

目錄

入門	2
連線設定	4
MySQL 一般設定	7
Oracle 一般設定	8
Oracle Basic 連線一般設定	9
Oracle TNS 連線一般設定	10
PostgreSQL 一般設定	11
SQLite 一般設定	12
SQL Server 一般設定	13
SSH 設定 (只限於 MySQL、Oracle、PostgreSQL 及 SQL Server 並只支援 SSH2 通訊協定)	14
SSH 通道的好處	15
密碼驗證	16
公開金鑰驗證	17
HTTP 設定 (只限於 MySQL、PostgreSQL 及 SQLite)	19
SSL 設定 (只限於 MySQL 及 PostgreSQL)	20
安裝 OpenSSL 及 MySQL 或 PostgreSQL	21
設定 MySQL 或 PostgreSQL 的 SSL 憑證	22
設定 Navicat 的用戶端憑證	25
進階設定	27
設定進階資料庫內容 (只限於 MySQL 及 PostgreSQL)	31
與資料庫或結構描述工作	32
MySQL 資料庫管理	33
Oracle 結構描述管理	35
PostgreSQL 資料庫管理	36
PostgreSQL 資料庫設計器	38
編輯 PostgreSQL 資料庫一般設定	39
SQLite 資料庫管理	40
SQL Server 資料庫管理	42
SQL Server 資料庫設計器	43
編輯 SQL Server 資料庫一般設定	44
編輯 SQL Server 資料庫檔案群組	46
編輯 SQL Server 資料庫檔案	47
編輯進階 SQL Server 資料庫內容	48
與資料庫或結構描述物件工作	53
檢視物件資訊	54

入門

要在 Navicat 開始 MySQL、Oracle、PostgreSQL、SQLite 或 SQL Server 資料庫的工作，你要首先用連線視窗建立一個或多個連線。如果你是一個伺服器新手和不肯定如何工作，你可能想看看：

- [MySQL 使用者手冊](#)
- [Oracle 資料庫文件](#)
- [PostgreSQL 使用者手冊](#)
- [SQLite 使用者手冊](#)
- [SQL Server MSDN Library](#)

按一下  或選擇檔案 ->  **新增連線** 來設置連線內容。

- [連線設定](#)

在建立連線後，你可以連線到資料庫，管理它的物件、資料表中的資料等。請看下面的說明，以了解如何用最簡單的方法執行這些操作。

- [與資料庫或結構描述工作](#)
- [與資料庫或結構描述物件工作](#)

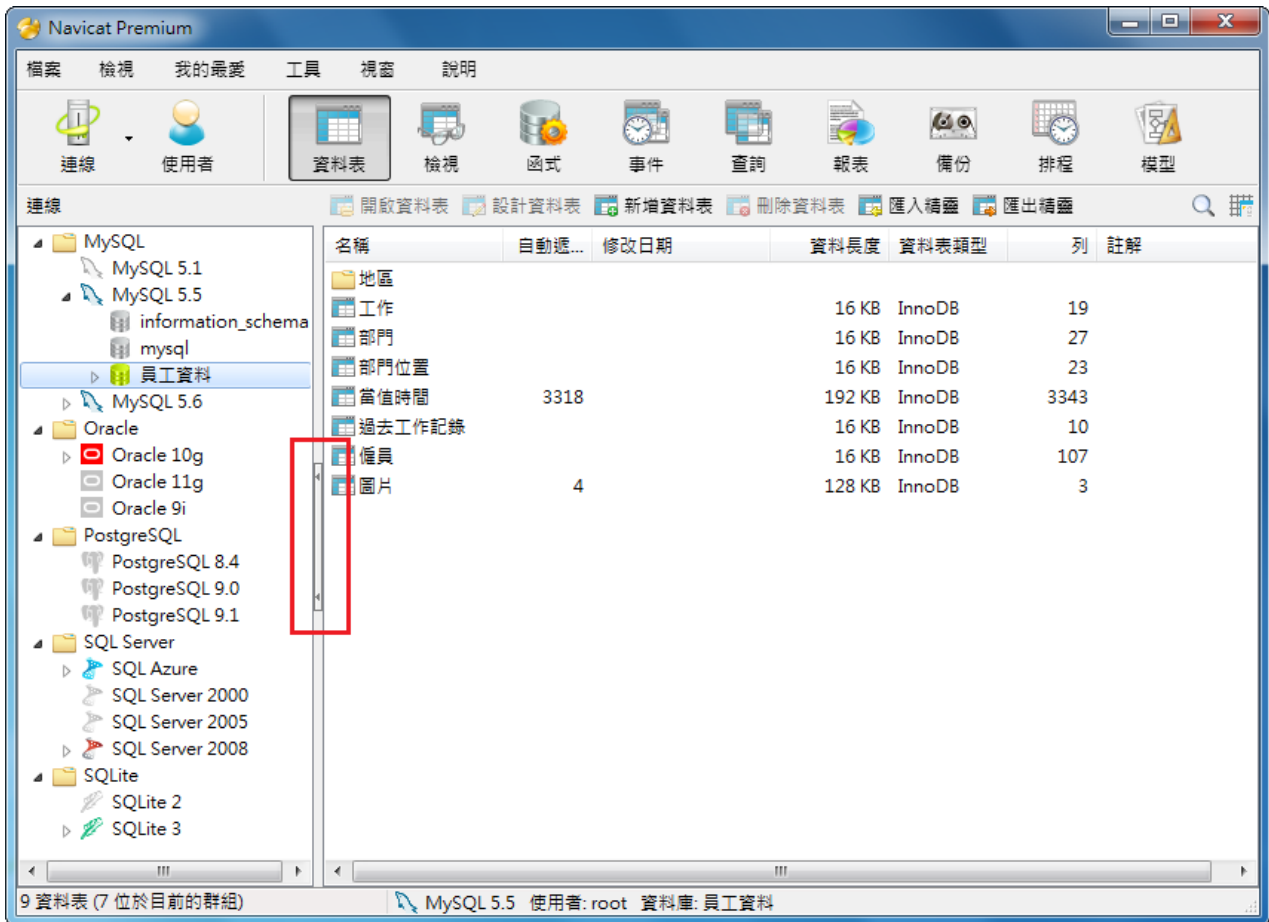
Navicat 瀏覽器！

Navicat 視窗包括一個導覽窗格（左邊的窗格）及一個物件窗格（右邊的窗格）。

導覽窗格一個是導覽連線、資料庫及資料庫物件的基本途徑。它採用樹狀結構，讓你透過彈現式功能表快捷及方便地使用資料庫和它們的物件。

物件窗格顯示開啟資料表、查詢等。在視窗頂部的工具列提供其他控制項，您可以用它來操作你的資料。



要檢視或隱藏此視窗，按一下下面的**紅色的標示**或在主功能表選擇檢視 -> **連線樹**。



連線設定

Navicat 集合實用工具來管理你的資料庫。要開始在 Navicat 管理你的資料庫，首先你需要做的是建立你的伺服器連線。

建立連線

Navicat 提供了三種典型的方法來建立你的連線，按一下  或選擇檔案 ->  **新增連線** 來開始設置。

- [MySQL 一般設定](#)
- [Oracle 一般設定](#)
- [PostgreSQL 一般設定](#)
- [SQLite 一般設定](#)
- [SQL Server 一般設定](#)
- [SSH 設定](#)（只限於 MySQL、Oracle、PostgreSQL 及 SQL Server）
- [HTTP 設定](#)（只限於 MySQL、PostgreSQL 及 SQLite）

注意：對於 MySQL 或 PostgreSQL 伺服器，一個常用的通訊協定 - **Secure Sockets Layer (SSL)** 是用於管理在網際網路上傳輸信息的安全（詳細資料請看 [SSL 設定](#)）。

Navicat 提供測試用的評估帳號。

遠端 MySQL 伺服器連線設定為：


- 主機名稱或 IP 位址：server1.navicat.com
- 埠：4406
- 使用者名稱：navicat
- 密碼：testnavicat

遠端 PostgreSQL 伺服器連線設定為：

- 主機名稱或 IP 位址：server1.navicat.com
- 埠：5432
- 初始資料庫：HR
- 使用者名稱：navicat
- 密碼：testnavicat


注意： Navicat 能讓你連線到不同平台上的遠端伺服器，即 Windows、Mac、Linux 及 UNIX。

建立一個與現有連線相同內容的新連線

- 在導覽窗格中的連線上按一下滑鼠右鍵並選擇  **複製連線...**。
- 新建立的連線將被命名為「連線名稱_copy」。

刪除連線

刪除一個連線

- 在導覽窗格中的連線上按一下滑鼠右鍵並選擇  **刪除連線**。
- 在對話視窗確認刪除。

開啟連線

開啟一個連線

- 在導覽窗格中按兩下以開啟連線。


關閉連線

關閉一個連線

- 在導覽窗格中的連線上按一下滑鼠右鍵並選擇  **關閉連線**。

編輯連線

編輯一個連線的資訊

- 如果連線開啟著，關閉它。
- 在連線上按一下滑鼠右鍵並選擇  **連線內容...**。

開啟連線設定儲存路徑

開啟一個連線設定儲存路徑

- 在導覽窗格中選擇連線。
- 在連線上按一下滑鼠右鍵並選擇 **前往設定儲存路徑** 或按 **Ctrl+G** 來開啟設定儲存路徑資料夾。

匯出連線設定

匯出連線設定

- 選擇檔案 -> **匯出連線...**。
- 選擇連線和匯出檔案路徑。

匯入連線設定

匯入連線設定

- 選擇檔案 -> **匯入連線...**。
- 指定連線設定檔案路徑。
- 如果連線已存在，在對話視窗確認取代或跳過。

取得連線資訊

取得一個連線的資訊

- 在導覽窗格中開啟連線。
- 在已開啟的連線上按一下滑鼠右鍵並選擇 **連線資訊...**。

MySQL 一般設定

下列的指示引導你通過建立一個新連線的程序。要成功地建立一個新的連線到本機或遠端 MySQL - 不管通過 SSL、SSH 或 HTTP，都要在相應的框中設定連線內容：連線名稱、主機名稱、埠號碼、使用者名稱及密碼。

預設情況下，MySQL 提供使用者名為「root」和密碼為空白。

連線名稱

一個最恰當地描述你的連線名稱。

主機名稱或 IP 位址

資料庫所在的主機名或伺服器的 IP 位址。

埠

連線到資料庫伺服器的 TCP/IP 埠。

使用者名稱

連線到資料庫伺服器的使用者名稱。

密碼

連線到伺服器的密碼。

你可以連線到你的 MySQL 遠端伺服器，但基於安全原因，本機遠端直接連線到 MySQL 伺服器已停用。因此，你可能不能使用 Navicat Premium 或其他類似的 MySQL 管理應用程式直接連線到遠端伺服器，除非已配置[使用者權限](#)。

如果你的互聯網服務供應商（ISP）不提供直接存取其伺服器，安全通道協定（SSH）或 HTTP 是另一種解決辦法。

Oracle 一般設定

下列的指示引導你通過建立一個新連線的程序。要成功地建立一個新的連線到本機或遠端 Oracle - 不管通過 SSH，都要在相應的框中設定連線內容：連線名稱、主機名稱、埠號碼、使用者名稱及密碼。

預設情況下，Oracle 在安裝時已建立一些使用者帳號。管理帳號：SYS、SYSTEM、SYSMAN 及 DBSNMP。樣本結構描述帳號：SCOTT、HR、OE、OC、PM、IX 及 SH。

Navicat 支援兩種 Oracle 伺服器連線：

- [Basic 連線](#)
- [TNS 連線](#)

如果你的互聯網服務供應商 (ISP) 不提供直接存取其伺服器，安全通道協定 (SSH) 是另一種解決辦法。

Oracle Basic 連線一般設定

連線名稱

一個最恰當地描述你的連線名稱。

連線類型

連線到伺服器的連線類型：**Basic** 或 TNS。

Basic

在 Basic 模式中，Navicat 通過 Oracle Call Interface (OCI) 連線到 Oracle。OCI 是一個應用程式介面，讓程式開發人員使用第三代語言原有程序或函式呼叫來存取 Oracle 資料庫及控制全部 SQL 語句執行的階段。OCI 是一個標準資料庫存取的函式庫和動態連結庫形式檢索函式。

主機名稱或 IP 位址

資料所在的主機名或伺服器的 IP 位址。

埠

連線到資料庫伺服器的 TCP/IP 埠。

Service Name/SID

設定使用者連線時用的 Service Name/SID。選擇相應的選項按鈕。

使用者名稱

連線到資料庫伺服器的使用者名稱。

密碼

連線到伺服器的密碼。

Oracle TNS 連線一般設定

連線名稱

一個最恰當地描述你的連線名稱。

連線類型

連線到伺服器的連線類型：Basic 或 **TNS**。

TNS

在 TNS 模式中，Navicat 使用在 tnsnames.ora 檔案中的別名項目通過 Oracle Call Interface (OCI) 連線到 Oracle 伺服器。OCI 是一個應用程式介面，讓程式開發人員使用第三代語言原有程序或函式呼叫來存取 Oracle 資料庫及控制全部 SQL 語句執行的階段。OCI 是一個標準資料庫存取的函式庫和動態連結庫形式檢索函式。

網路服務名稱

網路服務的名稱。

使用者名稱

連線到資料庫伺服器的使用者名稱。

密碼

連線到伺服器的密碼。

PostgreSQL 一般設定

下列的指示引導你通過建立一個新連線的程序。要成功地建立一個新的連線到本機或遠端 PostgreSQL - 不管通過 SSL、SSH 或 HTTP，都要在相應的框中設定連線內容：連線名稱、主機名稱、埠號碼、初始資料庫、使用者名稱及密碼。

預設情況下，PostgreSQL 提供使用者名稱為「postgres」和密碼為空白。

連線名稱

一個最恰當地描述你的連線名稱。

主機名稱或 IP 位址

資料庫所在的主機名或伺服器的 IP 位址。

埠

連線到資料庫伺服器的 TCP/IP 埠。

初始資料庫

連線時，使用者連線到的初始資料庫。

使用者名稱

連線到資料庫伺服器的使用者名稱。

密碼

連線到伺服器的密碼。

如果你的互聯網服務供應商（ISP）不提供直接存取其伺服器，安全通道協定（SSH）或 HTTP 是另一種解決辦法。

注意：基於安全原因，本機遠端直接連線到 PostgreSQL 伺服器已停用。因此，你可能不能使用 Navicat Premium 或其他類似的 PostgreSQL 管理應用程式直接連線到遠端伺服器。如需詳細資料，請參考下一段的伺服器管理。

伺服器管理：

預設情況下，PostgreSQL 只允許由使用 TCP/IP 連線的本機電腦連線。其他的機器將無法連接，除非你修改在 *postgresql.conf* 檔案中的 *listen_addresses*，通過修改 *\$PGDATA/pg_hba.conf* 檔案啟用基於主機的認證及重開伺服器。欲了解更多信息：[用戶端身份驗證](#)

SQLite 一般設定

下列的指示引導你通過建立一個新連線的程序。要成功地建立一個新的連線到本機或遠端 SQLite - 不管通過 HTTP，都要在相應的框中設定連線內容：連線名稱、類型及資料庫名稱。

連線名稱

一個最恰當地描述你的連線名稱。

類型

指定資料庫的類型。

現有的資料庫檔案

連線到 **資料庫檔案** 的現有資料庫。

新增 SQLite 3

在 **資料庫檔案** 建立新的 SQLite 3 資料庫。

新增 SQLite 2

在 **資料庫檔案** 建立新的 SQLite 2 資料庫。

資料庫檔案

指定初始資料庫檔案。如果 HTTP 通道 開啟，你需要輸入資料庫在你網絡伺服器的絕對檔案路徑。

SQL Server 一般設定

下列的指示引導你通過建立一個新連線的程序。要成功地建立一個新的連線到本機或遠端 SQL Server - 不管通過 SSH，都要在相應的框中設定連線內容：連線名稱、主機名稱及驗證類型。

連線名稱

一個最恰當地描述你的連線名稱。

主機名稱或 IP 位址

資料庫所在的主機名或伺服器的 IP 位址。

驗證

SQL Server 使用兩種方式來驗證連線到 SQL Server 資料庫：SQL Server Authentication 及 Windows Authentication。

SQL Server Authentication

SQL Server Authentication 使用登入記錄來驗證連線。每次使用者連線，都必須提供他們的使用者名稱及密碼。

使用者名稱

連線到資料庫伺服器的使用者名稱。

密碼

連線到伺服器的密碼。

Windows Authentication

當一個使用者通過一個視窗使用者帳號連線，SQL Server 會使用作業系統中 Windows 主要代符來驗證帳號名稱及密碼。這意味著使用者身份是由 Windows 確認。SQL Server 不會要求輸入密碼，並且不執行身份驗證。

如果你的互聯網服務供應商 (ISP) 不提供直接存取其伺服器，安全通道協定 (SSH) 是另一種解決辦法。

SSH 設定（只限於 **MySQL、Oracle、PostgreSQL** 及 **SQL Server** 並只支援 **SSH2** 通訊協定）

Secure SHell (SSH) 是一個通過網絡登入其他電腦的程式，在遠端伺服器執行命令，和從一台機器移動檔案到另一台。在不安全的網絡中，它提供兩台主機之間強大認證和安全加密的的通訊，被稱為 **SSH Port Forwarding (Tunneling)**。通常情況下，它是使用為一個 Telnet 的加密版本。

在一個 Telnet 階段作業，所有的通訊，包括使用者名稱和密碼，會用純文字傳輸，讓任何人都能監聽你的階段作業及竊取密碼或其他資訊。這種階段作業也容易受到階段作業劫持，一旦你驗證，惡意使用者就能接管這種階段作業。**SSH** 的目的是防止這種漏洞，並允許你在不影響安全性的情況下存取遠端伺服器的 shell。

- [SSH 通道的好處](#)。

為確保進來的連線請求是由你發出，**SSH** 能夠使用密碼，或公開及私密金鑰對（也稱為公開金鑰）驗證機制。

- [密碼驗證](#)。
- [公開金鑰驗證](#)。

注意：請確保在 Linux 伺服器的參數 - 「AllowTcpForwarding」設定值為「yes」，否則，會停用 **SSH port forwarding**。要尋找路徑：[/etc/ssh/sshd_config](#)。在預設情況下，**SSH port forwarding** 應該已啟用。請仔細檢查該值的設定。

****** 即使伺服器支援 **SSH** 通道，然而，如果 **port forwarding** 被停用，**Navicat** 就無法通過 **SSH** 埠 22 連線。

SSH 通道的好處

SSH 有一個極好功能叫做 **SSH Port Forwarding**，有時也被稱為 **SSH 通道**，它允許你建立一個安全的階段作業，然後通過它打開隨心所欲的 **TCP 連線**。通道可以隨時建立，幾乎不需要任何努力及編製程式，這令到它們非常有吸引力。在無數不同的方式，**SSH Port Forwarding** 可以用於安全通訊。

許多提供伺服器託管的託管公司會封鎖託管公司網絡以外存取伺服器，以及只授予存取給本機 (**localhost**) 使用者連線。



使用 **SSH** 的多個好處：

- 當伺服器埠被封鎖時，連線到使用了防火牆的伺服器。
- 自動驗證使用者，沒有發送純文字的密碼，以防止竊取密碼。
- 多個強大的認證方法，防止安全威脅如欺騙的身份。
- 安全和快速的加密和壓縮資料。
- 安全檔案傳輸。

密碼驗證

使用這種模式下，SSH 是幾乎和 telnet 程式相同。當你進行連接時，它會詢問你的密碼。輸入它後，你會登入或被拒絕。你的密碼會首先加密並由網絡傳送，然後在遠端主機解密。這是鼓勵大多數使用者使用的模式，因為它不需要額外的設置或配置。

下列指示將引導你完成設定使用密碼驗證的 SSH 連線。要成功地建立一個 SSH 連線，在相應的框中設定 SSH 連線內容：主機名稱或 IP 位址、埠號碼、使用者名稱、驗證方法及密碼。

1. 按一下  或選擇檔案 ->  新增連線 來設定連線內容。
2. 選擇 SSH 索引標籤並啟用 使用 SSH 通道。
3. 填寫所需的資訊：

主機名稱或 IP 位址

SSH 伺服器的主機。

埠

SSH 伺服器的埠，預設情況下它是 22。

使用者名稱

在 Linux 機器的使用者。（這是一個 Linux 使用者。它不是一個資料庫伺服器使用者。）

驗證方法

在 密碼驗證 和公開金鑰驗證間選擇。

密碼

這是一個 Linux 使用者的密碼。



一般 進階 SSL SSH HTTP

使用 SSH 通道

主機名稱或 IP 位址: SSH_伺服器_IP_位址

埠: 22

使用者名稱: SSH_登入名稱

驗證方法: 密碼

密碼: ●●●●●●●●

儲存密碼

4. 在一般設定頁面的 Navicat 主機名是由你的資料庫託管公司提供，應設定和 SSH 伺服器相對。



公開金鑰驗證

公開金鑰驗證是基於電子簽名的使用並提供最佳的安全驗證。

要使用公開金鑰驗證，有四項需要：

- 在你連線到的遠端伺服器必須有你的公開金鑰。
- 在你的本機用戶端必須有你的私密金鑰。
- 遠端伺服器必須配置為允許你使用你的公開金鑰登入。
- 本機用戶端必須配置為當你登入遠端伺服器時使用你的私密金鑰。

下列指示將引導你完成配置使用公開金鑰驗證的 **SSH** 連線。要成功地建立一個 **SSH** 連線，在相應的框中設定 **SSH** 連線內容：主機名稱或 **IP** 位址、埠號碼、使用者名稱、驗證方法、私密金鑰及密碼短語。

1. 按一下  或選擇檔案 ->  **新增連線** 來設定連線內容。
2. 選擇 **SSH** 索引標籤並啟用 **使用 SSH 通道**。
3. 填寫所需的資訊：

主機名稱或 **IP** 位址

SSH 伺服器的主機。

埠

SSH 伺服器的埠，預設情況下它是 22。

使用者名稱

在 Linux 機器的使用者。（這是一個 Linux 使用者。它不是一個資料庫伺服器使用者。）

驗證方法

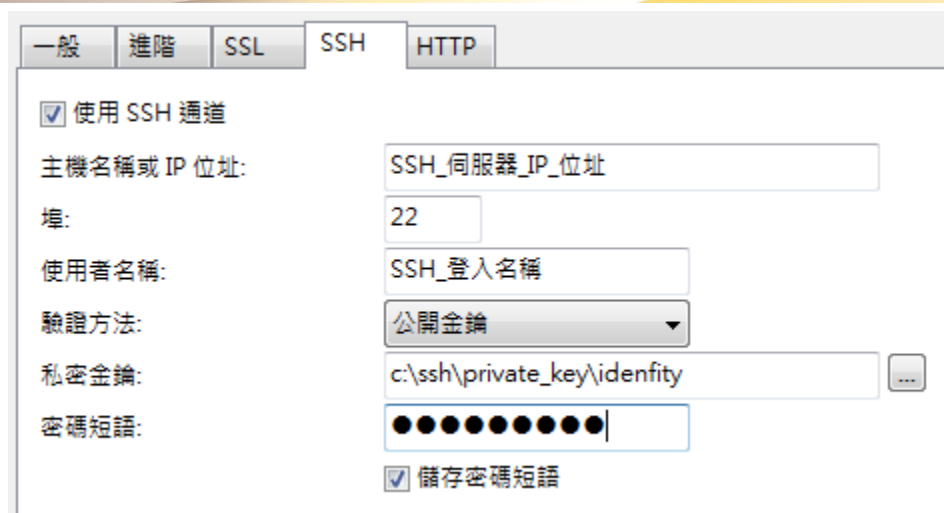
在密碼驗證和 **公開金鑰** 驗證間選擇。

私密金鑰

這是和你的公開金鑰一起使用。私密金鑰應該只有你可讀。

密碼短語

密碼短語和密碼完全一樣，除了它應用於你產生的金鑰而不是帳號。密碼短語的長度是 1024 字元以下。



4. 在一般設定頁面的 Navicat 主機名是由你的資料庫託管公司提供，應設定和 SSH 伺服器相對。

HTTP 設定（只限於 MySQL、PostgreSQL 及 SQLite）

HTTP 通道是一種連線到伺服器的方法，使用與網絡伺服器相同的通訊協定（<http://>）和相同的埠（埠 80）。這是當你的互聯網服務供應商不允許直接連線，但允許建立 HTTP 連線時使用。

建立 HTTP 連線的步驟：

1. 上傳通道指令碼

若要使用此連線方法，首先需要做的就是上傳通道指令碼到你的伺服器位於的網絡伺服器。

注意： `ntunnel_mysql.php`、`ntunnel_pgsql.php` 或 `ntunnel_sqlite.php` 是在 Navicat 的安裝資料夾中。

2. 設定 HTTP 通道

下列指示將引導你完成設定 HTTP 連線程序。

- i. 按一下  或選擇檔案 ->  **新增連線** 來設定連線內容。
- ii. 選擇 HTTP 索引標籤並啟用 **使用 HTTP 通道**。
- iii. 輸入通道指令碼的網址，例如：`http://www.navicat.com/ntunnel_mysql.php`。
- iv. 如果你的伺服器安裝了 ModSecurity，你可以勾選 **用 base64 編碼傳出查詢** 選項。
- v. 如果通道指令碼是在一個密碼保護的伺服器或你要通過代理伺服器來存取網際網路，你可以在 **驗證** 或 **代理伺服器** 索引標籤提供所需的詳細驗證資料。
- vi. 在一般設定頁面的 Navicat 主機名是由你的資料庫託管公司提供，應設定和 HTTP 伺服器相對。

注意： HTTP 通道和 SSH 通道不能同時運作。當你選擇了 HTTP 通道，SSH 通道停用，反之亦然。

SSL 設定（只限於 MySQL 及 PostgreSQL）

Secure Sockets Layer(SSL) 是一個通過網際網路傳輸私人文件的協定。為了安全連線，首先你需要做的是安裝 OpenSSL 函式庫和下載資料庫來源。

建立 SSL 連線及 Navicat 日 步驟：

1. [安裝 OpenSSL 及 MySQL 或 PostgreSQL](#)。
2. [設定 MySQL 或 PostgreSQL 的 SSL 憑證](#)。
3. [設定 Navicat 的用戶端憑證](#)。

注意： PostgreSQL 8.4 或以上版本支援。

安裝 OpenSSL 及 MySQL 或 PostgreSQL

安裝 OpenSSL

1. 下載 OpenSSL - <http://www.openssl.org>
2. Linux 命令：`[zcat 0.96l.tar.gz | tar xvf -]`
3. Linux 命令：`[./config]`
4. Linux 命令：`[make]`
5. Linux 命令：`[make install]`

安裝 MySQL

1. 下載 MySQL - <http://www.mysql.com>
2. Linux 命令：`[./configure --with -vio --with -openssl]`
3. Linux 命令：`[make]`
4. Linux 命令：`[make install]`

注意：請確保 MySQL 伺服器支援 OpenSSL 使用查詢語句：`[SHOW VARIABLES LIKE 'have_openssl'];` - 回傳值 = YES

安裝 PostgreSQL

1. 下載 PostgreSQL - <http://www.postgresql.org>
2. Linux 命令：`[./configure --with-openssl]`
3. Linux 命令：`[gmake]`
4. Linux 命令：`[gmake install]`

注意：請確保 PostgreSQL 伺服器支援 OpenSSL 使用查詢語句：`[SHOW ssl;]` - 回傳值 = ON

設定 MySQL 或 PostgreSQL 的 SSL 憑證

要建立伺服器或用戶端憑證，用 root 登入 Linux 伺服器並使用 Shell 命令如下：

MySQL

1. `DIR=`pwd`/openssl`
2. `PRIV=$DIR/private`
3. `mkdir $DIR $PRIV $DIR/newcerts`
4. `cp /usr/share/ssl/openssl.cnf $DIR`
5. `replace ./demoCA $DIR -- $DIR/openssl.cnf`
6. 產生 Certificate Authority (CA)

```
/usr/local/ssl/bin/openssl req -new -x509 -keyout $PRIV/cakey.pem -out $DIR/cacert.pem -config $DIR/openssl.cnf
```

注意：如果「PEM」是需要的，請通過以下步驟輸入另外的「PEM pass」。

7. 建立伺服器的請求和鑰匙

```
/usr/local/ssl/bin/openssl req -new -keyout $DIR/server-key.pem -out $DIR/server-req.pem -days 3600 -config $DIR/openssl.cnf
```

8. 從鑰匙移除密碼短語（非必須的）

```
/usr/local/ssl/bin/openssl rsa -in $DIR/server-key.pem -out $DIR/server-key.pem
```

9. 簽署伺服器憑證

```
/usr/local/ssl/bin/openssl ca -policy policy_anything -out $DIR/server-cert.pem -config $DIR/openssl.cnf -infiles $DIR/server-req.pem
```

10. 建立用戶端的請求和鑰匙

```
/usr/local/ssl/bin/openssl req -new -keyout $DIR/client-key.pem -out $DIR/client-req.pem -days 3600 -config $DIR/openssl.cnf
```

11. 從鑰匙移除密碼短語（非必須的）

```
/usr/local/ssl/bin/openssl rsa -in $DIR/client-key.pem -out $DIR/client-key.pem
```

12. 簽署用戶端憑證

```
/usr/local/ssl/bin/openssl ca -policy policy_anything -out $DIR/client-cert.pem  
-config $DIR/openssl.cnf -infile $DIR/client-req.pem
```

13. 建立一個 **my.cnf** 檔案來測試憑證。儲存於 **/etc** 或 MySQL 資料目錄（通常在來源安裝的 **/usr/local/var**）

my.cnf 範例內容：

```
[client]  
ssl-ca=$DIR/cacert.pem  
ssl-cert=$DIR/client-cert.pem  
ssl-key=$DIR/client-key.pem  
[mysqld]  
ssl-ca=$DIR/cacert.pem  
ssl-cert=$DIR/server-cert.pem  
ssl-key=$DIR/server-key.pem
```

14. 要開始 MySQL 程式類型：

```
/usr/local/libexec/mysqld -u mysql &
```

or

```
/usr/local/sbin/mysqld -u &
```

PostgreSQL

1. 要建立一個伺服器的快速自簽署憑證，使用以下的 `OpenSSL` 命令：

```
openssl req -new -text -out server.reqm
```

2. 填寫 `openssl` 要求的資訊。請確認你輸入的本機主機名稱為「`Common Name`」；要求的密碼可以留空。程式會產生一個密碼保護的鑰匙；它不會接受一個少於 `4` 個字元長度的密碼短語。要移除密碼短語（你必須，如果你想要自動啟動的伺服器），執行命令：

```
openssl rsa -in privkey.pem -out server.key  
rm privkey.pem
```

3. 輸入舊的密碼短語來解開現有的鑰匙。現在做的：

```
openssl req -x509 -in server.req -text -key server.key -out server.crt
```



4. 將憑證變為自簽署憑證和複製鑰匙及憑證到伺服器將在那裏尋找他們。最後做：

```
chmod og-rwx server.key
```

設定 Navicat 的用戶端憑證

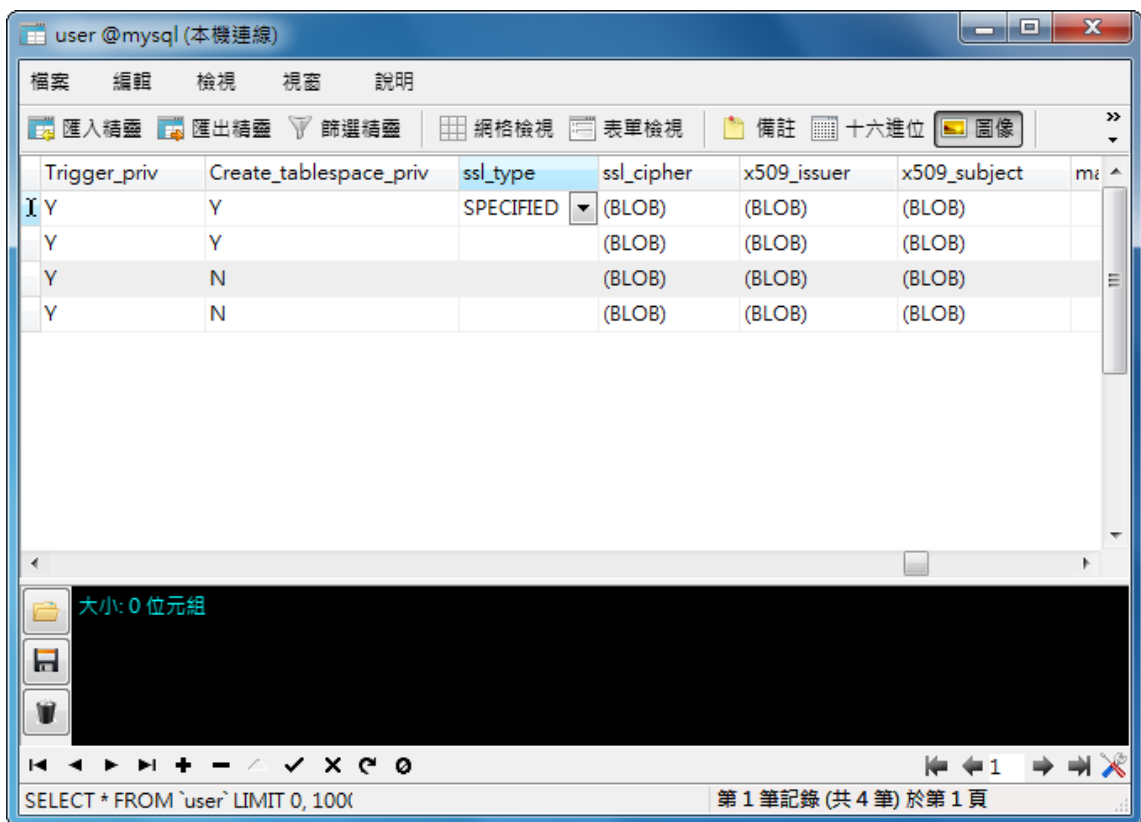
下列指示將引導你完成配置 Navicat 和 MySQL 或 PostgreSQL 伺服器之間使用 SSL 的連線程序。要成功地建立一個 SSL 連線，請填寫步驟 1 及步驟 2，並在相應的框中設定連線內容。

MySQL



1. 按一下  或選擇檔案 ->  新增連線 來設定連線內容。
2. 選擇 SSL 索引標籤並啟用 使用 SSL。
3. 提供驗證資料，填寫所需的資訊：

用戶端鑰匙、用戶端憑證 及 CA 憑證 通常儲存在你的伺服器 - `/usr/local/openssl`。請從遠端伺服器複製他們到本機電腦。指定密碼檢索表（非必須的）只需要在 `ssl_type` 欄位設定為「**SPECIFIED**」- [`ssl_type` 可以在名為「mysql」的系統資料庫中找到 -> 「user」資料表]。指定密碼檢索表的範例是「EDH-RSA-DES-CBC3-SHA」，可以填寫在上面顯示的連線內容或下面的「mysql」資料庫 -> 「user」資料表 -> 「ssl_cipher」 blob 欄位。

注意：你可以儲存你的指定密碼檢索表在一個文字檔案，以便載入到到「ssl_cipher」blob 欄位。



PostgreSQL

1. 按一下  或選擇檔案 ->  **新增連線** 來設定連線內容。
2. 選擇 **SSL** 索引標籤並啟用 **使用 SSL**。
3. 選擇 **SSL 模式**。

require - 只嘗試一個 SSL 連線。

verify-ca - 只嘗試一個 SSL 連線，並驗證伺服器憑證是由受信任的 CA 發行。

verify-full - 只嘗試一個 SSL 連線，驗證伺服器憑證是由受信任的 CA 發行及伺服器主機名稱符合憑證內的。

4. 要提供驗證資料，啟用 **使用驗證** 並填寫所需的資訊：

用戶端鑰匙、**用戶端憑證** 及 **CA 憑證** 通常儲存在你的伺服器 - `/usr/local/openssl`。請從遠端伺服器複製他們到本機電腦。

憑證撤銷清單 指定 SSL 憑證撤銷清單 (certificate revocation list - CRL) 的檔案路徑。

對於 PostgreSQL 伺服器，OpenSSL 支援範圍廣泛不同強度的密文和驗證演算法。當密文清單可以在 OpenSSL 設定檔案中指定，你可以透過修改 `postgresql.conf` 中的 `ssl_ciphers` 指定資料庫要使用的密文。

進階設定

根據你的需要自訂連線選項。詳細介紹如下：

設定儲存路徑

當建立一個新的連線，Navicat 將建立一個子資料夾在設定儲存路徑內。大多數檔案都儲存在該子資料夾：

Navicat 物件	伺服器類型	副檔名
查詢	全部	.sql
匯出查詢結果設定檔	MySQL	.npeq
	Oracle	.nopeq
	PostgreSQL	.nppeq
	SQLite	.nlpeq
	SQL Server	.nmpeq
匯出檢視結果設定檔	MySQL	.npev
	Oracle	.nopev
	PostgreSQL	.nppev
	SQLite	.nlpev
	SQL Server	.nmpev
備份	MySQL、PostgreSQL 及 SQLite	壓縮 (.psc)、未經壓縮 (.psb)
備份設定檔	MySQL	.npb
	PostgreSQL	.nppb
	SQLite	.nlpb
報表	全部	.rtm
匯入精靈設定檔	MySQL	.npi
	Oracle	.nopi
	PostgreSQL	.npqi
	SQLite	.nlpi
	SQL Server	.nmpi
匯出精靈設定檔	MySQL	.npe
	Oracle	.nope
	PostgreSQL	.nppe
	SQLite	.nlpe
	SQL Server	.nmpe

匯出具體化檢視設定檔	Oracle	.nopem
ER 圖表檔案	全部	.ned

其他檔案均位於 **profiles** 目錄。要尋找路徑，選擇工具 -> 選項 -> 其他 -> 設定檔儲存路徑。

其他設定檔	伺服器類型	副檔名
資料傳輸	MySQL	.npt
	Oracle	.nopt
	PostgreSQL	.nppt
	SQLite	.nlpt
	SQL Server	.nmpt
	Premium (跨伺服器)	.napt
資料同步	MySQL	.npd
	Oracle	.nopd
	PostgreSQL	.nppd
	SQLite	.nlpd
	SQL Server	.nmpd
結構同步	MySQL	.nps
	Oracle	.nops
	PostgreSQL	.npps
	SQL Server	.nmps
批次作業	Premium (跨伺服器)	.napj
模型檔案	全部	.ndm
虛擬群組	全部	vgroup.xml - 儲存如何分類物件。

提示： 你的所有連線設定都儲存在登錄程式。

參見：
記錄檔案

MySQL

編碼

當使用 MySQL 字元集沒有使用，選擇一個代碼頁來和 MySQL 伺服器溝通。

保持連線間隔 (秒)

此選項允許你通過 ping 來保持連線。你可以在編輯欄位設定 ping 與 ping 之間的時間隔。

使用 MySQL 字元集

如果使用 MySQL 4.1 或以上的版本，應該啟用此選項。

使用壓縮

此選項讓你使用壓縮通訊協定。使用它當用戶端和伺服器支援 **zlib** 壓縮，以及用戶端需要壓縮。

自動連接

勾選了這個選項，當程式啟動時，Navicat 會自動開啟連線和顯示登記的資料庫。

使用名稱管道、通訊端

勾選了這個選項，Navicat 使用本機連線的通訊端檔案。

Oracle

角色

指示資料庫使用者用 **Default**、**SYSOPER** 或 **SYSDBA** 系統權限連線。

作業系統驗證

勾選了這個選項，Oracle 資料庫使用 Windows 使用者登入證書來驗證資料庫使用者。

自動連接

勾選了這個選項，當程式啟動時，Navicat 會自動開啟連線和顯示登記的資料庫。

PostgreSQL

保持連線間隔 (秒)

此選項允許你通過 ping 來保持連線。你可以在編輯欄位設定 ping 與 ping 之間的時間隔。

自動連接

勾選了這個選項，當程式啟動時，Navicat 會自動開啟連線和顯示登記的資料庫。

SQLite

自動連接

勾選了這個選項，當程式啟動時，Navicat 會自動開啟連線和顯示登記的資料庫。

已加密

當連線到已加密的 SQLite 資料庫，啟用這個選項並提供 **密碼**。

已附加的資料庫

在連線中附加或卸除資料庫。

SQL Server

初始資料庫

連線時，使用者連線到的初始資料庫。

保持連線間隔 (秒)

此選項允許你通過 ping 來保持連線。你可以在編輯欄位設定 ping 與 ping 之間的時間隔。

使用加密

這選項讓你使用加密。

自動連接

勾選了這個選項，當程式啟動時，Navicat 會自動開啟連線和顯示登記的資料庫。

設定進階資料庫內容（只限於 MySQL 及 PostgreSQL）

設定進階資料庫內容，這不是強制性的。要開始使用進階資料庫設定，勾選 **使用進階連線**。詳細介紹如下：

顯示已選擇的資料庫

在導覽窗格中顯示已選擇的資料庫處於 **關閉** 狀態

- 在資料庫清單中按一下合意的資料庫，核取方塊將顯示為

在導覽窗格中顯示已選擇的資料庫處於 **開啟** 狀態

- 在資料庫清單中按兩下合意的資料庫，核取方塊將顯示為

加入隱藏的資料庫

加入一個隱藏的資料庫

- 按一下 **加入資料庫到清單** 按鈕。
- 輸入資料庫名稱。
- 在資料庫清單中選擇新加入的資料庫。
- 輸入你的互聯網服務供應商提供的 **使用者名稱** 和 **密碼**。



移除資料庫


移除一個資料庫

- 在資料庫清單中選擇要移除的資料庫。
- 按一下 **從清單中移除資料庫** 按鈕。

注意： 資料庫只從資料庫清單中移除，它仍然存在於伺服器中。

與資料庫或結構描述工作

在你建立連線後，你的資料庫或結構描述會出現在左邊的導覽窗格。如果在選項視窗勾選 **在連線樹中顯示物件** 選項，所有資料庫或結構描述物件也會顯示在窗格。要連線到一個資料庫或結構描述，只需在窗格中按兩下它。如果連線成功，該資料庫或結構描述的節點會變成  或 ，並展開顯示其物件的樹。



在 **Oracle** 伺服器，當你建立一個使用者時，你也同時建立那個使用者的結構描述。結構描述是一個使用者建立資料庫物件（例如資料表、檢視、觸發器、等）的邏輯容器。結構描述名稱是和使用名稱一樣，並可以明確指由使用者擁有的物件。其他使用者結構描述顯示在  **結構描述** 下面。

MySQL 資料庫管理

Navicat 提供全部管理及導覽資料庫所需的工具。請注意，要在 Navicat 中開始資料庫的工作，你需要建立連線。

建立資料庫

建立一個資料庫

- 在導覽窗格中按兩下以開啟連線。
- 在已開啟的連線上按一下滑鼠右鍵並選擇  新增資料庫...。
- 或在現有的資料庫上按一下滑鼠右鍵並選擇  新增資料庫...。
- 填寫所需的資訊：

輸入資料庫名稱

設定新的資料庫名稱。

字元集


字元集指定預設的資料庫字元集。

排序規則

排序規則指定預設的資料庫排序規則。

刪除資料庫


刪除一個資料庫

- 在導覽窗格中的資料庫上按一下滑鼠右鍵並選擇  刪除資料庫。
- 在對話視窗確認刪除。


注意：此操作是不可逆轉的。

開啟資料庫

開啟一個隱藏資料庫


- 在導覽窗格中按兩下以開啟連線。
- 在已開啟的連線上按一下滑鼠右鍵並選擇  開啟資料庫。
- 輸入資料庫名稱。

開啟一個顯示在導覽窗格的資料庫

- 在導覽窗格中按兩下以開啟資料庫。
- 或
- 在資料庫上按一下滑鼠右鍵並選擇  開啟資料庫。

關閉資料庫

關閉一個資料庫

- 在導覽窗格中的資料庫上按一下滑鼠右鍵並選擇  關閉資料庫。

編輯資料庫

編輯一個資料庫

- 在導覽窗格中的資料庫上按一下滑鼠右鍵並選擇  資料庫內容...。

請注意，這個時候 MySQL 不支援通過它的介面重新命名資料庫。

存取儲存資料庫的目錄。預設情況下，全部資料庫儲存於 MySQL 安裝資料夾內一個名為 **data** 的目錄。
例如：**C:\mysql5\data**。


注意：你必須停止 MySQL，然後才能重新命名資料庫。

Oracle 結構描述管理

在 Navicat 中開始結構描述的工作，你需要建立連線。

開啟結構描述

開啟一個顯示在導覽窗格的結構描述

- 在導覽窗格中按兩下以開啟結構描述。
- 或
- 在結構描述上按一下滑鼠右鍵並選擇  開啟結構描述。

關閉結構描述

關閉一個結構描述



- 在導覽窗格中的結構描述上按一下滑鼠右鍵並選擇  關閉結構描述。

PostgreSQL 資料庫管理

在 Navicat 中開始資料庫的工作，你需要建立連線。


建立資料庫

建立一個新的資料庫

- 在導覽窗格中按兩下以開啟連線。
- 在已開啟的連線中按一下滑鼠右鍵並選擇  新增資料庫...。
- 或在現有的資料庫中按一下滑鼠右鍵並選擇  新增資料庫...。
- 在資料庫設計器中適當的索引標籤編輯資料庫內容。


編輯資料庫

編輯一個現有的資料庫（管理它的一般等）

- 在導覽窗格中的資料庫中按一下滑鼠右鍵並選擇  資料庫內容...。
- 在資料庫設計器中適當的索引標籤編輯資料庫內容。


刪除資料庫

刪除一個資料庫


- 在導覽窗格中的資料庫中按一下滑鼠右鍵並選擇  刪除資料庫。
- 在對話視窗中確認刪除。

開啟資料庫

開啟一個隱藏資料庫

- 在導覽窗格中按兩下以開啟連線。
- 在已開啟的連線中按一下滑鼠右鍵並選擇  開啟資料庫。
- 輸入資料庫名稱。

開啟一個顯示在導覽窗格的資料庫

- 在導覽窗格中按兩下以開啟資料庫。
- 或在資料庫中按一下滑鼠右鍵並選擇  開啟資料庫。

關閉資料庫

關閉一個資料庫

- 在導覽窗格中的資料庫上按一下滑鼠右鍵並選擇  關閉資料庫。

PostgreSQL 資料庫設計器

資料庫設計器 是與資料庫工作的 Navicat 基本工具。它允許你建立新的資料庫和編輯現有的資料庫內容。

- [編輯資料庫一般設定](#)
- 編輯資料庫註解

編輯 PostgreSQL 資料庫一般設定

要建立一個資料庫，你一定要有 **可以建立資料庫**（`usecreatedb`）權限。如何設定使用者內容，請參照角色編輯器或使用者編輯器。

資料庫名稱

定義資料庫的名稱。

編碼

定義資料庫的編碼。如果省略，預設為範本資料庫的編碼。

擁有者

定義資料庫的擁有者。如果省略，預設為執行命令的使用者。只有超級使用者可以建立其他使用者擁有的資料庫。

範本

由一個範本資料庫建立資料庫。

注意：在複製操作的期間，來源資料庫必須是空閒的（沒有資料修改交易進行中）。`CREATE DATABASE` 將檢查在開始操作時沒有階段作業（除本身）連線到來源資料庫。但這並不保證當複製進行時不會有變更，導致複製的資料庫不一致。因此，建議視用作範本的資料庫為唯讀。

資料表空間



定義資料庫的資料表空間。如果省略，預設為 `pg_default`。

SQLite 資料庫管理

Navicat 提供全部管理及導覽資料庫所需的工具。請注意，要在 Navicat 中開始資料庫的工作，你需要建立連線。

附加資料庫

附加一個資料庫

- 在導覽窗格中按兩下以開啟連線。
- 在已開啟的連線上按一下滑鼠右鍵並選擇  附加資料庫。
或
- 在現有的資料庫上按一下滑鼠右鍵並選擇  附加資料庫。
- 填寫所需的資訊：

資料庫檔案

設定資料庫的檔案路徑。

資料庫名稱


輸入 Navicat 顯示的資料庫名稱。

已加密

啟用此選項，並提供連線到已加密的 SQLite 資料庫的 密碼。


開啟資料庫

開啟一個顯示在導覽窗格的資料庫

- 在導覽窗格中按兩下以開啟資料庫。
或
- 在資料庫上按一下滑鼠右鍵並選擇  開啟資料庫。

卸除資料庫

卸除一個資料庫

- 在導覽窗格中的資料庫上按一下滑鼠右鍵並選擇  卸除資料庫。

加密資料庫

加密一個資料庫

- 在導覽窗格中的資料庫上按一下滑鼠右鍵並選擇 **加密資料庫**。
- 輸入密碼。

解密資料庫

解密一個資料庫

- 在導覽窗格中的資料庫上按一下滑鼠右鍵並選擇 **解密資料庫**。
- 在對話視窗確認解密。

關閉資料庫

關閉一個資料庫



- 在導覽窗格中的資料庫上按一下滑鼠右鍵並選擇  **關閉資料庫**。

SQL Server 資料庫管理

要在 Navicat 中開始資料庫的工作，你需要建立連線。


建立資料庫

建立一個新的資料庫

- 在導覽窗格中按兩下以開啟連線。
- 在已開啟的連線中按一下滑鼠右鍵並選擇  新增資料庫...。
- 或在現有的資料庫中按一下滑鼠右鍵並選擇  新增資料庫...。
- 在資料庫設計器中適當的索引標籤編輯資料庫內容。


編輯資料庫

編輯一個現有的資料庫（管理它的一般等）

- 在導覽窗格中的資料庫中按一下滑鼠右鍵並選擇  資料庫內容...。
- 在資料庫設計器中適當的索引標籤編輯資料庫內容。


刪除資料庫

刪除一個資料庫

- 在導覽窗格中的資料庫中按一下滑鼠右鍵並選擇  刪除資料庫。
- 在對話視窗中確認刪除。


開啟資料庫

開啟一個資料庫

- 在導覽窗格中按兩下以開啟連線。
- 或在已開啟的連線中按一下滑鼠右鍵並選擇  開啟資料庫。

關閉資料庫

關閉一個資料庫

- 在導覽窗格中的資料庫中按一下滑鼠右鍵並選擇  關閉資料庫。

SQL Server 資料庫設計器

資料庫設計器 是與資料庫工作的 Navicat 基本工具。它允許你建立新的資料庫和編輯現有的資料庫內容。

- [編輯資料庫一般設定](#)
- [編輯資料庫檔案群組](#)
- [編輯資料庫檔案](#)
- [編輯進階資料庫內容](#)
- 編輯資料庫註解（SQL Azure 不支援）
- 資料庫 SQL 預覽

編輯 SQL Server 資料庫一般設定

SQL Server 的選項

資料庫名稱

定義資料庫的名稱。

擁有者

選擇資料庫的擁有者。

排序規則

選擇資料庫的預設排序規則。排序規則名稱可以是 Windows 排序規則名稱或 SQL 排序規則名稱。若未指定，會將 SQL Server 執行個體的預設排序規則指派給資料庫。資料庫快照集中無法指定排序規則名稱。

復原模式

控制資料庫復原選項及磁碟 I/O 錯誤檢查。

FULL

在媒體失敗之後，利用交易記錄備份來提供完整復原。如果資料檔損毀，媒體復原可以還原所有已認可的交易。

BULK_LOGGED

提供媒體失敗之後的復原功能，針對特定大規模或大量作業，結合最好的效能及最少的記錄使用空間。

SIMPLE

提供使用最少記錄空間的簡單備份策略。當伺服器失敗復原不再需要記錄空間，會自動重複使用這個記錄空間。

相容性層級

選擇資料庫相容的 SQL Server 版本。

SQL Azure 的選項

資料庫名稱

定義資料庫的名稱。

版本

選擇資料庫的版本：web 或 business。

最大大小

選擇資料庫的最大大小。

編輯 SQL Server 資料庫檔案群組

SQL Azure 不支援這個索引標籤。

檔案群組

加入或刪除一個檔案群組。PRIMARY 檔案群組不可以刪除。

檔案資料流檔案群組

加入或刪除一個檔案資料流檔案群組。

注意： SQL Server 2008 或以上版本支援。

編輯 SQL Server 資料庫檔案

SQL Azure 不支援這個索引標籤。

資料庫檔案

名稱

指定檔案的邏輯名稱。

類型

選擇檔案類型。

檔案群組

選擇檔案群組。

檔案目錄

當你建立檔案時作業系統所使用的路徑。

檔案名稱

當你建立檔案時作業系統所使用的檔案名稱。

大小

指定檔案的大小。

允許自動成長

勾選這個選項，如果你想允許自動成長。

成長

指定檔案的自動成長遞增。

最大大小

指定檔案所能成長的大小上限。

無限制

指定檔案可成長直到磁碟已滿。在 SQL Server 中，指定為無限成長的記錄檔，大小上限是 2 TB，資料檔的大小上限是 16 TB。

編輯進階 SQL Server 資料庫內容

SQL Azure 不支援這個索引標籤。

狀態

資料庫唯讀

如果勾選這個選項，使用者可以從資料庫中讀取資料，但不能修改它。

資料庫狀態

選擇資料庫的狀態。

OFFLINE

關閉資料庫，並將它完整關機，再標示為離線。資料庫在離線狀態時，無法修改。

ONLINE

資料庫在開啟狀態，可供使用。

EMERGENCY

資料庫標示為 `READ_ONLY`、停用記錄功能，而且只有系統管理員（`sysadmin`）固定伺服器角色的成員能夠存取它。**EMERGENCY** 主要用於進行疑難排解。

限制存取

控制使用者對資料庫的存取權。

SINGLE_USER

指定每次只能有一位使用者存取資料庫。

RESTRICTED_USER

`RESTRICTED_USER` 只允許 `db_owner` 固定資料庫角色以及資料庫建立者（`dbcreator`）和系統管理員（`sysadmin`）固定伺服器角色的成員連接到資料庫，但並不限制他們的數目。

MULTI_USER

允許所有具備適當權限來連接資料庫的使用者。

加密已啟用

勾選這個選項，如果你想加密資料庫。

注意： SQL Server 2008 或以上版本支援。

SQL

ANSI Null 預設值

勾選這個選項，如果你想決定預設值是 NULL。

ANSI Nulls 已啟用

如果勾選這個選項，所有對於 Null 值的比較，都會得出 UNKNOWN。

ANSI 填補已啟用

如果勾選這個選項，在轉換或插入到 varchar 或 nvarchar 資料類型之前，會將字串填補到相同的長度。

ANSI 警告已啟用

如果勾選這個選項，當發生除以零之類的狀況，或彙總函式中出現 NULL 值時，便會發出錯誤或警告。

算術中止已啟用

如果勾選這個選項，在查詢執行期間，當發生溢位或除以零的錯誤時，查詢會結束。

串連 Null 產生 Null

如果勾選這個選項，當任何一個運算元是 NULL 時，串連作業的結果都會是 NULL。

數值捨入中止

如果勾選這個選項，當運算式中遺失有效位數時，會產生錯誤。

引號識別元已啟用

如果勾選這個選項，可以利用雙引號來含括分隔的識別元。

遞迴觸發器已啟用

如果勾選這個選項，允許遞迴引發 AFTER 觸發器。

指標

認可時關閉資料指標已啟用

如果勾選這個選項，關閉認可或回復交易時在開啟狀態的任何資料指標。

預設資料指標

LOCAL

當指定 **LOCAL**，而且資料指標並未在建立時定義為 **GLOBAL** 時，資料指標的範圍僅限於資料指標建立時所在之批次、預存程序或觸發器的本機範圍。資料指標名稱只在這個範圍內有效。批次、預存程序或觸發器內的區域資料指標變數或是預存程序 **OUTPUT** 參數可以參考資料指標。當批次、預存程序或觸發器結束時，除非在 **OUTPUT** 參數中傳回資料指標，否則會隱含地取消配置資料指標。如果在 **OUTPUT** 參數中傳回資料指標，當最後一個參考資料指標的變數取消配置或離開範圍時，便會取消配置資料指標。

GLOBAL

當指定 **GLOBAL**，而且資料指標並未在建立時定義為 **LOCAL** 時，資料指標的範圍便是連接的全域範圍。連接所執行的任何預存程序或批次內都可以參考資料指標名稱。

自動

自動關閉

如果勾選這個選項，資料庫會完整關閉，而當最後一位使用者結束之後，便會將它的資源釋放出來。

自動建立統計資料

如果勾選這個選項，查詢最佳化工具會視需要針對查詢述詞中的單一資料欄建立統計資料，以便改善查詢計畫和查詢效能。

自動壓縮

如果勾選這個選項，資料庫檔案是定期壓縮的候選項。

自動更新統計資料

指定當查詢使用統計資料而且這些統計資料可能已過期時，查詢最佳化工具就會更新這些統計資料。

自動非同步更新統計資料

指定 **AUTO_UPDATE_STATISTICS** 選項的統計資料更新是非同步的。在查詢最佳化工具編譯查詢之前，它不會等候統計資料更新完成。

注意： SQL Server 2005 或以上版本支援。

復原

頁面確認

探索 I/O 路徑錯誤所造成的損毀資料庫頁面。磁碟 I/O 路徑錯誤可能是資料庫損毀問題的原因，這通常是因為電源故障或頁面寫入磁碟時磁碟硬體故障所造成。

NONE

寫入資料庫頁面不會產生 CHECKSUM 或 TORN_PAGE_DETECTION 值。在讀取期間，SQL Server 不會驗證總和檢查碼或損毀頁，即使頁首中有 CHECKSUM 或 TORN_PAGE_DETECTION 值也是如此。

TORN_PAGE_DETECTION

將每個 512 位元組磁區的 2 位元模式儲存在 8 KB 資料庫頁面上，當頁面寫入磁碟時，便將它儲存在資料庫頁首。

CHECKSUM

計算整個頁面內容的總和檢查碼，當頁面寫入磁碟時，將值儲存在頁首。

Service Broker

注意：SQL Server 2005 或以上版本支援。

Broker 已啟用

指定針對指定的資料庫啟用 Service Broker。啟動訊息傳遞，而且 sys.databases 目錄檢視中的 is_broker_enabled 旗標會設為 true。資料庫會保留現有的 Service Broker 識別元。

Honor Broker 優先權

傳送作業會將指派給交談的優先權等級列入考量。具有高優先權等級之交談的訊息會在被指派低優先權等級之交談的訊息之前傳送。

注意：SQL Server 2008 或以上版本支援。

變更追蹤

注意：SQL Server 2008 或以上版本支援。

變更追蹤已啟用

啟用資料庫的變更追蹤。當您啟用變更追蹤時，也可以設定 AUTO_CLEANUP 和 CHANGE RETENTION 選項。

保留週期

指定在資料庫中保存變更追蹤資訊的最小週期。只有當 AUTO_CLEANUP 值為 ON 時，才會移除資料。

自動刪除

在經過了指定的保留週期後，將會自動移除變更追蹤資訊。

其他

注意：SQL Server 2005 或以上版本支援。

已啟用跨資料庫擁有權鏈結

如果勾選這個選項，資料庫可以是跨資料庫擁有權鏈結的來源或目標。

可信任

如果勾選這個選項，使用模擬內容的資料庫模組（如使用者定義函數或預存程序）能夠在資料庫之外存取資源。

已啟用日期相互關聯的最佳化

SQL Server 會維護資料庫中 FOREIGN KEY 條件約束所連結且有 datetime 資料欄的任意兩份資料表之間的相互關聯統計資料。

參數化

SIMPLE

根據資料庫的預設行為，將查詢參數化。

FORCED

SQL Server 會將資料庫中的所有查詢參數化。

VarDecimal 儲存已啟用

表示 decimal 和 numeric 資料類型是使用 Vardecimal 儲存格式來儲存。

與資料庫或結構描述物件工作

你可以透過導覽窗格的節點存取你已選擇資料庫的資料庫或結構描述物件。根據你連線到的伺服器版本，支援的物件將會出現在物件樹。要開啟特定的物件，按一下滑鼠右鍵來開啟適當的編輯器。在物件上按一下滑鼠右鍵來顯示彈現式功能表，讓你在已選擇的物件或資料庫執行各種操作。

檢視物件資訊

要查看物件的資訊，只需簡單地在導覽窗格或物件窗格選擇一個物件並點選 **檢視 -> 物件資訊** 或在物件窗格中的物件上按一下滑鼠右鍵並選擇 **物件資訊** 來開啟 **物件資訊**。

- **一般**
以網格顯示物件的資訊。
- **DDL**
顯示物件的 DDL 語句。
- **使用**
顯示目前物件使用的物件。
注意：只限於 Oracle、PostgreSQL 及 SQL Server。
- **被使用**
顯示哪個物件使用目前的物件。
注意：只限於 Oracle、PostgreSQL 及 SQL Server。
- **物件**
顯示物件的資料表空間。
注意：只限於 Oracle 及 PostgreSQL。
- **預覽**
顯示查詢的 sql 語句。
- **成員屬於**
顯示使用者或角色分配到的角色。
注意：只限於 Oracle、PostgreSQL 及 SQL Server。
- **成員**
顯示角色的成員。
注意：只限於 Oracle、PostgreSQL 及 SQL Server。