sql server 2008 连接到服务器

安装vs会自带sqlserver的数据库。但是没有数据库的管理界面。是sqlserver的精简版。

服务器名称选择默认，一般是localhost或127.0.0.1或者.都可以。然后是加上我们的数据库的实例。默认的实例名称是不用加上去的。

身份验证，选择window身份验证和sqlserver身份验证的区别。链接本机是没有任何区别，链接同桌的机器上的数据库就有区别了。

第一步（SQL2005、SQL2008）：

开始-->程序-->Microsoft SQL Server 2008（或2005）-->配置工具-->SQL Server 配置管理器-->SQL Server网络配置-->MSSQLSERVER(这个名称以具体实例名为准) 的协议-->TCP/IP-->右键-->启用

第二步：

SQL2005:

开始-->程序-->Microsoft SQL Server 2005-->配置工具-->SQL Server 2005外围应用配置器-->服务和连接的外围应用配置器 -->Database Engine -->远程连接，选择本地连接和远程连接并选上同时使用Tcp/Ip和named pipes.

（附：如何配置 SQL Server 2005 以允许远程连接：[http://support.microsoft.com/kb/914277/zh-cn](http://support.microsoft.com/kb/914277/zh-cn" \t "http://blog.csdn.net/htl258/article/details/_blank) ）

SQL2008:

打开SQL Server Management Studio-->在左边[对象资源管理器]中选择第一项(主**[数据库](http://lib.csdn.net/base/mysql" \o "MySQL知识库" \t "http://blog.csdn.net/htl258/article/details/_blank)**引擎)-->右键-->方面-->在方面的下拉列表中选择[外围应用配置器]-->将RemoteDacEnable置为True.

Express：

如果XP有开防火墙，在例外里面要加入以下两个程序:  
C:/Program Files/Microsoft SQL Server/MSSQL.1/MSSQL/Binn/sqlservr.exe,  
C:/Program Files/Microsoft SQL Server/90/Shared/sqlbrowser.exe

第三步：（开启server服务。）

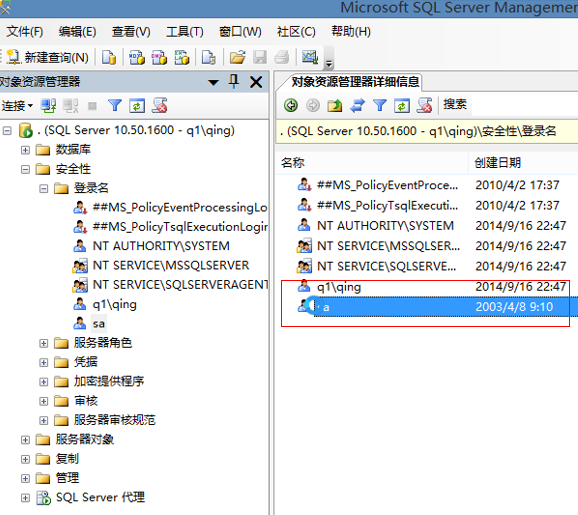
开始-->程序-->Microsoft SQL Server 2008（或2005）-->配置工具-->SQL Server 配置管理器-->SQL Server 服务-->右击SQL Server(MSSQLSERVER) （注：括号内以具体实例名为准）-->重新启动

在公司里，我们连的数据库，是连在防止服务器上的sqlserver数据库。

、

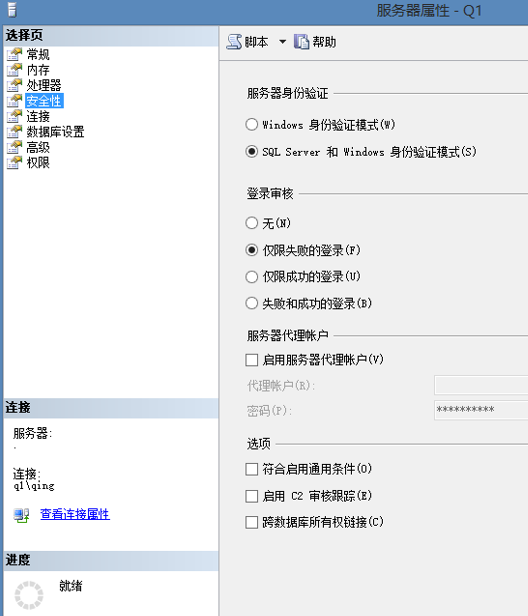
1.先用windows链接进入，然后改用户名。

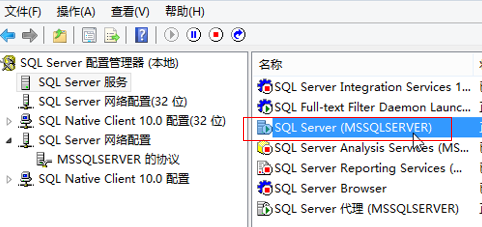
这里修改登录名。

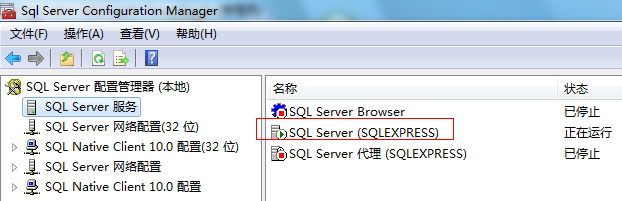


如果不记得密码，在此设置密码。状态授予和启用。

2.







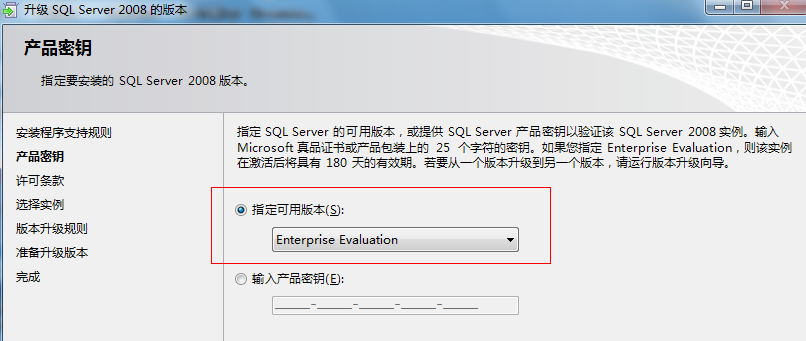
SQLExpress是简介版，上面sqlserver（mssqlserver）是完整版。

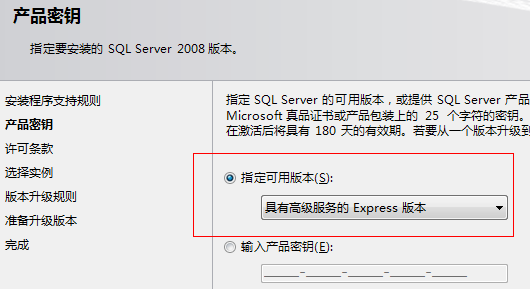
<https://jingyan.baidu.com/article/1e5468f928e106484961b7b0.html>

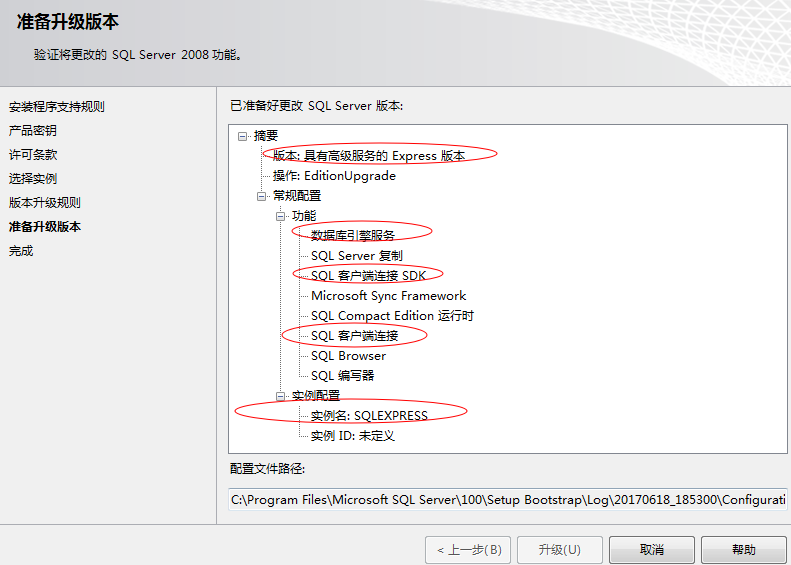
<https://jingyan.baidu.com/article/1e5468f928e106484961b7b0.html>

# vs2010自带sql2008数据库使用方法 添加管理界面

1.安装好VS2010后，开始—>Microsoft SQL Server —>配置工具—>SQL Server 安装中心—> 维护—>版本升级（可能要重启），一直下一步，**当中要选择“具有高级服务的Express版本”确认完成升级之后**，记得关闭SQL Server 安装中心（以免与第二步中的安装中心混淆）



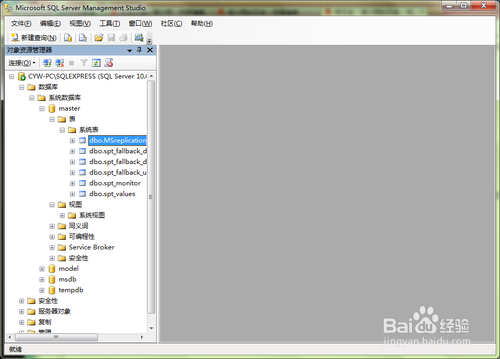




2点击刚刚下载文件SQL Server Management Studio。会自动弹出新的SQL Server 安装中心的界面，选择: 安装—>“全新SQL Server 独立安装或向现有安装添加功能”，然后一路下一步。

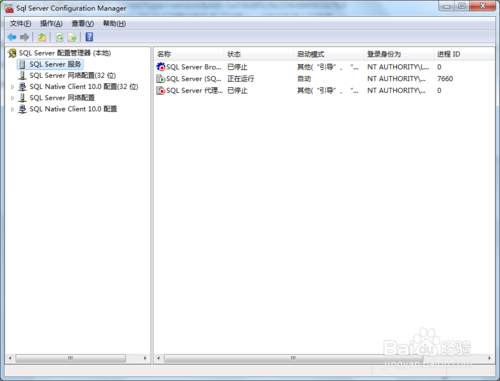
[](http://jingyan.baidu.com/album/1e5468f928e106484961b7b0.html?picindex=2" \t "https://jingyan.baidu.com/article/_self)

3完成之后，开始菜单中Microsoft SQL Server 2008中就多了“SQL Server Management Studio”这一项。打开后在，服务器名称填写：“你计算机的名字\SQLEXPRESS”

[](http://jingyan.baidu.com/album/1e5468f928e106484961b7b0.html?picindex=3" \t "https://jingyan.baidu.com/article/_self)

4

4启动SQLEXPRESS的步骤：开始—>Microsoft SQL Server —>配置工具—>SQL Server配置管理器—>SQL Server服务（左侧）—>SQL Server（SQLEXPRESS）（右侧）

[](http://jingyan.baidu.com/album/1e5468f928e106484961b7b0.html?picindex=4" \t "https://jingyan.baidu.com/article/_self)

数据库介绍与分类... 1

1.1 数据库介绍... 2

1.1.1什么是数据库... 2

1.2数据库的种类... 2

1.2.1关系型数据库介绍... 2

1.2.2非关系型数据库介绍... 3

1.3 常用关系型数据库产品介绍... 4

1.3.1 Oracle数据库... 4

1.3.2 MySQL数据库... 5

1.3.3 MariaDB数据库... 5

1.3.4 SqlServer数据库... 6

1.3.5 Access数据库... 6

1.3.6 其他不常用数据库... 6

1.4 常用非关系型数据库产品介绍... 6

1.4.1 Memcached（key-value）... 6

1.4.2 Redis（key-value）... 7

1.4.3 MongoDB（Document-oriented）... 8

1.4.4 Cassandra（Column-oriented）... 8

1.4.5 其他不常用非关系型数据库... 8

1.5 数据库相关知识... 9

1.5.1数据库软件企业应用排名参考... 9

## 1.1      ****数据库介绍****

### 1.1.1什么是数据库

简单的说，数据库（database）就是一个存放数据的仓库，这个仓库是按照一定的数据结构（数据结构是指数据的组织形式或数据之间的联系）来组织、存储的，我们可以通过数据提供的多种方法来管理数据库里的数据。

当人们收集了大量的数据后，应该把它们保存起来进入近一步的处理，进一步的抽取有用的信息。当年人们把数据存放在[文件柜](http://www.baidu.com/s?wd=%E6%96%87%E4%BB%B6%E6%9F%9C&hl_tag=textlink&tn=SE_hldp01350_v6v6zkg6" \t "http://www.mamicode.com/_blank)中，可现在随着社会的发展，数据量急剧增长，现在人们就借助计算机和[数据库技术](http://www.baidu.com/s?wd=%E6%95%B0%E6%8D%AE%E5%BA%93%E6%8A%80%E6%9C%AF&hl_tag=textlink&tn=SE_hldp01350_v6v6zkg6" \t "http://www.mamicode.com/_blank)科学的保存大量的数据，以便能更好的利用这些数据资源。

## ****1.2数据库的种类****

    数据库通常分为层次式数据库、网络式数据库和关系式数据库三种。而不同的数据库是按不同的[数据结构](http://baike.baidu.com/view/9900.htm" \t "http://www.mamicode.com/_blank)来联系和组织的。而在当今的互联网中，最常见的数据库模型主要是两种，即关系型数据库和非关系型数据库。

### 1.2.1关系型数据库介绍

（1）、关系型数据库的由来

    虽然网状数据库和层次数据库已经很好的解决了数据的集中和共享问题，但是在数据库独立性和抽象级别上扔有很大欠缺。用户在对这两种数据库进行存取时，仍然需要明确数据的存储结构，指出存取路径。而关系型数据库就可以较好的解决这些问题。

（2）、关系型数据库介绍

  关系型数据库模型是把复杂的数据结构归结为简单的二元关系（即二维表格形式）。在关系型数据库中，对数据的操作几乎全部建立在一个或多个关系表格上，通过对这些关联的表格分类、合并、连接或选取等运算来实现数据库的管理。

  关系型数据库诞生40多年了，从理论产生发展到现实产品，例如：Oracle和MySQL，Oracle在数据库领域上升到霸主地位，形成每年高达数百亿美元的庞大产业市场。

（3）、关系型数据库表格之间的关系举例

|  |  |  |
| --- | --- | --- |
| 学生表 | | |
| 学号 | 姓名 | 年龄 |
| S001 | 张三 | 24 |
| S002 | 李四 | 25 |
| S003 | 王二麻子 | 26 |

|  |  |  |
| --- | --- | --- |
| 课程表 | | |
| 课程号 | 课程名 | 学分 |
| C001 | Linux | 25 |
| C002 | Python | 25 |
| C003 | MySQL | 25 |

|  |  |  |
| --- | --- | --- |
| 选课表 | | |
| 学号 | 课程号 | 成绩 |
| S001 | C001 | 90 |
| S002 | C001 | 97 |
| S003 | C001 | 94 |

### 1.2.2非关系型数据库介绍

#### 1.2.2.1 非关系型数据库诞生背景

NoSQL，泛指非关系型的数据库。随着互联网[web2.0](http://baike.baidu.com/view/733.htm" \t "http://www.mamicode.com/_blank)网站的兴起，传统的关系数据库在应付web2.0网站，特别是超大规模和高并发的[SNS](http://baike.baidu.com/subview/8258/5896174.htm" \t "http://www.mamicode.com/_blank)类型的web2.0纯[动态网](http://baike.baidu.com/view/528572.htm" \t "http://www.mamicode.com/_blank)站已经显得力不从心，暴露了很多难以克服的问题，而非关系型的数据库则由于其本身的特点得到了非常迅速的发展。NoSql数据库在特定的场景下可以发挥出难以想象的高效率和高性能，它是作为对传统关系型数据库的一个有效的补充。

   NoSQL(NoSQL = Not Only SQL )，意即“不仅仅是[SQL](http://baike.baidu.com/view/34.htm" \t "http://www.mamicode.com/_blank)”，是一项全新的数据库革命性运动，早期就有人提出，发展至2009年趋势越发高涨。NoSQL的拥护者们提倡运用非关系型的数据存储，相对于铺天盖地的[关系型数据库](http://baike.baidu.com/view/495393.htm" \t "http://www.mamicode.com/_blank)运用，这一概念无疑是一种全新的思维的注入。

#### 1.2.2.2 非关系型数据库种类

（1）、键值存储数据库（key-value）

 键值数据库就类似传统语言中使用的哈希表。可以通过key来添加、查询或者删除数据库，因为使用key主键访问，所以会获得很高的性能及扩展性。

键值数据库主要使用一个哈希表，这个表中有一个特定的键和一个指针指向特定的数据。Key/value模型对于IT系统来说的优势在于简单、易部署、高并发。

典型产品：Memcached、Redis、MemcacheDB

（2）、列存储（Column-oriented）数据库

列存储数据库将数据存储在列族中，一个列族存储经常被一起查询的相关数据，比如人类，我们经常会查询某个人的姓名和年龄，而不是薪资。这种情况下姓名和年龄会被放到一个列族中，薪资会被放到另一个列族中。

这种数据库通常用来应对分布式存储海量数据。

典型产品：Cassandra、HBase

（3）、面向文档（Document-Oriented）数据库

文档型数据库的灵感是来自于Lotus Notes办公软件，而且它同第一种键值数据库类似。该类型的数据模型是版本化的文档，半结构化的文档以特定的格式存储，比如JSON。文档型数据库可以看作是键值数据库的升级版，允许之间嵌套键值。而且文档型数据库比键值数据库的查询效率更高。

面向文档数据库会将数据以文档形式存储。每个文档都是自包含的数据单元，是一系列数据项的集合。每个数据项都有一个名词与对应值，值既可以是简单的数据类型，如字符串、数字和日期等；也可以是复杂的类型，如有序列表和关联对象。数据存储的最小单位是文档，同一个表中存储的文档属性可以是不同的，数据可以使用XML、JSON或JSONB等多种形式存储。

典型产品：MongoDB、CouchDB

（4）、图形数据库

图形数据库允许我们将数据以图的方式存储。实体会被作为顶点，而实体之间的关系则会被作为边。比如我们有三个实体，Steve Jobs、Apple和Next，则会有两个“Founded by”的边将Apple和Next连接到Steve Jobs。

典型产品：Neo4J、InforGrid

## ****1.3**** ****常用关系型数据库产品介绍****

### 1.3.1 Oracle数据库

ORACLE[数据库系统](http://baike.baidu.com/view/7809.htm" \t "http://www.mamicode.com/_blank)是美国ORACLE公司（甲骨文）提供的以[分布式数据库](http://baike.baidu.com/view/68389.htm" \t "http://www.mamicode.com/_blank)为核心的一组软件产品，是目前最流行的客户/服务器(CLIENT/SERVER)或B/S[体系结构](http://baike.baidu.com/view/1188494.htm" \t "http://www.mamicode.com/_blank)的数据库之一。比如SilverStream就是基于数据库的一种中间件。ORACLE数据库是目前世界上使用最为广泛的[数据库管理系统](http://baike.baidu.com/view/68446.htm" \t "http://www.mamicode.com/_blank)，作为一个通用的[数据库系统](http://baike.baidu.com/view/7809.htm" \t "http://www.mamicode.com/_blank)，它具有完整的[数据管理](http://baike.baidu.com/view/14717.htm" \t "http://www.mamicode.com/_blank)功能；作为一个关系数据库，它是一个完备关系的产品；作为[分布式数据库](http://baike.baidu.com/view/68389.htm" \t "http://www.mamicode.com/_blank)它实现了[分布式处理](http://baike.baidu.com/view/185352.htm" \t "http://www.mamicode.com/_blank)功能。但它的所有知识，只要在一种机型上学习了ORACLE知识，便能在各种类型的机器上使用它。

Oracle数据库最新版本为Oracle Database 12c。Oracle数据库12c 引入了一个新的多承租方架构，使用该架构可轻松部署和管理数据库云。此外，一些创新特性可最大限度地提高资源使用率和灵活性，如Oracle Multitenant可快速整合多个数据库，而Automatic Data Optimization和Heat Map能以更高的密度压缩数据和对数据分层。这些独一无二的技术进步再加上在可用性、安全性和大数据支持方面的主要增强，使得Oracle数据库12c 成为私有云和公有云部署的理想平台。

### 1.3.2 MySQL数据库

MySQL（发音为"my ess cue el"，不是"my sequel"）是一种[开放源代码](http://baike.baidu.com/view/1708.htm" \t "http://www.mamicode.com/_blank)的关系型[数据库管理](http://baike.baidu.com/view/600155.htm" \t "http://www.mamicode.com/_blank)系统（RDBMS），MySQL[数据库系统](http://baike.baidu.com/view/7809.htm" \t "http://www.mamicode.com/_blank)使用最常用的数据库管理语言--[结构化查询语言](http://baike.baidu.com/view/595350.htm" \t "http://www.mamicode.com/_blank)（SQL）进行数据库管理。

由于MySQL是[开放源代码](http://baike.baidu.com/view/1708.htm" \t "http://www.mamicode.com/_blank)的，因此任何人都可以在General Public License的许可下下载并根据个性化的需要对其进行修改。MySQL因为其速度、可靠性和适应性而备受关注。大多数人都认为在不需要[事务](http://baike.baidu.com/view/121511.htm" \t "http://www.mamicode.com/_blank)化处理的情况下，MySQL是管理内容最好的选择。

MySQL这个名字，起源不是很明确。一个比较有影响的说法是，基本指南和大量的库和工具带有前缀“my”已经有10年以上，而且不管怎样，MySQL AB创始人之一的Monty Widenius的女儿也叫My。这两个到底是哪一个给出了MySQL这个名字至今依然是个迷，包括开发者在内也不知道。

### 1.3.3 MariaDB数据库

MariaDB数据库管理系统是MySQL的一个分支，主要由开源社区在维护，采用GPL授权许可。开发这个分支的原因之一是：甲骨文公司收购了MySQL后，有将MySQL闭源的潜在风险，因此社区采用分支的方式来避开这个风险。 MariaDB的目的是完全兼容MySQL，包括API和命令行，使之能轻松成为MySQL的代替品。在存储引擎方面，使用XtraDB（英语：XtraDB）来代替MySQL的InnoDB。 MariaDB由MySQL的创始人Michael Widenius（英语：Michael Widenius）主导开发，他早前曾以10亿美元的价格，将自己创建的公司MySQL AB卖给了SUN，此后，随着SUN被甲骨文收购，MySQL的所有权也落入Oracle的手中。MariaDB名称来自Michael Widenius的女儿Maria的名字。

### 1.3.4 SqlServer数据库

SQL Server是由Microsoft开发和推广的[关系数据库管理系统](http://baike.baidu.com/view/687948.htm" \t "http://www.mamicode.com/_blank)（DBMS），它最初是由Microsoft、Sybase和Ashton-Tate三家公司共同开发的，并于1988年推出了第一个OS/2版本。Microsoft SQL Server近年来不断[更新](http://baike.baidu.com/view/23174.htm" \t "http://www.mamicode.com/_blank)版本，1996年，Microsoft 推出了SQL Server 6.5版本；1998年，SQL Server 7.0版本和用户见面；SQL Server 2000是Microsoft公司于2000年推出，目前最新[版本](http://baike.baidu.com/view/26783.htm" \t "http://www.mamicode.com/_blank)是2012年3月份推出的SQL SERVER 2012。

### 1.3.5 Access数据库

Microsoft Office Access是微软把数据库引擎的[图形用户界面](http://baike.baidu.com/view/185360.htm" \t "http://www.mamicode.com/_blank)和[软件](http://baike.baidu.com/view/37.htm" \t "http://www.mamicode.com/_blank)开发[工具](http://baike.baidu.com/subview/74746/11097695.htm" \t "http://www.mamicode.com/_blank)结合在一起的一个[数据库](http://baike.baidu.com/view/1088.htm" \t "http://www.mamicode.com/_blank)管理系统。它是[微软](http://baike.baidu.com/view/2353.htm" \t "http://www.mamicode.com/_blank)[OFFICE](http://baike.baidu.com/subview/5819/5989804.htm" \t "http://www.mamicode.com/_blank)的一个成员, 在包括[专业版](http://baike.baidu.com/view/1060712.htm" \t "http://www.mamicode.com/_blank)和更高版本的[office](http://baike.baidu.com/view/5819.htm" \t "http://www.mamicode.com/_blank)版本里面被单独出售。2012年12月4日,最新的[微软](http://baike.baidu.com/view/2353.htm" \t "http://www.mamicode.com/_blank)Office Access 2013在微软Office 2013里发布,微软Office Access 2010 是前一个版本。

MS ACCESS以它自己的格式将[数据存储](http://baike.baidu.com/view/551712.htm" \t "http://www.mamicode.com/_blank)在基于Access Jet的数据库[引擎](http://baike.baidu.com/view/53607.htm" \t "http://www.mamicode.com/_blank)里。它还可以直接导入或者[链接](http://baike.baidu.com/view/147669.htm" \t "http://www.mamicode.com/_blank)[数据](http://baike.baidu.com/subview/38752/8058697.htm" \t "http://www.mamicode.com/_blank)(这些数据存储在其他应用程序和数据库)。

[软件](http://baike.baidu.com/view/37.htm" \t "http://www.mamicode.com/_blank)开发人员和[数据](http://baike.baidu.com/view/38752.htm" \t "http://www.mamicode.com/_blank)架构师可以使用Microsoft Access开发应用软件,“高级用户”可以使用它来构建软件[应](http://baike.baidu.com/view/294827.htm" \t "http://www.mamicode.com/_blank)用程序。和其他办公应用程序一样，ACCESS支持Visual Basic宏语言,它是一个面向对象的编程语言,可以引用各种对象，包括DAO(数据访问对象),ActiveX数据对象,以及许多其他的[ActiveX](http://baike.baidu.com/view/28141.htm" \t "http://www.mamicode.com/_blank)组件。可视对象用于显示表和报表，他们的方法和属性是在VBA编程环境下，VBA代码模块可以声明和调用Windows[操作系统](http://baike.baidu.com/view/880.htm" \t "http://www.mamicode.com/_blank)函数。

### 1.3.6 其他不常用数据库

    DB2，PostgreSQL，Informix，Syase等。

## ****1.4**** ****常用非关系型数据库产品介绍****

### 1.4.1 Memcached（key-value）

**memcached**是一套分布式的快取系统，当初是Danga Interactive为了LiveJournal所发展的，但被许多软件（如MediaWiki）所使用。这是一套开放源代码软件，以BSD license授权协议发布。[1]

memcached缺乏认证以及安全管制，这代表应该将memcached服务器放置在防火墙后。[1]

memcached的API使用32位元的[循环冗余校验](http://baike.baidu.com/view/1647314.htm" \t "http://www.mamicode.com/_blank)（CRC-32）计算键值后，将资料分散在不同的机器上。当表格满了以后，接下来新增的资料会以LRU机制替换掉。由于memcached通常只是当作快取系统使用，所以使用memcached的应用程式在写回较慢的系统时（像是后端的数据库）需要额外的程式码更新memcached内的资料[1]

memcached 是以LiveJournal 旗下Danga Interactive 公司的Brad Fitzpatric 为首开发的一款软件。已成为mixi、hatena、Facebook、Vox、LiveJournal等众多服务中提高Web应用扩展性的重要因素。许多Web应用都将数据保存到[RDBMS](http://baike.baidu.com/view/57768.htm" \t "http://www.mamicode.com/_blank)中，应用服务器从中读取数据并在[浏览器](http://baike.baidu.com/view/7718.htm" \t "http://www.mamicode.com/_blank)中显示。但随着数据量的增大、访问的集中，就会出现RDBMS的负担加重、[数据库](http://baike.baidu.com/view/1088.htm" \t "http://www.mamicode.com/_blank)响应恶化、网站显示延迟等重大影响。

这时就该memcached大显身手了。memcached是高性能的分布式[内存](http://baike.baidu.com/view/1082.htm" \t "http://www.mamicode.com/_blank)[缓存服务器](http://baike.baidu.com/view/2396471.htm" \t "http://www.mamicode.com/_blank)。一般的使用目的是，通过[缓存](http://baike.baidu.com/view/907.htm" \t "http://www.mamicode.com/_blank)[数据库](http://baike.baidu.com/view/1088.htm" \t "http://www.mamicode.com/_blank)查询结果，减少数据库访问次数，以提高动态Web应用的速度、提高可扩展性。

Memcached 的守护进程（daemon ）是用C写的，但是[客户端](http://baike.baidu.com/view/930.htm" \t "http://www.mamicode.com/_blank)可以用任何语言来编写，并通过memcached协议与守护进程通信。但是它并不提供[冗余](http://baike.baidu.com/view/104445.htm" \t "http://www.mamicode.com/_blank)（例如，复制其hashmap条目）；当某个服务器S停止运行或崩溃了，所有存放在S上的键/值对都将丢失。

Memcached由Danga Interactive开发，其最新版本发布于2010年，作者为Anatoly Vorobey和Brad Fitzpatrick。用于提升LiveJournal . com访问速度的。LJ每秒[动态页面](http://baike.baidu.com/view/2065821.htm" \t "http://www.mamicode.com/_blank)访问量几千次，用户700万。Memcached将[数据库](http://baike.baidu.com/view/1088.htm" \t "http://www.mamicode.com/_blank)负载大幅度降低，更好的分配资源，更快速访问。

### 1.4.2 Redis（key-value）

redis是一个key-value[存储系统](http://baike.baidu.com/view/51839.htm" \t "http://www.mamicode.com/_blank)。和Memcached类似，它支持存储的value类型相对更多，包括string(字符串)、list([链表](http://baike.baidu.com/view/549479.htm" \t "http://www.mamicode.com/_blank))、set(集合)、zset(sorted set --有序集合)和hash（哈希类型）。这些[数据类型](http://baike.baidu.com/view/675645.htm" \t "http://www.mamicode.com/_blank)都支持push/pop、add/remove及取交集并集和差集及更丰富的操作，而且这些操作都是原子性的。在此基础上，redis支持各种不同方式的排序。与memcached一样，为了保证效率，数据都是缓存在内存中。区别的是redis会周期性的把更新的数据写入磁盘或者把修改操作写入追加的记录文件，并且在此基础上实现了master-slave(主从)同步。

Redis 是一个高性能的key-value数据库。 redis的出现，很大程度补偿了[memcached](http://baike.baidu.com/view/794242.htm" \t "http://www.mamicode.com/_blank)这类key/value存储的不足，在部 分场合可以对关系数据库起到很好的补充作用。它提供了Java，C/C++，C#，PHP，JavaScript，Perl，Object-C，Python，Ruby，Erlang等客户端，使用很方便。[1]

Redis支持主从同步。数据可以从主服务器向任意数量的从服务器上同步，从服务器可以是关联其他从服务器的主服务器。这使得Redis可执行单层树复制。从盘可以有意无意的对数据进行写操作。由于完全实现了发布/订阅机制，使得从数据库在任何地方同步树时，可订阅一个频道并接收主服务器完整的消息发布记录。同步对读取操作的可扩展性和数据冗余很有帮助。

### 1.4.3 MongoDB（Document-oriented）

MongoDB是一个基于分布式文件存储的数据库。由[C++](http://baike.baidu.com/view/824.htm" \t "http://www.mamicode.com/_blank)语言编写。旨在为WEB应用提供可扩展的高性能数据存储解决方案。

MongoDB 是一个介于[关系数据库](http://baike.baidu.com/view/68348.htm" \t "http://www.mamicode.com/_blank)和非关系数据库之间的产品，是非关系数据库当中功能最丰富，最像关系数据库的。他支持的数据结构非常松散，是类似[json](http://baike.baidu.com/view/136475.htm" \t "http://www.mamicode.com/_blank)的[bson](http://baike.baidu.com/view/3463259.htm" \t "http://www.mamicode.com/_blank)格式，因此可以存储比较复杂的数据类型。Mongo最大的特点是他支持的查询语言非常强大，其语法有点类似于面向对象的查询语言，几乎可以实现类似关系数据库单表查询的绝大部分功能，而且还支持对数据建立[索引](http://baike.baidu.com/view/262241.htm" \t "http://www.mamicode.com/_blank)。

### 1.4.4 Cassandra（Column-oriented）

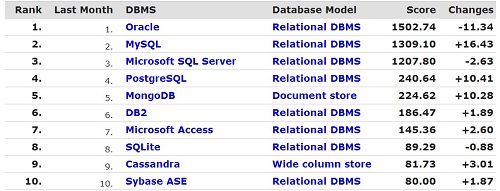
Cassandra是一个混合型的非关系的数据库，类似于Google的BigTable。其主要功能比Dynamo （分布式的Key-Value[存储系统](http://baike.baidu.com/view/51839.htm" \t "http://www.mamicode.com/_blank)）更丰富，但支持度却不如文档存储MongoDB（介于关系数据库和非关系数据库之间的开源产品，是非关系数据库当中功能最丰富，最像关系数据库的。支持的[数据结构](http://baike.baidu.com/view/9900.htm" \t "http://www.mamicode.com/_blank)非常松散，是类似json的bjson格式，因此可以存储比较复杂的数据类型）。Cassandra最初由Facebook开发，后转变成了开源项目。它是一个网络社交云计算方面理想的数据库。以Amazon专有的完全分布式的Dynamo为基础，结合了Google BigTable基于列族（Column Family）的数据模型。P2P去中心化的存储。很多方面都可以称之为Dynamo 2.0。[

### 1.4.5 其他不常用非关系型数据库

    HBase、MemacheDB、BerkeleyDB、Tokyo Cabinet

## ****1.5**** ****数据库相关知识****

### 1.5.1数据库软件企业应用排名参考



1. 所谓的封装

将不变的代码写死封装起来。将变化的东西通过传入参数来封装起来。

# **[C#中partial关键字](http://www.cnblogs.com/peasana/archive/2012/03/21/2409528.html)**

**1.** **什么是局部类型？**C# 2.0 引入了局部类型的概念。局部类型允许我们将一个类、结构或接口分成几个部分，分别实现在几个不同的.cs文件中。  
  
局部类型适用于以下情况：  
  
(1) 类型特别大，不宜放在一个文件中实现。  
(2) 一个类型中的一部分代码为自动化工具生成的代码，不宜与我们自己编写的代码混合在一起。  
(3) 需要多人合作编写一个类。  
  
局部类型是一个纯语言层的编译处理，不影响任何执行机制——事实上C#编译器在编译的时候仍会将各个部分的局部类型合并成一个完整的类。

   public partial class Program  
   {  
     static void Main(string[] args)  
     {  
     }  
   }  
  
   partial class Program  
   {   
     public void Test()  
     {   
     }  
   }

视图：

对select语句的封装；

是一个表吗?

是以个结果集吗？

有什么作用？

复杂代码的简化，复杂sql语句的提炼。代码的重用。

主要用来做什么；

复杂select语句的提炼。可以做删除，修改，增加吗？

存储过程：

C#里有很多代码是重复使用的，我们给他封装。封装后的东西叫方法。

比较完整的逻辑操作，

一段逻辑操作封装叫存储过程。比如删除，修改，增加。

也可以叫封装一段select语句，没有问题。

服务器对象。

--------------创建存储过程-----------------

1 CREATE PROCEDURE Get\_Data

2 (

3 @Dealer\_ID VARCHAR(**50**)

4 )

5 AS

6 SELECT \* FROM myData WHERE Dealer\_ID = @Dealer\_ID

****例3：一个高效的数据分页的存储过程 可以轻松应付百万数据****

CREATE PROCEDURE pageTest --用于翻页的测试  
--需要把排序字段放在第一列

(  
@FirstID nvarchar(20)=null, --当前页面里的第一条记录的排序字段的值  
@LastID nvarchar(20)=null, --当前页面里的最后一条记录的排序字段的值  
@isNext bit=null, --true 1 :下一页；false 0:上一页  
@allCount int output, --返回总记录数  
@pageSize int output, --返回一页的记录数  
@CurPage int --页号（第几页）0：第一页；-1最后一页。  
)

AS

if @CurPage=0--表示第一页  
begin  
--统计总记录数  
select @allCount=count(ProductId) from Product\_test   
  
set @pageSize=10  
--返回第一页的数据  
select top 10   
ProductId,  
ProductName,  
Introduction   
from Product\_test order by ProductId   
end

else if @CurPage=-1--表示最后一页

select \* from   
(select top 10 ProductId,  
ProductName,  
Introduction

from Product\_test order by ProductId desc ) as aa   
order by ProductId  
else

begin   
if @isNext=1  
--翻到下一页  
select top 10 ProductId,  
ProductName,  
Introduction  
from Product\_test where ProductId > @LastID order by ProductId   
else  
--翻到上一页  
select \* from  
(select top 10 ProductId,  
ProductName,  
Introduction  
from Product\_test where ProductId < @FirstID order by ProductId desc) as bb order by ProductId  
end