中国人民大学数据挖掘中心出品

MySQL 数据库自学教程（1.0 版）

——MySQL 入门及其可视化管理工具 Navicat 的使用

我们的网站: **http://rucdmc.net/**

2014.11

1、MySQL 介绍

MySQL 是一个小型关系型数据库管理系统，开发者为瑞典 MySQL AB 公司。在 2008 年 1 月 16 号被 Sun 公司收购。而 2009 年,SUN 又被 Oracle 收购.对于 Mysql 的前途,没有任何人抱乐观的态度.目前 MySQL 被广泛地应用在 Internet 上的中小型网站中。由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，许多中小型网站为了降低网站总体拥有成本而选择了 MySQL 作为网站数据库。但是我们从事数据分析时，主要是利用 MySQL 的查询功能对数据表格进行查询汇总。特别是在对处理万行以上的数据时，MySQL 不仅比 Excel 的运算速度来得快，而且还能完成许多 Excel 所不能完成的连接查询的功能。

2、MySQL 的下载、安装及配置

MySQL 下载

我们这里选择 MySQL 5.6.14 作为我们的安装版本，作为初学者，我们建议大家安装 windows 版的 MySQL。

自 MySQL 版本升级到 5.6 以后，其安装及配置过程和原来版本发生了很大的变化，下面详细介绍 5.6 版本 MySQL 的下载、安装及配置过程。

图 2.1 MySQL5.6

目前针对不同用户，MySQL 提供了 2 个不同的版本：

- MySQL Community Server：社区版，该版本完全免费，但是官方不提供技术支持。
- MySQL Enterprise Server：企业版，它能够高性价比的为企业提供数据仓库应用，功能齐全，但是该版本需付费使用，官方提供电

话及文档等技术支持。

对我们学习 MySQL 数据库而言，下载免费的社区版的即可。目前最新的 MySQL 版本为 MySQL 5.6，可以在官方网站（http://dev.mysql.com/downloads/windows/installer/5.6.html）上面下载该软件。现在 MySQL 的最新版本可能显示为 5.6.21，没有关系，它与 5.6.14 没有太大的不同，大家可以直接安装最新的版本。

图 2.2 在 MysQL 官网的下载页面上选择 MySQL 版本

MySQL 下载完成后，找到下载到本地的文件，按照下面所示的步骤双击进行安装：

MySQL 的安装过程及配置过程

有关于 MySQL 的安装及配置过程，网上有着详细的图文并茂的过程，所以安装过程可以详见下面连接，文档可以免费下载：
http://wenku.baidu.com/link?url=gGD3Q183fv0SOXD6YR7NJlwm7nCzdEI OahRnzEojvATAGzAeXFh6MFWgZdtMKQ9UOXe3acARSeFPhhiZ4Y9C7lP3vbvcLVyMwgpN98jC

【常见故障】处理 3306 端口被占用而无法启动 MySQL 服务的问题

安装完 MySQL 之后便可以启动 MySQL，如果大家遇见 MySQL 无法打开的情况，则要去电脑的任务管理器里查看 MySQL 的服务是

否已经启动。具体步骤为，打开任务管理器（Ctrl+Alt+Delete），选择“进程”标签，查看 MySQL 服务进程 mysqld.exe 是否已经启动，如下图所示。

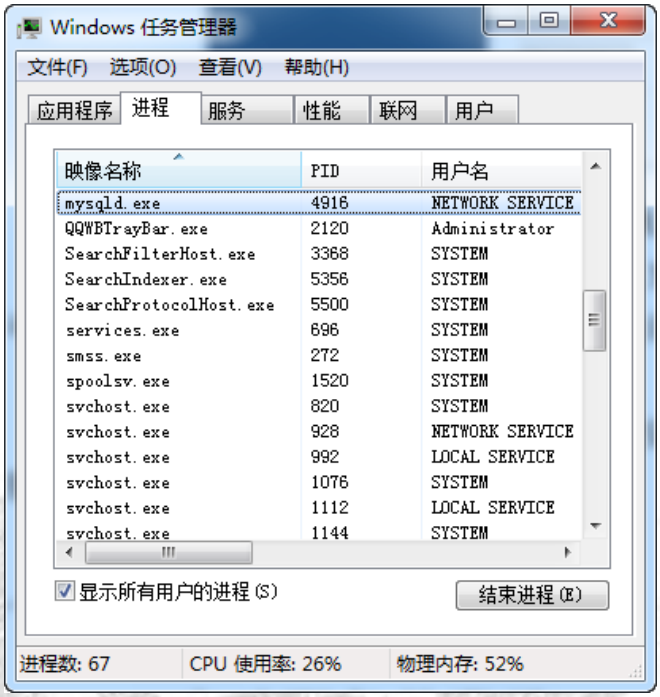

图 2.3 任务管理器窗口

如果没有遭到启动的 mysqld.exe，则进入“服务”标签，遭到 MySQL 所对应的服务，看是否显示正在运行，如果服务暂停，则右键选择启动即可。

另外，MySQL 在安装时默认使用 3306 的端口，但是电脑中可能有些其他先安装的程序会占用这个窗口（例如迅雷之类的软件），这个时候 MySQL 会由于端口被占用而无法启动服务。我们可以关闭在 3306 上运行的程序再启动 MySQL 的服务即可。具体操作步骤如下：

1、点击开始->运行->在输入框中输入 cmd，进入 DOS 界面。

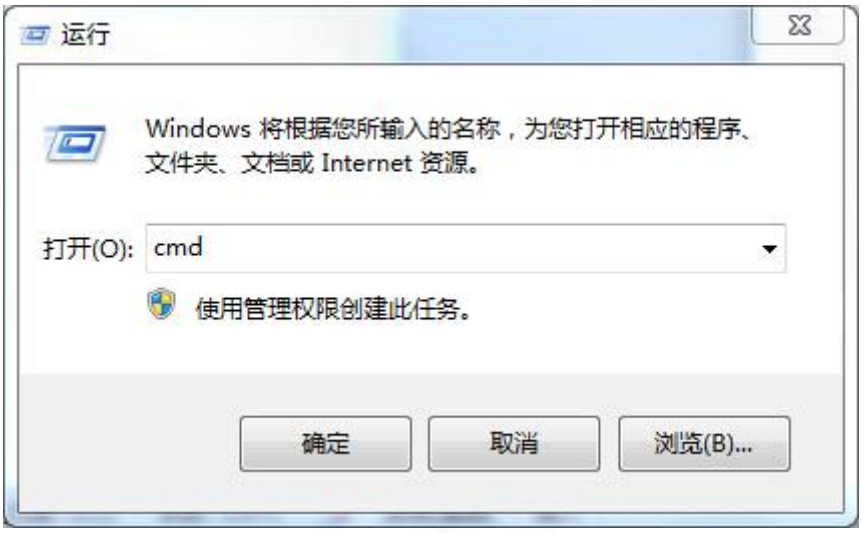

图 2.4 进入 DOS 界面

3、 输入命令 netstat -aon|findstr "3306"命令，查看 3306 端口都被哪个进程占用了。最后一列是进程的 PID，可以看见现在电脑的 3306 端口正在被 PID 为 6328 的程序占用。

图 2.5 查看占用端口程序的进程 PID

4、 输入命令 tasklist|findstr "3628"来查看 PID 为 3628 的进程是哪个程序。

5、 可以看见，我现在电脑里占用 3306 端口的正是 MySQL 程序，如果发现是其他程序，则可以在任务管理器的进程里结束该程序的进程，并在服务标签里启动 MySQL 服务即可。

3、MySQL 可视化管理工具 Navicat

MySQL 安装好后就可以在 DOS 界面下利用命令行进行操作了，但是为了使初学者能更快地掌握 MySQL 的掌握，我们推荐安装 MySQL 的可视化管理工具 Navicat 进行操作，这样就可以避免大量使用命令行操作，可视化的操作对于初学者来说可能更容易接受一些。当然，MySQL 用熟了之后，可以尝试在 DOS 界面下采用命令行操作，具体操作教程可以参见一本经典的 MySQL 入门教材《MySQL 必知必会》。

MySQL 的可视化管理软件有很多种，而 Navicat 是其中一种快速、可靠并价格便宜的数据库管理工具，其实它就是一个软件，以直觉化的图形用户界面来对本机或远程的 MySQL、SQL Server、SQLite、Oracle 及 PostgreSQL 数据库进行管理及开发。网络上有许多有关于 Navicat 的教程，我们选取了以下网络资料。

Navicat 下载地址：http://www.cr173.com/soft/26935.html

（这里用的是 8.2 的版本，因为我们后面演示的安装及使用教程也是 8.2 版本的，大家也可以去官网下载最新版的进行安装）

Navicat 安装教程：

http://xiaosuncunzhang.blog.51cto.com/317407/680107

Navicat 使用教程：

http://xiaosuncunzhang.blog.51cto.com/317407/680228

4、MySQL 查询语言基础教程（原创）

(此处基于你已经熟悉了 Navicat 的 MySQL 可视化界面操作)

准备工作：

1、 **创建数据库：**打开 Navicat，在左侧数据库列表菜单中单击右键，选择新建数据库，将数据库名称命名为 student，字符集选择最底下 utf-8，这样可以识别数据表中的中文。

2、 **导入数据表：**点击界面上方的“导入向导”，选择 Excel 文件，分别导入学生，课程，成绩三个表（三张表格在“要用的数据表”文件夹中）。在源表-目标表那一步要勾选创建表，并为三张表的目

标表重新命名，防止覆盖已存在的表。

3、 **删去首记录：**双击打开导入的数据表，发现数据表的第一行数据是行名，简单的处理方法就是在第一行数据的最左边单机右键，选择删去该条记录。

4、 **创建查询：**点击页面上方的“查询”，再点击创建查询，然后就可以输入查询语句啦~

最最常用的 select 查询语句：

单表查询

（只用一张表的情况下进行查询，大家可以输入相应语句，并用左键拖选拉黑该行语句，并右键选择“运行所选”，输出结果后，点击查询窗口中的“查询编辑器”菜单，可以返回编辑窗口。）

1、 select + 函数

查询当前时间：select now(); #now()是自带的时间函数，每行语句以英文分号结束。

查询当前年份：select year(now());

查询大写字母 A 的 ASCII 码：select ASCII('A');

2、 select+列名+from+表名

查询学生表中的所有列：select * from 学生; #*是通配符的一种，表示所有列名。

查询学生的姓名和性别：select 姓名,性别 from 学生; #多个列名间用英文逗号隔开

将上述查询的结果作成一张新表并存起来，且命名为“学生性别”：

create table 学生性别(select 姓名,性别 from 学生);

#在表格界面右键刷新便能见到新表

3、 select+列名+from+表名+where+条件

查询学生表中所有男同学的信息：select * from 学生 where 性别='男'; #字符要用单引号括起来

查询学生表中所有年龄不超过 26 岁的学生的学号和姓名以及出身年份： select 学号,姓名,year(出生日期) as 出生年份 from 学生 where (year(now())-year(出生日期))<=26;

#year()函数取出年份信息，as 可以创建一个新列名。

4、 ***select+列名+from+表名+where+条件+group by +分组列名***

查询班级中男女生的数量：select 性别,count(*) as 人数 from 学生 group by 性别;

#count()用来统计行数，通配符*也可以换成学号，结果一样。

查询班级男生中各年份出生的人数：

select year(出生日期) as 出身年份,性别, count(学号) as 人数 from 学生 where 性别='男' group by year(出生日期);

5、 ***select+列名+from+表名+where+条件+group by +分组列名+order by+排序列名***

将所有学生的学号按降序排列：select * from 学生 order by 学号 desc; #默认情况下是按升序排列的，在后面加上‘desc’则按降序排列

查询班级女生中各月份的出生人数，并按月份大小排序：select month(出生日期) as 出生月份,性别, count(*) as 人数 from 学生 where 性别='女' group by month(出生日期) order by month(出生日期);

多表查询：

1、 ***我想知道成绩表中的那些学号都是对应着哪些人（两表连接）：***

select 学生.姓名,成绩.* from 学生,成绩 where 学生.学号=成绩.学号;

#当有多张表进行连接查询的时候，列名须说明是哪张表里头的，用“表名.列名”形式表示，如果想让姓名在结果的第二列，则 select 后的语句改为“成绩.学号,学生.姓名,成绩.课程号,成绩.成绩”。where 后面跟着是关联的条件，大家可以把 where 语句即之后的条件去掉，看看二者的结果有何不同。

2、 ***我想知道成绩表中的那些不及格的都是哪些人（两表连接加条件）：***

select 学生.姓名,成绩.* from 学生,成绩 where 学生.学号=成绩.学号 and 成绩.成绩<60;

3、 我想知道每个学生都选了什么课，且每门课的成绩是多少（三表连接）：

select 学生.学号,学生.姓名,课程.课程名,成绩.成绩 from 学生,成绩,课程 where 学生.学号=成绩.学号 and 成绩.课程号=课程.课程号;

#多表连接，一定要考虑表和表连接的条件（某个你中有我、我中有你的列名），这里三表连接需要两个连接条件，用关系运算符 and 表示同时要满足两个条件。

4、我想知道每个学生的总成绩是多少（用 sum()函数）,并按总分的降序排列：

select 学生.学号,学生.姓名,sum(成绩.成绩) as 总分 from 学生,成绩,课程 where 学生.学号=成绩.学号 and 成绩.课程号=课程.课程号 group by 学生.学号 order by 总分 desc;

5、 我想知道每门课程的平均分是多少（用 avg()函数），并按平均分的升序排列：

select a.课程号,a.课程名,avg(b.成绩) as 平均分 from 课程 as a,成绩 b where a.课程号=b.课程号 group by a.课程名 order by 平均分;

#有时候我们为了使表名的输入更加简洁，经常在 from 语句后面重新为表名另起一个别名。例如“课程 as a”，之后在该语句的其他地方凡是需要输入表名“课程”的均可以以 a 替代，有时为了更加简便，as 还可以省略掉，例如“成绩 b”

6、 我想知道每门课程的最高分是多少？（max()函数）

select a.课程号,a.课程名,max(b.成绩) as 最高分 from 课程 as a,成绩 b where a.课程号=b.课程号 group by a.课程名;

#要求最低分则可以改用 min()函数

#这里出现了一个问题，让我们仔细观察成绩表中，06010 这个考生在 A102 英语考试中成绩为 100，但是这里查询出来的结果却显示为英语最高分为 99，再检查语句没有错误之后，那问题出现在哪里呢？问题出在数据的格式上。在导入数据时，我们默认将所有数据都存为 varchar 字符格式，这在用 max()比大小的时候会出问题，出现 99>100 的现象，所以我们应该在成绩表上单击右键，

选择“设计表”，将成绩这项的格式改为整形“int”即可。

7、 *我想知道每门课程的最高分都是谁？（子查询）*

分两步完成：这里用一个 select 语句不好处理，我们可以分两步解决，先得到各科最高分表然后替换原来的课程表进行相关查询。

create table 各科最高分(select a.课程号,a.课程名,max(b.成绩) as 最高分 from 课程 as a,成绩 b where a.课程号=b.课程号 group by a.课程名); #创建新表“各科最高分”

select a.学号,a.姓名,c.最高分 from 学生 a,成绩 b,各科最高分 c where a.学号=b.学号 and b.课程号=c.课程号 and b.成绩=c.最高分; #三表连接求每科最高分是谁

子查询完成：我们可以将上述两个 select 语句写成一条语句，将前一个 select 的结果直接作为后一个 select 要查询的表，这样减少了中间创建表的环节。

select a.学号,a.姓名,c.最高分 from 学生 a,成绩 b, (select a.课程号,a.课程名,max(b.成绩) as 最高分 from 课程 as a,成绩 b where a.课程号=b.课程号 group by a.课程名) c where a.学号=b.学号 and b.课程号=c.课程号 and b.成绩=c.最高分;

8、 *左连接，处理两表不对称的连接特别管用。（左连接用于处理左多右少的情况）*

预备工作：在成绩表中添加两行，假设 06111 和 06222 是两位转专业过来的同学，成绩表里有他们的学号、课程号、和成绩，但是班级的名单里还没有他们的学号、姓名和个人信息。我们可以双击打开成绩表，然后再左下方点加号，添加记录：

学号	课程号	成绩
06111	A101	75
06222	A102	85

重复步骤 1 的工作，我们想知道这些学号都对应着什么姓名，但是 06111 和 06222 虽然没有姓名但是我们也希望他们保留在结果里，这时就是某一表的数据多于另一表可又要显示多表的数据的情况，这时可以考虑用左连接：

select 成绩.学号,学生.姓名,成绩.课程号,成绩.成绩 from 成绩 left join 学生 on 学生.学号=成绩.学号;

from 多表 left join 少表 on 连接条件，结果如下：

06012	刘辉	A101	45
06012	刘辉	A102	66
06012	刘辉	A103	85
06008	张良	A103	62
06111	(Null)	A101	75
06222	(Null)	A102	85

练习：

1、 查询学生表中所有女同学的信息。
2、 查询学生表中所有下半年出生的学生信息。
3、 查询学生表中张姓学生的信息。
4、 按院系来统计学生表中各个院系有多少人，并将结果以人数由少到多降序排列。
5、 查询高等数学这门课所有成绩及格的学生。
6、 查询每个院系英语这门课的平均分是多少。
7、 查询每个学生的加权平均分是多少（以学分未权值），结果按加权平均分由高到低排序。
8、 我想知道每门课程的最低分都是谁。

（由于数据量较小，查询结果如果与自己手工计算出来的结果一样便说明你的查询写对了。）

5、MySQL 查询语言进阶教程（原创）

(此处在查询窗口中进行纯命令行操作)

现在我们有某音乐网站某一天的实时数据，记录着该天访问该网站的用户的行为信息。数据主要存储在两张表里头，user.txt 和 song.txt，两个文件都是以 utf-8 进行编码的。

user.txt 变量说明：
sectionid:记录用户行为的 id。
uid:标识用户的 id。
datetime:用户发生行为的时间和日期。
type:用户具体的操作行为，v 是浏览（试听）歌曲，d 是下载歌

曲，s 是搜索歌曲。

sid:歌曲的编号。

属性与属性之间以“{}”作为分隔符，真实数据如下：

```
sectionid{}uid{}datetime{}type{}sid
1003541{}347759{}20110701 000000{}d{}36169946
1010391{}352872{}20110701 000015{}d{}36233221
1001051{}346041{}20110701 000018{}v{}361538
```

song.txt 变量说明：

sid:歌曲的编号。

song:歌曲名称。

singer:歌手（有缺失）。

属性与属性之间也以“{}”作为分隔符，真实数据如下：

```
sid{}song{}singer
361402{}怎样(mvp情人){}
36162969{}心只有你{}刘德华
36133898{}心经{}陈奕迅
```

1、 第一步当然是将数据导入数据库中，这里我们教大家如何用命令导入txt数据。在导入数据之前，我们先要在数据库中创建两张表格，分别命名为usertable和songtable,如果你使用的是命令行界面则直接输入下列代码，如果你使用的是Navicat数据库管理软件，则可以在新建的查询窗口中输入下列代码。

```
create table usertable(
secid varchar(255),
uid varchar(255),
datetime varchar(255),
type char(1),
sid int
);
create table songtable(
sid int,
song varchar(255),
singer varchar(255)
);
```

#创建user表，为每个属性指定数据类型，这里我们都采用字符型

的数据类型。常见的字符型数据类型分两种，一种是char,一种是varchar,他们的区别是char的长度是固定的，而varchar的长度是可以变化的, 比如，存储字符串“abc"，对于char(20)，你存储的字符也要占20个字节(包括17个空字符)，而同样的varchar(20)则只占用3个字节的长度，20只是最大值，当你存储的字符小于20时，按实际长度存储。varchar的最大储存长度为255，即能储存255个字符或127个汉字，我们这里不妨将其取作最大长度255。这里的sid之所以设为整型，是因为后面要利用它进行两表连接，我们尝试过将它设为varchar字符串的形式，但是好像它不太认“=”这个运算符TAT，造成无法连接。

2、 利用load()函数分别导入txt数据。命令如下：

#导入user.txt文件，xxxx为具体的路径名称。

```
load data local infile "xxx\\song22.txt" into table songtable fields terminated by "{}" ignore 1 lines;
```

#导入song.txt文件

```
load data local infile 'C:\\xxxx \\songdata\\song.txt' into table songtable fields terminated by '{}' ignore 1 lines;
```

#local表示导入的是本地的数据集
#fields terminated by用来指定分隔符，如果没有指定分隔符，则默认情况下以一个制表符为分隔符，以回车'\n'为换行符。
#ignore 1 lines表示忽略数据的第一行，因为建立表格的时候已包含列名。

3、 了解数据的基本情况，简单计算两张表各有多少行的数据。

```
select count(*) from usertable;
select count(*) from songtable;
```

#两条语句可以同时运行，但是每一个select语句只返回一个查询结果，两个语句之间要用英文的分号间隔开，不然会出错。从结果可以看出，usertable表中有9917行数据，songtable表中有2038行数据。

4、 查询usertable中一共有多少个用户

```
select count(distinct uid) from usertable;
```

#usertable中用uid来标识用户的id，数据中有重复，所以需使用distinct来对uid进行去重处理。从结果看出，共有8031个用户id数。

5、 查询usertable中每个用户浏览了多少次，下载了多少次，搜索了多少次。

```
select
uid,
sum(if(type='v',1,0))as vnum,
sum(if(type='d',1,0))as dnum,
sum(if(type='s',1,0))as snum
from usertable group by uid;
```

#if 函数的三个参数分别为条件，条件为真返回第一个值，条件不成立返回第二个值

6、 处理 songtable 中的空值，将 singer 属性值为空的记为“无名氏”。

```
update songtable set singer='无名氏' where char_length(singer)=1;
```

#varchar 类型字符串的空值其实是个空字符串，它的长度为 1，所以利用 char_length 函数可以计算出字符串的长度。

7、 将两表按 sid 连接，尝试分别用内连接，左连接和全连接，看看结果有何不同。

#SQL 拥有四种连接：左外连接，右外连接、内连接和全连接；内连接对典型的连接运算，使用像“=”或“>”之类的比较运算符，外连接可以是左连接、右连接和全连接。

内连接：（我们最常见的连接,将结果保存为 usersongtable 表）

```
create table usersongtable(select
usertable.*,songtable.song,songtable.singer  from usertable,
songtable where usertable.sid=songtable.sid);
```

左连接：

```
select usertable.*,songtable.song,songtable.singer  from usertable
left join songtable on usertable.sid=songtable.sid;
```

全连接：

```
select usertable.*,songtable.*  from usertable left join songtable on
usertable.sid=songtable.sid
union
select usertable.*,songtable.*  from usertable right join songtable
on usertable.sid=songtable.sid;
```

#mysql 中没有全连接，可通过 union 连接

8、 查询该天最受欢迎的前 20 首歌曲

```
select song,count(song) as num from usersongtable group by song
order by num desc
limit 20;
```

#这里不管用户的type是什么，认为出现越多条记录的歌曲即越受欢迎的歌曲。那么，首先通过查询建立排序序列，然后利用limit函数选出前20位

9、 查询该天最受欢迎的前 20 名歌手

```
select singer,count(singer) as num from usersongtable group by singer
order by num desc
limit 20;
```

#方法与第8题一样

10、 查询每首歌曲该天的浏览量，下载量和搜索量。

```
select
sid,
song,
sum(if(type='v',1,0))as vnum,
sum(if(type='d',1,0))as dnum,
sum(if(type='s',1,0))as snum
from usersongtable
group by sid;
```

11、 查询每个时段的浏览量，下载量和搜索量。

```
select
substr(datetime,10,2) as hourtime,
sum(if(type='v',1,0))as vnum,
sum(if(type='d',1,0))as dnum,
sum(if(type='s',1,0))as snum
from usertable
group by substr(datetime,10,2);
```

#substr 函数用以截取字符串，substr(str,pos,len)第一个参数为指定字符串，第二个参数为开始位置，第三个参数为长度

6、其他参考材料

以上便是我们 DMC 整理并编写的 MySQL 自学教程，我们的出发点是令计算机新手们更快地掌握 MySQL 这个数据分析处理的工具。但是如果想深入学习 MySQL 我们还是建议在阅读完我们的自学教程之后再继续专门找一本入门的 MySQL 的书籍进行深入的学习，书中毕竟比我们这里写得详细些。下面我们推荐一个前人在自学过程中认为比较好的 MySQL 的自学视频和自学书籍，希望对您的学习有所帮助。

视频教程推荐：

有些新手可能更习惯于通过看视频来学习，这里我们推荐后盾网的 MySQL 教程（向军老师，一共八集），大家可以自行上网搜索。第八集可以不看，前七集每天一集，一周正好看完。看完视频以后该懂的都懂了，要熟悉的话就自己找些练习做就行。

书籍材料推荐：

《MySQL 必知必会》是英国人 Ben Forta 写的，网上可以下载 pdf 版。这本十六开的书写的很简练，每章也就几页纸，该有的都提到了。看书快的人两天就可以读完了，书中 1-17 章、19-22 章是重点内容，其他的比较深，可以等以后用到了再看。