day 01

**课程分为5个阶段：**

1. **服务器端开发：***登录微信时，账号的保存，或者是玩游戏，有多人对战。只要不是在自己电脑上的东西都是服务器上*

项目导入/数据库/JS基础/nodeJS（后端开发）

1. **前端核心技术：**HTML / *AJAX（用于取数据）*/ CSS*（网页的美容师）/* bootstrap*（响应式，一套网页适用于不同终端）*
2. **前端进阶技术：**JS高级 / DOM&BOM*（浏览器端交互效果技术） /* JQuery*（简化前端的DOM&BOM的操作）*
3. **H5移动端开发**：VueJS框架*（让项目更快速高效）* / H5 / 微信开发*（微信小程序）*/ HybridAPP
4. **框架技术：**AngularJS / ReactJS / Python

Windows系统常用的快捷键

Alt+tab切换窗口（向右切换）

Alt+shift+tab（向左）

Window+D隐藏/显示桌面

Window+R 输入cmd，打开‘运行’窗口；输入mspaint，打开画图，输入calc打开计算器

Window+E 打开文件资源管理器

替换单击右键

Ctrl+S保存

Alt+F4关闭当前窗口

在文件资源管理器里，打开查看，勾选文件扩展名，即可显示文件扩展名

**一．软件工程。**

*所谓软件，并不是要安装的才叫软件。*

*最开始软件被称为程序。*

**程序设计**

*Ada-＞bug-＞debug（修复漏洞）*

*计算机普及后出现软件这个概念，出现软件设计这个概念*

**软件设计**

*IBM-＞Dos-＞管理磁盘-＞Microsoft-＞Bill Gates-＞Windows*

**软件工程**

*在给定成本，进度的前提下，开发出具有有效性，可靠性，可维护性，可重用，可移植，可追踪的软件产品。*

**二．软件的生命周期（软件的流程）**

**1.软件定义期**

**（1）可行性研究阶段**

**技术、人力、设备、时间、资金、回报、政策、风俗**

**（2）需求分析阶段**

**功能性需求（商品展示功能、支付功能）和非功能性需求**

***学子商城功能性需求***

***前台：***[*www.codeboy.com*](http://www.codeboy.com)

***后台：***[*www.codeboy.com/admin/login.html*](http://www.codeboy.com/admin/login.html)

1. ***前台子系统：用户模块（注册/登录），商品模块，购物车模块***
2. ***后台子系统：用户模块（管理用户），商品模块（删除商品，修改商品价格），订单模块***
3. ***移动端子系统：用户模块（注册/登录），商品模块，购物车模块（功能和前台子系统是一样的）***

**2.软件开发期**

**（3）概要设计阶段——架构师（子系统，模块，各自功能，模块间的接口）**

**（4）详情设计阶段——设计师（页面，主体内容，属性，对象，方法）**

**（5）编码实现阶段——程序员（UI设计：网页的效果图→前端：把效果图转换为HTML/CSS/JS代码→后端：为前端页面提供所需要的数据）**

**（6）软件测试阶段——软件测试工程师**

**3.软件维护期**

**（7）项目部署阶段：将项目部署到服务器**

**（8）项目维护阶段**

**Web项目中的服务器**

**Server：为客户端提供各种服务的功能强大的计算机**

**（1）服务器地址：域名/IP地址**

***输入一个网址就会访问一个服务器，服务器的名字叫做域名，服务器的地址叫IP地址。通过域名或者IP地址都可以访问相应的服务器，如baidu.com***

**（2）对应服务的端口号**

***如网页服务（又叫web服务，端口号：443,80）.每个服务都有一个端口号。在网页后输入冒号然后输入端口号***

**（3）提供该服务使用的协议**

***使用网页服务需要使用到协议。协议相当于与服务员沟通时所用到的语言，不同的服务有不同的协议，如http***

**访问自己电脑上的web服务器**

**启动web服务器/webServer/start.cmd**

<http://127.0.0.1:80> **通过自己电脑的IP地址**

<http://localhost:80> **通过域名访问**

**以上服务器运行的软件**

**http://nodejs.org/en/**

**查看是否已安装软件 node –v**

**获取自己的电脑在局域网的IP地址**

**在命令行下输入ipconfig，查看 以太网适配器 以太网 ipv4地址**

<http://176.221.11.18:80>

**课后任务：1）复习今天内容，2）同桌之间，一个人开启服务器，另一个人使用局域网IP去访问**

**3）预习MySQL数据库**

day 02

1. **数据的储存形式：**a)文件（word/excel/文本文档..：存储比较小的数据） b)内存（文件一般时存相对少的数据，内存只能在程序运行/计算机运行的时候，往往存储临时数据）c）第三方服务器（阿里云，只有使用权没有管理权，银行数据不会托管给第三方服务器） d)数据库服务器
2. **什么是数据库？**按照特定的形式来组织、储存数据，目的是为了删改查等对数据的操作
3. 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）*
4. 关系型数据库逻辑结构

|  |
| --- |
| Server（数据库服务器）-＞database（数据库）-＞Table（数据表，多张表，每个表代表一类数据-＞Row（行）-＞column（列） |
|  |

常见的关系型数据库：SQLite 微型数据库，常用于移动设备

MySQL 开源（免费）中小型数据库，可用于各种操作

PostgreSQL开源（免费）中小型数据库

SQL Server （Microsoft开发）开发的收费中型数据库，只适用于Windows系统

Oracle 该公司开发的中大型数据库，可用于各种操作系统

1. **MySQL数据库**

Oracle：MySQL

Martin：MariaDB

XAMPP：服务器套装（包含有多个服务器mysql，Apache）

服务器端 mysqld.exe

客户端 mysql.exe

1. **MySQL部署结构**

服务器端：负责存储/维护数据——类比银行服务器

C：/xampp/mysql/bin/mysqld.exe 启动文件

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——类比ATM机

C：/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

简写形式：mysql –uroot（root:MySQL提供的一个用户）

6.**mysql常用管理命令（加分号）**

Show databases; 显示服务器上所有数据库（连接以后所有的命令都应该加分号）

Quit; 作用是退出服务器连接

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； 显示当前表一共多少列

*练习：mysql、performance schema数据库下有多少张表*

1. **SQL语句：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。

两种执行模式：

交互模式：户端输入一行点击回车，服务器端就会执行一行，只适用

临时性的查看数据。

脚本模式：客户端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。用于批量操作数据。

Mysql –uroot>拖拽文件或者写文件路径 回车

SQL命令语法规范

**（1）一个SQl命令可以跨越多行，以分号结尾**

**（2）加入某一行语句出现语法错误，则此条语句以及后面所有语句不会执行**

**（3）SQl命令不区分大小写，习惯数据库上的关键字用大写，非关键字用小写（如固定搭配的show databases就是关键字）**

**（4）单行注释#，多行注释/\*……\*/，注释的内容不会被服务器执行**

**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. **创建保存数据的表**

**CREATE TABLE student(**

**Sid INT, #integer 整数**

**Name VARCHAR(8),**

**sex VARCHAR(1), #m男 f女**

**score INT**

**(5)插入数据**

**INSERT INTO student VALUES(‘4’,’king’,’m’,’78’);**

**#修改数据**

**(6)查询数据**

**SELECT \* FROM student;（脚本里面不写，否则乱码）**

**（7）修改数据**

**UPDATE student SET sex=”f” WHERE sid=”4”**

**（8）删除数据**

**DELETE FROM student WHERE sid=”4”**

**\*\*\*最后一个不加分号**

day 03

*标准SQL语句分类*

*DDL：data Define language 定义数据结构*

*Drop，create，alter（修改）*

*DML：data manipulate language 操作数据*

*Insert，update，delete*

*DQL：data query language 查询数据*

*Select*

*DCR：data control language 控制用户权限*

*Grant（授权），revoke（收权）*

*两种情况的值可以用 #1/0表示，如sex，isonline*

1. 计算机如何存储字符
2. 如何存储英文字符

ASCII：总共有128个，对所有的英文字母和符号进行了编码

Latin-1：总共有256个，兼容ASCII码，同时对欧洲符号进行了编码

1. 如何存储中文字符

GB2312：对常用的6000多汉字进行的编码，兼容ASCII

GBK：对2万多汉字进行了编码，兼容GB2312

BIG5：台湾繁体字编码兼容ASCII

Unicode：对世界上主流国家常用的语言进行了编码，兼容ASCII码，不兼容GB2312，GBK，BIG5；具体分为UTF-8、UTF-16、UTF-32

*MySQL（德国）出现中文乱码的问题，是因为默认使用Latin-1*

1. 解决MySQL乱码

a脚本另存为的编码为UTF-8

b客户端连接服务器端使用的编码Set names UTF8；

c 服务器端创建数据库使用的编码create database xz charset=UTF8;

练习：编写脚本文件01\_sina.sql，先丢弃再创建数据库sina，进入数据库sina，设置编码为utf8，创建保存新闻数据的表news，包含编号nid，标题title，发表时间ctime，内容content，

坐着author，来源origin，插入若干条数据

1. mysql中的列类型

创建数据表的时候，指定的列可以存储的数据类型

1. 数值型（引号可加可不加） **TINYINT微整型**，占一个字节（BYTE），范围：-128~127

*1TB=1024GB；1GB=1024MB；1MB=1024KB；1KB=1024BYTE；1BYTE=8BIT（位）*

*每一位储存的值只能是0或1 1 10 11 100 101*

**SMALLINT 小整型**，占两个字节，范围：-32768~32767

**INT 整型**，占4个字节 范围-2147483648~2147483647

**BIGINT 大整型**，占8个字节

**Float 单精度浮点型，**占4个字节，最大值3.4E38（3.4\*10的38），可能产生计算误差

**Double 双精度浮点型，**占8个字节，可能产生计算误差

**DECIMAL（M,D） 定点小数**，不会产生计算误差，M表示总的有效位数，D表示小数点后的有效位数（7,2）

**BOOL 布尔型，**只有两组值 TRUE/1，FALSE/0（TRUE和FALSE不能加引号，真正存储数据的时候，列类型会变为TINYINT）

1. 日期时间型（必须加引号）

**DATE 日期型** 格式2019-10-20

**TIME 时间型** 格式14:34:20

**DATETIME 日期时间型** 2019-12-20 15:50:20

1. 字符串型（必须加引号）

**VARCHAR（M） 变长字符串** 不会产生空间浪费，操作速度相对慢，M最大值是65535

**CHAR（M） 定长字符串** 可能产生空间浪费，操作速度相对快，M最大值是255，用于存储手机号码，身份证号码等固定长度的数据

**TEX（M）大型变长字符串** M最多2G

***练习1：编写脚本文件02\_xz.sql，先丢弃再创建数据库xz，设置储存的编码utf8，进入该数据库，创建保存笔记本数据的表laptop，包含lid，标题title，价格price***

***lid INT,***

***title VARCHAR(128),***

***price DECIMAL(7,2),***

***stockcount SMALLINT, 库存***

***shelftime DATE, 上架时间***

***islndex BOOL 是否在线***

***2，编写脚本03\_tedu.sql,先丢弃再创建数据库tedu，设置编码utf8，进入该数据库，创建表dept，包含did，部门名称dname，员工数量empcount***

***Did INT;***

***Dname VARCHAR(8);***

***Empcount员工数量 SMALLINT;***

***sex BOOL ,性别可以用1,0 也可以用TRUE FALSE。***

***birthday DATE***

***salary DECIMAL(7,2)***

1. 列约束

MySQL对要插入的数据进行特定的验证，只有满足条件才允许插入数据中，否则被认为是非法的插入

例如一个人的性别只能是女或者是男，一个人的成绩范围是0—100分

1. 主键约束 PRIMARY KEY

声明了主键约束的列上不能插入重复的值

一个表里只能出现一次（不能编号和姓名都使用，一般加在编号，会自动排序，好处是加快查找速度）

主键列不能使用null值

*NULL 表示空，在插入数据的时候，无法确定要保存的值，例如员工的工资，生日等*

课后任务

1. 复习今天内容
2. 练习。编写脚本文件 ，先丢弃再创建数据库xz，设置编码utf8，进入该数据库，创建保存笔记本分类的表laptop\_family，包含fid，fname分类名称，

Laptopcount分类数量

10 联想 2 20 戴尔 2 30 小米 3

创建保存笔记本的表laptop，包含lid，title，price，spec（规格），detail，shelftime上架时间，isonsale，familyid所属分类编号

set names utf8;

drop database if exists xz;

create database xz charset=utf8;

use xz;

create table laptop\_family(

fid smallint primary key,

fname varchar(20),

laptopcount INT

);

insert into laptop\_family values(10,'联想',2);

insert into laptop\_family values(20,'戴尔',2);

insert into laptop\_family values(30,'小米',3);

create table laptop(

lid INT primary key,

title varchar(30),

price decimal(7,2),

spec varchar(50),

detail varchar(1000),

shelftime date,

isonsale bool,

familyid smallINT

);

insert into laptop values(1,'华为','5000','4G内存','粉色','2015-08-23','true','23');

insert into laptop values(3,'oppo','5000','6G内存','粉色','2015-08-23','true','25');

insert into laptop values(2,'vivo','5000','8G内存','粉色','2015-08-23','true','26');

day 04

3列约束

（1）主键约束 PRIMARY KEY

声明了主键约束的列上不能插入重复的值

一个表里只能出现一次（不能编号和姓名都使用，一般加在编号，会自动排序，好处是加快查找速度）

主键列不能使用null值

（2）非空约束 not null

声明了非空列数的列上禁止插入null值

（3）唯一约束 unique

唯一约束的列上不能插入重复的值，但是允许插入null，甚至多个null

（4）检查约束 check

可以对插入的数据进行自定义的验证

如score tinyint check（score>=0 and score<=100）

Mysql不支持检查约束，会降低数据的插入速度，增加服务器的压力

（5）默认值约束default

*如price decimal（7,2）not null default 3000；*

*可以使用default关键字设置默认值，有两种方式：(1)insert into laptop values(‘’,’’,’’,default);*

*(2)Insert into laptop(lid,title) values(8,’xiaoxin’); 其他的列全为默认值*

*练习：设置分类表中，笔记本数量默认值为1，分别使用两种方式运用默认值*

（6）外键约束 foreign key

Foreign key(familyID) reference laptop\_family(fid)

声明了外界约束的列，取值必须在另一个表的主键列上出现过，两者列类型必须保持一

致，允许出现null，甚至多个null

**MySQL的自增列**

Auto\_increment:自动增长

Lid int primary key auto\_increment

加入一个列声明了自增列，无需手动赋值，直接赋值为null，会获取当前的最大值，然后加1

注意：只适用于整数型的主键列上；

允许手动赋值；

*编写脚本文件01\_tedu.sql，先丢弃在创建数据库tedu，设置编码utf8，创建表dept，包含（did，dname部门名称），插入以下数据*

*10 研发部 20 市场部 30运营部 40 测试部*

*创建保存员工数据的表emp，包含eid，ename姓名，sex，birthday，salary，deptld所属部门编号，插入若干条数据*

**简单查询**

1. **查询特定的列**

示例：查询所有员工的编号和姓名 select \* from emp 如果是特定的列，就是select eid,ename from emp

Select ename,sex,birthday,salary from emp

1. **给列取别名**

示例：查询所有员工的姓名和工资，使用汉字别名 select ename AS 姓名,salary AS 工资 from emp;

练习：查所有员工姓名，性别，生日，编号，使用汉字别名 select ename AS 姓名,sex AS 性别,birthday 生日,eid 编号 from emp;（注：As关键字可省略，但需保留空格）

练习：查询员工的编号和姓名，使用一个字母作为别名 select eid a,ename b from emp;

1. **合并相同的项**

示例：查询出员工都分布在哪些部门 select distinct deptid from emp;

练习：查询公司都有哪些性别的员工 select distinct sex from emp;

1. **查询时执行计算**

示例：18+23-56/19+38\*21 select 18+23-56/19+38\*21;

练习：查询所有员工的姓名及其年薪 select ename,salary\*12 from emp;

练习：假设每个员工的工资增加500，年终奖5000，查询所有员工的姓名及其年薪，给列起汉字别名 select (salary+500)\*12+5000 年薪,ename 别名 from emp;

1. **查询的结果集排序**

示例：查询所有的部门，结果集按照部门编号降序排列 Select \* from dept order by did desc; #descendant

查询所有部门，结果集按照部门编号升序排列 select \* from dept order by did asc;

练习：查询所有员工，结果集按照工资升序排列 select \* from emp order by salary asc;

练习：查询所有员工，结果集按照年龄从小到大排序 select \* from emp order by birthday desc;

练习：查询所有员工，结果集按照姓名升序排列 select \* from emp order by ename asc; （升序asc可省略,也就是默认按照升序排）

练习：查询所有员工，结果集按照工资降序排列，如果工资相同，按照姓名升序排列 select \* from emp order by salary desc,ename;

练习：查询所有员工，结果集按照生日的降序排列，女员工显示在前面 select \* from emp order by sex,birthday desc;

1. **条件查询**

查询出姓名叫King的员工的所有列 select \* from emp where ename='king';

查询出编号为5的员工的所有列 select \* from emp where eid='5';

查询20号部门下员工的所有列 select \* from emp where deptid=20;

查询出所有女员工 select \* from emp where sex=0;

查询出工资在5000以上的员工有哪些 select \* from emp where salary>=5000;

查询出1991-1-1后出生的员工有哪些 select \* from emp where birthday>'1991-1-1';

查询出不在10号部门的员工有哪些 select \* from emp where deptid!=10;

查询出没有明确部门的员工有哪些 select \* from emp where deptid is null;（null是空，不能用=或者！=）

查询出有明确部门的员工有哪些 select \* from emp where deptid is not null;

查询出工资在6000以上的男员工有哪些 select \* from emp where sex=1 and salary>6000;

查询出工资在5000~7000之间的员工有哪些 select \* from emp where salary>=5000 and salary<=7000; 也可select \* from emp where salary between 5000 and 7000;  *(between and只能用于大于等于的) ->* not between 5000 and 7000;

查询出工资在5000以下和7000以上的员工有哪些 select \* from emp where salary<5000 or salary>7000;

查询出1993年出生的员工有哪些 select \* from emp where birthday between '1993-1-1' and '1993-12-31';

查询出20号部门和30号部门的员工有哪些 select \* from emp where deptid=20 or deptid=30; select \* from emp where deptid in(20,30);（满足其一) ->not in()

1. **模糊条件查询**

示例：查询姓名中含有字母e的员工有哪些 select \* from emp where ename like '%e%'; %表示匹配任意字符个数是零个或者多个

查询出姓名中以e结尾的员工有哪些 select \* from emp where ename like '%e';

查询出姓名中倒数第二位字符为e的员工 select \* from emp where ename like '%e\_'; \_表示可以匹配1个字符 =1。*以上两个匹配字符必须结合like使用*

1. **分页查询**

假如查询结果有太多的数据，一次性显示不完，则可使用分页查询。需要两个条件：当前的页码，每页的数据量

select \* from emp limit start,count

start:代表从哪一条数据开始查询

count:是每页的数据量

假设每一页显示5条记录

第一页：select \* from emp limit 0,5;

第二页：select \* from emp limit 5,5

第三页：select \* from emp limit 10,5;

第四页：select \* from emp limit 15,5;

第五页：select \* from emp limit 20,5;

练习：假设每一页显示6条记录，查询前三页

第一页:select \* from emp limit 0,6;

第二页：select \* from emp limit 6,6;

第三页：select \* from emp limit 12,6;

***\*\*limit后面的两个值不能加引号，只能是数值***

作业：查询工资在8000以上的女员工的姓名，性别，工资，结果集按照工资降序排列，获取前两个。

select ename,sex,salary from emp where salary>8000 and sex=0 order by salary desc;

day 05

1. **复杂查询**
2. **聚合查询/分组查询**

示例：查询所有员工的数量 select count(ename) fron emp

聚合函数

函数：是一个功能体，提供若干个数据，产出特定的结果

练习：查询女员工的数量，使用编号这一列 select count(eid) from emp where sex=0;

查询男员工的数量，使用部门编号这一列 select count(\*) from emp where sex=1; 用\*是因为有的部门编号为空

Count()/sum()/avg()/max()/min()

查询所有员工工资总和 select sum(salary) from emp

查询所有男员工的平均工资 select avg(salary) from emp where sex=1; select sum(salary)/count(\*) from emp where sex=1;

查询出工资最高的员工，查询出工资最低的员工 select max(salary) from emp; select min(salary) from emp;

查询出年龄最大的员工的生日 select max(birthday) from emp;

示例：查询出每个部门的员工最高工资select deptid,max(salary) from emp **group by** deptid; （分组查询特点：分组条件和聚合函数）

查询出男女员工数量，最高工资，平均工资 select sex,count(\*),max(salary),avg(salary) from emp group by sex;

year()获取中其中的年份 month() 获取日其中的月份

查询出1991年出生的员工哪些？ Select \* from emp year(birthday)=1991

查询出3月份生日的员工？ Select \* from emp where month(birthday)=3;

1. **子查询**

示例：查询出比Tom工资高的员工有哪些

步骤1 ：查询出Tom 的工资是多少 Select salary from emp where ename=’tom’;

步骤2：查询出工资高于6000的员工有哪些 select \* from emp where salary>6000;

综合：select \* from emp where salary>(Select salary from emp where ename=’tom’);(这种嵌套称为子查询)

练习：查询出和Tom同一年出生的员工有哪些 select \* from emp where year(birthday)=(select year(birthday) from emp where ename=’tom’);

练习：查询出研发部所有员工 select \* from emp where deptid=(select did from dept where dname=’研发部’);

1. **多表查询**

示例：查询出所有员工的姓名及其部门名称 select ename,dname from emp,dept 产生笛卡尔积 解决：添加条件

Select ename,dname from emp,dept where deptid=did; 存在的问题：无法查询出没有部门的员工，也无法查询出没有员工的部门

1. 内连接 select ename,dname from emp **inner join dept on deptid=did;** 和之前的查询结果一致

2）左外连接 --左侧表中所有记录都显示

select ename,dname from emp **left** **outer join** dept **on** deptid=did; *outer可省略* 左侧表中所有记录都显示

3）右外连接 --右侧表中所有记录都显示

select ename,dname from emp **right join** dept **on** deptid=did;

*4）全连接 显示左侧和右侧中所有记录 full join mysql不支持*

Union 合并相同记录 union all 不合并相同的记录