1. **数据的存储形式**

**文件/内存（存储临时数据）/第三方服务器/数据库服务器**

1. **什么是数据库**

**数据库是按照特定的形式来组织、存储数据，目的是为了后期的增删改查等对数据的操作。**

1. **数据库的发展历史**

**网状数据库->层次型数据库->关系型数据库->非关系型数据库（NoSQL）**

1. **关系型数据库逻辑结构**

|  |
| --- |
| **Server->Database->Table->Row->Column**  **服务器 数据库 数据表 行 列** |

1. **MySQL数据库**

**Oracle:MySQL**

**Martin:MariaDB**

**XAMPP**

|  |
| --- |
| **服务器套装，包含多个服务器mysql,apache** |

1. **MySQL部署结构**

**服务器端；负责存储/维护数据 —— 银行服务器**

**D：/xampp/mysql/bin/mysqld.exe 启动文件**

**d-> deamon 精灵/守护者**

**确保3306端口不被占用**

**客户端：负责向服务器端发起增删改查 —— ATM机**

**D：/xampp/mysql/bin/mysql.exe**

1. **使用客户端连接服务器端（不加分号）**

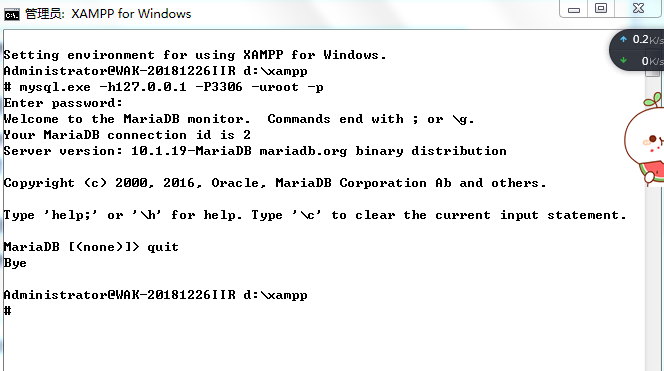
**mysql.exe -h127.0.0.1 -P3306 -uroot -p**

**-h host 主机（服务器的域名/IP地址）**

**-P port 端口号**

**-u user 用户**

**-p password 密码**

****

**mysql -uroot 简写形式**

1. **mysql常用管理命令（加分号结束）**

**show databases; 显示服务器上当前所有数据库**

**quit; 退出服务器连接**

**use 数据库名称; 进入指定的数据库**

**show tables; 显示当前数据库中所有的表**

**desc 表名称; 描述表中都有哪些列（表头）**

1. **SQL语句**

**SQL:Structured Query Language, 结构化查询语言，用于操作关系型数据库服务器，对数据执行增删改查。**

**SQL语句两种执行方式**

1. **交互模式：客户端输入一行，点击回车，服务器端执行一行。适用于临时性的查看数据**
2. **脚本模式：客户端把要执行的命令写在一个脚本文件中，一次性提交给服务器执行。适用于批量操作数据**

**mysql -uroot<D:/xampp/.../02.sql 回车**

**SQL命令语法规范：**

1. **每条SQL命令可以跨越多行，遇到分号作为结束**
2. **假如某一行命令出现语法错误，则此条语句以及后边所有的语句不会再执行**
3. **SQL命令不区分大小写，习惯上数据库关键字用大写，非关键字用小写**
4. **SQL命令可以使用单行注释（#...）和多行注释（/\*...\*/），注释的内容不会被服务器所执行**

**常用的SQL命令**

1. **丢弃指定的数据库，如果存在的话**

**DROP DATABASE IF EXISTS jd;**

1. **创建新的数据库**

**CREATE DATABASE jd;**

1. **进入创建的数据库**

**USE jd;**

1. **创建保存学生数据的表student**

**CREATE TABLE student(**

**sid INT, #integer**

**name VARCHAR(8),**

**sex VARCHAR(1), #m->男 f->女**

**score INT**

**);**

1. **插入数据**

**INSERT INTO student VALUES(‘1’,’tom’,’m’,’85’);**

1. **查询数据**

**SELECT \* FROM student;**

1. **修改数据**

**UPDATE student SET name='zhang',score='97' WHERE sid='4';**

1. **删除数据**

**DELETE FROM student WHERE sid='3';**