**SQL Part 1**

1. **SQL Language**
2. **Provides the following capabilities:**

* **Data Definition Language (DDL)**
* **To define and set up the database**
* **CREATE, ALTER, DROP**
* **Data Manipulation Language (DML)** **数据操作语言**
* **To maintain and use the database**
* **SELECT, INSERT, DELETE, UPDATE**
* **Data Control Language (DCL)** **数据控制语言**
* **To control access to the database**
* **GRANT, REVOKE**
* **Other Commands**
* **Administer the database**
* **Transaction Control**
* **START TRANSACTION**
* **BEGIN, END**

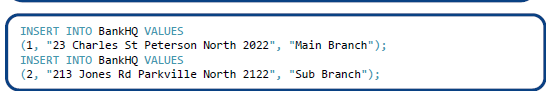
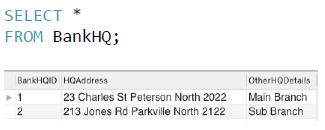
1. **How We Use SQL**

* **In Implementation of the database**
* **Take the tables we design in physical design**
* **Implement these tables in the database using create commands**

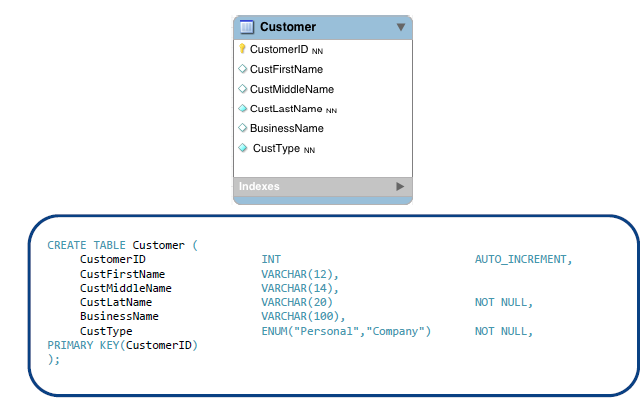
**使用create命令在数据库中实现这些表**

* **In Use of the database**
* **Use SELECT commands to read the data from the tables, link the tables together, etc.**
* **Use ALTER, DROP commands to update the database**
* **Use INSERT, UPDATE, DELETE commands to change data in the database**

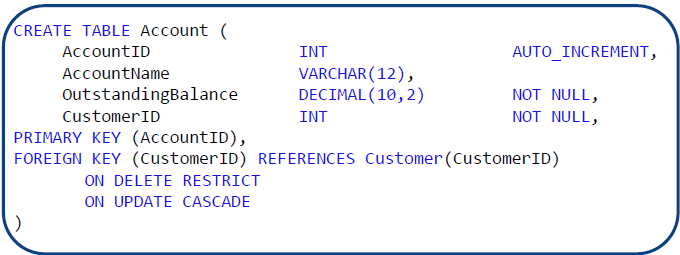
1. **CREATE, INSERT, NULL**
2. **SQL in Development Process**

* ****
* ****
* ****
* 

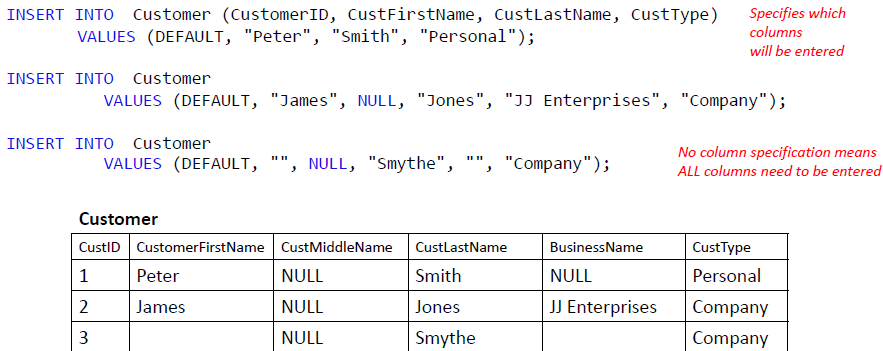
1. **CREATE Table: Review**

* ****

1. **CREATE Statement (with FK)**

* ****

1. **Insert Data**

* ****

1. **SELECT**
2. **The SELECT Statement**

* **A cut down version of the SELECT statement MySQL**
* **SELECT [ALL | DISTINCT] select\_expr [, select\_expr ...]**
* **List the columns (and expressions) that are returned from the query**

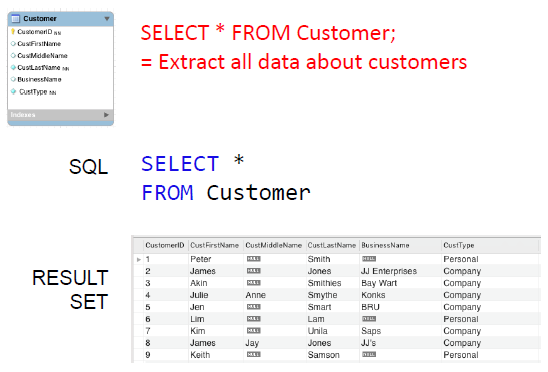
**列出查询返回的列(和表达式)**

* **[FROM *table\_references*]**
* **Indicate the table(s) or view(s) from where the data is obtained**

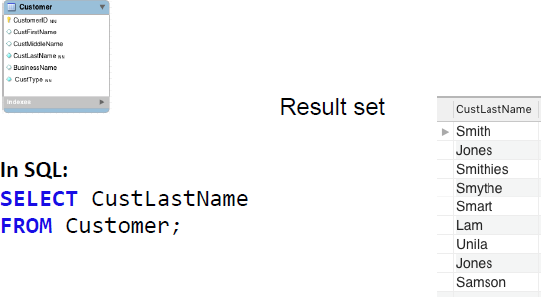
**指示从哪里获得数据的表或视图**

* **[WHERE *where\_condition*]**
* **Indicate the conditions on whether a particular row will be in the result指示特定行是否将出现在结果中的条件**
* **[GROUP BY {*col\_name | expr* } [ASC | DESC], ...]**
* **Indicate categorisation of results表明结果的分类**
* **[HAVING *where\_condition* ]**
* **Indicate the conditions under which a particular category (group) is included in the result指示特定类别(组)在何种条件下包含在结果中**
* **[ORDER BY {*col\_name | expr | position* } [ASC | DESC], ...]**
* **Sort the result based on the criteria根据标准对结果进行排序**
* **[LIMIT {[*offset ,*] *row\_count | row\_count* OFFSET *offset*}]**
* **Limit which rows are returned by their return order (i.e. 5 rows, 5 rows from row 2)** **根据返回顺序限制返回的行(即5行，从第2行开始的5行)**

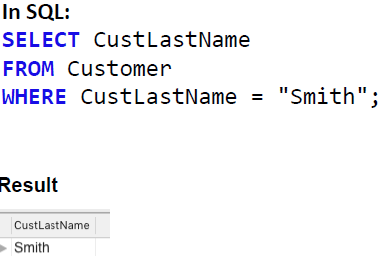
1. **Order is important! E.g. HAVING cannot go before GROUP BY or WHERE**
2. **SELECT Examples**

* ****

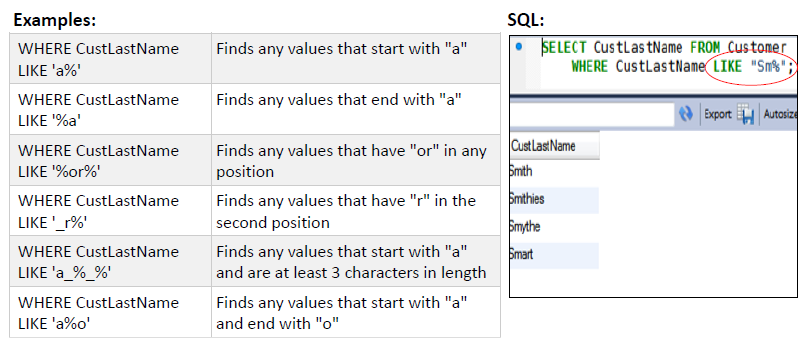
1. **SELECT Examples: Projection**

* ****

1. **SELECT Examples: Selection**

* ****

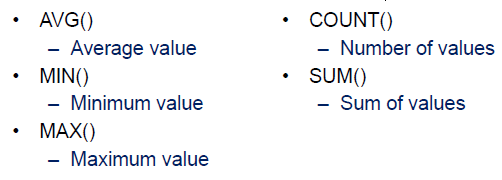
1. **SELECT Examples: LIKE clause**

* **In addition to arithmetic expressions, string conditions are specified with the LIKE clause除了算术表达式之外，字符串条件也用LIKE子句指定**
* ****
* ****

1. **Aggregate Functions聚集函数**

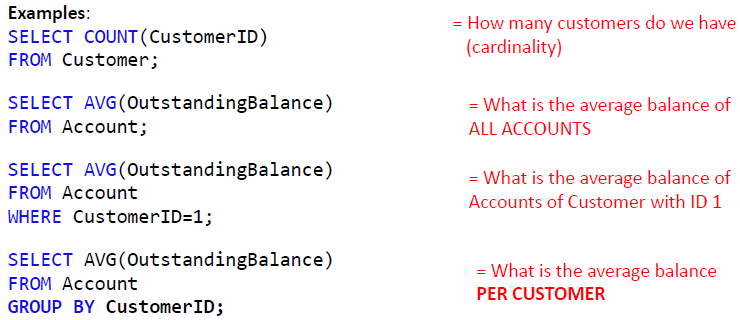
* **Aggregate functions operate on the (sub)set of values in a column of a relation (table) and return a single value**

**聚合函数对关系(表)的一列中的(子)值集进行操作，并返回单个值**

* ****
* **N.B. All of these except for COUNT() ignore null values and return null if all values are null. COUNT() counts the rows not the values and thus even if the value is NULL it is still counted.**

**注意:除了COUNT()之外，所有这些方法都忽略空值，如果所有值都为空则返回空值。COUNT()计数的是行而不是值，因此即使值是NULL，它仍然被计数。**

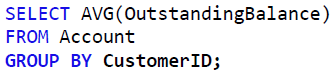
1. **Aggregate Examples: COUNT and AVG**

* ****
* ****

1. **GROUP BY / HAVING**

* **Group by groups all records together over a set of attributes**

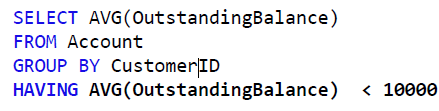
**通过一组属性将所有记录分组**

* **Frequently used with aggregate functions**
* **Example:**
* **What is the average balance PER CUSTOMER?**
* ****
* **The only way to put a selection condition over a group by statement is by using having clause**

**将选择条件置于group by语句之上的唯一方法是使用having子句**

* **Example:**
* **What is the exact average balance per customer for customers whose average balance is under 10000?**

**对于平均余额低于10000的客户，每个客户的确切平均余额是多少?**

* ****

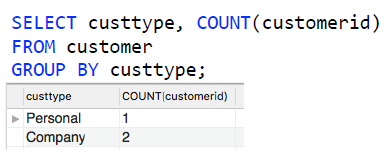
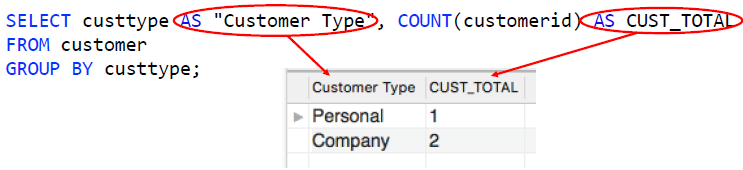
1. **Changing Column Heading in Output在输出中更改列标题**

* **We can rename the column name of the output by using the AS clause**

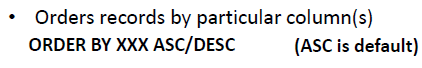
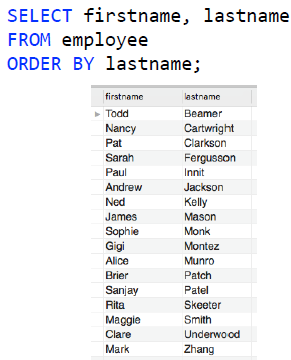
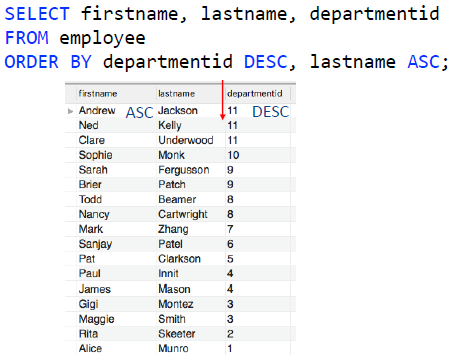
**可以使用AS子句重命名输出的列名**

* **If it contains a gap it must be in straight double quotes**

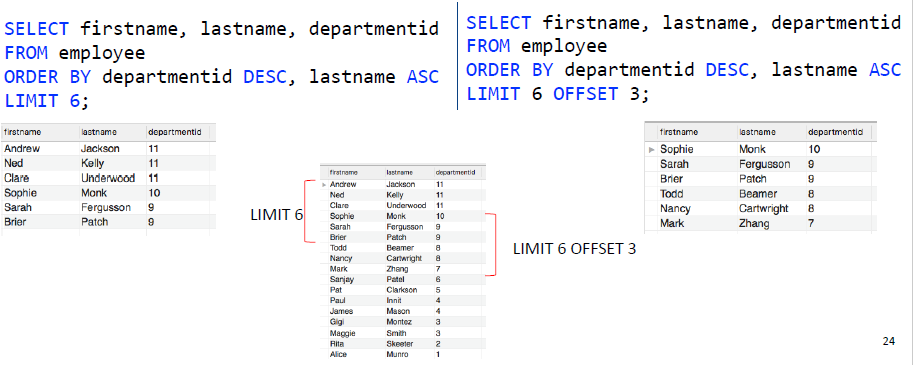
**如果它包含一个空白，它必须在直双引号**

* ****
* ****

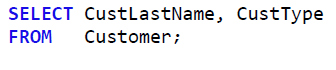
1. **ORDER BY**

* ****
* ****
* ****
* **DESC-降序**
* **ASC-升序**

1. **LIMIT and OFFSET限制和偏移**

* **LIMIT N-limits the output size**
* **OFFSET N-skips first N records跳过前N条记录**
* ****

1. **Joining Tables**
2. **Table name qualification and aliases表名限定和别名**

* **A simple SQL query using the SELECT statement**
* ****
* **Whenever we use a column name in a Query, we can qualify (prefix) with the appropriate table name**

**每当在Query中使用列名时，都可以使用适当的表名限定(前缀)**

* ****
* **We can use an alias for table name我们可以为表名使用别名**
* ****
* **The alias only applies to the current SQL statement (not remembered)**
* **Once an alias is used within an SQL statement, you can cannot refer to the original tablename within that SQL statement**

**一旦在SQL语句中使用了别名，就不能在该SQL语句中引用原始表名**

* **The usefulness of this feature becomes apparent when SQL statements refer to columns from two or more tables**

**当SQL语句引用两个或多个表中的列时，这个特性的用处就很明显了**

1. **Joining tables together**

* ****
* **the result set is a cross product or Cartesian Product**

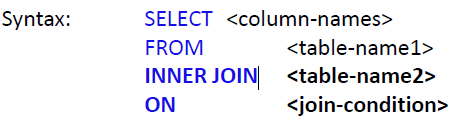
**结果集是外积或笛卡尔积**

* ****
* **Every row in the first table has been matched with every row in the second table 第一个表中的每一行都与第二个表中的每一行相匹配**

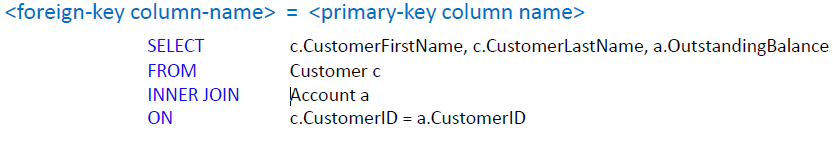
1. **Joins: INNER JOIN**

* **An Inner Join returns a result set that contains only data that satisfies a Foreign Key-Primary Key condition**

**Inner Join返回只包含满足外键-主键条件的数据的结果集**

* ****
* **The <join condition> is normally in the format**

**<连接条件>通常采用这种格式**

* ****

1. **Joins: Different Types**

* **Inner/ Equi join:**
* **Joins the tables over keys**
* 
* **Natural Join:**
* **Joins the tables over keys. The condition does not have to be specified (natural join does it automatically), but key attributes have to have the same name.**

**通过键连接表。条件不必指定(自然连接会自动指定)，但关键属性必须具有相同的名称。**

* ****
* **Outer join:**
* **Joins the tables over keys**
* **Can be left or right (see difference)**
* **Includes records that don’t match the join from the other table**

**包括与来自另一个表的联接不匹配的记录**

* ****
* ****