

# 在 SSMS 中設定新使用者、Schema 擁有權與角色成員資格

本文將詳細說明在 SQL Server Management Studio (SSMS) 中如何為新使用者設定 Schema 擁有權 (User-Owned Schema) 和角色成員資格 (Membership)，並解釋兩者的差異。

## 一、建立新使用者

### 步驟

1. 在物件總管中，展開資料庫 → 安全性
2. 右鍵點擊「使用者」資料夾
3. 選擇「新增使用者」
4. 填寫使用者資訊：

### 新增使用者視窗的屬性說明

屬性欄位	說明
使用者類型	可選擇 SQL 使用者、使用 Windows 身分驗證的使用者等選項
使用者名稱	新使用者在資料庫中的名稱
登入名稱	與此資料庫使用者關聯的 SQL Server 登入帳戶
結構描述擁有者	指定此使用者是否擁有資料庫中的結構描述
角色成員	指定此使用者所屬的資料庫角色
安全物件	設定此使用者對特定物件的權限

## 二、設定 Schema 擁有權 (User-Owned Schema)

### 方法一：建立使用者時直接設定

1. 在新增使用者視窗中，點擊「結構描述擁有者」頁籤
2. 點擊「新增結構描述」按鈕
3. 輸入新 Schema 名稱，例如「MarketingData」
4. 點擊「確定」，此使用者將成為這個新 Schema 的擁有者

### 方法二：先建立 Schema 再指定擁有者

1. 在物件總管中，右鍵點擊「資料表結構描述」資料夾
2. 選擇「新增資料表結構描述」
3. 輸入 Schema 名稱，例如「SalesData」
4. 在「結構描述擁有者」下拉選單中，選擇先前建立的使用者

5. 點擊「確定」完成設定

## Schema 屬性說明

屬性	說明
名稱	Schema 的名稱，用於物件命名空間
擁有者	擁有此 Schema 的資料庫使用者或角色
預設結構描述	是否為特定使用者的預設 Schema
權限	其他使用者或角色對此 Schema 的權限設定

## 三、設定角色成員資格 (Membership)

### 方法一：建立使用者時指定角色

1. 在新增使用者視窗中，點擊「角色成員」頁籤
2. 勾選要加入的角色，例如「db\_datareader」、「db\_datawriter」等
3. 點擊「確定」完成設定

### 方法二：為現有使用者加入角色

1. 在物件總管中，展開資料庫 → 安全性 → 角色 → 資料庫角色
2. 右鍵點擊要加入的角色，例如「db\_datareader」
3. 選擇「屬性」
4. 在「成員」頁籤中，勾選要加入此角色的使用者
5. 點擊「確定」完成設定

## 角色屬性說明

屬性	說明
角色名稱	資料庫角色的名稱
擁有者	擁有此角色的資料庫主體
成員	屬於此角色的使用者或其他角色
安全物件	指定此角色對資料庫物件的權限

## 四、實例比較：Schema 擁有權 vs 角色成員資格

### 實例一：行銷分析師 (MarketingAnalyst)

Schema 擁有權設定：

```
使用者：MarketingAnalyst
擁有的 Schema：MarketingData
```

在此設定下，MarketingAnalyst 擁有 MarketingData Schema，因此對該 Schema 中的所有物件都有完整控制權，可以建立、修改、刪除物件，並管理 Schema 中物件的權限。

角色成員資格設定：

```
使用者：MarketingAnalyst
角色成員：Marketing_Readers, Marketing_Reports
```

在此設定下，MarketingAnalyst 透過成為 Marketing\_Readers 和 Marketing\_Reports 角色的成員，獲得這兩個角色所具有的權限。這些權限可能跨越多個 Schema，但通常僅限於特定操作（如讀取、執行報表）。

### 實例二：資料科學家 (DataScientist)

Schema 擁有權設定：

```
使用者：DataScientist
擁有的 Schema：AnalyticsModels
```

DataScientist 對 AnalyticsModels Schema 有完整控制權，可以建立自己的分析模型和演算法。

角色成員資格設定：

```
使用者：DataScientist
角色成員：Analytics_Users, db_datareader
```

透過 Analytics\_Users 角色，DataScientist 獲得對分析相關物件的存取權。而作為 db\_datareader 的成員，他可以讀取整個資料庫中的所有資料，但沒有寫入權限。

## 五、Schema 擁有權與角色成員資格的主要差異

特性	Schema 擁有權 (User-Owned Schema)	角色成員資格 (Membership)
範圍	限於特定 Schema 及其物件	可跨多個 Schema 和物件
權限類型	提供對 Schema 中物件的完整控制權	提供根據角色定義的特定權限
管理方式	每個 Schema 只能有一個擁有者	一個角色可以有多個成員
繼承性	使用者自動繼承對所擁有 Schema 物件的權限	使用者繼承角色的所有權限
主要用途	資料庫物件的組織和命名空間	使用者權限的集中管理

特性	Schema 擁有權 (User-Owned Schema)	角色成員資格 (Membership)
安全性層級	物件層級	可跨越多個物件和操作的功能層級

## 六、使用時機建議

### 使用 Schema 擁有權的情境

- 需要將不同業務領域的資料結構隔離
- 團隊或部門需要管理自己的資料模型
- 需要實施精細的資料和物件版本控制
- 多租戶應用程式中需要隔離不同客戶資料

### 使用角色成員資格的情境

- 需要為多個使用者指派相同權限集
- 按功能職責管理權限（如讀者、撰寫者、管理員）
- 實施權限階層或繼承
- 當權限需要經常變更時，集中管理更有效率

在實際應用中，通常會同時使用這兩種機制，以實現全面且精細的資料庫安全性控制。