复习：

day01：

软件过程：

软件定义期：可行性研究、需求分析

软件开发期：概要设计、详细设计、编码实现、测试

软件维护期：项目部署、项目维护

day02：

MySQL中常用的SQL语句：

show databases;

use 库名;

show tables;

desc 表名;

show warnings; #显示上一条语句产生的警告

show create table 表名; #显示创建表时所用的语句

-------------------------------------

drop database if exists 库名;

create database 库名 charset=utf8;

use 库名;

create table 表名( 列名 列类型 列约束 );

insert into 表名 values('值');

delete from 表名 where 条件;

update 表名 set 列名='值' where 条件;

select \* from 表名;

day03:

列类型：

数值类型：tinyint/smallint/int/bigint/decimal/bool

日期时间类型：date/time/datetime

字符串类型：char/varchar/text

列约束：

唯一/非空/**主键**/默认值/检查/外键

day04：

简单查询：

查询特定的列 select ename,sex from emp

给列取别名 select ename as n,sex s from emp

显示不同的值 select distinct sex from emp

执行运算 select salary\*12 from emp

单条件查询 select \* from emp where salary>6000

多条件查询 select \* from emp where deptId=10 and sex=0

模糊条件查询 select \* from emp where ename like '%E\_\_'

复杂查询：

今日目标：

(1)MySQL中的查询

(2)WEB服务器

|  |
| --- |
| Order: 订单、排序 |
| Set：设置、集合 |
| Table：表、桌子 |

1.MySQL中的简单查询 —— 查询结果的排序

示例：查询出所有员工信息，要求按工资由小到大排序

SELECT \* FROM emp

ORDER BY salary ; #默认是由小到大

#ORDER BY salary ASC; #ascendant，升序

示例：查询出所有员工信息，要求按工资由大到小排序

SELECT \* FROM emp

ORDER BY salary DESC ; #descendant，降序

练习：查询出所有员工信息，按照姓名由小到大排序

SELECT \* FROM emp

ORDER BY ename ;

练习：查询出所有员工信息，按照姓名由大到小排序

SELECT \* FROM emp

ORDER BY ename DESC;

练习：查询出所有员工信息，按照年龄由大到小排序

SELECT \* FROM emp

ORDER BY birthday ASC ; #生日由小到大排列

练习：查询10号部门中员工的所有信息，按工资由大到小排列

SELECT \* FROM emp

WHERE deptId=10

ORDER BY salary DESC ;

#注意：where子句和order by子句的顺序问题

练习：查询出所有员工信息，按工资由大到小排列；工资相同的，再按姓名由小到大排列

SELECT \* FROM emp

ORDER BY salary DESC, ename ASC ;

练习：查询所有员工信息，按生日由大到小排列——前提是：所有女员工必须排在所有男员工之前

SELECT \* FROM emp

ORDER BY sex ASC , birthday DESC ;

2.MySQL中的简单查询 —— 分页查询 —— 重点&面试重点

分页查询：用于分批次的显示大量的数据。语法：

SELECT .... FROM ... WHERE ... ORDER BY ....

LIMIT start, count ;

(1)start：是一个数字，表示从哪一行开始获取数据

(2)count：是一个数字，表示一次最多获取的行数，即页面大小

假设每页显示5行记录，则

第1页： LIMIT 0, 5

第2页： LIMIT 5, 5

第3页： LIMIT 10, 5

第4页： LIMIT 15, 5

第5页： LIMIT 20, 5

....

第n页： LIMIT (n-1)\*5, 5

练习：分页查询出员工信息，页面大小为5，显示第1页

SELECT \* FROM emp LIMIT 0, 5 ;

练习：分页查询出员工信息，页面大小为5，显示第2页

SELECT \* FROM emp LIMIT 5, 5 ;

练习：分页查询出员工信息，页面大小为5，显示第3页

SELECT \* FROM emp LIMIT 10, 5 ;

练习：分页查询出员工信息，页面大小为5，显示第4页

SELECT \* FROM emp LIMIT 15, 5 ;

3.MySQL复杂查询 —— 子查询

示例：查询出“研发部”所有员工的信息

步骤1：查询研发部的部门编号 => 10

SELECT did FROM dept WHERE dname='研发部';

步骤2：查询部门编号为10的员工信息

SELECT \* FROM emp WHERE deptId = 10;

综合上述两步为一条语句：

SELECT \* FROM emp WHERE deptId =( #父查询

SELECT did FROM dept WHERE dname='研发部' #子查询

) ;

练习：查询出工资比TOM高的所有员工信息

步骤1：查询TOM的工资 => 6000

SELECT salary FROM emp WHERE ename='TOM';

步骤2：查询工资大于6000的员工信息

SELECT \* FROM emp WHERE salary>6000;

综合上述两步：

SELECT \* FROM emp WHERE salary>(

SELECT salary FROM emp WHERE ename='TOM'

);

练习：查询出比TOM年长的所有员工信息

步骤1：查询TOM的生日 => 1990-1-1

步骤2：查询出生日小于1990-1-1的员工所有信息

综合上述两步：

练习：查询出与TOM同年出生的所有员工信息

步骤1：查询TOM的生日中的年份 => 1990，提示：用year( )运算

步骤2：查询出生日中年份等于1990的员工所有信息

综合上述两步：