初级 SQL

1、**数据定义语言（DDL）**：提供定义关系模式、产出关系模式以及修改关系模式的命令（即创建表、删除表、修改表的属性、约束条件等）。

2、**数据操作语言（DML）**：提供查插删改的命令，是对元组上的操作。

3、几种数据类型介绍：

char(n)：固定长度字符串，缺少的用空格补全（一般不用）。

varchar(n)：可变长度字符串，最大长度为 n。

numeric(p,d)：定点数，一共 p 位数字（加上一个符号位），其中 d 位在小数点右边。

4、select 不自动去重，可以用 select distinct 显示指明去重。

select 子句可带含有+、-、*、/运算符的算术表达式，如 select salary*1.1 。

5、select…from…where…理解顺序：首先是 from，然后是 where，最后是 select。

6、连接方式：

from r1,r2 笛卡尔积，r1 的每个元组与 r2 所有元组进行连接

from r1 natural join r2 自然连接，只考虑 r1 和 r2 上都出现的属性上取值相同的元组对，且相同属性在结果中只出现一次。（注意：ri、r2 所有的相同属性都会考虑到）

from r1 join r2 using （A） 自然连接的构造形式，只考虑 A 属性上的取值相同，而其他相同属性不考虑。

7、字符串中包含’应写为’’，如 ‘it’’s me’ 。

8、模式匹配（like）：

% ——用来匹配任意字符创

_ ——匹配任意一个字符

如：**’_%’表示匹配至少含一个字符的字符串**。

like 运算中使用 eclipse 关键字来定义转义字符，转义字符放在特殊字符前表示该特殊字符被当做普通字符。

如：like ‘ab\%cd%’ eclipse ‘\’ 匹配所有以“ab%cd”。

9、order by 子句默认使用升序，desc 表示降序，asc 表示升序（如：order by salary desc）。

10、并运算（union）、交运算（intersect）、差运算（except）自动去重。要想保留重复后面加 all。

11、算术表达式的任一输入为空，则结果为空（如：null + 1 = null）。

涉及空值的任何比较运算的结果视为 unknown（如：1 < null 的结果为 unknown）。

distinct 子句认为 null 和 null 是相同的，但是谓词中“null=null”会返回 unknown，而不是 true。

12、聚集函数不去重，用 distinct 显示去重（如：count (distinct ID)）。

13、**select…from…where…group by…having…理解顺序：from -> where -> group by -> having -> select。**

14、在分组查询时，一个很重要的事情是保证出现在 select 语句中但没有被聚集的属性只能是出现在 group by 子句中的哪些属性。

15、聚集函数忽略空值。

16、使用了来自外岑查询相关名称的子查询被称作相关子查询。

如：“找出在 2009 年秋季学期和 2010 年春季学期同时开课的所有课程”

```
select course_id
from section as S
where semester = ‘Fall’ and year = 2009 and
      exists ( select *
               from section as T
```

```
        where semester = 'Spring' and year = 2010 and
              S.course_id = T.course_id ) ;
```

中级 SQL

1、外连接：

自然连接由于空值可能会造成某些元组缺失，比如学生信息（student）和选课信息（takes）的自然连接中，若某位同学没有选课，则该同学的信息不回保留在结果中。

左外连接（left outer join）：保留运算左边的所有元组，右边没有匹配元组的话用 null 填充。

右外连接（right outer join）：保留运算右边的所有元组，左边没有匹配元组的话用 null 填充。

全外连接（full outer join）：保留两个关系中的所有元组，未匹配元组用 null 填充。

例："student natural left outer join takes"表示 student 所有元组都会保留，若某位同学没有选课，则该同学的选课信息用 null 填充。

2、视图：

对权限问题进行了维护，提高了安全性。

用于定义视图的实际关系改变，视图也跟着改变，这种视图称为物化视图（可以及时更新也可以周期性维护）。

允许视图名出现在任何关系名可以出现的地方，但是对于视图的插删改操作要符合逻辑，否则会出错（一般不允许对视图进行修改）。

3、事务：一组具有原子性的操作，要么都完成，要么都没完成。一旦中间某一操作出现问题则会滚到事务未开始前的状态。

4、空值不等于其他任何值（null != null），所以 unique 约束的候选码上可以出现多个 null。

5、参照完整性约束添加级联删除（on delete cascade）、级联更新（on update cascade）子句后，当某一删除或更新操作会违反参照完整性约束时，这一操作不会被系统拒绝，而是会同样删除或更新参照该元组的元组或值。

6、默认情况下，被授予权限的用户/角色无权把得到的权限再授予给另外的用户/角色，如果想有拥有这种权限，在相应的 grant 命令后加上 with grant option 子句。

从一个用户/角色那里收回权限，那么该用户/角色授予出去的此权限也会被收回，这一行为称为级联收回，在大多数数据库系统中，级联是默认行为。

高级 SQL

1、JDBC 标准定义了 Java 程序连接数据库服务器的应用程序接口。具体实现由个数据库产品来实现，每个支持 JDBC 的数据库产品都会提供一个 JDBC 驱动程序，这个驱动程序必须被动态加载才能实现 Java 对数据库的访问。

例如：可以使用"Class.forName("oracle.jdbc.driver.OracleDriver")"来完成对 oracle 驱动程序的加载。

2、关闭数据库连接是很重要的，因为数据库连接的个数是有限制的。

3、使用预备语句（PreparedStatement）可以防止 SQL 注入问题。

4、默认情况下，每个 SQL 语句都被作为一个自动提交的独立的事务来对待。JDBC 的 Connection 接口的 setAutoCommit()允许打开或关闭这种行为。

5、一个触发器的动作可能会引发另一个触发器，在最坏的情况下，甚至会导致一个无限的触发链。有其他候选方法时最好别用触发器，很多触发器的应用都可以用适当的存储过程来替换。