纯数字或关键字，如果是关键字需用单引号标注。

进入某数据库：use *mudidb*;

更改数据库名称：新建一个数据库，倒入原有数据，删除旧数据库。耗时费力，起名时需谨慎。

删除数据库：drop database *mudidb*; 误删后无法恢复，慎用。

**表操作：**

建表：CREATE TABLE tbl\_user(id int(3), name varchar(8), password varchar(20));

重命名表：ALTER TABLE tbl\_user RENAME TO tbl\_member;

RENAME TABLE tble\_user TO tbl\_member;

删表：DROP TABLE tbl\_user;

增：ALTER TABLE tbl\_user ADD email varchar(20) not null;

删：ALTER TABLE tbl\_user DROP email;

改：ALTER TABLE tbl\_user CHANGE id newid int(3);

查：

**数据库物理文件：**

创建数据库后：

.opt 🡪 记录数据库配置。

创建一个表后，生成两个文件：

.frm 🡪记录表的结构信息。

.ibd 🡪记录表的数据、索引信息。

MySQL数据类型

**数值类型：**

整数：

|  |  |
| --- | --- |
| TINYINT | 1 byte |
| SMALLINT | 2 byte |
| MEDIUMINT | 3 byte |
| INT | 4byte |
| BIGINT | 8byte |

alter table tbl\_user add intsix int(6) zerofill; 一共6位，不足前面补0。

浮点数: float (4 byte), double (8 byte), decimal (M+D) M<65, D<30

alter table tbl\_user add float53 float(5, 2); 一共5位，小数点后2位。

浮点数存放的是近似值，定点数存放的是精确值。

BIT: 1bit

BOOL: 1bit

**字符类型：**

|  |  |
| --- | --- |
| char | 固定长字符串，255字符 |
| varchar | 可变长字符串，255字符 |
| tiny text | 255字符 /8 byte |
| text | 65535字符 /16 byte |
| medium text | 24 byte |
| long text | 32 byte |
| blob | Binary Large Objects, 65535字符 /16byte |
| medium blob |  |
| long blob |  |
| enum |  |
| set |  |

**日期和时间类型：**

|  |  |  |
| --- | --- | --- |
| DATA | yyyy-mm-dd | 1000-01-01, 9999-12-31 |
| DATETIME | yyyy-mm-dd hh:mm:ss | 1000-01-01 00:00:00, 9999-12-31 23:59:59 |
| TIMESTAMP | yyyy-mm-dd hh:mm:ss | 1970-01-01 00:00:00UTC, present |
| TIME | hh:mm:ss | -838:59:59, 838:59:59 |
| YEAR | yy or yyyy | 70-69, 1901-2155 |

联表查询和部分SQL函数的使用

内链接：

select \* from table1 inner join table2 on condition;

外右连接：

select \* from table1 right join table2 on condition;

外左链接：

select \* from table1 left join table2 on condition;

全链接：MySQL没有提供全连接指令，所以需要联合左右外连接，以实现。

select \* from table1 right join table2 on condition

union

select \* from table1 left join table2 on condition;

查询包含特定信息：

select \* from table1 where column1 like ‘%mudi%’;

升降序显示查询结果order语句：

select \* from table1 order by id asce limit 0,5;

select \* from table2 order by name desc limit 2,7;

SQL函数：

select min(age) from staff;

select max(salary) from staff;

select avg(salary) from staff;

select sum(salary) from staff;

select count(id) from staff;

select count(id) from staff group by age;

利用反射完成初级DAO

**编写DAO意义：**

解决大量冗余工作，提高开发效率；

提高代码整洁性、规范性。

**编写反射DAO的流程：**

创建数据库

创建项目、包、基础实体类

编写万能DAO前提条件：表明与类名一致，列名与类属性一致

创建JDBC连接类

编写2个表的查询方法

对比查询方法，建立利用反射编写思路