

Tutorial – Google Drive

Vodalok

November 24, 2015

Contents

Lab time.....	1
About This Document	1
Tutorial – Google Drive	2
Additional Information.....	9

Lab time

時間估計 30 分鐘。

About This Document

Google Drive APIs 是利用 REST (Representational State Transfer) 風格來操作的 API，有關 REST 請查閱：<https://zh.wikipedia.org/wiki/REST>，其正好對應 HTTP 的四種操作，分別是: GET、POST、PUT 以及 DELETE，對應到 Google Drive 的操作即是下載、上傳、更新以及刪除。

在這個教學裡面，將會完成一個 Google Drive 的範例程式，透過完成這個程式將會學習到：如何透過 NuGet 套件管理安裝 Google Drive API for .Net 並且透過利用 GoogleDriveService 來使用 OAuth 2.0 認證以取得對使用者的 Google Drive 的使用權，以得以對其做 Google Drive 的四種操作，分別是上傳 (Insert)、下載 (Get)、更新 (Update) 以及刪除 (Delete)。

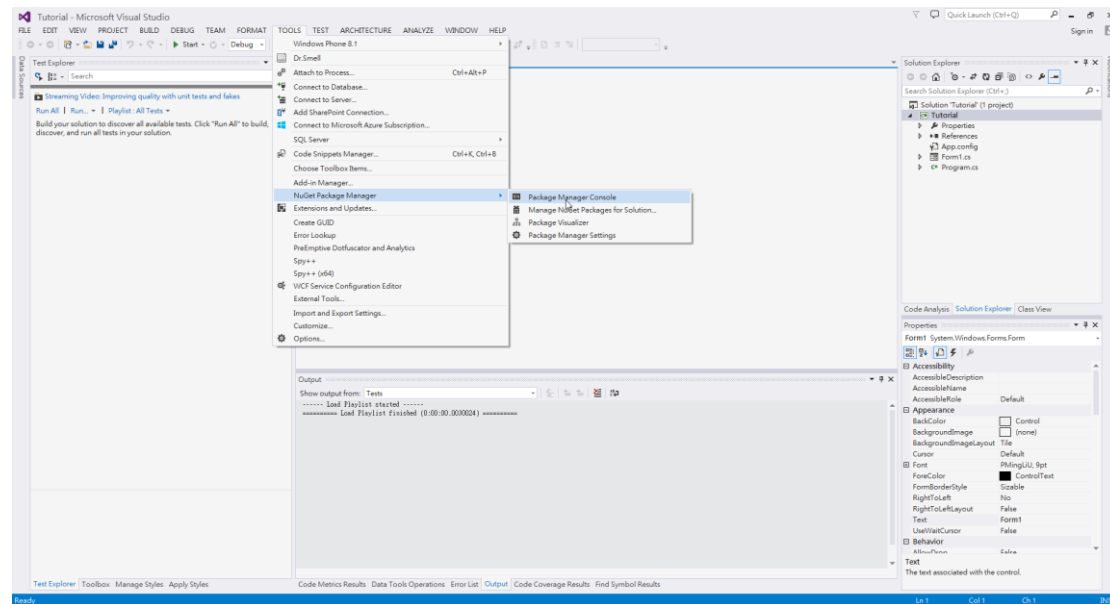
本教學中所用到的檔案，請自「Sample files」子目錄取得。

Tutorial – Google Drive

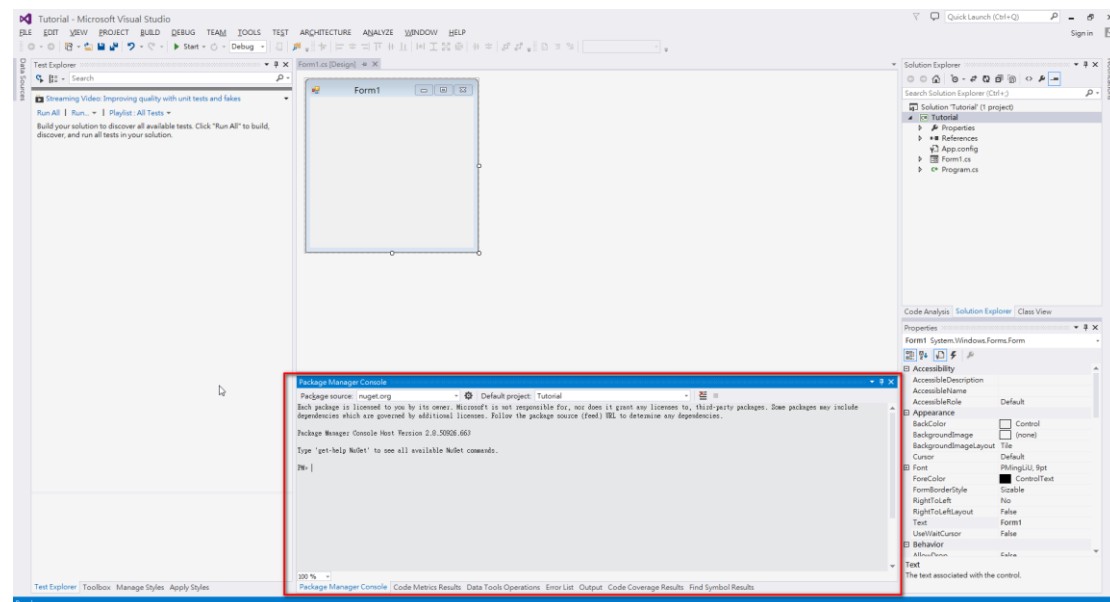
Step 1. 新增一個 Windows Form 應用程式專案

Step 2. 透過 NuGet 安裝 Google Drive API

Step 2.1 點選 TOOL -> NuGet Package Manager -> Package Manager Console (圖一)點選後會在下方開啟套件管理命令視窗(圖二)。



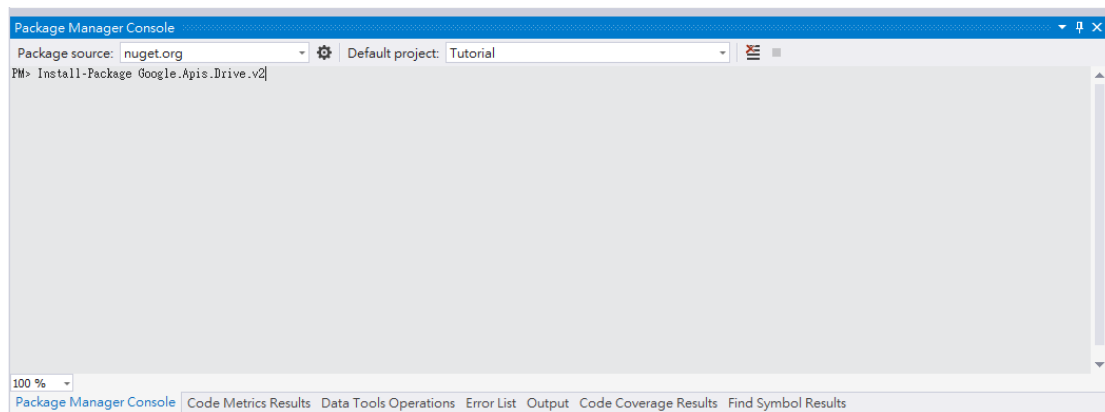
圖一



圖二

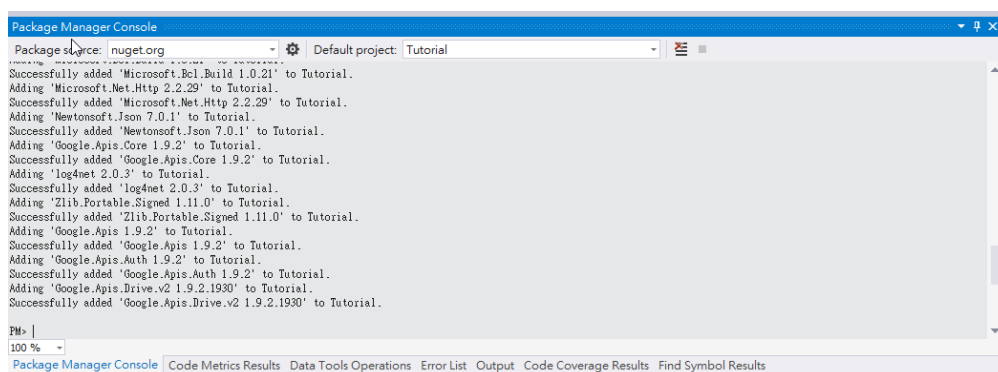
Step 2.2 在剛剛開啟的命令視窗，輸入如下的指令並按下 Enter 便會開始安裝 Google Drive API 參考組件及其相依的組件：

Install-Package Google.Apis.Drive.v2

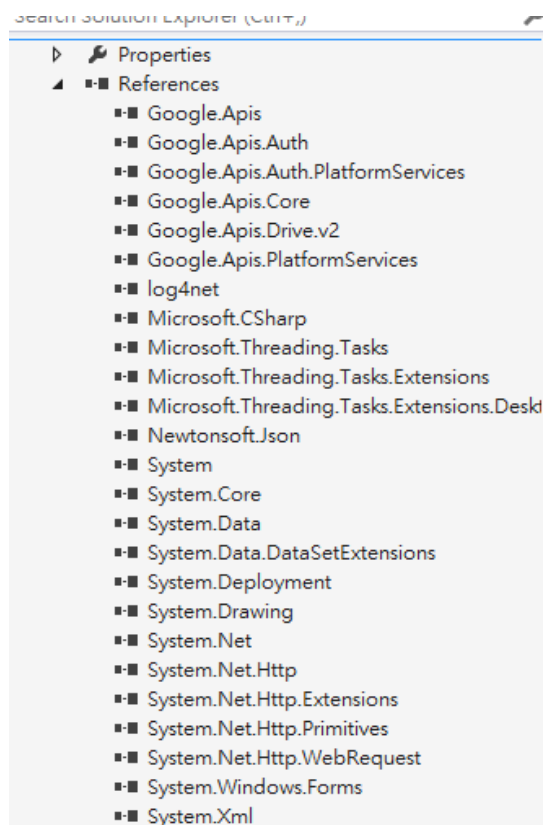


圖三

Step2.3 等待安裝完成...待 PM>顯示後即表示安裝完成 (假設沒有任何錯誤)，安裝後，會發現參考裡面多了 Google Drive 相關的參考



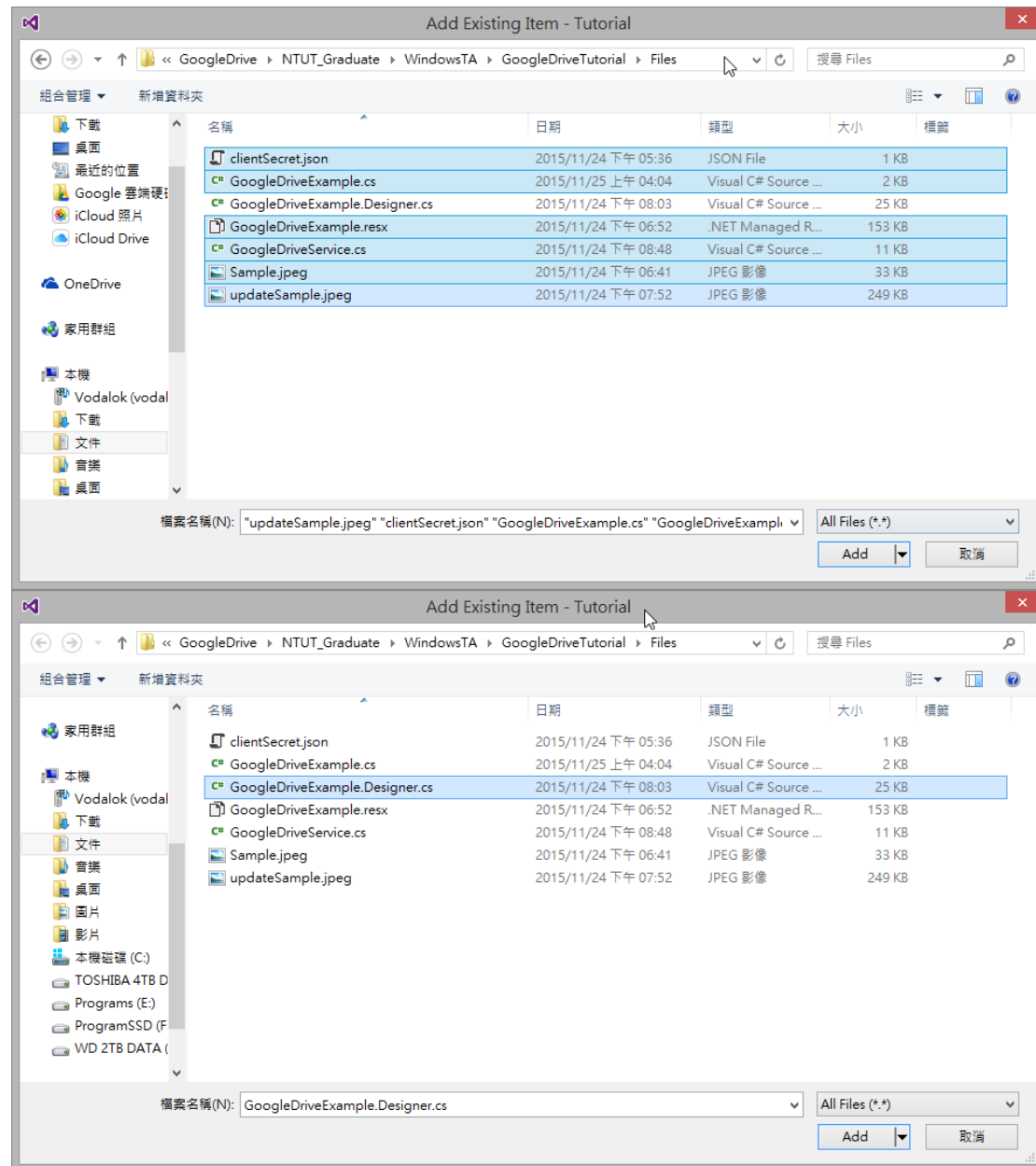
圖四



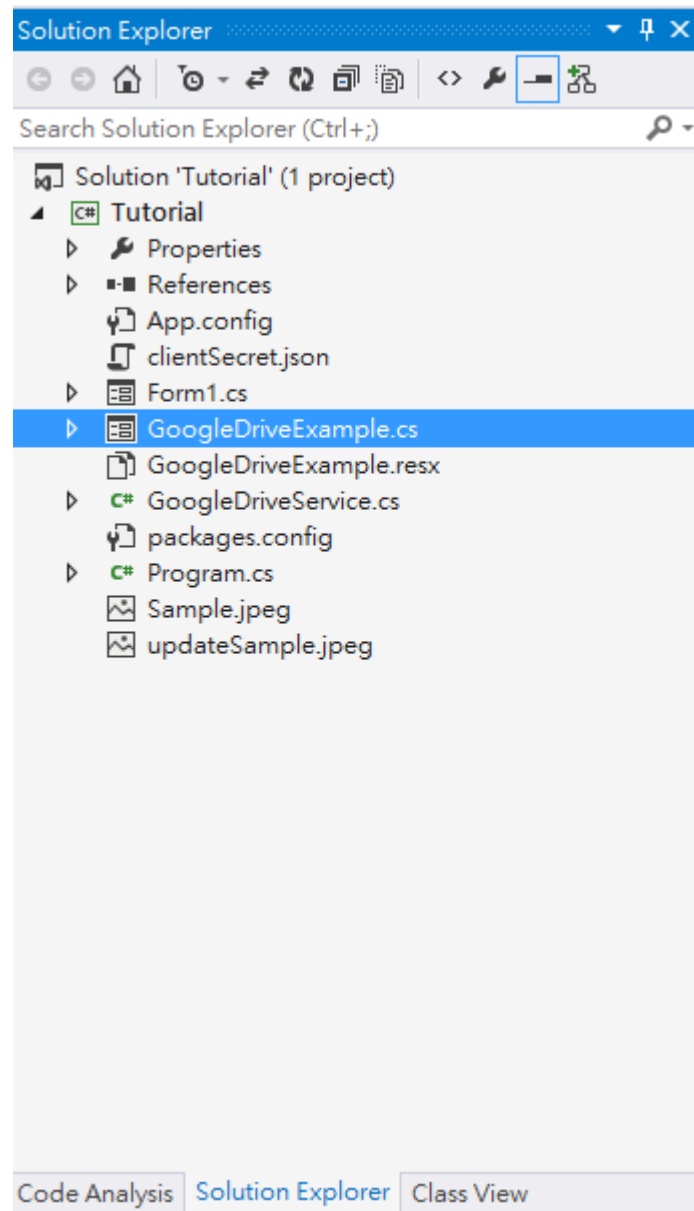
圖五

Step 2.4 至此，Google Drive API 的相關組件已經安裝完成。

Step 3. 將下載回來的檔案加至專案(下載連結在最上方)，請注意，請分兩次將 GoogleDriveExample.cs 與 GoogleDriveExample.Designer.cs 分別加入專案當中，否則 GoogleDriveExample.cs 會與 GoogleDriveExample.Designer.cs 分離，導致 Design 無法正常顯示(圖六)，加入後如圖七。

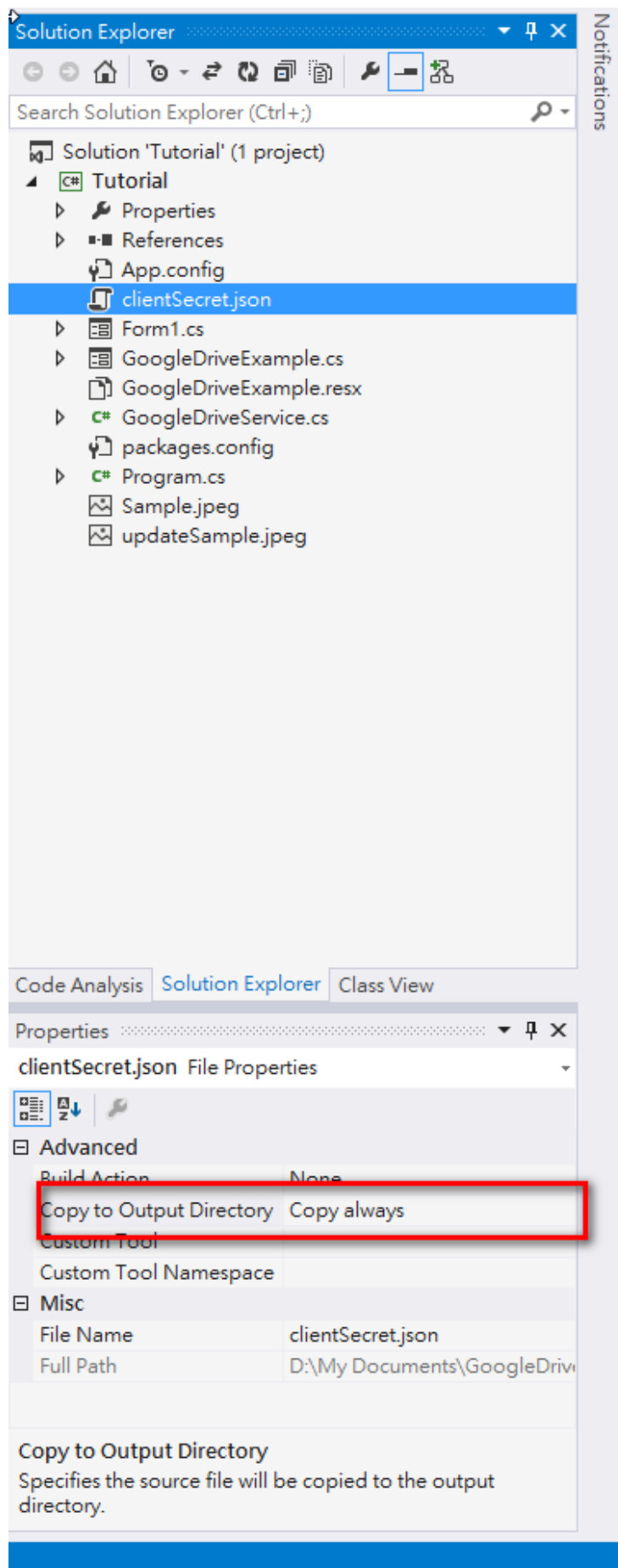


圖六



圖七

Step 4. 點選 clientSecret.json、Sample.jpeg 及 updateSample.jpeg，在屬性視窗，將 Copy to Output Directory 的選項改成 Copy Always (圖八)



圖八

Step 5. 加入程式碼

Step 5.1 在 Constructor 加入如下程式碼，對某些元件的屬性初始化，並且初始一 `GoogleDriveService` 並存至成員變數。

```
0reference
1public GoogleDriveExample()
2{
3    const string APPLICATION_NAME = "DrawAnywhere";
4    const string CLIENT_SECRET_FILE_NAME = "clientSecret.json";
5    InitializeComponent();
6
7    _uploadButton.Enabled = false;
8    _downloadButton.Enabled = false;
9    _downloadBrowseButton.Enabled = false;
10    _uploadFileOpenFileDialog.FileName = "";
11    _uploadFileOpenFileDialog.InitialDirectory = Directory.GetCurrentDirectory();
12    _downloadFolderBrowserDialog.SelectedPath = Directory.GetCurrentDirectory();
13    _service = new GoogleDriveService(APPLICATION_NAME, CLIENT_SECRET_FILE_NAME);
14}
15}
```

Step 5.2 在 `ClickBrowseUploadFileButton` 加入如下程式碼，當使用者點選 Upload 下的 Browse...按鈕，便會開啟檔案選擇對話視窗。

```
1reference
2private void ClickBrowseUploadFileButton(object sender, EventArgs e)
3{
4    DialogResult result = _uploadFileOpenFileDialog.ShowDialog();
5    if (result == System.Windows.Forms.DialogResult.OK)
6    {
7        _uploadButton.Enabled = true;
8        _uploadFilePathTextBox.Text = _uploadFileOpenFileDialog.FileName;
9        _uploadFileName = _uploadFileOpenFileDialog.FileName;
10    }
11}
12}
```

Step 5.3 在 `ClickUploadButton` 加入如下程式碼，當使用者選擇要上傳的檔案後，點選 Upload 鈕便會將檔案上傳至使用者的 Google Drive 根資料夾。

```
1private void ClickUploadButton(object sender, EventArgs e)
2{
3    const string CONTENT_TYPE = "image/jpeg";
4    _service.UploadFile(_uploadFileName, CONTENT_TYPE);
5}
6}
```

Step 5.4 在 `ClickListFileOnRootButton` 加入如下程式碼，當使用者點選 List Files On Root 按鈕時，會在 `ListBox` 顯示在 Google Drive 的根資料夾有的檔案的檔案名稱。

```
1private void ClickListFileOnRootButton(object sender, EventArgs e)
2{
3    const string DISPLAY_MEMBER = "Title";
4    const string VALUE_MEMBER = "Id";
5    const string FOLDER_MIME_TYPE = @"application/vnd.google-apps.folder";
6
7    List<Google.Apis.Drive.v2.Data.File> rootFoldersFiles = _service.ListRootFileAndFolder();
8    rootFoldersFiles.RemoveAll(removeItem =>
9    {
10        return removeItem.MimeType == FOLDER_MIME_TYPE;
11    });
12
13    _fileListBox.DisplayMember = DISPLAY_MEMBER;
14    _fileListBox.ValueMember = VALUE_MEMBER;
15    _fileListBox.DataSource = rootFoldersFiles;
16
17    _downloadBrowseButton.Enabled = true;
18}
19}
```

Step 5.5 在 ClickDownloadBrowseButton 加入如下程式碼，當使用者點選 Download 下的 Browse 按鈕，便會顯示 FolderBrowserDialog，讓使用者選擇要從 Google Drive 下載檔案到某一資料夾路徑。

```
private void ClickDownloadBrowseButton(object sender, EventArgs e)
{
    DialogResult result = _downloadFolderBrowserDialog.ShowDialog();
    if (result == System.Windows.Forms.DialogResult.OK)
    {
        _downloadPath = _downloadFolderBrowserDialog.SelectedPath;
        _downloadToTextBox.Text = _downloadPath;
        _downloadButton.Enabled = true;
    }
}
```

Step 5.6 在 ClickDownloadButton 加入如下程式碼，當使用者點選 Download 的按鈕時，會將在 ListBox 選取的檔案下載到使用者指定的資料夾。

```
private void ClickDownloadButton(object sender, EventArgs e)
{
    Google.Apis.Drive.v2.Data.File selectedFile = _fileListBox.SelectedItem as Google.Apis.Drive.v2.Data.File;
    _service.DownloadFile(selectedFile, _downloadPath);
}
```

Step 5.7 在 ClickUpdate 加入如下程式碼，當按下 Update 鈕時，程式會去尋找在跟資料夾名為 Sample.jpeg 的檔案，找到後會將 updateSample.jpeg 上傳至 Google Drive 並且更新在 Google Drive 上的 Sample.jpeg 的內容為 updateSample.jpeg 的內容(更新的是內容，也就是說在 Google Drive 上面的檔案名稱不會改變，還是一樣是 Sample.jpeg 只不過內容變為 updateSample.jpeg 的內容)，請注意，你可能需要修改 FILE_TITLE_ON_DRIVE 變數成有在你 Drive 上面的檔案名稱以讓程式能正確運作(假設 Drive 的根資料夾沒有 Sample.jpeg 就會發生錯誤)。你可以修改 UPDATE_FILE_NAME 成你想要的檔案名稱或路徑，讓程式更新所選擇的檔案。更新過程如下：首先會取得 Google Drive 的根資料夾的檔案列表(List)，再從這個列表尋找 Title 是 Sample.jpeg 的檔案，再利用這個檔案的 ID 去要求 Google Drive 更新對應 ID 的檔案。

```
private void ClickUpdateButton(object sender, EventArgs e)
{
    const string UPDATE_FILE_NAME = "updateSample.jpeg";
    const string CONTENT_TYPE = "image/jpeg";
    const string FILE_TITLE_ON_DRIVE = "Sample.jpeg";

    List<Google.Apis.Drive.v2.Data.File> fileList = _service.ListRootFileAndFolder();
    Google.Apis.Drive.v2.Data.File foundFile = fileList.Find(item => { return item.Title == FILE_TITLE_ON_DRIVE; });
    _service.UpdateFile(UPDATE_FILE_NAME, foundFile.Id, CONTENT_TYPE);
}
```

Step 5.8 在 ClickDeleteButton 加入如下的程式碼，當使用者按下 Delete 按鈕時，會刪除 Google Drive 根資料夾裡面名為 Sample.jpeg 的檔案。刪除的過程如下:首先會取得 Google Drive 的根資料夾的檔案列表(List)，再從這個列表尋找 Title 是 Sample.jpeg 的檔案，再利用這個檔案的 ID 去要求 Google Drive 刪除對應 ID 的檔案。


```

1 reference
private void ClickDeleteButton(object sender, EventArgs e)
{
    const string DELETE_FILE_ON_DRIVE_NAME = "Sample.jpeg";

    List<Google.Apis.Drive.v2.Data.File> fileList = _service.ListRootFileAndFolder();
    Google.Apis.Drive.v2.Data.File foundFile = fileList.Find(item => { return item.Title == DELETE_FILE_ON_DRIVE_NAME; });

    _service.DeleteFile(foundFile.Id);
}

```

Step 5.9 修改 Program.cs，讓程式一開始執行 GoogleDriveExample 的 Form。

```

static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new GoogleDriveUploader.View.GoogleDriveExample());
}

```

Step 6. 開始執行程式，一開始程式會開啟瀏覽器並且要求你登入 Google 帳戶並且允許程式取用你的 Google Drive，這只有在第一次執行程式時會要求，之後就不會再要求。允許之後，便可以回到程式，你可以試試看這個程式的各個功能。



Additional Information

1. 了解更多關於 Google Drive API 請參考 <https://developers.google.com/drive/web/about-sdk>
2. 如需了解 GoogleDriveService 請參閱 GoogleDriveService.cs，此檔案包含註解。

- The End -