1 SELECT语法

**SELECT field FROM tb\_name|(SELECT) [WHERE qualification] [GROUP BY field [HAVING qualification]] [ORDER BY field {ASC|DESC}] LIMIT**

SELECT: field | field AS alias\_name **字段别名** | 聚合

字段别名：**SELECT name AS n FROM student;**

聚合：AVG() SUM() MIN() MAX() COUNT()**个数和**

FROM: 要查询的对象，表，多个表，其他select语句

WHERE：资格条件

1 布尔关系表达式，= > < >= != Example: Age>=16

2 逻辑运算：**AND OR NOT XOR( 亦或 ) && || !**

3 **BETWEEN numb1 AND numb2**  数值一到数值2之间

4 模式匹配，**LIKE** %:表示任意长度的任意字符 \_：表示任意单个字符

Example: **WHERE name LIKE ‘Y%’;**

5 正则表达式匹配：**RLIKE|REGEXP**

6  **IN（）**  **WHERE age IN(18,20,22);**

7 判断是否为空.  **WHERE cid** **IS [NOT] NULL;**

GROUP BY: **一般和select聚合联用。并且，对分组的条件进行过滤要用HAVING。**

**SELECT AVG(age) FROM students GROUP BY gender;**

**SELECT COUNT(age) AS persons FROM students GROUP BY cid HAVING persons>=2;**

ORDER BY: field\_name {ASC|DESC} 默认为ASC也就是正序。

LIMIT: **LIMIT [offset,]count**

Offset: 偏移量，也就是略过几个

Count：取几个数值。

2 SELECT 查询语句类型

1 单表查询

**SELECT \* FROM tb\_name**

**SELECT field1,field2,…… FROM tb\_name 投影**

**SELECT \* FROM tb\_name WHERE qualification 选择**

**SELECT [DISTINCT] 相同的值只显示一次**

2 多表查询

分多种连接方式

1 交叉连接（笛卡尔积） **SELECT \* FROM tb\_name,tb\_name2;**

2 自然连接：**将两张表指定字段存在等值关系的行显示出来，非等值的不显示**

**SELECT \* FROM student,couse WHERE student.cid = couse.cid;**

3 外连接，**以某个表为标准，这个表指定字段的值若在另一个表中对应则显示，不对应则显示NULL**

左外连接：左表 LEFT JOIN 右表 ON 条件

右外连接：右表 RIGHT JOIN 左表 ON 条件

**SELECT s.name,c.name FROM student AS s LEFT JOIN couse AS c ON s.cid=c.cid**

4 自连接,**一般必须使用表别名。**

**SELECT s.name AS stu,c.name AS teacher FROM student AS s,student AS c WHERE c.tid = c.cid;**

3 子查询——**查询的对象不是表，而是select语句**

1 位于FROM语句后面

**SELECT name,age FROM (SELECT name,age FROM student) WHERE age>=20;**

2 位于WHERE语句后

**IN(): SELECT name FROM student WHERE age IN (SELECT age FROM tutor);**

**SELECT name FROM student WHERE age >(SELECT AVG(age) FROM student)**

4 联合查询

**UNION**

**(SELECT name,age FROM student) UNION (SELECT name,age FROM tutor);**