**三、SQL**

Sql语言本身不难，更多在于题目的灵活和解法的巧妙，建议多写多练，见过较多的题目情景，可以轻松熟练地应对sql题目，大部分的sql需求可通过构造辅助列，分组、开窗等方法解决。

**核心知识点**

**1、高频知识点 (必须掌握)**

**1) 查询语句(select)**

*select \* from <表名> (limit 10);*

基础查询语句，select \* from t 是从表t中获取全部信息，表量级较大时该语句运行较慢，慎重使用，加入limit n 为取该表中前n条数据信息，可用于表结构确认及局部信息确认。

*select <列名1>,<列名2> from <表名>;*

select <列名1>,<列名2> from <表名> 为从表中获取指定两列的数据

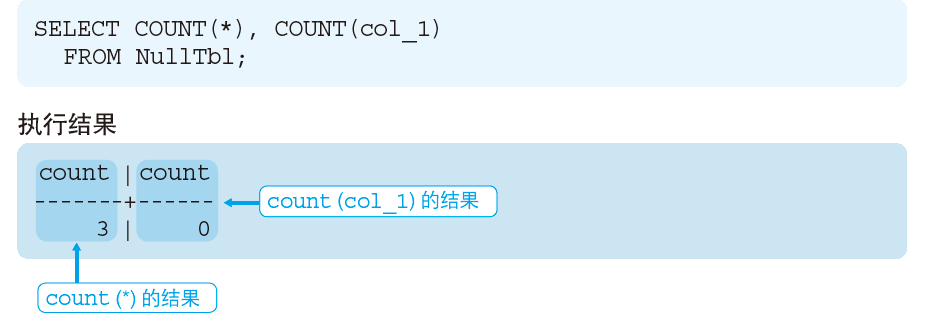
select语句为最常用的查询语句，在不更改原始表信息的情况下查询获取所需要的信息，除上面的基础用法外，可与条件过滤语句(where)结合获取指定条件下的数据信息(eg:select \* from <表名> where <列名1> = 1)，可与聚合函数、分组函数(group by)结合获取按需要方式分组后的统计结果(eg:select <列名1>,<列名2>,sum(<列名3>) from <表名> group by <列名1>,<列名2>)，可与连接函数(join)结合实现表之间的内连、外连，也可与窗口函数结合进行累计计算、分组排序、偏移分析等操作。

**2)聚合函数(sum/count/avg/max/min)**

聚合函数为sql中用于汇总的函数，它会对NULL以外的对象进行汇总（除count函数特殊外，若列中有null值，聚合函数会将null值排除在外，只计算非空值部分的汇总数据），常用的聚合函数有如下五种

COUNT： 计算表中的记录数（行数）

(若某列中包含null值，COUNT(\*)会得到包含NULL的数据行数，而COUNT(<列名>)会得到NULL之外的数据行数。)



SUM： 计算表中数值列中数据的合计值

AVG： 计算表中数值列中数据的平均值

MAX： 求出表中任意列中数据的最大值

MIN： 求出表中任意列中数据的最小值

使用聚合函数时，可用distinct函数删除重复值

eg：*SELECT COUNT(DISTINCT product\_type) FROM Product;*

**3)分组函数(group by)**

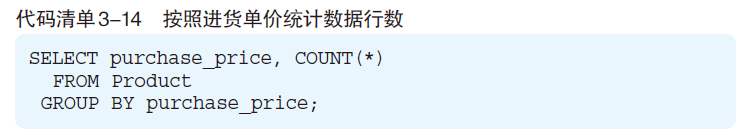
使用GROUP BY子句可以按特定列将表分割，常与聚合函数配合使用。

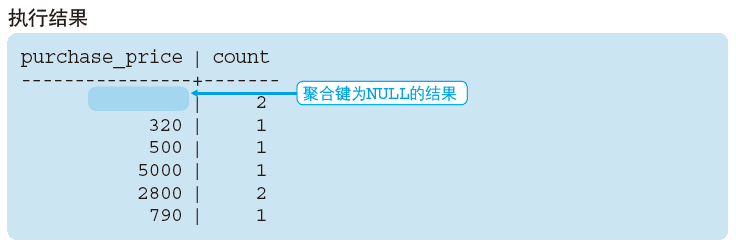
*SELECT <列名1>, <列名2>, <列名3>,count(列名4),sum(列名5),……*

*FROM <表名>*

*GROUP BY <列名1>, <列名2>, <列名3>;*

如分组列中有null值，分组汇总后null会以一种单独种类出现。





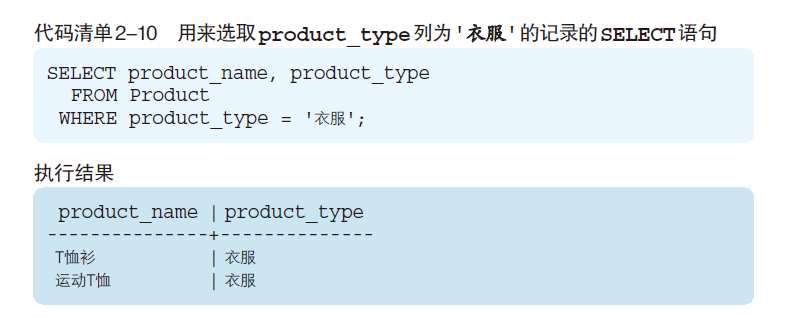
**3) 条件过滤语句(where/having)**

SELECT 语句通过WHERE 子句来指定查询数据的条件

*SELECT <列名>, ……*

*FROM <表名>*

*WHERE <条件表达式>;*



Where子句执行顺序在分组函数group by之前，如果需要根据分组后数据进行筛选利用having子句进行。

*SELECT <列名1>, <列名2>, <列名3>, ……*

*FROM <表名>*

*(WHERE <原表进行筛选的条件表达式>)*

*GROUP BY <列名1>, <列名2>, <列名3>, ……*

*HAVING <分组结果对应的条件>*

Having子句应用实例：<https://www.runoob.com/sql/sql-having.html>

**4) 联接函数(left join/right join/(inner)join/full (outer) join)**

常出现所需数据出现在不同表中的情况，通过联结函数我们可以将多张表连接起来从中取得需要的数据。

Join：<https://www.runoob.com/sql/sql-join.html>

Left join(左连接，最常用): <https://www.runoob.com/sql/sql-join-left.html>

Right join(右连接): <https://www.runoob.com/sql/sql-join-right.html>

Inner join(内连接，即取交集):<https://www.runoob.com/sql/sql-join-inner.html>

Full outer join(外连接，取并集): <https://www.runoob.com/sql/sql-join-full.html>

**5) 合并函数(union)**

Join函数完成多张表列的合并，union函数则完成不同表或查询结果行的合并，两部分利用union函数进行合并需保证其列结构完全相同。

Unoin(行合并):<https://www.runoob.com/sql/sql-union.html>

**6) 窗口函数( over(partiton by ))**

窗口函数语法解读 <https://zhuanlan.zhihu.com/p/92654574>

**2、拓展知识点(低频、难度较大，选择性掌握)**

**1) 基础表操作(建表create 删表 drop/truncate 插入insert 更新update)**

Create table 创建表 <https://www.runoob.com/sql/sql-create-table.html>

Delete 删除表中数据 <https://www.runoob.com/sql/sql-delete.html>

Drop table 删除表格 Drop table <表名>

Truncate table 删除表内全部数据，但并不删除表本身 Truncate table <表名>

Update 插入数据 <https://www.runoob.com/sql/sql-insert.html>

Update 更新数据 <https://www.runoob.com/sql/sql-update.html>

UPDATE *table\_name* SET *column1*=*value1*,*column2*=*value2*,...WHERE *some\_column*=*some\_value*;

**2) 模糊匹配(like)正则表达式(regexp)**

模糊匹配常用于按照所给的条件或者说要求进行大致程度匹配，如获取name列以g开头的所有行(SELECT \* FROM table WHERE name LIKE 'G%')，常用的函数为like/rlike/regexp，配合通配符使用。

Like ：<https://www.runoob.com/sql/sql-like.html>

SELECT *column\_name(s)*  
FROM *table\_name*  
WHERE *column\_name* LIKE *pattern*;

通配符：% — 代替一个或多个字符

\_ — 代替一个字符

通配符的使用及正则表达式(rlike/regexp)：<https://www.runoob.com/sql/sql-wildcards.html>

**3) 日期函数**

常用日期函数

[https://blog.csdn.net/Zhu\_Hua/article/details/81843393?ops\_request\_misc=%257B%2522request%255Fid%2522%253A%2522164766941816780271958362%2522%252C%2522scm%2522%253A%252220140713.130102334.pc%255Fall.%2522%257D&request\_id=164766941816780271958362&biz\_id=0&utm\_medium=distribute.pc\_search\_result.none-task-blog-2~all~first\_rank\_ecpm\_v1~rank\_v31\_ecpm-14-81843393.142^v2^pc\_search\_result\_control\_group,143^v4^control&utm\_term=sql+%E6%97%A5%E6%9C%9F%E5%87%BD%E6%95%B0&spm=1018.2226.3001.4187](https://blog.csdn.net/Zhu_Hua/article/details/81843393?ops_request_misc=%257B%2522request%255Fid%2522%253A%2522164766941816780271958362%2522%252C%2522scm%2522%253A%252220140713.130102334.pc%255Fall.%2522%257D&request_id=164766941816780271958362&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~first_rank_ecpm_v1~rank_v31_ecpm-14-81843393.142%5ev2%5epc_search_result_control_group,143%5ev4%5econtrol&utm_term=sql+%E6%97%A5%E6%9C%9F%E5%87%BD%E6%95%B0&spm=1018.2226.3001.4187)

<https://blog.csdn.net/qq_19865749/article/details/78535414?spm=1001.2101.3001.6650.8&utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7ERate-8.pc_relevant_antiscanv2&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7ERate-8.pc_relevant_antiscanv2&utm_relevant_index=12>

获取当前系统日期：select getdate();

日期加减：dateadd(初始日期(datetime格式),时间间隔(bigint格式) , 固定的转换间隔datepart (‘yyyy’-年,’mm’-月,’dd’-日))

*Eg: dateadd(to\_date('2020-01-01',’yyyy-mm-dd’), -1, 'mm')*

日期间隔：datediff(datetime date1, datetime date2, string datepart)

字符串转日期格式：to\_date('20080718', 'yyyymmdd')

from\_unixtime (unix\_timestamp ('2009-08-06'))

**4) 字符串函数**

常用字符串处理函数汇总 <https://blog.csdn.net/weixin_41999297/article/details/104341632?utm_medium=distribute.pc_relevant.none-task-blog-2~default~baidujs_title~default-0.pc_relevant_default&spm=1001.2101.3001.4242.1&utm_relevant_index=3>

**5) 列转行/列转行**

Group by + case when: <https://blog.csdn.net/wyqwilliam/article/details/82559023>

**6) 行转多列**

Lateral view explode <https://blog.csdn.net/weixin_36630761/article/details/76146842>

**7) 偏移函数lag()和lead()**

<https://blog.csdn.net/weixin_39415084/article/details/76429306>

**面试问题**

**1、高频面试问题 (重要！必须掌握)**

**1.1 SQL基础问题**

**1) sql执行顺序**

[**https://www.cnblogs.com/qanholas/archive/2010/10/24/1859924.html**](https://www.cnblogs.com/qanholas/archive/2010/10/24/1859924.html)

**2) where与on的区别、where与having的区别**

**3)几种join的含义与区别**

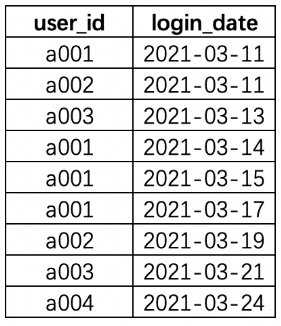
INNER JOIN： 内连接，只有进行连接的两个表中都存在与连接标准相匹配的数据才会被保留下来。   
LEFT JOIN: 左连接，左边表中符合 WHERE 子句的所有记录将会被返回，操作符右边表中如果没有符合 ON 后面连接条件的记录时，那么从右边表指定选择的列的值将会是 NULL。   
RIGHT JOIN: 右连接会返回右边表所有符合 WHERE 语句的记录。左表 中匹配不上的宇段值用 NULL 代替。   
FULL JOIN：完全外连接，取并集，将会返回所有表中符合 WHERE 语句条 件的所有记录。如果任一表的指定宇段没有符合条件的值的话，那么就使用 NULL 替代。

**4) join与union的区别**

**5)row\_number/rank/dense\_rank区别**

**1.2 代码问题**

现有一张用户登录表active\_user，包括两个字段：user\_id（用户ID）：varchar，login\_date（用户登录日期）：DATE



**1) 留存率/日活/月活**

根据用户登录表active\_user，求用户的次日留存率、日活跃用户数(日活率)、月活跃用户数(月活率) （快手、oppo）

**2) 活跃用户分析**

根据用户登录表active\_user，查询登录天数最多的3位用户的最近一次登录日期。输出user\_id（用户ID），login\_date（用户登录日期）。

**3) 连续时间问题**

根据用户登录表active\_user，求用户本月活跃天数、最大连续活跃天数

连续时间问题解法：

<https://zhuanlan.zhihu.com/p/349358841>

**4) 时间间隔问题**

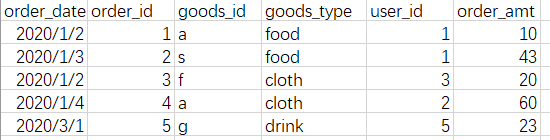
[**https://zhuanlan.zhihu.com/p/354816097**](https://zhuanlan.zhihu.com/p/354816097)

**5) 分组排序、窗口函数问题**

现有商品订单表（order\_info），有以下字段，求

a. 2020年1月销售金额最大的goods\_id

b.每个用户购买金额最大的商品种类goods\_type



**2、 拓展面试问题(低频、难度较大，选择性掌握)**

**1) 一段时间内最大登录人数问题**

<https://zhuanlan.zhihu.com/p/356037470>

**2) 随机抽样**

从数据集中随机抽取100条数据

Select \* from t order by rand() limit 100;

**推荐学习资料**

书籍：《Mysql必知必会》《SQL基础教程》

学习视频：尚硅谷 <https://www.bilibili.com/video/av49181542>

(视频讲的比较详细，内容较多，时间紧张的建议跳过视频)

代码练习网站：leecode 牛客网

Sql语法网站:<https://www.runoob.com/sql/sql-tutorial.html>