

一、语法和特点.

```
#进阶1: 基础查询
/*
语法:
select 查询列表 from 表名;

特点:
1.查询列表可以是: 表中的字段、常量值、表达式、函数
2.查询的结果是一个虚拟的表格。
*/
```

二、示例

```
#1.查询表中的单个字段
SELECT last_name FROM employees;

#2.查询表中的多个字段
SELECT last_name,salary,email FROM employees;

#3.查询表中的所有字段
#方式一: 双击
SELECT
`first_name`,
`last_name`,
`email`,
`job_id`
FROM
employees ;

#方式二: *
SELECT * FROM employees;
```

```
#4.查询常量值
SELECT 100;
SELECT 'john';
注意: 字符串和日期必须用单引号引起来, 数值不用引号.

#5.查询表达式
SELECT 100%98;

#6.查询函数
SELECT VERSION();
select 函数名(参数列表);

#7.为字段起别名
/*
便于理解;
如果要查询的字段有冲突的情况, 可以使用别名区分
*/

#方式一: AS
SELECT 100%98 AS 结果;
SELECT last_name AS 姓, first_name AS 名 FROM employees;

#方式二: 空格
SELECT last_name 姓 FROM employees;
```

```
#8.去重: DISTINCT
#案例: 查询员工表中所有涉及的部门编号
SELECT DISTINCT department_id FROM employees;

#9.+号的作用
/*
java中的+号作用:
①运算符: 两个操作数均为数值型
②连接符: 只要有一个操作数为字符串

mysql中+号的作用:
仅有一个功能: 运算符

select 100+90; 两个操作数均为数值型, 则做加法运算
select '123'+90; 其中一方为字符型, 试图将字符型转换为数值型
如果转换成功, 则基础做加法运算
select 'john'+90; 如果转换失败, 则将字符型数值转换为0
select null+10; 只要其中一方为null, 则结果肯定为null
*/

#案例: 查询员工名和姓连成一个字段, 并显示为 姓名
SELECT last_name+first_name AS 姓名
FROM employees; #显示为0

#使用concat函数做拼接
SELECT CONCAT('a','b','c') AS 结果;
SELECT CONCAT(last_name,first_name) AS 姓名 FROM employees;
```

10. 补充: concat 函数

功能: 拼接字符串

格式: concat(字符串1, 字符串2, ...);

11. 补充: ifnull 函数

功能: 判断某字段或表达式是否为null。如果为null, 返回指定值, 否则返回原本的值。

12. 补充: isnull 函数

功能: 判断某字段或表达式是否为null。如果为null, 返回1, 否则返回0