



# PayPal Express Checkout 整合指南

版本 001

2015 年 02 月 18 日

# 目錄

1. 開始使用 Express Checkout .....	3
1.1. 簡介 .....	3
1.2. Express Checkout 和購物車之間的關係 .....	5
1.3. 使用 Express Checkout 的先決條件 .....	5
1.4. Express Checkout 組成要素 .....	6
1.5. 實作最簡單的 Express Checkout 整合 .....	8
1.6. 排解 Express Checkout 整合的問題 .....	19
1.7. Express Checkout 功能 .....	21
2. Express Checkout 使用者介面要求 .....	26
2.1. PayPal Mark 標誌 .....	26
2.2. PayPal 結帳按鈕 .....	27
3. 自訂 Express Checkout .....	29
3.1. PayPal「審核」頁面上的訂單詳細資料 .....	29
3.2. 提供禮品選項 .....	33
3.3. 詢問買家是否同意收到促銷電子郵件 .....	34
3.4. 覆寫客服電話號碼 .....	35
3.5. 加入意見調查問題 .....	36
3.6. PayPal 頁面風格 .....	37
3.7. 變更地區設定 .....	39
3.8. 運送地址選項 .....	41
3.9. 自動填寫運送和聯絡資訊 .....	45
3.10. 買家在 PayPal 上付款 .....	46
4. 取得 API 憑證 (Credentials) .....	48
4.1. 建立和管理標準 API 憑證 .....	48
4.2. 憑證類型 .....	48
4.3. 建立標準 API 憑證 .....	49
4.4. 管理證書憑證 .....	50
5. 讓 Express Checkout 整合正式上線 .....	55
6. 相關的 API 操作 .....	56
6.1. Express Checkout 的「銷售」付款動作 .....	56
6.2. Express Checkout 的「授權」付款動作 .....	56
6.3. Express Checkout 的「訂單」付款動作 .....	57
6.4. 發放退款 .....	58
6.5. 使用 API 幂等 .....	59
7. 將 Express Checkout 與 PayPal SDK 整合 .....	61

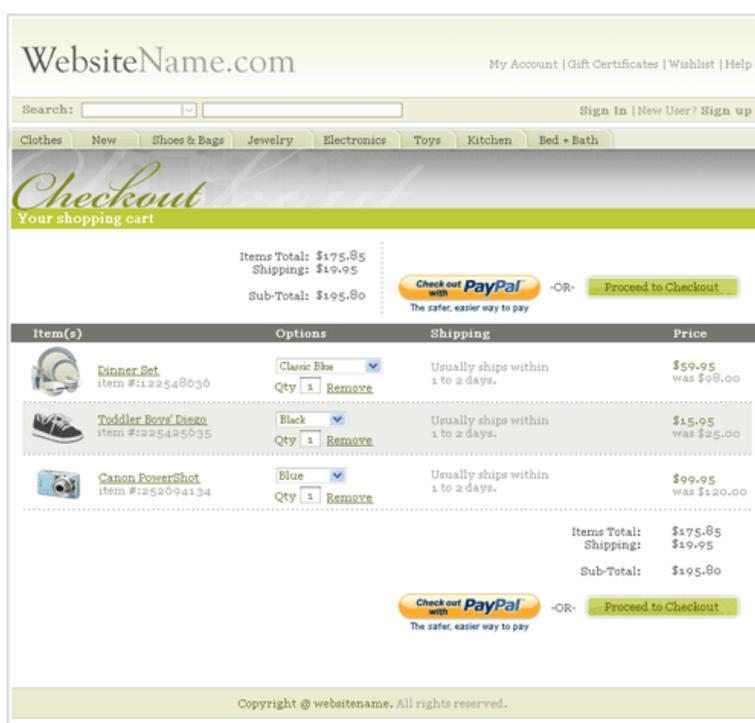
# 1. 開始使用 Express Checkout

Express Checkout 能讓買家快速輕鬆地使用 PayPal 支付購物款項。Express Checkout 能立即將所有交易詳細資料提供給買家，包括訂單詳細資料、保險選項及稅金總額。

研究結果顯示，在網站中新增「Express Checkout」按鈕，最多可提升提升銷售量 18%。

## 1.1. 簡介

「Express Checkout」按鈕提供買家另一種付款方式，並能配合你現有的線上支付方案。線上買家認為 PayPal 既方便又安全，因為他們可以使用 PayPal 餘額、銀行帳戶或信用卡輕鬆支付購物款項。下列網頁將「Express Checkout」按鈕與原有的結帳按鈕並列使用。



Express Checkout 可以：

- 接受買家使用 PayPal 帳戶支付購物款項。
- 減少買家重新輸入運送、帳單或付款資訊等個人資料的時間。
- 讓買家在完成交易後繼續停留在你的網站上。
- 讓買家註冊定期扣款的方案。
- 詳情請參閱：<https://www.paypal.com/webapps/mpp/express-checkout>

## Express Checkout 結帳流程

Express Checkout 不但讓買家能在線上輕鬆支付購物款項，也能讓你在接受 PayPal 購物款項的同時，充分控制買家進行結帳時的整個流程。

實作 Express Checkout 之前，請先想想買家原本的結帳流程。一般結帳流程可能有下列一連串的步骤：

圖 1：一般結帳流程



在一般的結帳流程中，買家會歷經下列步驟：

1. 在購物車頁面選擇結帳
2. 提供運送資料
3. 選擇付款方式並提供帳單 / 付款資料
4. 檢查訂單資訊是否正確，然後支付購物款項
5. 收到下單成功的確認訊息

在 Express Checkout 結帳流程中，買家仍必須先選擇結帳，但不需要輸入運送、帳單或付款資料，因為 PayPal 會自動填入他預先儲存在 PayPal 中的資料，這樣能夠簡化結帳流程，使結帳更快速。

下圖為 Express Checkout 的結帳流程：

圖 2：Express Checkout 結帳流程



在 Express Checkout 結帳流程中，買家會歷經下列步驟：

1. 按一下「Check out with PayPal」以選擇使用 Express Checkout
2. 登入 PayPal 以驗證用戶身份
3. 在 PayPal 的網站上檢查交易資料是否正確

**備註：**你也可選擇是否讓買家在你的網站上檢查訂單資料（上圖不包括此步驟）。你也可以加入其他結帳步驟，包括在「確認訂單」頁面加入進階銷售 (upselling)。

4. 買家確認訂單，並從你的網站支付購物款項
5. 收到下單成功的確認訊息

## 支援的國家 / 地區和貨幣

Express Checkout 能讓你接受來自許多國家與地區的交易款項。我們也為一些國家與地區提供當地語言版本的結帳流程。

若要了解 Express Checkout 所支援國家 / 地區和貨幣的詳細資料，請參閱 [PayPal 全球服務網站](#) 清單，並選擇特定國家 / 地區的 PayPal 網站。

你也可以把 API 操作中所使用到的 [貨幣代碼](#) 和 [國家 / 地區代碼](#) 的參考網頁加入瀏覽器書籤中。

## 1.2. Express Checkout 和購物車之間的關係

若你的網站原本沒有購物車的設計，你也沒有在網站中使用 Express Checkout，不妨考慮使用第三方的購物車。購物車軟體可讓買家將商品放進購物籃中，並在結帳時計算交易總金額。

PayPal 與許多家購物車廠商都有合作關係，他們的軟體都能與 PayPal 相容，為買家提供安全的購物環境。購物車廠商會向你說明如何將他們的購物車整合到你的網站上。若要了解 PayPal 支援哪些購物車，請參閱「[PayPal 合作夥伴目錄](#)」。

**重要資訊：**若你選用名單中某一家廠商的購物車，請勿聯絡 PayPal。PayPal 無法掌控任何一家購物車廠商，也無法協助你解決整合第三方購物車軟體或使用時可能遇到的問題。

## 1.3. 使用 Express Checkout 的先決條件

若想要使用 Express Checkout，先決條件包括擁有特定類型的 PayPal 帳戶，以及具備必要的程式設計能力和經驗。若你想要使用 Express Checkout，但是欠缺必要的能力或經驗，請考慮改用 [PayPal Payments Standard](#)（前身為 Website Payments Standard），或者由 [PayPal 合作夥伴](#) 提供的購物車。

若要使用 Express Checkout，你必須具備：

- **PayPal 商業帳戶。**透過 [商業帳戶](#)，PayPal 就能為你的商店向買家收取商品或服務的款項。PayPal 會管理這些交易，並針對每一筆買家所支付的購物款項，向你收取一筆小額固定費用及特定百分比的手續費。
- **包含兩個測試帳戶的 [Sandbox 帳戶](#)。**PayPal Sandbox 提供模擬 PayPal 的環境，你不必透過實際的交易，就能模擬將 Express Checkout 整合到網站後的情形。其中一個測試帳戶是商店帳戶（在 Sandbox 中稱為「賣家」），另一個則是買家帳戶。你在進行測試時，可以使用兩個以上的帳戶；不過你必須先擁有 Sandbox 帳戶，才能建立測試帳戶並進行測試。
- **編寫 HTML 程式碼的經驗。**你可能已經有自己的網站，並為你在網站上販售的商品或服務建立結帳機制。在這種情況下，你必須將「Pay with Express Checkout」按鈕和另一個稱為「Express Checkout Mark」的 PayPal 選項標誌放到你的網頁上。只要按一下「Pay with Express Checkout」按鈕，就會開始 PayPal 結帳流程。透過 Express Checkout Mark，即使買家不是從「Pay with Express Checkout」開始結帳流程，也可在指定付款方式的「付款」頁面中選用 Express Checkout。

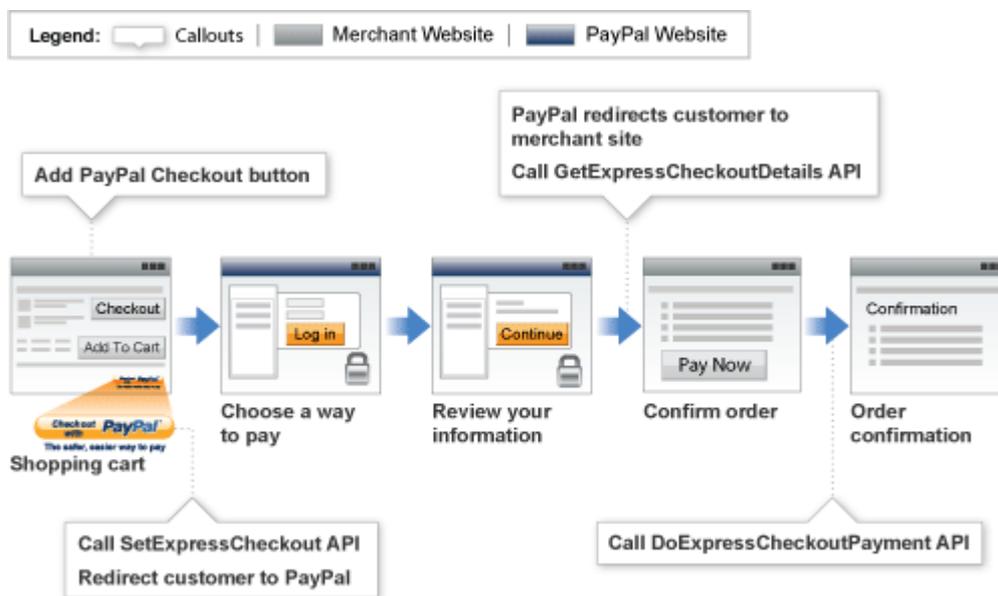
- 程式設計經驗。買家在點選按鈕時，你必須提供程式碼，以要求 PayPal 伺服器設定或處理交易和處理回應。PayPal 提供便於使用的 [Name-Value Pair 介面](#)，這種介面是建構在 HTTP 要求 / 回應模型上，另外也提供比較複雜的 [SOAP Web Services 介面](#)。此外，PayPal 也提供 [整合精靈](#)。你還可以選擇使用提供各種程式設計語言介面的 [軟體開發工具 \(SDK\)](#)，這些程式設計語言包括 PHP、Ruby、Java，以及 C# 等 .NET 程式語言。雖然你不必精通任何一種特定的程式語言，但你必須了解程式語言的邏輯，尤其是要求 / 回應模型、錯誤處理機制，以及編寫應用程式等級程式碼的訣竅。

**備註：**若你不太了解以上關於程式設計經驗先決條件的說明內容（例如你不清楚介面是什麼），你還是可以使用由 PayPal 或其他購物車廠商提供的 [Express Checkout](#)，或使用其他的 PayPal 產品，例如 [PayPal Payments Standard](#)。除非你具備足夠的程式設計經驗，否則請勿嘗試自行整合 [Express Checkout](#)。對程式設計或開發人員來說，[Express Checkout](#) 非常簡單，但是沒接觸過程式設計的新手可能會在完全沒有查覺到的情況下，造成交易或商品的損失。

## 1.4. Express Checkout 組成要素

你在實作 [Express Checkout](#) 結帳流程時，需要使用下列要素：「[Express Checkout](#)」按鈕、PayPal API 操作、PayPal 指令和 Token。

下列概念圖顯示你用來將 [Express Checkout](#) 整合到網站上的各項組成要素：



[Token](#) 是由 PayPal 指定的一個值，能將 API 操作和指令的執行作業與用戶結帳流程中的特定執行階段建立關聯。

**備註：**上圖未列出 [Token](#)。

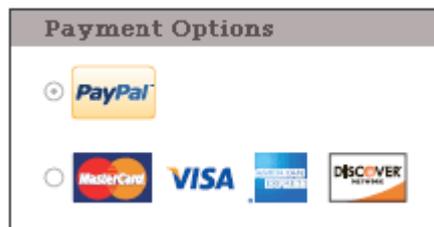
[「Express Checkout」按鈕](#)

PayPal 提供可讓你放在網站上的按鈕和圖片。

若要實作 Express Checkout 購物車流程，請將下面的按鈕放在購物車頁面上：



若要將 PayPal 設為 Express Checkout 結帳流程中的一種付款選項，請將 PayPal Mark 圖片加入你的付款選項清單中。PayPal 建議你使用選項按鈕（radio button）來讓買家選擇付款方式：



若要了解詳細資訊，請參閱下面的「

[Express Checkout 使用者介面要求](#)」一節。

## Express Checkout API 操作

PayPal 為 Express Checkout 提供三種 API 操作：建立交易、取得買家資訊，以及處理購物款項以完成交易。

API 操作	說明
<code>SetExpressCheckout</code>	用來建立 Express Checkout 交易。你可以指定使用特定資料，以自訂 PayPal 網站的外觀，以及 PayPal 網站要顯示的資料。你必須提供下列資料： <ul style="list-style-type: none"><li>買家登入 PayPal 並成功核准購物款項後，PayPal 重新導向買家瀏覽器的目的地網頁 URL</li><li>買家若放棄付款，PayPal 重新導向買家瀏覽器的目的地網頁 URL</li><li>訂單總金額或是預估總額，需要盡可能地精確。</li></ul>
<code>GetExpressCheckout</code>	用來向 PayPal 取得買家資訊，包括運送資料。
<code>DoExpressCheckoutPayment</code>	用來完成 Express Checkout 交易，包括計算出訂單的實際總金額。

若要了解關於這些 API 操作的詳細資料，請參閱下面的「[實作最簡單的 Express Checkout 整合](#)」一節。

## Express Checkout 指令

PayPal 提供一項指令，讓你在重新導向買家的瀏覽器至 PayPal 時使用。這項指令可以讓你的買家登入 PayPal，以核准 Express Checkout 購物款項。

你在重新導向買家的瀏覽器至 PayPal 時，必須為 Express Checkout 指定 `_ExpressCheckout` 指令。你也必須指定 `SetExpressCheckout API` 操作所傳回之用來識別交易的 Token。

**備註：**若要讓 PayPal 將買家的瀏覽器重新導向你的網站，必須先叫用 `SetExpressCheckout API`，並指定 PayPal 重新導向你的網站時使用的 URL。PayPal 會在買家使用 PayPal 支付購物款項時，將買家的瀏覽器重新導向成功 URL；若買家取消交易，則導向取消 URL。

若買家同意支付購物款項，PayPal 就會將買家的瀏覽器重新導向成功交易的 URL，其中包含下列資訊：

- 重新導向 PayPal 網頁時所用的 Token

- 買家專屬識別碼 (Payer ID)

若買家取消交易，PayPal 就會將買家的瀏覽器重新導向取消 URL，其中包含重新導向 PayPal 網頁時所用的 Token。

## Express Checkout Token 的使用

Express Checkout 會使用 Token 來控制存取 PayPal 的權限，並執行 Express Checkout API 操作。

SetExpressCheckout API 操作會傳回一個 Token，而其他 Express Checkout API 操作及 \_ExpressCheckout 指令會使用此 Token 來識別交易。Token 的有效期限大約是 3 小時。

## 1.5. 實作最簡單的 Express Checkout 整合

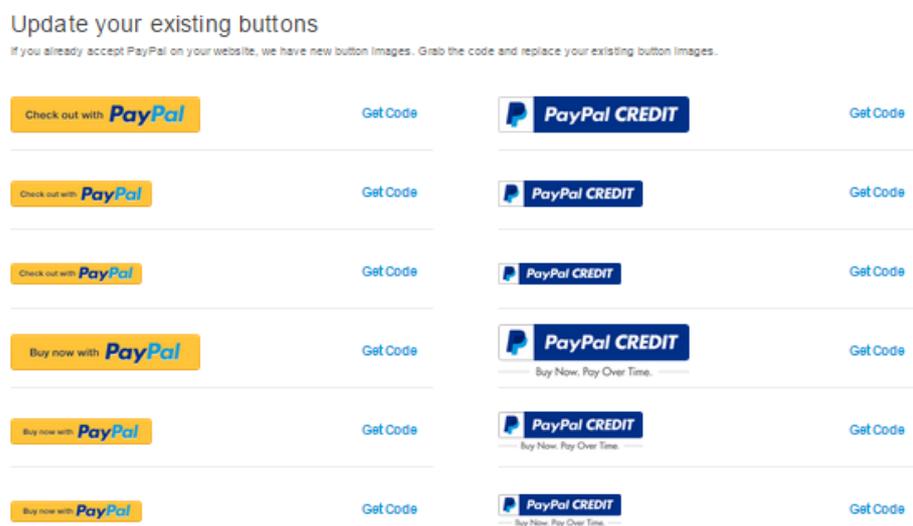
若要實作最簡單的 Express Checkout 整合，必須先建立一個 PayPal 按鈕，然後在買家點選此按鈕時呼叫下列 PayPal API 操作：SetExpressCheckout、DoExpressCheckoutPayment，通常還包括 GetExpressCheckoutDetails。

你也必須讓開始結帳流程的買家，選擇 PayPal 做為支付購物款項的方式。因此，你必須從兩處呼叫這些 API 操作，也就是在結帳頁面和付款方式頁面執行這些 API 操作。

取得「Express Checkout」按鈕和 PayPal Mark

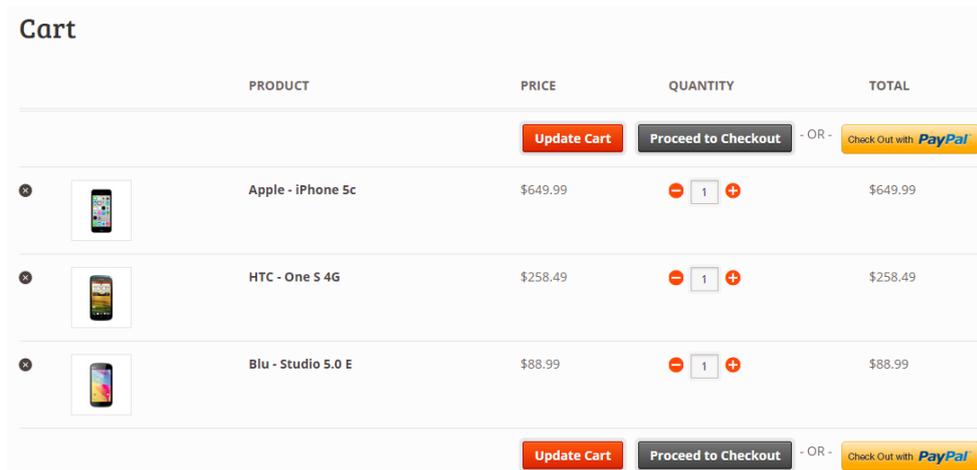
PayPal 要求你使用 PayPal 安全伺服器 (secure PayPal servers) 上的付款按鈕圖片。我們更新圖片時，新的圖片會自動顯示在你的應用程式中。請將「Express Checkout」按鈕置於您的結帳頁面上，並依以下指示取得「Express Checkout」按鈕：

1. 前往 PayPal 的「[按鈕程式碼](https://www.paypal.com/us/webapps/mpp/logos-buttons)」頁面 (<https://www.paypal.com/us/webapps/mpp/logos-buttons>)。你會看到類似下圖的畫面：

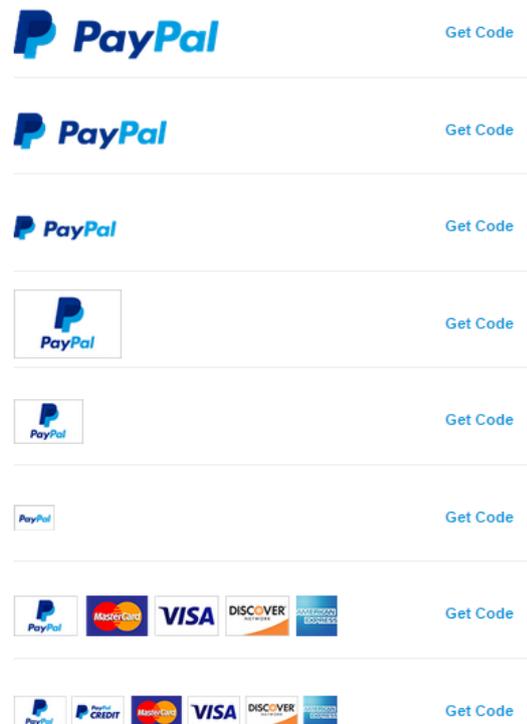


2. 選擇想要使用的按鈕大小，然後按一下旁邊的「Get Code」連結。
3. 找到你希望按鈕在結帳頁面上顯示的位置，將圖片原始碼貼到該位置的 HTML 原始碼中。

完成之後，當你在瀏覽器中開啟結帳頁面時，應該會看到「Check out with PayPal」按鈕，如下圖所示：



你必須將 PayPal Mark 圖片放在付款頁面上。取得此圖片的方法與取得按鈕的方法類似，你可以在剛才的「按鈕程式碼」頁面 (<https://www.paypal.com/us/webapps/mpp/logos-buttons>) 找到。



詳情請參閱「Express Checkout 使用者介面要求」一節。

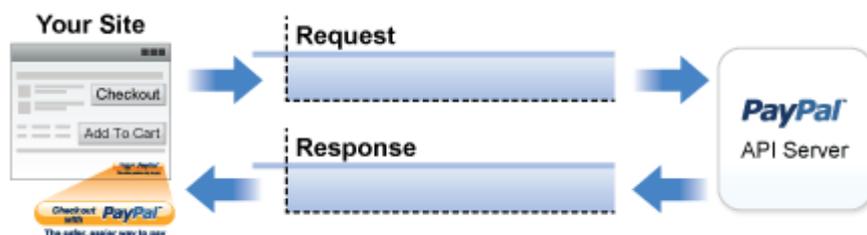
編寫程式碼前的準備工作

若你不清楚 PayPal API 的運作方式，請詳閱此主題的內容，以了解順利使用 PayPal [Name-Value Pair API](#) 所需的最基本資訊。

## PayPal API 主從式架構

PayPal API 採用主從式架構；在此架構中，你的網站是 PayPal 伺服器的用戶端。

你網站上的某個網頁會傳送要求給 PayPal API 伺服器，以在該伺服器上展開某個動作。PayPal 伺服器收到要求後會回應用戶端，確認已經採用戶端所要求的行動，或通知用戶端發生了錯誤。伺服器的回應內容可能還會包括與該要求相關的額外資訊。下圖顯示主從式架構的基本運作機制。



假如你想要從 PayPal 取得買家的運送地址，你可以發出要求指定執行某個 API 操作，以取得買家詳細資料。PayPal API 伺服器回應要求時，會告知要求是否成功。若操作成功，伺服器就會在回應中包含所要求的資訊，也就是買家的運送地址。若操作失敗，伺服器就會在回應中包含一則或多則錯誤訊息。

## 取得 API 憑證

若你要使用 PayPal API，就必須擁有 API 憑證，藉此表示你是有權執行多種 API 的 PayPal 商業帳戶持有人。雖然電子簽章 (Signature) 和證書 (Certificate) 皆可認證 API 的操作，但 PayPal 較推薦電子簽章。詳情請參閱「[建立和管理標準 API 憑證](#)」一節。

**重要資訊：**雖然電子簽章和證書皆可認證，但不能同時使用。

## 設定 Express Checkout 交易

若要設定 Express Checkout 交易，你必須叫用 `SetExpressCheckout` (NVP | SOAP) API 操作，提供足夠資料以展開付款流程，並在操作成功後重新導向 PayPal。

此範例假設你已經建立用來與 PayPal 伺服器通訊的機制，並擁有具備 API 憑證的 PayPal 商業帳戶，亦假設付款操作為最終銷售。

設定 Express Checkout 交易時，你指定 `SetExpressCheckout` 要求中的值，然後呼叫 API。指定的值會控制 PayPal 頁面流程，以及可供你和買家使用的選項。你應該從設定標準 Express Checkout 交易開始，此模式可以修改以加入其他選項。若要設定最簡單的標準 Express Checkout 交易：

1. 指定你想執行的 `SetExpressCheckout` API 操作，以及你想使用的 API 版本。

```
METHOD=SetExpressCheckout
```

```
VERSION=109.0
```

2. 指定你的 API 憑證。

使用下列參數來加上電子簽章：

```
USER=API_username
```

```
PWD=API_password
```

```
SIGNATURE=API_signature
```

在 Sandbox 中，你永遠可以使用下列電子簽章：

```
USER=sdk-three_api1.sdk.com
```

```
PWD=QFZCWN5HZM8VBG7Q
```

```
SIGNATURE=A-IzJhZZjhg29XQ2qnhapuwXIDzyAZQ92FRP5dqBzVes0kzbdUONzmOU
```

3. 指定交易金額；如果計價單位不是美元，也請指定貨幣。

若知道交易總金額，請加以指定；否則，請指定小計。不論使用何種貨幣，顯示格式的右側必須到小數點第二位，左側可選擇是否使用千位分隔符，但千位分隔符必須為逗號。

例如，2,000.00 歐元必須指定為 `2000.00` 或 `2,000.00`。你指定的金額不應超過任何所使用貨幣的每筆交易金額限制。

```
PAYMENTREQUEST_0_AMT=amount
```

```
PAYMENTREQUEST_0_CURRENCYCODE=currencyID
```

指定返回 URL。返回 URL 是你網站上的特定頁面；在買家登入 PayPal 並核准購物款項後，PayPal 就會將買家的瀏覽器重新導向此頁面。通常這會是你網站上的安全頁面 (`https://...`)。

**備註：**你可以利用 URL 來攜帶你的網站頁面之間的參數。例如，使用

```
https://www.yourcompany.com/page.html?param=value ...
```

 語法設定返回 URL，同時指定其他參數

(例如指定你網站系統上的訂單號碼)。只要這個返回 URL 頁面可以辨識，所指定的參數都是有效的參數。

```
RETURNURL=return_url
```

4. 指定取消 URL。

取消 URL 是你網站上的特定頁面；若買家沒有核准購物款項，PayPal 就會將買家的瀏覽器重新導向此頁面。通常會是你網站上的安全頁面 (`https://...`)，你可以藉此將買家的瀏覽器重新導向 PayPal。

**備註：**在 PayPal 重新導向取消 URL 後，你可以視需要傳送 SetExpressCheckout 的要求值，做為取消 URL 中的參數，使其變成你網站上可用的資訊。

```
CANCELURL=cancel_url
```

5. 指定付款操作。

雖然預設付款操作是「Sale」，但最好將付款操作明確指定為「Sale」（銷售）、  
「Authorization」（授權）或「Order」（訂單）。詳情請參閱「相關的 API 操作」一節。

```
PAYMENTREQUEST_0_PAYMENTACTION=Sale
```

若成功呼叫 `SetExpressCheckout` API，即可將買家的瀏覽器重新導向 PayPal，並使用  
`SetExpressCheckout` 回應中傳回的 Token，執行 `_express-checkout` 命令。

備註：下列範例使用的是 PayPal Sandbox 伺服器：

```
https://www.sandbox.paypal.com/cgi-bin/webscr?cmd=_express-checkout&token=tokenValue
```

### 取得 Express Checkout 交易詳細資料

若要取得 Express Checkout 交易的相關詳細資料，可以叫用 `GetExpressCheckoutDetails` (NVP | SOAP) API 操作。

此範例假設買家在 PayPal 檢閱交易後，PayPal 使用有效的 Token 重新導向買家的瀏覽器。

雖然你不需要叫用 `GetExpressCheckoutDetails` API 操作，但多數 Express Checkout 實作會進行這項操作取得買家相關資料。你在 `SetExpressCheckout` API 指定返回 URL，再從返回 URL 指定的頁面叫用 `GetExpressCheckoutDetails` API。你通常會在重新導向時叫用這項操作，並使用回應中的資料來填寫你的審核頁面。若要透過此 API 操作取得買家的運送地址和 Payer ID：

1. 指定你想執行的 `GetExpressCheckoutDetails` API 操作，以及你想使用的 API 版本。

```
METHOD=GetExpressCheckoutDetails  
VERSION=XX.0
```

2. 指定你的 API 憑證。

使用下列參數來加上電子簽章：

```
USER=API_username  
PWD=API_password  
SIGNATURE=API_signature
```

3. 指定在 PayPal 將買家的瀏覽器重新導向你的網站時傳回的 Token。

你在叫用 `SetExpressCheckout` API 時有指定重新導向的 URL，當 PayPal 將買家的瀏覽器重新導向這個 URL 時，也會在 HTTP 要求中回傳 `token` 參數。

4. 執行 `GetExpressCheckoutDetails` API 以取得買家資訊。
5. 取用 `GetExpressCheckoutDetails` API 傳回的欄位。

備註：回應中只會傳回已自動填寫的欄位。

## 完成 Express Checkout 交易

若要完成 Express Checkout 交易，你必須叫用 `DoExpressCheckoutPayment` (NVP | SOAP) API 操作。

此範例假設在你呼叫 PayPal `SetExpressCheckout` API 將買家的瀏覽器重新導向你的網站時，使用了有效的 Token。此外，你可能會選擇先呼叫 `GetExpressCheckoutDetails` API 再呼叫 `DoExpressCheckoutPayment` API。

最簡單的情況就是在呼叫 `SetExpressCheckout` API 時，設定訂單總金額。然而，若在呼叫 `SetExpressCheckout` API 時還不確定金額，也可以在 `DoExpressCheckoutPayment` 呼叫中變更金額。此範例假設處於最簡單的情況下，其中總金額是在呼叫 `SetExpressCheckout` API 時指定的返回 URL 頁面才能指定的。雖然還可以指定更多選項，但本範例並不多作指定。若要操作 Express Checkout 交易：

1. 指定你想執行的 `DoExpressCheckoutPayment` API 操作，以及你想使用的 API 版本。

```
METHOD=DoExpressCheckoutPayment  
VERSION=XX.0
```

2. 指定你的 API 憑證。

使用下列參數來加上電子簽章：

```
USER=API_username  
PWD=API_password  
SIGNATURE=API_signature
```

3. 指定在 PayPal 將買家的瀏覽器重新導向你的網站時傳回的 Token。

你在叫用 `SetExpressCheckout` API 時有指定重新導向的 URL，當 PayPal 將買家的瀏覽器重新導向這個 URL 時，也會在 HTTP 要求中回傳 `token` 參數。

```
TOKEN=tokenValue
```

4. 指定在 PayPal 將買家的瀏覽器重新導向你的網站時傳回的 Payer ID。

你在叫用 `SetExpressCheckout` API 時有指定重新導向的 URL，當 PayPal 將買家的瀏覽器重新導向這個 URL 時，也會在 HTTP 要求中回傳 Payer ID。你也可以呼叫 `GetExpressCheckoutDetails` API 來取得 Payer ID。

```
PAYERID=id
```

5. 指定訂單金額，包括運費、處理費和稅金；若計價單位不是預設貨幣（美元），也請指定貨幣。

在多數情況下，這筆金額與你在 `SetExpressCheckout` 呼叫中指定的金額相同，不過會另外計入運費和稅金。

```
PAYMENTREQUEST_0_AMT=amount
```

```
PAYMENTREQUEST_0_CURRENCYCODE=currencyID
```

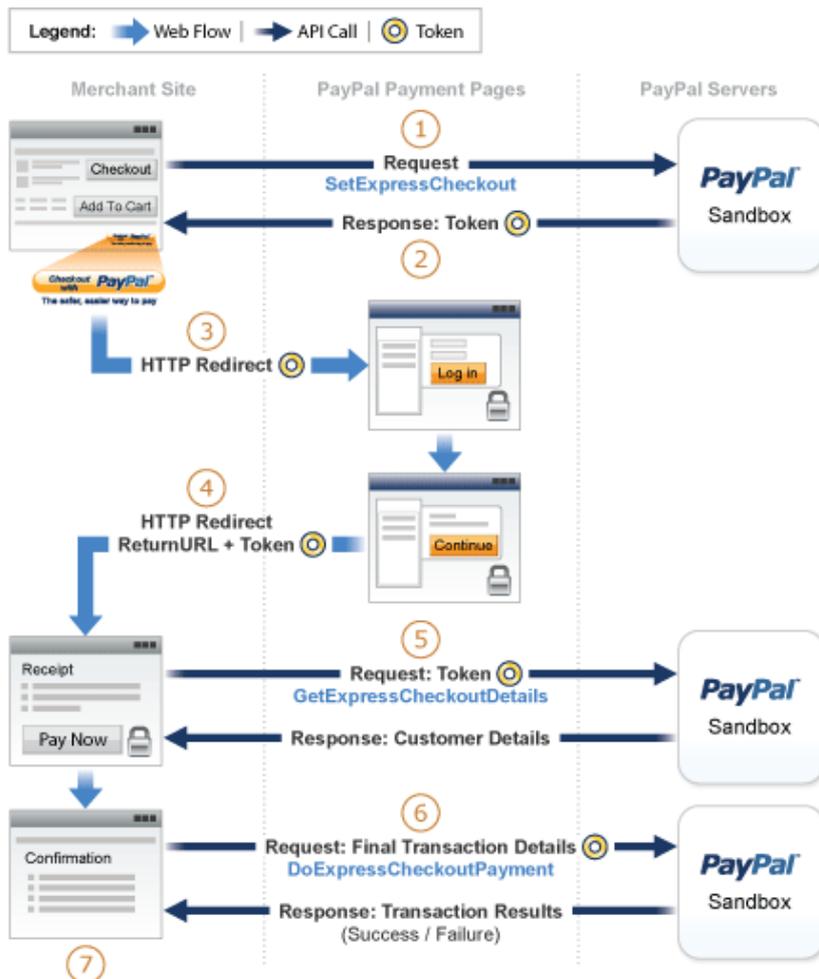
- 指定你在 `SetExpressCheckout` 中指定的同一個付款動作。

```
PAYMENTREQUEST_0_PAYMENTACTION=Sale
```

## 測試 Express Checkout 整合

你可以在 PayPal 提供的 Sandbox 測試環境裡測試 Express Checkout 整合。

此範例說明如何使用 HTTP 表單，並透過這些表單為 API 操作提供值，以便模擬你的網頁。你可以透過 PayPal 提供的 Sandbox 測試環境中進行初步的測試；不過若要進行更完整的測試，就需要使用包含實際程式碼的網頁來取代這些表單。下圖顯示 Express Checkout 的執行流程，此流程是使用 Sandbox 做為 API 伺服器。左邊的頁面代表你的網站。



下列各個步驟可以對應到上圖中的編號。請執行每個步驟中的動作，以測試 Express Checkout。

1. 在你的網站上叫用會在 **Sandbox** 中呼叫 `SetExpressCheckout API` 的表單。

若要叫用 API，請將表單欄位名稱設為與要設定欄位的 **NVP** 名稱相同、為這些欄位指定對應的值，然後將表單貼到 **PayPal Sandbox** 伺服器中，例如 `https://api-3t.sandbox.paypal.com/nvp`，如下列範例所示：

```
<form method=post action=https://api-3t.sandbox.paypal.com/nvp>
  <input type=hidden name=USER value=API_username>
  <input type=hidden name=PWD value=API_password>
  <input type=hidden name=SIGNATURE value=API_signature>
  <input type=hidden name=VERSION value=XX.0>
  <input type=hidden name=PAYMENTREQUEST_0_PAYMENTACTION
    value=Sale>
  <input name=PAYMENTREQUEST_0_AMT value=19.95>
  <input type=hidden name=RETURNURL
    value=https://www.YourReturnURL.com>
  <input type=hidden name=CANCELURL
    value=https://www.YourCancelURL.com>
  <input type=submit name=METHOD value=SetExpressCheckout>
</form>
```

**備註：**請使用擁有電子簽章之 **Sandbox** 商業測試帳戶的 API 使用者名稱。請查看 **Sandbox** 的「測試憑證」(Test Certificates) 頁籤，以取得電子簽章。若你沒有使用電子簽章，就必須使用另一個 **Sandbox** 伺服器。



**重要資訊：**我們在此範例中沒有建立安全連線，因此不應該實際用於 `paypal.com` 上。你必須在實作過程中保護 `USER`、`PWD` 和 `SIGNATURE` 的值。你可以將這些值存放在安全的地點（請勿將其存放於網路伺服器的文件根目錄），並設定檔案權限，如此只有執行你的電子商務應用程式的系統用戶能夠存取。

2. 檢視來自 `SetExpressCheckout API` 操作的回應字串。

PayPal 會回應一則類似以下所示的訊息。請注意，訊息中的 `ACK` 狀態應該要是為 `Success`，並且包含用於後續步驟的 `Token`。

```
TIMESTAMP=2007%2d04%2d05T23%3a23%3a07Z
&CORRELATIONID=63cdac0b67b50
```

```
&ACK=Success

&VERSION=XX%2e000000

&BUILD=1%2e0006

&TOKEN=EC%2d1NK66318YB717835M
```

3. 若操作成功，使用 Token 將你的瀏覽器重新導向 Sandbox 以便登入，如下所示：

```
https://www.sandbox.paypal.com/cgi-bin/webscr?cmd=_express-checkout&token=EC-1NK66318YB717835M
```

你可能需要將 URL 解碼，也就是反譯 URL 編碼，做法是以 ASCII 代碼取代 Hex 代碼，例如以連字號 (-) 取代 %2d。你在以 Sandbox 帳號登入之前請先登入 <https://developer.paypal.com>。接著，你應該登入代表買家的測試帳戶，而不是代表你（也就是商店）的賣家商業測試帳戶。

4. 登入買家測試帳戶後，確認帳戶的詳細資料。

確認詳細資料後，Sandbox 會將你的瀏覽器重新導向你在叫用 SetExpressCheckout API 操作時指定的返回 URL，如下列範例所示：

```
http://www.YourReturnURL.com/?token=EC-1NK66318YB717835M&PayerID=7AKUSARZ7SAT8
```

5. 在你的網站上叫用會在 Sandbox 中呼叫 `GetExpressCheckoutDetails API` 操作的表單：

```
<form method=post action=https://api-3t.sandbox.paypal.com/nvp

  <input type=hidden name=USER value=API_username>

  <input type=hidden name=PWD value=API_password>

  <input type=hidden name=SIGNATURE value=API_signature>

  <input type=hidden name=VERSION value=XX.0>

  <input name=TOKEN value=EC-1NK66318YB717835M>

  <input type=submit name=METHOD value=GetExpressCheckoutDetails>

</form>
```

若操作成功，`GetExpressCheckoutDetails API` 會傳回關於付款者的資訊，例如下列資訊：

```
TIMESTAMP=2007%2d04%2d05T23%3a44%3a11Z

&CORRELATIONID=6b174e9bac3b3

&ACK=Success

&VERSION=XX%2e000000
```

```
&BUILD=1%2e0006

&TOKEN=EC%2d1NK66318YB717835M

&EMAIL=YourSandboxBuyerAccountEmail

&PAYERID=7AKUSARZ7SAT8

&PAYERSTATUS=verified

&FIRSTNAME=...

&LASTNAME=...

&COUNTRYCODE=US

&BUSINESS=...

&PAYMENTREQUEST_0_SHIPTONAME=...

&PAYMENTREQUEST_0_SHIPTOSTREET=...

&PAYMENTREQUEST_0_SHIPTOCITY=...

&PAYMENTREQUEST_0_SHIPTOSTATE=CA

&PAYMENTREQUEST_0_SHIPTOCOUNTRYCODE=US

&PAYMENTREQUEST_0_SHIPTOCOUNTRYNAME=United%20States

&PAYMENTREQUEST_0_SHIPTOZIP=94666

&PAYMENTREQUEST_0_ADDRESSID=...

&PAYMENTREQUEST_0_ADDRESSSTATUS=Confirmed
```

6. 在你的網站上叫用會在 Sandbox 中叫用 `DoExpressCheckoutPayment API` 操作的表單：

```
<form method=post action=https://api-3t.sandbox.paypal.com/nvp>

  <input type=hidden name=USER value=API_username>

  <input type=hidden name=PWD value=API_password>

  <input type=hidden name=SIGNATURE value=API_signature>

  <input type=hidden name=VERSION value=XX.0>

  <input type=hidden name=PAYMENTREQUEST_0_PAYMENTACTION

    value=Sale>

  <input type=hidden name=PAYERID value=7AKUSARZ7SAT8>

  <input type=hidden name=TOKEN value= EC%2d1NK66318YB717835M>
```

```
<input type=hidden name=PAYMENTREQUEST_0_AMT value= 19.95>

<input type=submit name=METHOD value=DoExpressCheckoutPayment>

</form>
```

7. 檢視來自 `DoExpressCheckoutPayment` API 操作的回應字串。

若操作成功，回應中的 `ACK` 狀態應該會設為 `Success`，如下所示：

```
TIMESTAMP=2007%2d04%2d05T23%3a30%3a16Z

&CORRELATIONID=333fb808bb23

ACK=Success

&VERSION=XX%2e000000

&BUILD=1%2e0006

&TOKEN=EC%2d1NK66318YB717835M

&PAYMENTREQUEST_0_TRANSACTIONID=043144440L487742J

&PAYMENTREQUEST_0_TRANSACTIONTYPE=expresscheckout

&PAYMENTREQUEST_0_PAYMENTTYPE=instant

&PAYMENTREQUEST_0_ORDERTIME=2007%2d04%2d05T23%3a30%3a14Z

&PAYMENTREQUEST_0_AMT=19%2e95

&PAYMENTREQUEST_0_CURRENCYCODE=USD

&PAYMENTREQUEST_0_TAXAMT=0%2e00

&PAYMENTREQUEST_0_PAYMENTSTATUS=Pending

&PAYMENTREQUEST_0_PENDINGREASON=authorization

&PAYMENTREQUEST_0_REASONCODE=None
```

## 安全性問題

你必須隨時保護敏感資料的安全。敏感資料不只包括你的 API 憑證，也包括客戶瀏覽器中暴露的任何資料，例如儲存在 `Cookie` 中的交易相關資料。

- 在最簡單的範例中（例如 `PayPal` 為了示範如何使用 `Express Checkout` 而提供的範例），API 憑證可能會遭到暴露。因此，若你從範例或 `SDK` 中複製程式碼來使用，就應該隨時檢查你的網站是否有安全性問題，並在發佈網站前修正這些問題。
- 為與 `PayPal` 交易相關的所有已儲存資訊加密。例如，若你將訂單狀態資訊儲存在 `Cookie` 中，請務必加密這些資訊。

- 使用 HTTPS 等安全傳輸協定在你的網站和 PayPal 之間傳送資訊。請勿使用 HTTP 或不安全的 cURL。

## 1.6. 排解 Express Checkout 整合的問題

若你在進行整合時遇到問題，可以先採取一些措施來解決問題。若你採取這些措施後還是無法解決問題，也可以與商店技術支援 (MTS) 聯絡。

若你無法自行解決問題，就需要收集一些基本資訊（包括導致錯誤發生的動作的紀錄檔），再與 MTS 聯絡。你可以前往 <https://www.paypal.com/mts> 與 MTS 聯絡。

### 如何處理錯誤

回應訊息中會包含一個 `ACK` 值。除非 `ACK=Success`，否則你必須進一步查看是否有錯誤或警告訊息。

你必須查看來自 PayPal 伺服器的每則回應，以判斷是否發生了錯誤。由於警告和失敗值有很多個，因此最保險的方法，是查看回應中是否包含 `ACK=Success`。若 `ACK` 傳回任何其他的值，你必須檢查回應中是否包含錯誤代碼和錯誤訊息。

未成功的回應可能會包含一個以上的錯誤代碼和錯誤訊息。「NVP 錯誤」欄位的開頭是 `L_ERRORCODEn`，其中的 `n` 是從 0 到 9 的數字，不同的數字代表不同的錯誤類型。每個錯誤代碼包含兩種訊息，也就是 `L_SHORTMESSAGEn` 和 `L_LONGMESSAGEn`，其中的 `n` 與 `L_ERRORCODEn` 相同。在 `n`。「SOAP 錯誤」與 NVP 相對應的錯誤欄位是：`ErrorCode`、`ShortMessage` 和 `LongMessage`。

**重要資訊：**由於每個錯誤代碼不一定會與特定錯誤訊息配對，因此你必須在發生錯誤時同時參照錯誤代碼和錯誤訊息，以決定該採取什麼做法。

若要檢閱按照 ExpressCheckoutAPI 分群的錯誤代碼清單，請參閱 [商店 API 錯誤代碼](#)。

有些錯誤只是暫時的現象，因此你可以嘗試重新執行操作；例如一個指出 PayPal 發生問題的錯誤。若過了一個小時問題還是沒有解決，那麼問題可能與你實作 Express Checkout 的過程有關，因為 PayPal 伺服器幾乎是全天候提供服務。

有些錯誤指出買家的帳戶有問題，例如資金來源不再有效，或者買家的帳戶在某方面受到限制。由於這類問題可能代表有風險存在，因此建議你等到問題解決後再寄出商品。錯誤訊息中包含的資訊很完整，你可以根據這些資訊在網站上張貼相關訊息，告訴買家如何解決問題。通常你只要提示買家選擇另一種資金來源即可。這份「[說明指南](#)」提供一些資訊，告訴你在 PayPal 傳回錯誤代碼 10486 後，如何將買家重新導向。

其他錯誤可能指出你的整合過程發生問題，例如在你的網站上接受無效的輸入，並透過要求訊息將該輸入內容傳遞給 PayPal。你需要使用 Sandbox 進行充分的測試，以避免網站在發佈上線後才發生問題。

### 逾時

若某項 API 操作的完成狀態不明，或者買家在 PayPal 執行完成前就離開用來接收回應的頁面，就會發生逾時的狀況。若還沒收到有效的 Transaction ID，指出 PayPal 已經接受購物款項，你就不應該寄出商品。

若完成狀態不明，比較保險的做法是再執行 API 操作一次。若你先前執行的 API 操作是 `DoExpressCheckout`，可以執行 `GetExpressCheckoutDetails`，然後查看 `CheckoutStatus` 欄位。若出現的值不是 `PaymentCompleted`，表示付款並未完成。這時，你應該在呼叫 `DoExpressCheckoutPayment` 或 `GetExpressCheckoutDetails` 並收到有效的 Transaction ID 後，再寄出商品。

## 紀錄 API 的操作

每次執行 PayPal API 操作時，都應該紀錄要求和回應訊息中包含的基本資訊。你必須紀錄回應訊息中的 Correlation ID，此 ID 指出 PayPal API 的對應操作；若你需要 PayPal [商店技術支援](#) 協助你處理特定交易，就必須將此 ID 提供給對方。

所有針對 PayPal API 操作的回應都包含可能可以協助除錯的資訊。除了紀錄回應訊息中包含的 Correlation ID 以外，你還可以紀錄其他資訊，例如 Transaction ID 和時間戳記，以便你在 PayPal 網站上或透過 API 檢視某一筆交易。你可以實作用來紀錄完整要求和回應內容的「冗長」機制，但請務必不要紀錄要求中包含的密碼。

## 將值編碼和解碼

你必須將在 API 操作中傳送的所有值編碼和解碼。不過，請只將值編碼，而不要將 NVP 中的名稱和 SOAP 中的標籤編碼。

你必須在送出要求給 PayPal 時，將所有要求欄位中的值編碼，並將回應中所有欄位的值解碼。你必須將個別的值編碼和解碼；請勿將整個訊息內容編碼和解碼。瀏覽器通常會嘗試將重新導向瀏覽器或導離瀏覽器的訊息編碼和解碼；然而，你必須確認編碼和解碼的過程正確，而且只針對欄位中的值進行。

## 1.7. Express Checkout 功能

Express Checkout 功能包括設定 Express Checkout API 要求、自訂 PayPal 結帳頁面的方式，以及你可以指定的其他設定。

### 自訂 Express Checkout 使用者介面

你可以自訂 PayPal Express Checkout 頁面的外觀。有些變更會改變結帳流程。

Express Checkout 提供在買家於結帳過程中登入 PayPal 時，如何呈現結帳頁面的選項。有些選項會讓 PayPal 頁面看起來像你網站中原本的頁面，使客戶有一致的視覺體驗：

- 要顯示的商店標誌

- 購物車內容檢視區域邊框使用的漸層填充色彩
- 顯示 PayPal 內容時使用的語言
- 你的客服電話號碼

**備註：**以上所有自訂選項都可以在你 PayPal 帳戶的個人檔案中設定。只有在你想要覆寫個人檔案所提供的預設設定時，才需要在 Express Checkout API 操作中設定這些選項。

其他選項會簡化結帳流程，方法是允許買家在 PayPal 上完成購物款項支付流程，或變更結帳過程中要顯示的資訊類型。你可以在 PayPal 的「審核」頁面上進行下列設定：

- 包含每項商品的詳細資料
- 包含稅、保險、運費和運費折扣
- 說明頁面上顯示的總金額是實際應付金額，還是尚未計入稅款和運費等的估計金額
- 顯示給買家的訊息，例如說明你提供哪些運送選項的訊息
- 允許買家向你說明關於交易的詳細指示
- 為購物款項指定發票號碼

在特殊情況下，你還可以使用其他功能：

- 顯示和使用運送地址
- 選擇是否使用禮盒
- 讓買家選擇是否要收到你的促銷訊息
- 意見調查問題

詳情請參閱「[自訂 Express Checkout](#)」一節。

## 結算和取得的購物款項

Express Checkout 能让你即時收取購物款項，或稍後（例如在你寄出商品時）再取得購物款項。

Express Checkout 提供幾種方式，讓你將交易設為稍後再取得購物款項。

你通常可以即時接受購物款項並寄出商品，這種情況稱為 **Sale**。除了即時接受購物款項的選擇以外，Express Checkout 也允許你先授權購物款項，稍後再扣取款項，這種方式稱為 **Authorization**。授權在某些情況下相當實用，例如當你想在尚未寄出商品時保留買家的交易款項，並在商品寄出後才扣取購物款項時。你可以在必要時重新授權一次，例如當你無法在授權後 3 天內寄出商品時。

Express Checkout 提供另一個稱為「**Order**」的選項，你可以在單一授權不敷使用時，使用此選項。你可以建立多個授權，並在同一筆訂單中同時取得這些授權。這在某些情況下相當實用，例如當你將同一筆訂單中的商品分次寄出，而你需要時在每次寄出商品後取得款項時。

詳情請參閱「[相關的 API 操作](#)」一節。

## 退款

你可以發放最多等於購物款項總金額的全額或部分退款。你可以退還一開始取得的購物款項，或在稍後進行結算時再退款。

若交易是在退款期限（通常是 60 天）過後發生，你就無法使用 [RefundTransaction API](#) 操作來退款。詳情請參閱 [RefundTransaction \(NVP | SOAP\) API](#) 操作。

## 定時定額付款

**Express Checkout** 提供 [定時定額付款](#)，能讓你按照既定時程來管理訂閱和其他購物款項。若你獲得 PayPal 的允許使用 [reference transactions](#)，就可以按照不同的時程來安排可變的購物款項金額。

當你為買家提供定時定額付款時，需要建立一個「定時定額付款約定」。此付款約定包含與定時定額付款相關的資訊，包括選用 [試用期](#) 的詳細資料，以及固定的付款期間；而試用期和付款期間都包含與付款頻率和金額相關的資訊，包括運費和稅款（若有的話）。建立付款約定後，PayPal 會根據開始付款日期、付款頻率和付款金額，自動將付款排入佇列。如此一來，除非付款約定到期、付款失敗次數過多而無法繼續使用定時定額付款，或你取消付款約定，否則款項會持續定時定額支付。

定時定額付款是買家與商店在 PayPal 上設定結帳同意書時，由買家同意使用。在 **Express Checkout** 中，結帳同意書可以在買家第一次購物前或購物時設定；無論是哪一種情況，付款約定都是在你呼叫 **Express Checkout API** 操作時設定。

## 使用 Reference Transactions 功能來設定定時定額付款

你也可以使用 [reference transaction](#) 功能來設定定時定額或不定時不定額付款，這種方式能讓你按照不同的時程來管理不同交易金額的支付情形。參考交易是可以促使後續交易發生的一種金融交易；例如買家可以在你的網站上購物，並在稍後使用 PayPal Transaction ID（此處亦稱為「*reference transaction ID*」）來展開另一筆交易。

詳情請參閱 [Express Checkout reference transactions](#)。

## 行動裝置適用的 Express Checkout

PayPal 支援幾種行動裝置適用的 **Express Checkout** 實作方式。你可以提供完整的行動版網站，並將結帳按鈕放在該網站上，也可以建立行動電話應用程式（mobile APP），並將結帳按鈕整合到該應用程式中。

在行動裝置上，**Express Checkout** 提供專為快速結帳而打造、適合較小的行動裝置螢幕和鍵盤的付款頁面。你可以為買家提供在你的網站上或 PayPal 上付款的結帳方式。

詳情請參閱「[在行動裝置上使用 Express Checkout](#)」。

### 透過 Express Checkout 使用多方收款功能

多方收款功能可讓買家在同一個 **Express Checkout** 結帳流程中支付購物款項給多家商店。Mobile **Express Checkout** 不提供這項功能。

在多方收款功能中，商店扮演的是賣場代管者的角色。以線上旅行社為例，買家除了購買機票，還會向旅行社網站上列出的飯店、租車公司和娛樂場所等多家商店進行預訂。只要透過 **Express Checkout** 實作多方收款功能，賣場代管者就能接受 **PayPal** 做為付款方式。此外，代管者也會在 **PayPal** 的「審核」頁面上向買家顯示經過整合的訂單，摘要列出各項支出、行程安排及其他相關資訊。買家會在「交易詳細資料」頁面上和電子郵件訊息中看到由服務供應商直接提供的旅遊資訊（包括取消行程的費用）。

詳情請參閱「[Express Checkout 多方收款](#)」。

### 詐騙控管篩選條件

**Fraud Management Filters (FMF)** 為你提供能辨識出疑似詐騙交易的「篩選條件」。這些篩選條件可分成兩種類型：

- 基本篩選條件，用來篩選來源國家/地區和交易值等資料。**PayPal** 為商業帳戶和 **PayPal Payments Pro**（前身為 **Website Payments Pro**）帳戶提供基本篩選條件。
- 進階篩選條件，用來篩選信用卡和地址資訊、高風險指示清單和其他交易特徵等資料。**PayPal Payments Pro** 商店可以進行升級，以使用這些篩選條件。

**備註：**進階篩選條件可能需要另外付費才能使用。

若要進一步了解 **Fraud Management Filters**，請參閱「[Fraud Management Filters 整合指南](#)」。

### 事件通知

在多數情況下，你可以使用 **PayPal API** 操作來判斷自己需要哪些交易相關資訊。但是在某些情況下，你必須設定 **Instant Payment Notifications (IPN)**，例如當你需要收到關於糾紛申訴及其調解結果等動作的自動通知訊息時。

**IPN** 是 **PayPal** 用來向你通知各種事件的一種訊息服務，這些事件包括：

- 即時付款事件，包括 **Express Checkout** 付款、**Adaptive Payment**、信用卡直接付款，以及授權事件（也就是尚未扣取購物款項的銷售事件）。
- **eCheck** 付款事件和「待處理」(pending)、「已完成」(completed) 或「遭拒」(denied) 等相關狀態，以及因其他原因（例如因疑似詐騙而受到審核）而待處理的付款事件。
- 定時定額付款事件和訂閱行動
- 與交易相關的交易退單、糾紛申訴、撤銷款項和退款等事件

若要進一步了解 **IPN**，請參閱「[Instant Payment Notification 指南](#)」

### 動態圖片概覽

動態圖片能讓 **PayPal** 為廣告活動或事件打造專屬的 **Express Checkout** 按鈕。**Mobile Express Checkout** 不支援這項功能。

當你參與 PayPal 廣告活動或事件時，PayPal 會自動更新圖片，以呈現廣告活動相關資訊。廣告活動結束後，PayPal 會恢復使用預設的圖片。在廣告活動開始前、進行期間或結束後，你都不用負責排程或變更網站的應用程式碼。只要你設定使用動態圖片，PayPal 就會代替你處理這些活動。

詳情請參閱「[Express Checkout 動態圖片整合](#)」。

### Express Checkout 即時更新

即時更新功能可讓你建立包含運送資訊的回應訊息，讓你藉此提供運送、保險和稅款的資訊。行動裝置版的 Express Checkout 不提供這項功能。

Instant Update API 是傳送到你的回傳伺服器的一種伺服器呼叫，會即時更新 PayPal 頁面，並能改善你在「檢視你的資訊」頁面上的 Express Checkout 結帳體驗。

Instant Update API 能讓你指定一個 URL，讓 PayPal 透過此 URL 呼叫你的回傳伺服器並提供買家的運送地址，如此你就能為買家提供更詳細的運送、保險和稅款資訊。

詳情請參閱「[實作 Instant Update API](#)」。

### 商店優惠券和忠實顧客優惠方案支援

Express Checkout 支援商店優惠券和忠實顧客優惠方案。詳情請參閱「[如何在 Express Checkout 中使用優惠券和忠實顧客優惠方案](#)」。

## 2. Express Checkout 使用者介面要求

你在整合 Express Checkout 時，必須符合 PayPal 的使用者介面要求。若要讓買家知道你的網站接受 PayPal，你必須將 PayPal 按鈕和標誌圖片放在網站結帳流程中。

PayPal 要求你使用由安全 PayPal 伺服器代管的「Check out with PayPal」按鈕和 PayPal Mark 圖片。當 PayPal 代管的圖片更新時，變更會自動出現在你的網站上。請勿複製並在你的伺服器本機上管理 PayPal 圖片，因為使用過時的 PayPal 按鈕和圖片會使買家對你的網站失去信任。

若要實作 Express Checkout，你必須在購物車頁面將其做為結帳選項，並在付款選項頁面將其做為付款選項。

請務必在你的網站上使用下列必要的 PayPal 圖片：

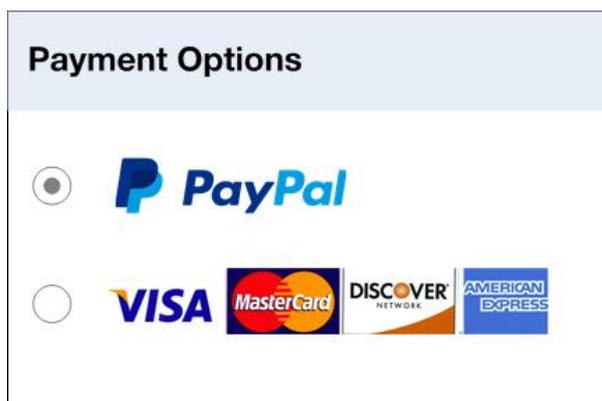
- **PayPal Mark 標誌**：放在付款方式頁面上。
- **PayPal 結帳按鈕**：放在購物車結帳頁面上。

### 2.1. PayPal Mark 標誌

若要取得用來加入 PayPal Mark 標誌的程式碼，請參閱 <https://www.paypal.com/brandassets/>。

#### 將 PayPal Mark 標誌加入你的付款方式頁面

若要將 PayPal 實作為付款選項，請將 PayPal Mark 圖片加入你的付款選項中。PayPal 建議你使用選項按鈕來讓買家選擇付款方式：



#### 從付款方式頁面開始 PayPal 結帳流程

在你的「付款方式」頁面上加入 PayPal Mark 做為付款選項之一，並將 PayPal 設為預設的選項。在此流程中，買家會從你的付款方式頁面展開 Express Checkout 結帳流程。

下圖顯示從你的付款方式頁面展開的 Express Checkout 結帳流程：



當買家選擇以 PayPal 支付購物款項時，就會展開 Express Checkout 結帳流程。買家登入 PayPal，並選擇交易資金來源和運送地址。接著，買家回到你的網站確認付款。此流程能讓買家在每次購物時，省去手動輸入付款詳細資料的步驟。不過，你也可以將「Check out with PayPal」按鈕放在比較前面的結帳步驟中，也就是放在購物車結帳頁面上，為買家節省更多時間。請參閱以下的「

從購物車結帳頁面開始 PayPal 結帳流程」一節。

## 2.2. PayPal 結帳按鈕

將「Check out with PayPal」按鈕放在你的購物車頁面上。



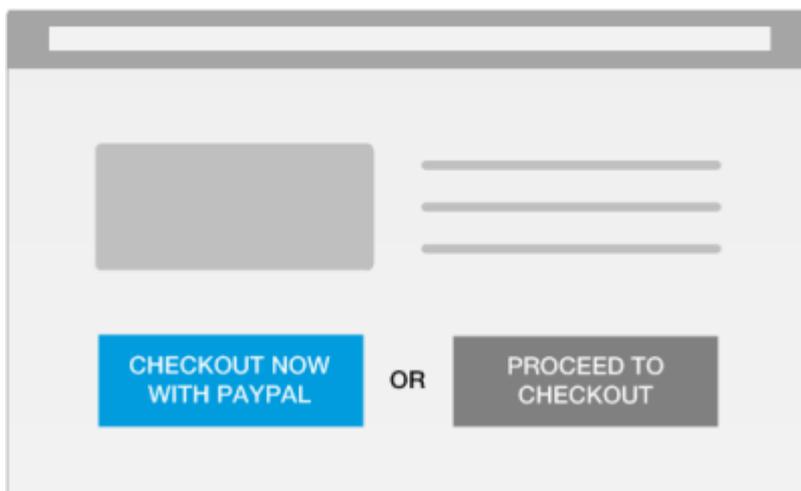
若要建立 Express Checkout 按鈕，請存取 Express Checkout 按鈕程式碼工具 (<https://www.paypal.com/us/webapps/mpp/logos-buttons>)。

### 將 PayPal 按鈕加入你的結帳頁面

「Check out with PayPal」按鈕應該：

- 隨時可以點按。
- 位於個人購物車結帳按鈕的正下方或旁邊。
- 在網站開始收集帳單或運送詳細資料，以及處理任何其他購物款項前顯示。
- 與一般結帳按鈕垂直對齊或水平對齊。
- 包含「[What is PayPal?](#)」(PayPal 是什麼?)連結。

若要了解如何將「Check out with PayPal」按鈕放到你的購物車結帳頁面上，請參閱下圖：



## 從購物車結帳頁面開始 PayPal 結帳流程

在你的購物車頁面中加入「Check out with PayPal」按鈕。買家只要按一下 Express Checkout 結帳流程。此付款流程比前文描述的流程快，因為它讓買家跳過用來收集運送和付款詳細資料的網站頁面，就能從你的購物車頁面直接展開 PayPal 付款。你不需要收集買家的運送資訊，因為 PayPal 會為你提供這筆資訊。

下圖顯示使用 Express Checkout 捷徑的結帳流程及好處：



當買家想在購物車結帳頁面上使用 PayPal 來結帳時，就會展開此流程。買家只要按一下「Check out with PayPal」按鈕，系統就會立刻將其重新導向 PayPal 網站。買家登入 PayPal 並檢視訂單後，就會回到你的網站最後一次確認訂單內容，並且可以在此步驟選擇是否要建立帳戶。

## 將 PayPal 按鈕加入結帳頁面的好處

使用「Check out with PayPal」*按鈕快速結帳*

你只要採用 [PayPal Mark 流程](#)，買家就可以在結帳流程的最後一個步驟，透過 PayPal 的安全網站付款給你，而且只要將「Check out with PayPal」按鈕放在比較前面的結帳步驟中，買家就不需要輸入運送和帳單資訊。這項整合能讓買家在不用手動輸入額外資料的情況下完成購物流程，因為 PayPal 會提供運送地址（非預設）、電子郵件和聯絡電話等詳細資料。買家只要檢查資訊是否正確，然後確認購買即可。如此一來就能簡化結帳流程，並協助提升網站和行動版網站的成交率。

你也可以選擇請買家在你的網站上輸入密碼（而不需要任何額外的資訊），以便建立帳戶。

## 3. 自訂 Express Checkout

你可以在 Express Checkout API 要求中設定選項，以變更結帳流程的外觀、行為和過程。

### 3.1. PayPal「審核」頁面上的訂單詳細資料

當買家登入 PayPal 以進行結帳時，你可以向買家顯示關於各項訂購商品的詳細資訊。PayPal 訂單詳細資料可以透過 53.0 版或較新版本的 API 來提供。

備註：`DoExpressCheckoutPayment` 要求包含與 `SetExpressCheckout` 中相同的訂單詳細資料。PayPal 建議你在兩個 API 呼叫中送出相同的參數。

下圖顯示你可以包含的所有詳細資料：

Descriptions	Amount
10% Decaf Kona Blend Coffee Item description: Size: 8.8-oz Item number: 623083 Item price: \$9.95 Quantity: 2	\$19.90
Coffee Filter Bags Item description: Size: Two 24-piece... Item number: 623084 Item price: \$39.70 Quantity: 2	\$79.40
Item total	\$99.30
Tax	\$2.58
Shipping and handling	\$5.99
Shipping discount	-\$3.00
Insurance	\$1.00
<b>Total \$105.87 USD</b>	

(1) – 商品名稱。商品名稱欄位可以指出特定商品，以便與訂單中列出的其他商品區別。

(2) – 商品說明。此欄位指出買家要購買的是多項商品中的哪一項。舉例來說，同一種商品可能有不同的尺寸。若買家知道某項商品的尺寸，就能判斷自己是否選到適合的商品。若商品是 eBay 上的拍賣商品，建議你在此欄位中加註「eBay 商品」字樣。

- (3) – 商品編號。每項商品都可以加上商品編號，以便進一步識別。若商品是 eBay 上的拍賣商品，建議你在此欄位中填寫 eBay 商品編號。
- (4) – 商品單價。此欄位指出一份商品的實際價格，可以是正數或負數，但不可以為 0。
- (5) – 商品份數。此欄位指出買家訂購的商品份數。PayPal 會將商品單價乘以商品份數，並將計算結果顯示在「金額」(6) 欄中。你也可以顯示關於訂單的其他詳細資訊：
- (7) – 商品總金額和稅款，分別是同一筆訂單中所有商品的總金額以及稅款。
- (8) – 運費和手續費，也就是運費加上手續費的總金額。  
備註：你必須自行決定實際的運費和手續費。
- (9) – 運費折扣。若買家享有運費折扣，此欄位中的值就會顯示為負值。
- (10) – 保險。若運送有附加保險，此欄位就會顯示保險費。
- (11) – 總計。此欄位顯示訂單的總金額，包括運費、手續費、稅款，以及與價格優惠相關的其他項目。  
備註：「輸入禮券、優惠券或折扣代碼」連結能讓買家兌換 PayPal 發行的禮券、優惠券或折扣代碼，但無法讓買家兌換你所發行的獎勵代碼。

### 給商店的特別說明

你可以允許買家將關於訂單的特別說明傳送給你。對想要購買客製化商品的買家而言，這項功能特別實用。買家也可能因為人不在國內，而希望告訴你晚一點再寄出商品。

備註：若要使用這項功能，請務必詳細閱讀買家傳送的說明。

這項功能會以連結的形式顯示在「檢視你的資訊」頁面上。當買家按一下「新增」，就會開啟「給賣家的備註」文字方塊；買家可以在這裡輸入給商店的特別說明，然後按一下「儲存」即可。特別說明會包含在針對 `GetExpressCheckoutDetails` 和 `DoExpressCheckoutPayment` 的回應中一起傳回給你。

### 將訂單詳細資料整合到 Express Checkout 購物流程中

若要將訂單詳細資料整合到結帳流程中，請將下列任何一個 Express Checkout 參數傳送給 `SetExpressCheckout`。

NVP 欄位	SOAP 欄位	說明和意見
<code>L_PAYMENTREQUEST_n_NAMEm</code>	<code>Name</code>	商品名稱。
<code>L_PAYMENTREQUEST_n_NUMBERm</code>	<code>Number</code>	商品編號。

L_PAYMENTREQUEST_n_DESCm	Description	商品說明。為了保留回溯相容性，DESC (NVP) 和 OrderDescription (SOAP) 欄位仍舊存在。不過，L_DESCn 和 Description 能讓你為買家購買的各項商品提供更精確的說明內容，例如「hiking boots」(登山鞋)或「cooking utensils」(野炊鍋)，而不是像「camping supplies」(露營用品)這種指涉範圍較廣的說明內容。
L_PAYMENTREQUEST_n_AMTm	Amount	商品單價。PayPal 會將商品單價乘以商品份數（下一個參數），並將計算結果顯示在購物車內容檢視區域的「金額」欄中。商品單價可以是正數或負數，但不得為 0。舉例來說，你可以填入負數來表示這筆訂單享有折扣。
L_PAYMENTREQUEST_n_QTYm	Quantity	商品份數。
PAYMENTREQUEST_n_ITEMAMT	ItemTotal	同一筆訂單中所有商品的總金額。
PAYMENTREQUEST_n_TAXAMT	TaxTotal	同一筆訂單中所有商品的總稅款。
PAYMENTREQUEST_n_SHIPPINGAMT	ShippingTotal	同一筆訂單中的運費總金額 (8)。PayPal 會計算運費總金額和手續費。雖然你稍後可能會變更這筆金額，但請盡量提供最準確的運費金額。
PAYMENTREQUEST_n_HANDLINGAMT	HandlingTotal	同一筆訂單中的手續費總金額。
PAYMENTREQUEST_n_SHIPDISCAMT	ShippingDiscount	同一筆訂單中的運費折扣。這筆金額應該設為負數。
PAYMENTREQUEST_n_INSURANCEAMT	InsuranceTotal	同一筆訂單中的運送保險總金額。

PAYMENTREQUEST\_n\_AMT

OrderTotal

訂單的總金額，包括運費、手續費、稅款，以及可用點數等任何其他價格優惠項目。

ALLOWNOTE

AllowNote

若你隨下列任兩個商品參數一同傳送通用訂單說明參數 (PAYMENTREQUEST\_n\_DESC)，訂單說明值就不會顯示。

- L\_PAYMENTREQUEST\_n\_NAMEm
- L\_PAYMENTREQUEST\_n\_NUMBERm
- L\_PAYMENTREQUEST\_n\_DESCm

下列範例說明如何在針對 SetExpressCheckout 的呼叫中設定商品參數。

要求參數

```
[requiredSecurityParameters]
&METHOD=SetExpressCheckout
&RETURNURL=http://...
&CANCELURL=http://...
&PAYMENTREQUEST_0_PAYMENTACTION=Sale
&L_PAYMENTREQUEST_0_NAME0=10% Decaf Kona Blend Coffee
&L_PAYMENTREQUEST_0_NUMBER0=623083
&L_PAYMENTREQUEST_0_DESC0=Size: 8.8-oz
&L_PAYMENTREQUEST_0_AMT0=9.95
&L_PAYMENTREQUEST_0_QTY0=2
&L_PAYMENTREQUEST_0_NAME1=Coffee Filter bags
&L_PAYMENTREQUEST_0_NUMBER1=623084
&L_PAYMENTREQUEST_0_DESC1=Size: Two 24-piece boxes
&L_PAYMENTREQUEST_0_AMT1=39.70
&L_PAYMENTREQUEST_0_QTY1=2
&PAYMENTREQUEST_0_ITEMAMT=99.30
&PAYMENTREQUEST_0_TAXAMT=2.58
```

```
&PAYMENTREQUEST_0_SHIPPINGAMT=3.00
&PAYMENTREQUEST_0_HANDLINGAMT=2.99
&PAYMENTREQUEST_0_SHIPDISCAMT=-3.00
&PAYMENTREQUEST_0_INSURANCEAMT=1.00
&PAYMENTREQUEST_0_AMT=105.87
&PAYMENTREQUEST_0_CURRENCYCODE=USD
&ALLOWNOTE=1
```

回應參數

```
[successResponseFields]
&TOKEN=EC-17C76533PL706494P
```

### 3.2. 提供禮品選項

你可以在 PayPal 上為買家提供禮品選項。若要使用這項功能，你必須實作商品詳細資料。

備註：禮品選項可以透過 61.0 版或較新版本的 API 來提供。

你可以啟用下列任何禮品選項：

- 送禮賀詞：這項功能會顯示一個文字方塊，讓買家在其中輸入送禮賀詞。
- 禮品收據：這項功能提供一個勾選方塊，讓買家勾選是否要包含禮品收據。
- 禮盒包裝：這項功能提供一個勾選方塊，讓買家勾選是否要使用禮盒包裝。禮盒包裝功能可以包含用來描述包裝方式的標籤，例如「禮盒加蝴蝶結」。你也可以選擇向買家顯示禮盒包裝所需的費用。

下列 `SetExpressCheckout` 的要求範例中設定了這些選項：

要求參數

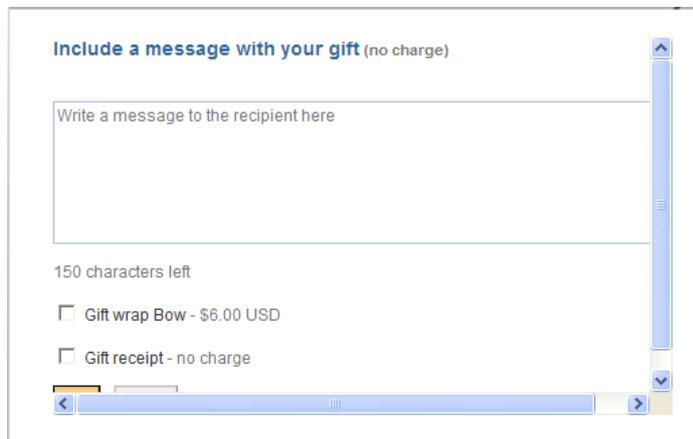
```
[requiredSecurityParameters]
&METHOD=SetExpressCheckout
&RETURNURL=https://...
&CANCELURL=https://...
&PAYMENTREQUEST_0_AMT=10.00
&PAYMENTREQUEST_0_CURRENCYCODE=USD
&PAYMENTREQUEST_0_PAYMENTACTION=Sale
```

```
&GIFTMESSAGEENABLE=1
&GIFTRECEIPTENABLE=1
&GIFTWRAPENABLE=1
&GIFTWRAPNAME="Bow and Ribbon"
&GIFTWRAPAMOUNT=6.00
```

回應參數

```
[successResponseFields]
&TOKEN=EC-17C76533PL706495P
```

下圖顯示買家看到的禮品選項畫面。



備註：你也可以在個人檔案中設定此選項。詳情請參閱「[商店設定和管理指南](#)」。

### 3.3. 詢問買家是否同意收到促銷電子郵件

你可以在 PayPal 頁面上詢問買家是否同意收到促銷電子郵件。PayPal 會透過針對 `GetExpressCheckoutDetails` 和 `DoExpressCheckoutPayment` 的回應，傳回買家的電子郵件地址。

備註：詢問買家是否同意收到促銷電子郵件的功能，可以透過 61.0 版或較新版本的 API 來提供。

若要取得買家的電子郵件地址，請在 `SetExpressCheckout` 呼叫中將 `BUYEREMAILOPTINENABLE` 欄位設為 `1`。

要求參數

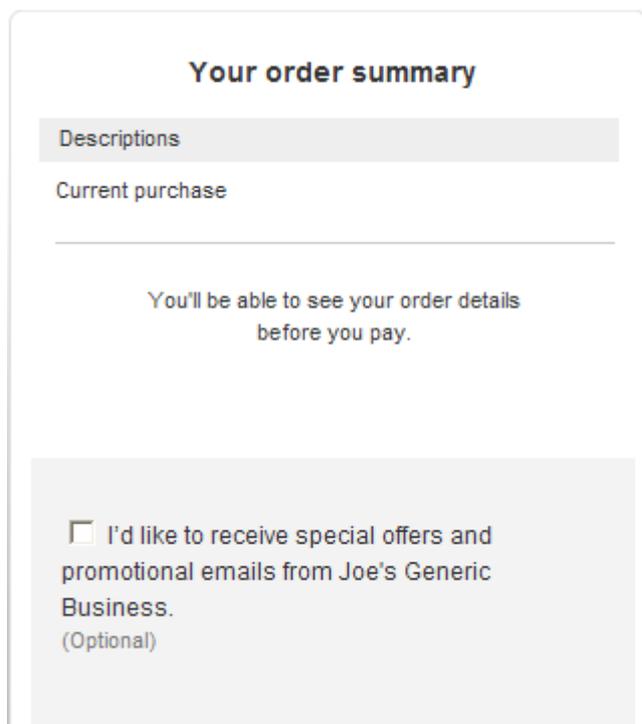
```
[requiredSecurityParameters]
&METHOD=SetExpressCheckout
&RETURNURL=https://...
&CANCELURL=https://...
```

```
&PAYMENTREQUEST_0_AMT=10.00
&PAYMENTREQUEST_0_CURRENCYCODE=USD
&PAYMENTREQUEST_0_PAYMENTACTION=Sale
&BUYEREMAILOPTINENABLE=1
```

回應參數

```
[successResponseFields]
&TOKEN=EC-17C76533PL706495P
```

下圖顯示買家在購物車內容檢視區域中看到的詢問畫面。



備註：你也可以在個人檔案中設定這項功能。詳情請參閱「[商店設定和管理指南](#)」。

### 3.4. 覆寫客服電話號碼

你可以在個人檔案中設定客服電話號碼，以在 Express Checkout 頁面上向買家顯示該號碼。若你要覆寫此號碼，可以在 `SetExpressCheckout` 要求中指定另一個號碼。

備註：這項功能可以透過 61.0 版或較新版本的 API 來提供。

只要向買家顯示客服電話號碼，你就能透過電話快速回應買家的問題。若要以 Express Checkout 頁面上顯示的其他號碼覆寫你在個人檔案中設定的客服電話號碼，請在

`SetExpressCheckoutSetExpressCheckout` 的呼叫中設定 `CUSTOMERSERVICENUMBER` 欄位。

要求參數

```
[requiredSecurityParameters]
&METHOD=SetExpressCheckout
&RETURNURL=https://...
&CANCELURL=https://...
&PAYMENTREQUEST_0_AMT=10.00
&PAYMENTREQUEST_0_CURRENCYCODE=USD
&PAYMENTREQUEST_0_PAYMENTACTION=Sale
&CUSTOMERSERVICENUMBER=1-800-FLOWERS
```

回應參數

```
[successResponseFields]
&TOKEN=EC-17C76533PL706494P
```

備註：若要進一步了解如何在 PayPal 的「個人檔案」頁面上設定客服電話號碼，請參閱「[商店設定和管理指南](#)」。

### 3.5. 加入意見調查問題

你可以在 PayPal 頁面加入一則意見調查問題。PayPal 會透過針對 `GetExpressCheckoutDetails` 和 `DoExpressCheckoutPayment` 的回應，傳回買家選擇的答案選項。

備註：這項功能可以透過 61.0 版或較新版本的 API 來提供。

意見調查問題會以文字字串的格式，顯示在 PayPal 的「檢視你的資訊」頁面上。買家只要選擇下拉式選單中的答案選項，就可以回答問題。

- 若要顯示意見調查問題，請在 `SetExpressCheckout` 的呼叫中將 `SURVEYENABLE` 欄位設為 `1`。
- 意見調查問題請設定 `SURVEYQUESTION` 字串。
- 至少提供兩個可讓買家擇一的 `L_SURVEYCHOICEn` 答案選項。

下列的要求範例中設定了這些欄位：

要求參數

```
[requiredSecurityParameters]
&METHOD=SetExpressCheckout
&RETURNURL=https://...
&CANCELURL=https://...
```

```
&PAYMENTREQUEST_0_AMT=10.00  
&PAYMENTREQUEST_0_CURRENCYCODE=USD  
&PAYMENTREQUEST_0_PAYMENTACTION=Sale  
&SURVEYENABLE=1  
&SURVEYQUESTION="How did you hear about us?"  
&L_SURVEYCHOICE0="friend"  
&L_SURVEYCHOICE1="newspaper ad"
```

*回應參數*

```
[successResponseFields]  
&TOKEN=EC-17C76533PL706495P
```

買家會在購物車內容檢視區域中看到意見調查問題。

備註：你也可以在個人檔案中設定這項功能。詳情請參閱「[商店設定和管理指南](#)」。

### 3.6. PayPal 頁面風格

你可以自訂頁面風格或個別頁面的風格特徵，以改變 PayPal 頁面的整體外觀。

你在個人檔案中自訂頁面風格，然後在呼叫 `SetExpressCheckout` 時傳送完成的頁面風格名稱。個別頁面風格特徵通常也可以在你的個人檔案中自訂。不過你也可以呼叫 `SetExpressCheckout`，並動態傳送個別網頁的特徵。

#### 自訂頁面風格

當買家登入 PayPal 進行結帳時，你可以讓買家所看見 PayPal 頁面的外觀看起來和你網站上的網頁類似。你可以自訂這些頁面特徵中的任何項目，並將結果儲存為「頁面風格名稱」。你最多可以定義三個獨特的頁面風格名稱，其中可以指定的特徵如下所示：

- 標題圖片
- 標題邊框色彩
- 標題底色
- 頁面底色

若要了解如何自訂頁面風格和建立頁面風格名稱，請參閱「[商店設定和管理指南](#)」。

若要在針對 `SetExpressCheckout` 的呼叫中設定自訂頁面風格：

1. 在 `SetExpressCheckout` 的呼叫中包含頁面風格名稱 `PAGESTYLE` 參數（選擇性）。

2. 將帳戶中定義的頁面風格名稱填入 `PAGESTYLE` 參數中。

下列範例將 `PAGESTYLE` 設為頁面風格名稱：

要求參數

```
[requiredSecurityParameters]
&METHOD=SetExpressCheckout
&RETURNURL=https://...
&CANCELURL=https://...
&PAYMENTREQUEST_0_AMT=10.00
&PAYMENTREQUEST_0_CURRENCYCODE=USD
&PAYMENTREQUEST_0_PAYMENTACTION=Sale
&PAGESTYLE=TestMerchant
```

回應參數

```
[successResponseFields]
&TOKEN=EC-17C76533PL706494P
```

## 個別頁面風格特徵

你通常是在「帳戶個人檔案」的自訂付款頁面中，為 PayPal 頁面建立自訂頁面風格。若你不想使用「帳戶個人檔案」選項，可以使用程式碼中的變數來指定這些個別頁面風格特徵：

- 標誌圖片：標誌圖片的連結 URL。
- 購物車內容檢視區域邊框色彩：你的主要指定色彩。PayPal 會將你指定的色彩與白色融合成漸層色彩，並用來填滿購物車內容檢視區域的邊框。

若要定義商店標誌圖片：

1. 製作寬 90 像素、高 60 像素的標誌圖片，並將其儲存為有效的影像格式，例如 .gif、.jpg 或 .png。
2. 將圖片的連結 URL 儲存在安全 (https) 伺服器上，如此買家的網頁瀏覽器才不會顯示付款頁面包含不安全項目的訊息。
3. 在針對 `LOGOIMG` 參數使用。 `SetExpressCheckout` .

`SetExpressCheckout` 的呼叫中，將該 URL 指派給在針對 `CARTBORDERCOLOR` 參數設為該色彩的 6 位數 Hex 值。下列範例自訂了標題圖片，並加入了邊框色彩。 `SetExpressCheckout` . The follow `SetExpressCheckout` 的呼叫中，將 `ers]`

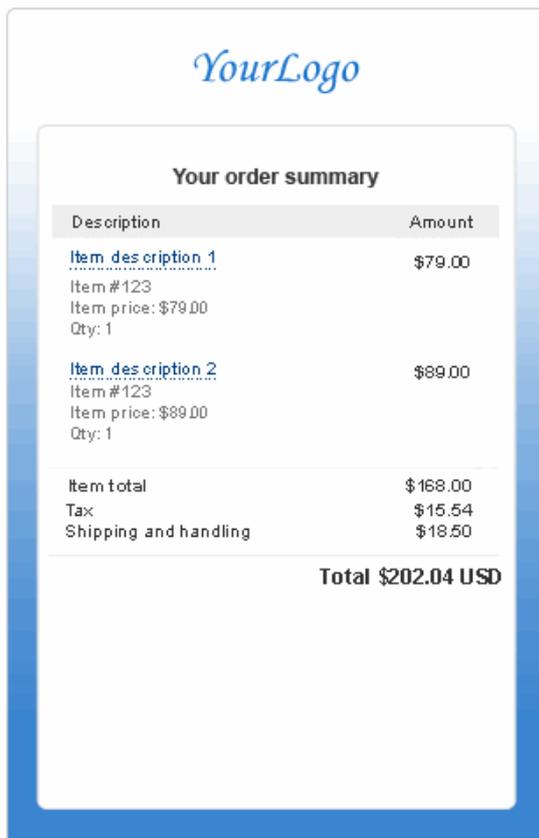
```
&METHOD=SetExpressCheckout
&RETURNURL=https://...
&CANCELURL=https://...
&PAYMENTREQUEST_0_AMT=10.00
&MAXAMT=...
&PAYMENTREQUEST_0_PAYMENTACTION=Sale
&LOGOIMG=https://www.../YourLogo.gif
&CARTBORDERCOLOR=0000CD
```

回應參數

```
[successResponseFields]
```

```
&TOKEN=EC-17C76533PL706494P
```

下圖顯示包含自訂標誌和邊框色彩的購物車內容檢視區域，這些項目是在針對 `SetExpressCheckout` 的呼叫中設定。



### 3.7. 變更地區設定

你可以變更 PayPal 頁面的地區設定，使其符合你的網站使用的語言。

若要變更顯示在 PayPal 頁面上的語言，請將 `SetExpressCheckout` 呼叫的 `LOCALECODE` 參數設為下列其中一個允許使用的值：

- `AU` - 澳洲
- `AT` - 奧地利
- `BE` - 比利時
- `BR` - 巴西
- `CA` - 加拿大 `CH` - 瑞士
- `CN` - 中國
- `DE` - 德國
- `ES` - 西班牙
- `GB` - 英國
- `FR` - 法國
- `IT` - 義大利
- `NL` - 荷蘭
- `PL` - 波蘭
- `PT` - 葡萄牙
- `RU` - 俄羅斯
- `US` - 美國
- PayPal 也針對某些國家專屬的語言，提供以下由 5 個字元組成的代碼：
  - `da_DK` - 丹麥語（僅限丹麥）
  - `he_IL` - 希伯來語（全部）
  - `id_ID` - 印尼語（僅限印尼）
  - `ja_JP` - 日語（僅限日本）
  - `no_NO` - 挪威語（僅限挪威）
  - `pt_BR` - 巴西葡萄牙語（僅限葡萄牙和巴西）
  - `ru_RU` - 俄語（僅限立陶宛、拉脫維亞和烏克蘭）
  - `sv_SE` - 瑞典語（僅限瑞典）
  - `th_TH` - 泰語（僅限泰國）
  - `tr_TR` - 土耳其語（僅限土耳其）
  - `zh_CN` - 簡體中文（僅限中國）
  - `zh_HK` - 繁體中文（僅限香港）
  - `zh_TW` - 繁體中文（僅限台灣）

下列範例將 `LOCALCODE` 設為 `ES`（西班牙）。

要求參數

```
[requiredSecurityParameters]
```

```
&METHOD=SetExpressCheckout
```

```
&RETURNURL=https://...
```

```
&CANCELURL=https://...  
&PAYMENTREQUEST_0_AMT=10.00  
&PAYMENTREQUEST_0_CURRENCYCODE=EUR  
&PAYMENTREQUEST_0_PAYMENTACTION=Sale  
&LOCALECODE=ES
```

回應參數

```
[successResponseFields]  
&TOKEN=EC-17C76533PL706494P
```

下圖為當 PayPal 登入頁面之 LOCALECODE 設為 `ES` 時的樣子：

### 3.8. 運送地址選項

你可以在 `SetExpressCheckout` 要求中指定使用下列選項：

- 要求使用經過確認的地址。
- 不要在審核頁面上顯示運送地址。
- 在審核頁面上顯示替代地址。

## 經過確認的地址

「經過確認的地址」是經過 PayPal 確認屬於 PayPal 帳戶持有人的運送地址。若要受到「PayPal 賣家交易安全保障」的保護，你必須要求買家提供經過確認的運送地址。

備註：由於許多買家偏好讓商品寄到未經確認的地址（例如寄送禮物給某人），因此除非你販售的是高風險商品，否則 PayPal 不建議你要求買家提供經過確認的地址。若你還是希望買家提供經過確認的地址，那麼請勿設定 `ADDROVERRIDE`。

若想要買家提供經過確認的運送地址，請確認該地址與 PayPal 中儲存的地址相同。你可以在你的帳戶個人檔案中確認這一點，如「[商店設定和管理指南](#)」中所述。另外，你也可以在針對 `SetExpressCheckout` 的呼叫中設定旗標，如下所示：

1. 在針對 `SetExpressCheckout` 的呼叫中包含 `REQCONFIRMSHIPPING` 參數（選擇性）。
2. 將 `REQCONFIRMSHIPPING` 設為 `1`。

下列範例說明如何要求運送地址必須經過確認。

備註：`REQCONFIRMSHIPPING` 的值會覆寫你在商店帳戶個人檔案中的設定。

### 要求參數

```
[requiredSecurityParameters]
&METHOD=SetExpressCheckout
&RETURNURL=https://...
&CANCELURL=https://...
&PAYMENTREQUEST_0_AMT=10.00
&PAYMENTREQUEST_0_PAYMENTACTION=Sale
&REQCONFIRMSHIPPING=1
&PAYMENTREQUEST_0_SHIPTOSTREET=1 Main St
&PAYMENTREQUEST_0_SHIPTOCITY=San Jose
&PAYMENTREQUEST_0_SHIPTOSTATE=CA
&PAYMENTREQUEST_0_SHIPTOCOUNTRYCODE=US
&PAYMENTREQUEST_0_SHIPTOZIP=95131
&PAYMENTREQUEST_0_SHIPTOPHONENUM=408-967-4444
```

### 回應參數

```
[successResponseFields]
&TOKEN=EC-6UA07551EA393551U
```

## 避免顯示買家的運送地址

你可以避免在 PayPal 頁面上顯示買家的運送地址。建議你在下列情況下再這樣做：

- 你銷售的產品或服務不需要運送時。
- 你希望完全由自己處理地址的問題，而不希望讓買家從其 PayPal 通訊錄中選擇地址時。

若要避免顯示買家的運送地址，請在針對 `NOSHIPPING` 參數設為 `1`。如此一來，`SetExpressCheckout` 的呼叫中就不會顯示運送地址。下列範例避免運送地址顯示在頁面上。

### 要求參數

```
[requiredSecurityParameters]
&METHOD=SetExpressCheckout
&RETURNURL=https://...
&CANCELURL=https://...
&PAYMENTREQUEST_0_AMT=10.00
&PAYMENTREQUEST_0_CURRENCYCODE=USD
&PAYMENTREQUEST_0_PAYMENTACTION=Sale
&NOSHIPPING=1
&PAYMENTREQUEST_0_SHIPTONAME=J Smith
&PAYMENTREQUEST_0_SHIPTOSTREET=1 Main St
&PAYMENTREQUEST_0_SHIPTOCITY=San Jose
&PAYMENTREQUEST_0_SHIPTOSTATE=CA
&PAYMENTREQUEST_0_SHIPTOCOUNTRYCODE=US
&PAYMENTREQUEST_0_SHIPTOZIP=95131
&PAYMENTREQUEST_0_SHIPTOPHONENUM=408-967-4444
```

### 回應參數

```
[successResponseFields]
&TOKEN=EC-17C76533PL706494P
```

當 `NOSHIPPING` 設為 `1` 時，PayPal 的「檢視你的資訊」頁面就不會顯示運送地址。

## 覆寫運送地址

你可以覆寫買家儲存在 PayPal 的運送地址，例如在你的網站已經要求買家填寫運送地址的情況下進行覆寫運送地址。若覆寫儲存在 PayPal 中的運送地址，就會以你在針對 `SetExpressCheckout` 的呼叫中指定的運送地址取代 PayPal 中的地址。買家無法編輯遭到覆寫的地址。

備註：若你想要覆寫地址，PayPal 建議你不要按照先前「[經過確認的地址](#)」一節中的說明來要求買家提供經過確認的地址。

若要覆寫運送地址：

1. 在針對 `SetExpressCheckout` 的呼叫中將 `ADDROVERRIDE` 參數設為 1。
2. 在針對 `SetExpressCheckout` 的呼叫中，將下列運送地址參數設為你想要為新地址使用的地址值。
  - `PAYMENTREQUEST_0_SHIPTONAME`
  - `PAYMENTREQUEST_0_SHIPTOSTREET`
  - `PAYMENTREQUEST_0_SHIPTOCITY`
  - `PAYMENTREQUEST_0_SHIPTOSTATE` (選用)
  - `PAYMENTREQUEST_0_SHIPTOCOUNTRYCODE`
  - `PAYMENTREQUEST_0_SHIPTOZIP`
  - `PAYMENTREQUEST_0_SHIPTOSTREET2` (選用)

下列範例以所顯示的地址值覆寫了運送地址。

要求參數

```
[requiredSecurityParameters]
&METHOD=SetExpressCheckout
&RETURNURL=https://...
&CANCELURL=https://...
&PAYMENTREQUEST_0_AMT=10.00
&PAYMENTREQUEST_0_CURRENCYCODE=USD
&PAYMENTREQUEST_0_PAYMENTACTION=Sale
&ADDROVERRIDE=1
&PAYMENTREQUEST_0_SHIPTOSTREET=1 Second St
&PAYMENTREQUEST_0_SHIPTOSTREET2=Ste 210
&PAYMENTREQUEST_0_SHIPTOCITY=San Jose
&PAYMENTREQUEST_0_SHIPTOSTATE=CA
&PAYMENTREQUEST_0_SHIPTOCOUNTRYCODE=US
&PAYMENTREQUEST_0_SHIPTOZIP=95131
```

```
&PAYMENTREQUEST_0_SHIPTOPHONENUM=408-967-4444
```

回應參數

```
[successResponseFields]
```

```
&TOKEN=EC-57K68322WE343022B
```

PayPal 的「檢視你的資訊」頁面顯示你在 `SetExpressCheckout` 要求中指定的運送地址參數值。

### 3.9. 自動填寫運送和聯絡資訊

PayPal 能根據針對 `SetExpressCheckout` 的呼叫中包含的買家運送聯絡資訊，替買家自動填寫表單欄位。

當你透過針對 `SetExpressCheckout` 的呼叫傳送買家的運送地址、電話號碼和電子郵件地址時，PayPal 就會在 PayPal 的「選擇付款方式」頁面上，將這些資訊自動填入金融卡或信用卡表單欄位中。

針對 `SetExpressCheckout` 的呼叫傳送完成後，系統就會將買家重新導向 PayPal。在「選擇付款方式」頁面上，擁有 PayPal 帳戶的買家可以使用電子郵件地址和密碼登入 PayPal。沒有 PayPal 帳戶的買家則可以使用金融卡或信用卡來支付購物款項，並自行填寫運送和聯絡資訊。

請參閱下列的 `SetExpressCheckout` 範例：

要求參數

```
[requiredSecurityParameters]
```

```
&METHOD=SetExpressCheckout
```

```
&RETURNURL=https://...
```

```
&CANCELURL=https://...
```

```
&PAYMENTREQUEST_0_AMT=10.00
```

```
&PAYMENTREQUEST_0_PAYMENTACTION=Sale
```

```
&PAYMENTREQUEST_0_SHIPTOSTREET=1 Main Street
```

```
&PAYMENTREQUEST_0_SHIPTOCITY=San Jose
```

```
&PAYMENTREQUEST_0_SHIPTOSTATE=CA
```

```
&PAYMENTREQUEST_0_SHIPTOCOUNTRYCODE=US
```

```
&PAYMENTREQUEST_0_SHIPTOZIP=95131
```

```
&PAYMENTREQUEST_0_EMAIL=jsmith01@example.com
```

```
&PAYMENTREQUEST_0_SHIPTOPHONENUM=408-559-5948
```

回應參數

```
[successResponseFields]
```

```
&TOKEN=EC-6UA07551EA393551U
```

下圖顯示「使用金融卡或信用卡付款」區段中的買家運送和聯絡資訊欄位填妥後的畫面。

▶ Pay with my PayPal account

▼ Pay with a debit or credit card  
If you don't have a PayPal account

Country

Credit card number

Payment types    

Expiration date mm / yy  /

CSC   
[What is this?](#)

Billing information J Smith  
1 Main St  
San Jose, CA 95131  
United States  
[Change](#)

Contact information 408-559-5948  
jsmith01@example.com  
[Change](#)

▶ Save your information with PayPal

This transaction will appear on your statement as JOESGENERIC.

[Review and Continue](#)

*Secure payments by* **PayPal**

[Cancel and return to Joe's Generic Business](#)

### 3.10. 買家在 PayPal 上付款

有了 Express Checkout，你就可以縮短結帳流程，讓買家在 PayPal 上完成購物流程。接著，你可以略過訂單確認頁面。

選擇付款方式通常是買家完成購物前的最後一個步驟。若你在買家從 PayPal 回到你的網站後不需要收集額外的資訊，就可以讓買家略過你網站上的確認訂單頁面。若你收集到的額外資訊對付款並沒有影響，PayPal 建議你在買家完成購物流程「之後」再收集這些額外資訊。

你將使用者重新導向 PayPal 時使用的 `useraction` URL 參數，會決定買家將在 PayPal 或在你的網站上完成付款。若你將 `useraction` 設為 `commit`，PayPal 就會將「檢視你的資訊」頁面上的按鈕文字設為「Pay Now」。這段文字能讓買家知道，一旦點擊這個按鈕，付款就會完成。



PayPal 將買家重新導向你的網站後，請呼叫 `GetExpressCheckoutDetails` 和 `DoExpressCheckoutPayment`，讓 PayPal 順利完成付款流程。你不需要等買家做出互動動作，就可以呼叫 `DoExpressCheckoutPayment`。請使用 `GetExpressCheckoutDetails` 回應中的資訊來填寫你的訂單確認頁面。

在 Express Checkout 的重新導向 URL 中設定讓買家在 PayPal 頁面上完成付款

下列範例是讓買家在 PayPal 上付款的重新導向程式碼：

```
https://www.paypal.com/cgi-bin/webscr?cmd=_express-checkout&useraction=commit&token=valueFromSetExpressCheckoutResponse
```

## 4. 取得 API 憑證 (Credentials)

若你要使用 PayPal API，就必須擁有 API 憑證，藉此表示你是有權執行多種 PayPal API 的商業帳戶持有人。雖然使用電子簽章 (signature) 或證書 (certificate) 都可以，PayPal 還是建議使用電子簽章。

 重要：即使你可以擁有電子簽章和證書，但兩者不能同時使用。

若要進一步了解如何建立標準 API 憑證，請參閱下一個小節。

### 4.1. 建立和管理標準 API 憑證

呼叫 PayPal 標準 API 時，必須使用一組「API 憑證」來授權每個要求。PayPal 會將一組 API 憑證與特定 PayPal 帳戶相關聯，而你可以為任何 PayPal 商業帳戶產生憑證。

本指南說明在針對實際 PayPal 環境執行 API 呼叫時，如何建立所需的憑證。若要了解如何產生和使用 Sandbox 測試用憑證，請參閱「[測試標準 API 呼叫](#)」。

### 4.2. 憑證類型

標準 API 支援兩種不同類型的憑證：

- **電子簽章 (Signature) API 憑證**
- **證書 (Certificate) API 憑證**

每個憑證組都包含三個憑證值：

憑證組別	憑證名稱
「電子簽章」組	<ul style="list-style-type: none"><li>• API 用戶名稱</li><li>• API 密碼</li><li>• 電子簽章</li></ul>
「憑證」組	<ul style="list-style-type: none"><li>• API 用戶名稱</li><li>• API 密碼</li><li>• 憑證</li></ul>

雖然兩組憑證都可以用來授權給標準 API 平台，但為了方便使用，PayPal 建議你使用電子簽章憑證。不過在某些情況下，你還是有可能需要使用憑證來授權 API 呼叫。若要進一步了解憑證，請參閱後面的「[管理憑證](#)」小節。

**備註：**你將呼叫授權給「*Adaptive*」API（Adaptive Account、Adaptive Payment、Invoicing 服務或 Permissions 服務）時，是使用相同的憑證。不過 Adaptive 平台除了你的電子簽章或憑證憑證值以外，也會要求你提供 appID。

## 4.3. 建立標準 API 憑證

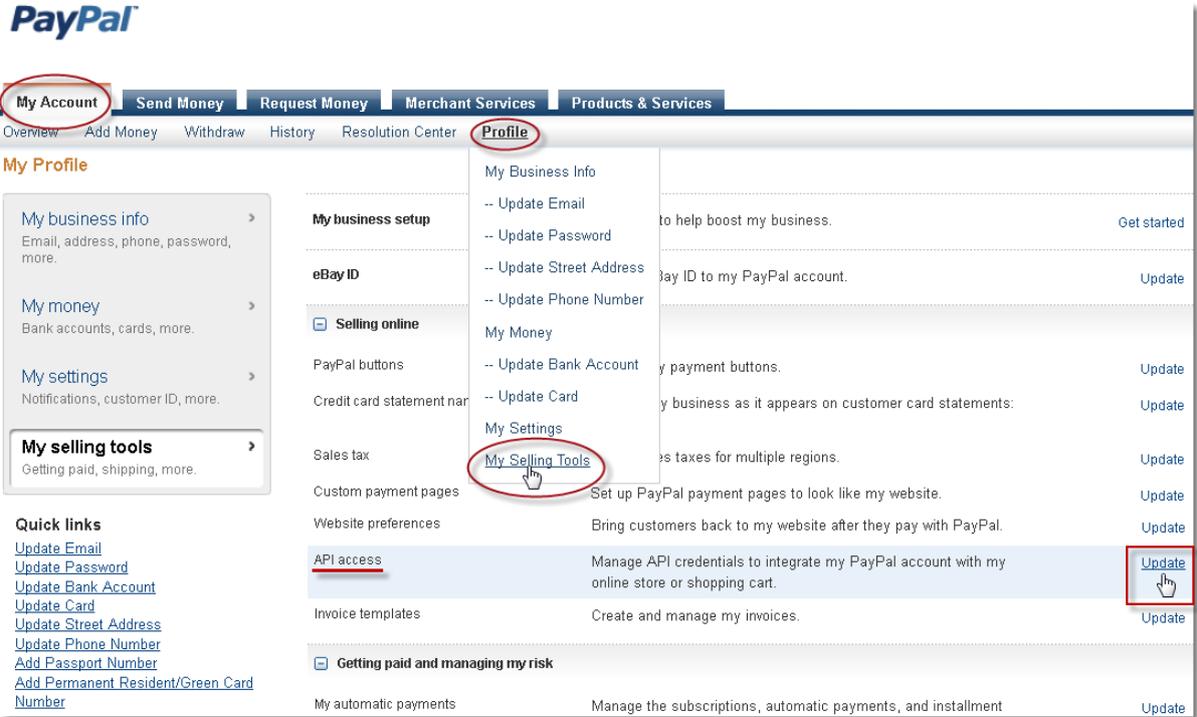
按照下列步驟從 PayPal.com 網站建立實際標準 API 憑證：

### 1. 登入 PayPal.com

你必須擁有 PayPal 商業帳戶，才能製作針對實際 PayPal 伺服器的呼叫。使用商業帳戶的登入帳號與密碼來登入 PayPal。

### 2. 導覽至「API 存取」頁面

選擇「我的帳戶」>「個人檔案」>「我的銷售工具」，然後按一下「更新（API 存取）」連結：

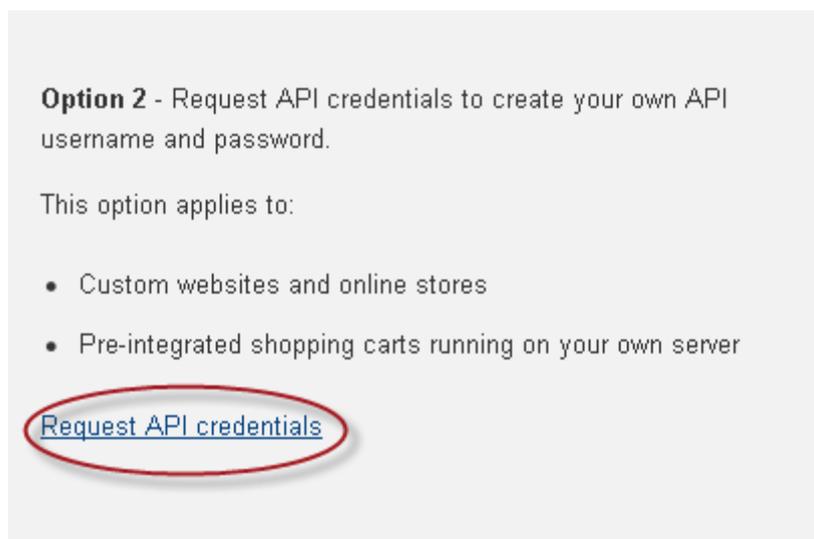


The screenshot shows the PayPal 'My Profile' page. The 'My Account' tab is selected. The 'Profile' sub-tab is active. The 'My selling tools' section is expanded, showing a dropdown menu with 'My Selling Tools' selected. The 'API access' section is visible, with an 'Update' button highlighted in a red box.

Section	Description	Action
My business setup	My business info	Get started
	-- Update Email	
	-- Update Password	
eBay ID	-- Update Street Address	Update
	-- Update Phone Number	
Selling online	My Money	
PayPal buttons	-- Update Bank Account	Update
Credit card statement name	-- Update Card	Update
	My Settings	
Sales tax	My Selling Tools	Update
Custom payment pages	Set up PayPal payment pages to look like my website.	Update
Website preferences	Bring customers back to my website after they pay with PayPal.	Update
API access	Manage API credentials to integrate my PayPal account with my online store or shopping cart.	Update
Invoice templates	Create and manage my invoices.	Update
Getting paid and managing my risk		
My automatic payments	Manage the subscriptions, automatic payments, and installment	Update

### 3. 產生憑證組

1. 在 API 存取頁按一下「**要求 API 憑證**」：



備註: 如果你已經產生過一組憑證, 你將必須刪除現有的憑證組, 才能產生新的憑證組。

2. 選擇「**要求 API 電子簽章**」, 然後按一下「**同意並提交**」, 以產生一組電子簽章憑證。

以下範例顯示一組電子簽章憑證 (證書憑證範例請參閱下一節) :

**PayPal**

My Account | Send Money | Request Money | Merchant Services | Products & Services

Overview | Add Money | Withdraw | History | Resolution Center | Profile

**View or Remove API Signature** [Back to My Profile](#)

**Developers:** Do not share your credential information with anyone. Store in a secure location with limited access.

**For preconfigured shopping carts:** Copy and paste the API username, password, and signature into your shopping cart configuration or administration screen.

**For building custom shopping carts:** Store the following credential information in a secure location with limited access.

Credential	API Signature
API Username	krich_api1.ebay.com
API Password	R2CV5XCLEYEMQ7RH
Signature	ACxwNDtvkO2nsXs2fa4ldAjRq98TA204gsd2ODrvR4mTZY.e.nYU15Sh
Request Date	Mar 29, 2013 08:48:34 PDT

#### 4.4. 管理證書憑證

證書憑證和上述的電子簽章憑證很像，不過其產生和使用方式稍有不同。

**備註：**若你的 API 憑證即將到期，請跳到後面的「

[更新 API 證書憑證](#)」小節。

API 證書憑證的建立方式與電子簽章憑證很像。下列步驟說明兩者之間的差異：

1. 在「[要求 API 憑證](#)」頁面上，選擇「[要求 API 證書憑證](#)」選項按鈕，然後按一下「[同意並提交](#)」按鈕（詳情請參閱以上的[步驟 3](#)）。

**Request API Credentials** [Back to My Profile](#)

API credentials consist of three elements:

- An API username
- An API password
- Either an API signature or an API SSL client-side certificate

If you're using a shopping cart or solution provider, ask whether you need an API signature or a certificate.

**Request API signature** if your shopping cart or solution provider has asked for an API username, password, and signature, or if you're developing a custom shopping cart.

**Request API certificate** if your shopping cart or solution provider requires a file-based certificate.

Need help deciding which credential is right for your needs? [Learn more](#)

By clicking **Agree and Submit**, I agree to the [API License Agreement and Terms of Use](#).

**Agree and Submit** Cancel

如此就會顯示「[下載或移除 API 證書憑證](#)」頁面。

2. 按一下[下載證書憑證](#)。

憑證會下載成一個名為「cert\_key\_perm.txt」的檔案。請務必將這個檔案儲存在安全的位置。

PayPal 會將 API 證書憑證檔案的格式設定為 PEM。檔案中包含你的「公用憑證」和相關的「私人金鑰」。雖然 PEM 憑證不是採用人類可以讀取的格式，但這支檔案並未經過加密。

## 更新 API 證書憑證

API 證書憑證的效期是從建立日期算起後 10 年，過後就會失效。為了預防 API 服務中斷，你必須在憑證到期前進行更新。

憑證更新流程中會產生新的憑證，你可以安裝新的憑證，以取代任何即將到期的憑證。

1. 登入你的 PayPal 帳戶，並前往「**管理 API 證書憑證**」頁面。  
若要這樣做，請選擇「**我的帳戶**」>「**個人檔案**」>「**我的銷售工具**」、按一下「**更新 (API 存取)**」連結，然後按一下「**檢視 API 證書憑證**」。
2. 在「**管理 API 憑證**」頁面上查看 API 憑證的狀態是「**有效**」或「**即將到期**」。

My Account Send Money Request Money Merchant Services Products & Services

Overview Add Money Withdraw History Resolution Center Profile

### Manage API certificate [Back to My Profile](#)

**Developers:** Do not share your credential information with anyone. Store in a secure location with limited access.

**Warning:** Your API certificate is about to expire. Please renew your certificate and download the new one. You can delete your old certificate by clicking **Remove**.

Credential: API Certificate  
Registrant: Julie James  
Joe's Generic Business,  
Napa, CA US

API username: schandrasekaran-us\_api1.paypal.com  
API password: 4AK8KBHSKUQE6JK8

**API certificate**

Fingerprint: 6CC5F19AB23DB0C45026DFA45D21057D  
[What is this?](#)

Request date: Mar 10, 2003 21:05:05 PST

Expiration date: Mar 7, 2013 21:05:05 PST Expires in 23 days **Renew Certificate**

**Remove Certificate** **Download Certificate**

3. 若憑證狀態是「**即將到期**」，請按一下「**更新憑證**」按鈕，  
如此就會產生另一個處於「**有效**」狀態的憑證。新舊憑證或同時顯示在「**管理 API 憑證**」頁面上。

My Account Send Money Request Money Merchant Services Products & Services

Overview Add Money Withdraw History Resolution Center Profile

### Manage API certificate [Back to My Profile](#)

You renewed your API certificate. Your new expiration date is Feb 9, 2023 22:41:21 PST. You can upload your certificate to any sites where you've provided your API certificate credentials in the past.

**Developers:** Do not share your credential information with anyone. Store in a secure location with limited access.

Credential:	API Certificate
Registrant:	Julie James Joe's Generic Business, Napa, CA US
API username:	schandrasekaran-us_api1.paypal.com
API password:	4AK8KBHSKUQE6JK8

**API certificate - Active**

Fingerprint:	DA9610E00EA1B8F1AD7711FCEDCE1971
<a href="#">What is this?</a>	
Request date:	Feb 11, 2013 22:41:21 PST
Expiration date:	Feb 9, 2023 22:41:21 PST <span style="background-color: #90EE90;">Expires in 3649 days</span>

**API certificate - Expires soon**

Fingerprint:	6CC5F19AB23DB0C45026DFA45D21057D
<a href="#">What is this?</a>	
Request date:	Mar 10, 2003 21:05:05 PST
Expiration date:	Mar 7, 2013 21:05:05 PST <span style="background-color: #FF8C00;">Expires in 23 days</span>

- 在標示為「**有效**」的憑證上按一下「**下載憑證**」，然後按照之前「**管理證書憑證**」小節中的步驟，下載和使用更新的憑證。

匯入新的 API 憑證後，請測試你的整合，以確保整合與新憑證能夠互相搭配。將你的新 API 憑證散佈給所有受影響的合作夥伴。舊憑證一旦到期，請按一下與該憑證相關的「**移除憑證**」按鈕將其移除。

### 加密憑證

用於 Java、.NET 和 Classic ASP 的 SDK 需要進行先加密成 PKCS12 格式，才能用於 SDK（請注意，PayPal PHP SDK 不需要 SSL 加密）。

**提示：**若你使用了加密機制，請務必同時加密 Sandbox 憑證和實際環境的 API 憑證。

加密憑證的步驟中需要用到 OpenSSL 加密工具。Unix 使用者的作業系統中可能已經安裝這項工具，Windows 使用者則需要下載 OpenSSL。接受 OpenSSL 的預設安裝選項。

1. 開啟命令提示字元。  
在 Windows 中，選擇「開始」>「程式集」>「附屬應用程式」>「命令提示字元」。
2. 確認 OpenSSL bin 目錄位於你的系統路徑中。如果不是，請將其加入你的系統路徑中。
3. 變更你想加密之憑證 (cert\_key.pem.txt) 的目錄位置，然後執行下列指令：

```
openssl pkcs12 -export -in cert_key.pem.txt -inkey cert_key.pem.txt -out paypal_cert.p12
```

**備註：**加密憑證時，系統會提示你輸入你用來解密檔案的密碼。請在「輸入匯出密碼」提示中輸入密碼，並請務必將其儲存在安全的位置。

以上流程會建立一個名為「paypal\_cert.p12」的檔案，這就是你的加密 API 憑證。

### 為 .NET 或 Classic ASP 安裝憑證

若你使用的是 NET SDK 或 Classic ASP SDK，則你需要安裝加密後的憑證。

若你是使用用於 NET SDK 來進行開發，Windows 系統會要求你匯入憑證並授權給你的網站使用者。Microsoft 提供能在單一指令中完成這些任務的公用程式。此公用程式叫做「Windows HTTP 服務憑證設定工具」（又叫 WinHttpCertCfg.exe）可從 Microsoft 免費下載。

若要使用此公用程式，你需要系統管理員的權限。在命令提示字元中輸入下列指令，並如下文所述改用適當的字串：

```
WinHttpCertCfg -i \<paypal_cert.p12> -p \<privateKeyPassword> -c LOCAL_MACHINE\my -a \<userName>
```

- 將 <paypal\_cert.p12> 換成你之前產生的 PKCS12 加密 API 憑證名稱。
- 將 <privateKeyPassword> 換成你用來加密憑證的密碼。
- 將 <userName> 換成執行你應用程式的使用者名稱。
  - 若是 ASP.NET 應用程式，此值就是 ASPNET。
  - 在 Windows IIS 5（預設設定）中，此值是 IWAM\_，其中底線部分是適當的電腦名稱。
  - 在 Windows IIS 6（預設設定）中，此值是“NETWORK SERVICE”（請務必包括英文括號）。

## 5. 讓 Express Checkout 整合正式上線

在你的應用程式與 PayPal Sandbox 共同運作，並且你已經準備移到實際環境下之後，請使用下列檢查清單來確保沒有遺漏正式上線前的任何步驟。

1. 建立並設定有效的 PayPal 帳戶。備註：若你代表另一家商店執行 Express Checkout 或其他 PayPal API 操作，則必須先取得許可，才能在實際環境中執行這些操作。換句話說，你代為執行 API 操作的商店必須為你提供許可，好讓你在實際環境中執行操作。
2. 確認有效帳戶的個人檔案設定與 Sandbox 帳戶的個人檔案設定相符，或者你知道並同意兩者設定不一致。
3. 設定有效 PayPal 帳戶的憑證。API 憑證與帳戶有關；因此正式運作的憑證與 Sandbox 的憑證不同。你必須為有效帳戶取得不同的電子簽章或下載不同的證書。
4. 若應用程式使用 PayPal SDK，請建立內含有效帳戶詳細資料的 API Profile 物件。你必須將「environment」欄位指定為 live，而且若你使用證書憑證，就必須包括有效帳戶的 API 用戶名稱、API 密碼，以及正式運作 API 證書的路徑。
5. 將 PayPal 的 IP 位址加入防火牆或其他網路裝置所需的任何受信任 IP 位址清單。詳情請參閱上線檢查清單：

<https://developer.paypal.com/webapps/developer/docs/classic/lifecycle/goingLive/>

## 6. 相關的 API 操作

若要建立最簡單的 Express Checkout 整合，請將付款操作指定為「Sale」（銷售），以便立即接受購物款項。你也可以設定稍後才收取購物款項，或退還購物款項。

### 6.1. Express Checkout 的「銷售」付款動作

「銷售」付款動作代表用來完成特定金額購物的一次付款事件。

「銷售」是預設的 Express Checkout 付款動作，不過你也可以在 `SetExpressCheckout` 和 `DoExpressCheckoutPayment` 要求中指定下列動作：

```
PAYMENTREQUEST_n_PAYMENTACTION=Sale
```

「銷售」是最直截了當的付款動作。若交易（包括商品的運送）可以立即完成，請選擇此付款動作。若要使用此付款動作：

- 當你叫用 `DoExpressCheckoutPayment` API 操作時，必須知道付款的最終金額
- 你打算立即執行訂單，例如有數位檔案形式的商品，或有可立即寄出之庫存商品的情況

執行 `DoExpressCheckoutPayment` API 操作後，付款就會完成，你不需要再進行其他動作。你使用此付款動作時，無法取得其他款項或使款項的任何部分無效。

### 6.2. Express Checkout 的「授權」付款動作

「授權」付款動作代表同意付款，並將買家的資金擱置最多 3 天的動作。

若要設定「授權」，請在 `SetExpressCheckout` 和 `DoExpressCheckoutPayment` 要求中指定下列付款動作：

```
PAYMENTREQUEST_n_PAYMENTACTION=Authorization
```

「授權」能讓你取得你在 `DoExpressCheckoutPayment` 要求中所指定金額的 115%，或最多比該金額多出 \$75 美元的多筆款項。若你需要在取得款項之前先寄出商品，或若你因故無法立即接受款項，請選擇此付款動作。

可擱置資金的「信任期」（款項先被保留住）是 3 天，授權維持有效的「有效期」則是 29 天。你可以在 29 天的有效期中，最多重新授權一次 3 天的信任期。

你可以使授權失效；在這種情況下，你在 `DoExpressCheckoutPayment` 要求中所指定金額的尚未取得部分就會失效，無法再取得。若你完全沒有取得款項，則整筆款項就會失效，你也無法取得任何款項。

表 1：API 操作 Express Checkout

---

API 操作

說明

---

DoCapture

取得授權的款項。

---

DoReauthorization

重新授權款項。

---

DoVoid

將訂單或授權作廢。

### 6.3. Express Checkout 的「訂單」付款動作

「訂單」付款動作表示買家同意在最多 29 天內，支付一或多筆高達指定總金額的授權金額。

若要設定「訂單」，請在 `SetExpressCheckout` 和 `DoExpressCheckoutPayment` 要求中指定下列付款動作：

```
PAYMENTREQUEST_n_PAYMENTACTION=Order
```

建立 PayPal 訂單，表示買家已同意購買，但沒有擱置資金。建立「訂單」付款動作後，你就可以直接進行資金取得的動作，不必經過額外的授權程序；你也可以建立授權 API 呼叫，以將資金擱置到你準備取得該筆資金為止。由於「訂單」付款動作不會將資金擱置，因此建議你按照第二種方式來建立授權，而不要直接取得「訂單」。

「訂單」能讓你在 29 天內建立多個授權。你可以為每個授權取得多筆款項，其金額最高為你在 `DoExpressCheckoutPayment` 要求中指定金額的 115%（不能超過 \$75 美元）。

**備註：**你最多可以為每個「訂單」建立 10 個授權。若你發現帳戶限制每個「訂單」的授權數，請與 [PayPal 客戶支援部門](#) 聯絡，要求增加授權數（每個「訂單」最多 99 個）。

接受不同類型商品訂單的商店通常會利用此技巧，這些商品中有些可能在下單時還無法寄出。當商品可以寄出時，就會進行後續的基本授權。這些授權能確保客戶仍舊擁有用來購買每項商品的資金。此付款動作的彈性最大，並且應該在銷售無法滿足你的需求，或一次授權加上一次重新授權還是無法滿足你的需求時使用。訂單符合需求的情況包括：

- 回單：在回單中，可提供的商品會立即寄出，剩餘的商品則會在可提供時寄出，因此可能需要運送兩次
- 分單：在分單中，商品是分次寄出，也許寄到不同的地址，這時你會想要針對每次寄出的商品分別取得購物款項
- 批發商出貨：由其他廠商出貨，並且你代替這些廠商接受購物款項

你不能進行重新授權。需要重新授權，例如當授權的擱置期或有效期到期時，只要再建立一個授權即可。

你可以使「訂單」或從「訂單」建立的授權失效。若你使授權失效，你在 `DoExpressCheckoutPayment` 要求中所指定金額的尚未取得部分就會失效，無法再取得。若你完全沒有取得款項，則整筆款項就會失效，你也無法取得任何款項。

表 2：Express Checkout 中與「訂單」付款動作相關的 API 操作

API 操作	說明
<code>DoAuthorization</code>	授權款項。
<code>DoCapture</code>	取得授權的款項。
<code>DoVoid</code>	將訂單或授權作廢。

## 6.4. 發放退款

你可以使用 `RefundTransaction` API 來發放退款。

使用 `RefundTransaction` API 發放一筆或多筆與交易相關的退款，例如由取得購物款項所建立的交易。交易的識別方式是依據取得購物款項時 PayPal 指派的 Transaction ID。

**備註：**若交易發生時已超過退款期限，便無法發放退款；退款期限通常是 60 天。

允許的退款金額最多可達原來交易的總金額。若你指定全額退款，就會退還整筆金額。若你指定部分退款，則必須指定要退款的金額、貨幣和退款說明。當你呼叫 `RefundTransaction` API 時，PayPal 的回應會包含與退款相關的另一個 Transaction ID（非原來交易），以及其他退款相關資料。資料內容如下：

- 退還給買家的退款總額
- 退還給你原本交易手續費的退款金額
- 從你的餘額中扣除的退款淨額

若要發放退款：

1. 在 `RefundTransaction` 要求中，指定要退還的購物款項的交易 ID。

```
TRANSACTIONID=transaction_id
```

2. 指定退款種類，有 `Full` 或 `Partial` 兩種。

```
REFUNDTYPE=Full
```

或者

```
REFUNDTYPE=Partial
```

3. 部分退款應指定退款金額，含使用的貨幣。

```
AMT=amount
```

```
CURRENCYCODE=currencyID
```

4. 部分退款應指定備忘錄說明。

```
NOTE=description
```

5. 執行 `RefundTransaction` 操作。
6. 檢查 `RefundTransaction` 回應中的認可狀態，確定操作成功。

## 6.5. 使用 API 幂等

你可以使用 `MsgSubID`（訊息提交編號）來協助追蹤等待中或失敗的要求。

`MsgSubID` 已經加入到下列 API 的要求和回應中：

- `DoAuthorization` (NVP | SOAP)
- `DoCancel` (銷售點)
- `DoCapture` (NVP | SOAP)
- `DoExpressCheckoutPayment` (NVP | SOAP)
- `DoReauthorization` (NVP | SOAP)
- `DoReferenceTransaction` (NVP | SOAP)
- `DoVoid` (NVP | SOAP)
- `RefundTransaction` (NVP | SOAP)

在要求失敗或你不確定原始要求的結果時，幂等相當實用。幂等也能協助將要求承載與回應承載相關聯。幂等能避免出現重複的要求，因為先前隨附 `MsgSubID` 而傳送的要求，將傳回使用相同 `MsgSubID` 之前一個要求的最新狀態。反之，沒有隨附 `MsgSubID` 的要求將重複進行要求。

以下是幂等派上用場的案例：

- 若隨附 API 要求逾時，用戶端應用程式可以使用隨附的 `MsgSubID` 重新嘗試原始要求。若要求已經處理完成，PayPal 就會提供要求的最新狀態，並可能傳回 `11607` 警告代碼（特定訊息提交編號出現重複的要求）。
- 若用戶端應用程式同時傳送兩個具備相同 `MsgSubID` 的 API 要求，PayPal 會處理第一個要求，另一個要求可能會失敗並傳回 `11604` 錯誤代碼（訊息提交編號的要求已經在處理中）。

#### 附註

- 幂等可以在 API 版本 `74.0` 或較新版本中使用
- 在 `DoExpressCheckoutPayment` 呼叫中，你可以使用 `Token` 來取代 `MsgSubID`。若是多筆款項，則應該合併使用 `Token` 和 `Express Checkout PaymentRequestID` 欄位，以取代 `MsgSubID`。

#### 範例

下列範例將 `MsgSubID` 用做 `DoAuthorization` 要求的一部分：

```
<transactionID>0-87H32160HB8486131</transactionID>
<transactionEntity>Order</transactionEntity>
<amount currencyID="USD">10</amount>
<msgSubID>81d4fae-7dec-11d0-a765-00a0c91e6bf6</msgSubID>
```

#### 使用備註

- 每個要求都必須使用不同的 `MsgSubID`，而且最好是每種 API 呼叫類型（例如 `DoAuthorization`、`DoCapture`）都使用不同的 `MsgSubID`。
- PayPal 建議使用 UUID 標準來為你的要求指派 `MsgSubID`，因為此標準符合 `MsgSubID` 38 個半形字元的長度限制。
- 幂等只有在原始要求成功時才適用。若原始要求造成錯誤，系統就不會儲存原始要求。
- PayPal 保留讓 `MsgSubID` 在 13 天後到期的權利。
- PayPal 會提供要求在目前的狀態，而不是起始要求的狀態，例如進行付款的起始要求（狀態為「`Complete`」）。款項會在稍後退還。若重新提交使用原始 `MsgSubId` 的原始要求，則回應中會指出付款狀態為「`Refunded`」。
- 後續回應中的結果將與原始交易回應不同，並且將傳回「`SuccessWithWarning`」，而不是「`Success`」。

## 7. 將 Express Checkout 與 PayPal SDK 整合

你可以選擇使用 PayPal SDK，以取代將訊息建立為 NVP 字串或 SOAP 結構的做法。這些 SDK 能讓你使用偏好的程式設計語言來撰寫程式碼，例如 Java、PHP、Ruby 等。

SDK 除了能讓你使用自己選擇的程式設計語言以外，還能讓你輕鬆執行常見的任務，例如編碼和解碼訊息，以及管理憑證。若要進一步了解 SDK 並下載適合你的環境的 SDK，請參閱：

- [商店 API SDK](#)