

[Open in app ↗](#)

搜索



寫



如何使用 C# 和 .NET 6 創建簡單的 Slackbot



賈斯汀·格伯 · 跟隨

11分鐘閱讀 · 5月2022 · <>



...

項目詳情

本教程將使用 ASP.NET Core Web API 建立 Slackbot。

將在此 API 上創建兩個 API 端點。

第一個終結點 /Submit

第一個端點是 POST 方法，Slackbot 將在其中將消息發佈到 Slack 頻道。消息和 Slack 頻道將在請求中指定。讀取器可以修改其現有應用程式，以向此終結點發出 HTTP 請求。它將在 Slack 頻道中發送消息。這對於記錄嚴重錯誤、提供狀態更新等非常有用。

The top half of the image shows a screenshot of a Slack channel named '#tennis'. The channel has a message from the user stating 'You created this channel today. This is the very beginning of the # tennis channel.' Below this, there are options to 'Add description', 'Add people', and 'Send emails to channel'. The bottom half shows a screenshot of a web browser displaying the Slack API documentation on Swagger UI. The title is 'SlackAPI 1.0 OAS3'. It shows two POST methods: '/Slack/Submit' and '/Slack/Event'.

第二個終結點/事件

當使用者在 Slack 頻道上創建消息時，Slack 將代表他們向第二個端點發出 HTTP 請求。如果讀者還不知道，這稱為 webhook。第二個 API 端點將接收在 Slack 頻道上創建的消息。根據消息的內容，API 將使用 Slackbot 將消息寫回 Slack 頻道。在下面的示例中，API 中的代碼查找包含單詞 ping 的消息並回復back pong。整潔！這對於運行內部報告、實時監控工具、提供即時應用程式狀態更新等非常有用。

tennis

You created this channel today. This is the very beginning of the # tennis channel.

[Add description](#) [Add people](#) [Send emails to channel](#)

Today ▾

 **Justin Gerber** 10:34 PM

joined #tennis along with Slack Bot.

B I S | ⌂ | ⌂ ⌂ | ⌂ ⌂ | ⌂ ⌂

|Message #tennis I

+ | ⌂ ⌂ | ☺ ☺ | ☺ ☺ | Aa

▶ | ▾

自主技術

本教程將使用 AWS 和 Elastic Beanstalk 服務來部署 API。筆者認為本教程非常適合不熟悉AWS的讀者。如果讀者已經知道如何部署可公開訪問且未在 localhost 上運行的API，則他們可能能夠設法不使用 AWS。請注意，除非 API 公開託管在某個位置，否則第二個端點將不起作用。在本教程中，作者將展示 AWS 和 Elastic Beanstalk 服務。ECS 是另一個不錯的選擇的 AWS 服務，但這裡不會使用它。

在開始之前，需要向 soxtoby 索此專案大量使用該存儲庫。如果你覺得本教程有說明，一定要給他的專案一顆星。

GitHub - soxtoby/SlackNet：適用於 .NET 的綜合性 Slack API 用戶端

一個易於使用且全面的 API，用於在 .NET 中編寫 Slack 應用。有兩個主要的 NuGet 包可用於...

[github.com](https://github.com/soxtoby/SlackNet)

先決條件

- Slack 和管理員許可權
- 一個 AWS 帳戶
- .NET 6
- 視覺工作室 2022
- AWS Toolkit for Visual Studio 2022

在設置上述先決條件時需要說明？作者有一個關於這方面的教程。可以在這裡找到它。

添加 AWS Toolkit for Visual Studio 2022

AWS Toolkit for Visual Studio 是在 Microsoft Windows 上運行的 Microsoft Visual Studio 的擴展，可...

[medium.com](#) 媒體網站

可選，但建議 先決條件

- GoDaddy 功能變數名稱（截至 11 年 99 月 09 日，每年 03.2022 美元）

在購買 GoDaddy 功能變數名稱時需要說明？作者有一個可以提供幫助的教程，可以在這裡找到[_](#)。

從 GoDaddy 購買功能變數名稱

本教程將提供有關如何從 GoDaddy 購買功能變數名稱的詳細資訊。這個過程非常簡單。網域是...

[medium.com](#) 媒體網站

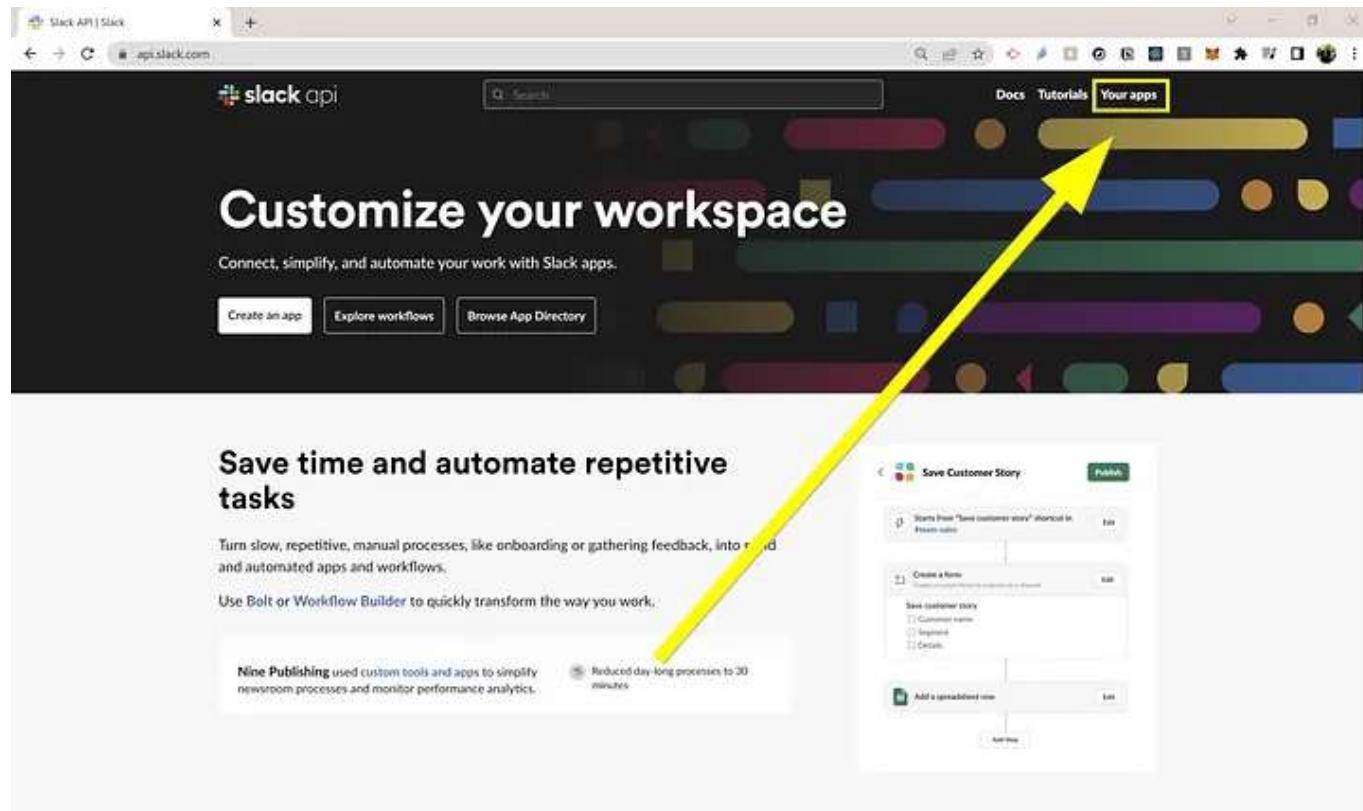
好的，事不宜遲，讓我們開始吧！

Slack 應用

第一個任務是創建一個 Slack 應用程式。

首先，導航到 <https://api.slack.com>。

按兩下頂部工具列上的「您的應用」。



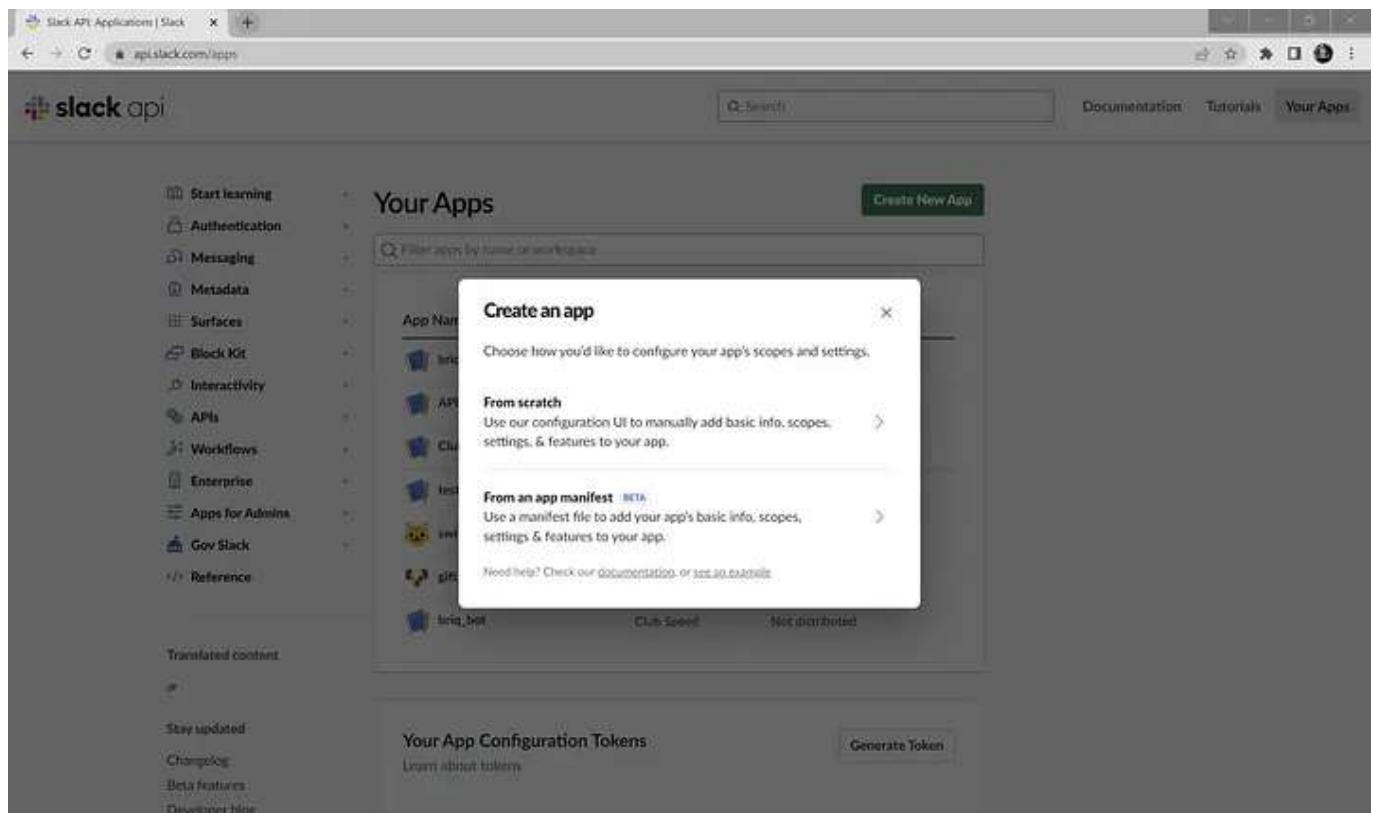
按兩下登錄 Slack 帳戶超連結。登錄。

The screenshot shows the Slack API Applications interface. On the left is a sidebar with various API categories like Start learning, Authentication, Messaging, etc. The main area is titled 'Your Apps' and contains a message: 'You'll need to sign in to your Slack account to create an application.' Below this is a section for 'Your App Configuration Tokens' with a 'Generate Token' button. At the bottom, there's a note: 'Don't see an app you're looking for? Sign in to another workspace.' A large yellow arrow points from the bottom right towards the 'Create New App' button.

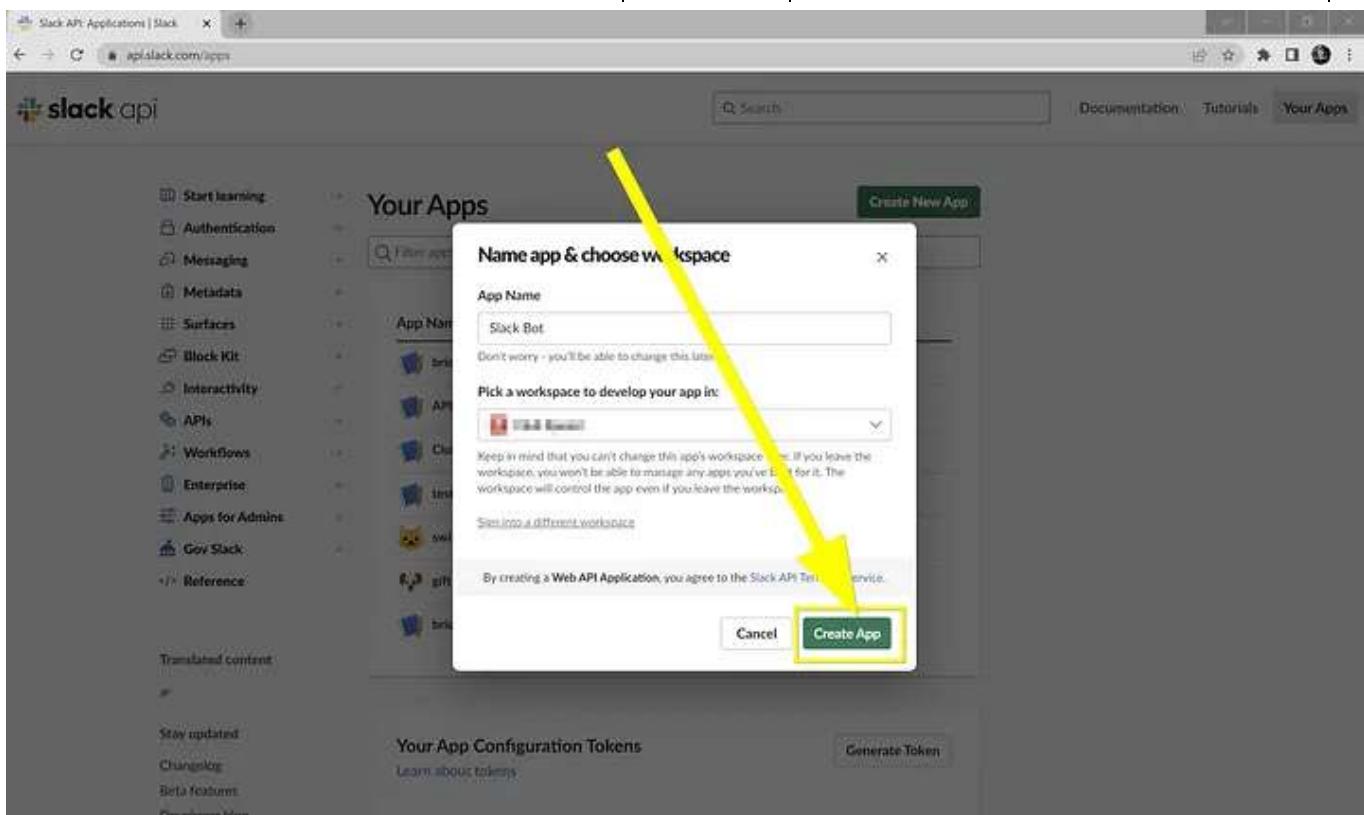
按兩下建立新應用程式按鈕。

This screenshot shows the same Slack API Applications interface after a user has signed in. The 'Your Apps' section now lists several existing applications. A yellow arrow points to the 'Create New App' button at the top right of the 'Your Apps' section. The sidebar and other UI elements remain the same as in the first screenshot.

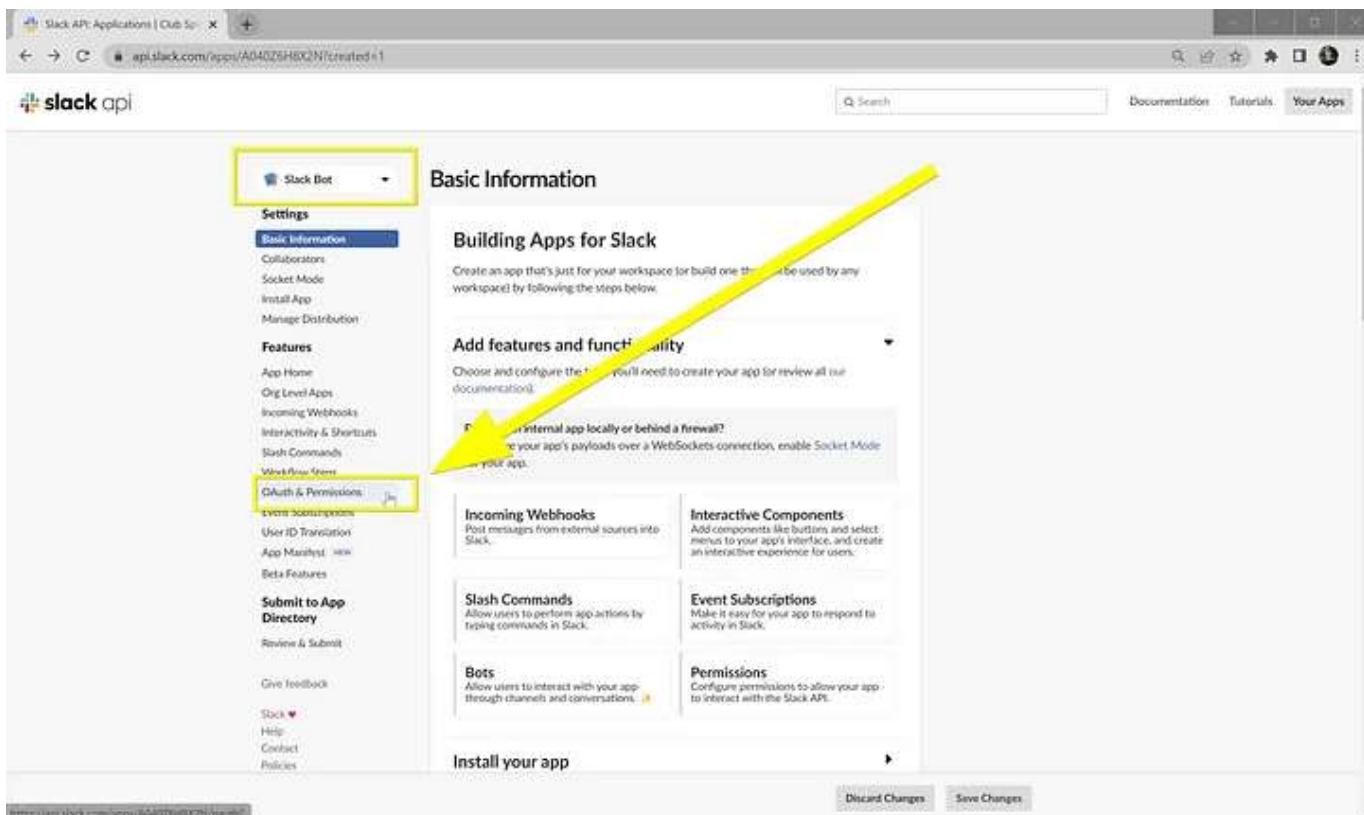
按兩下從頭開始選項。



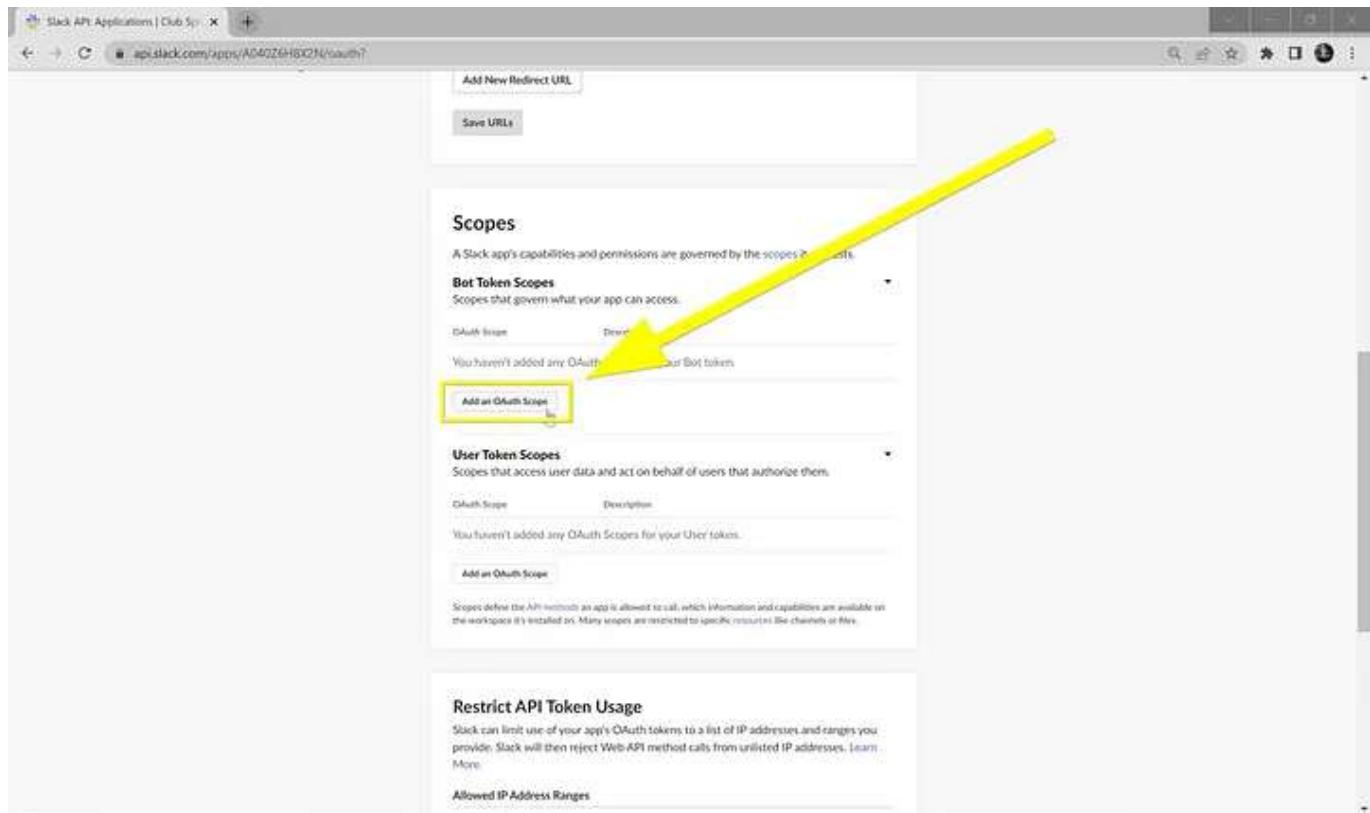
提供 Slack 機器人的應用名稱。此外，請確保在「選擇要在其中開發應用的工作區」下拉清單中選擇一個 Slack 工作區。按兩下建立應用程式按鈕。



應載入基本資訊頁面。在左上角，應該有一個下拉清單，其中選擇了 Slack 機器人。點擊左側選單中的 Oauth & Permissions。



在「範圍」部分中，按兩下機器人令牌範圍部分下的「添加 OAuth 範圍」按鈕。



選擇 chate : write 選項。這將授予 Slackbot 寫入許可權。

Scopes

A Slack app's capabilities and permissions are governed by the scopes it requests.

Bot Token Scopes

Scopes that govern what your app can access.

OAuth Scope	Description
chat:history	View messages and other content in direct messages that Slack Bot has been added to.
chat:read	View basic information about direct messages that Slack Bot has been added to.
chat:write	Send messages as Slack Bot
chat:write:custom	Send messages as Slack Bot with a customized username and avatar.
chat:write:public	Send messages to channels Slack Bot isn't a member of.
commands	

Add an OAuth Scope

Scopes define the API methods an app is allowed to call, which information and capabilities are available on the workspace it's installed on. Many scopes are restricted to specific resources like channels or files.

Restrict API Token Usage

Slack can limit use of your app's OAuth token to a list of IP addresses and ranges you provide. Slack will then reject Web API method calls from unlisted IP addresses. Learn More.

選擇im : write 選項。這將向使用者授予 Slackbot 寫入許可權。

Scopes

A Slack app's capabilities and permissions are governed by the scopes it requests.

Bot Token Scopes

Scopes that govern what your app can access.

OAuth Scope	Description
chat:write	Send messages as @slack_bot

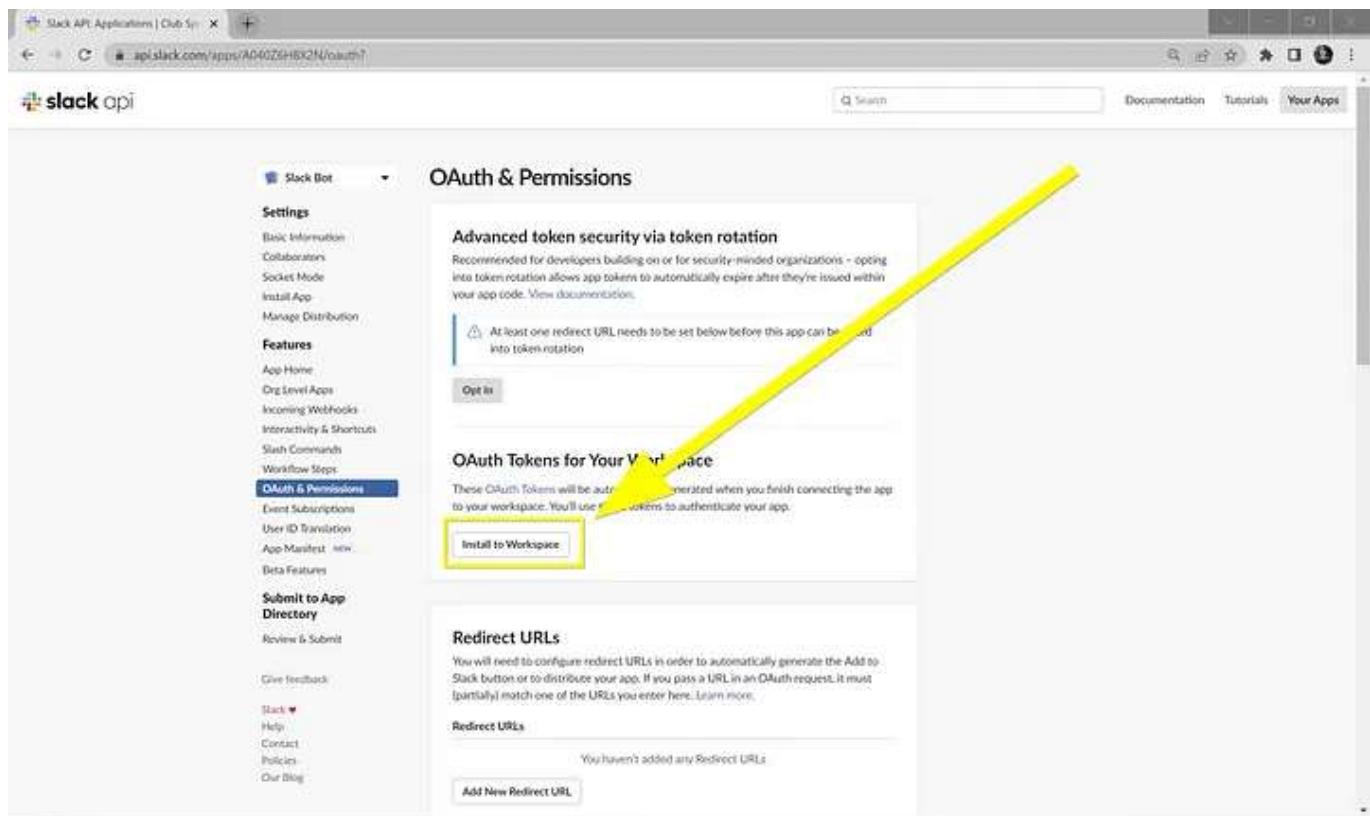
Add permission by Scope or API method...

Manage private channels that Slack Bot has been added to and create new ones:

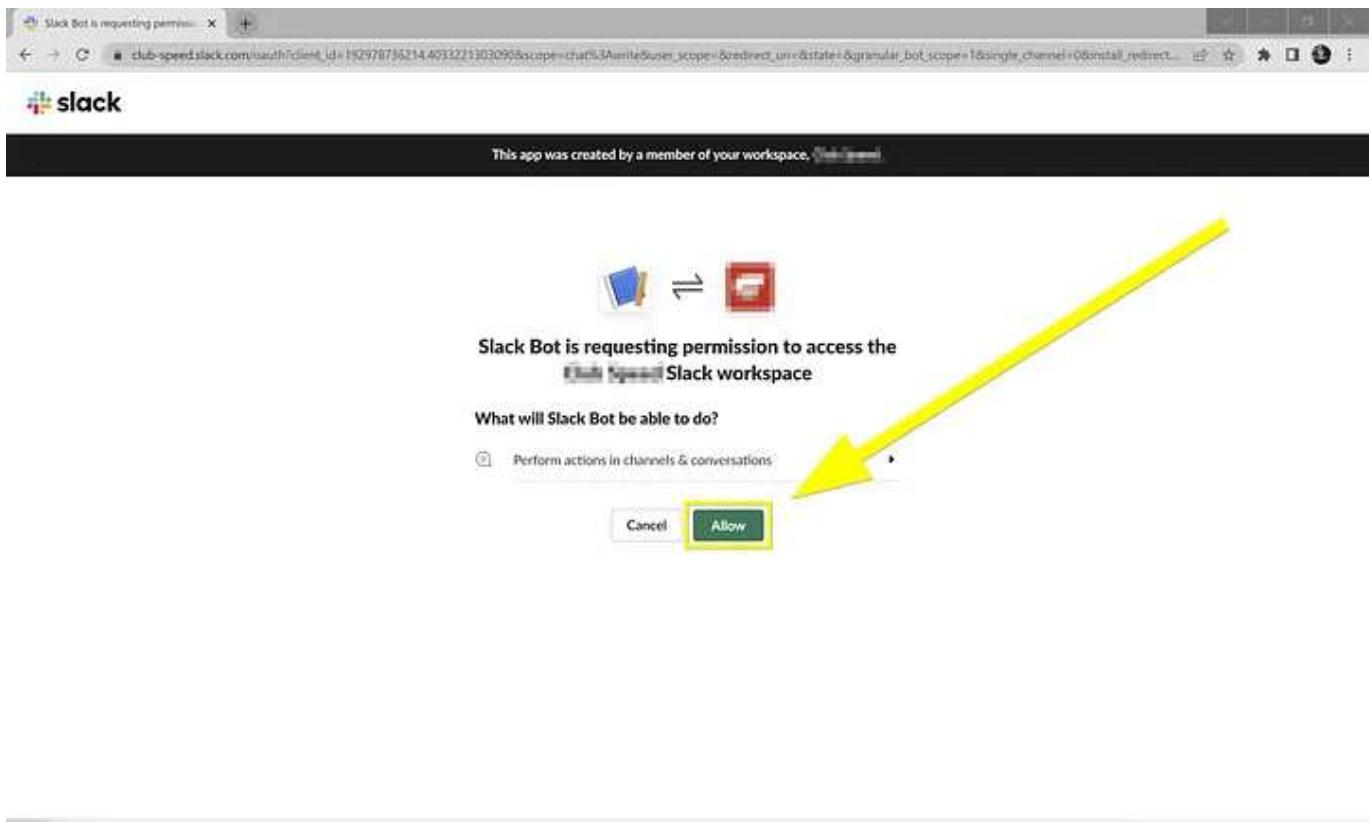
- im:history
- View messages and other content in direct messages that Slack Bot has been added to.
- im:read
- View basic information about direct messages that Slack Bot has been added to
- im:write**
- Start direct messages with people**
- incoming-webhook
- Pint messages to specific channels in Slack
- links:embed:write
- Embed video player URLs in messages and app surfaces

Add an OAuth Scope

向上滾動到工作區的 OAuth 令牌部分。按兩下安裝到工作區按鈕。



按兩下允許按鈕。



可以看到機器人使用者 OAuth 令牌。將此值複製到記事本或類似內容中。它將在後面的部分中使用。

OAuth & Permissions

Advanced token security via token rotation
Recommended for developers building on or for security-minded organizations - opting into token rotation allows app tokens to automatically expire after they're issued within your app code. [View documentation](#).

OAuth Tokens for Your Workspace
These tokens were automatically generated when you installed the app to your team. You can use these to authenticate your app. [Learn more](#).

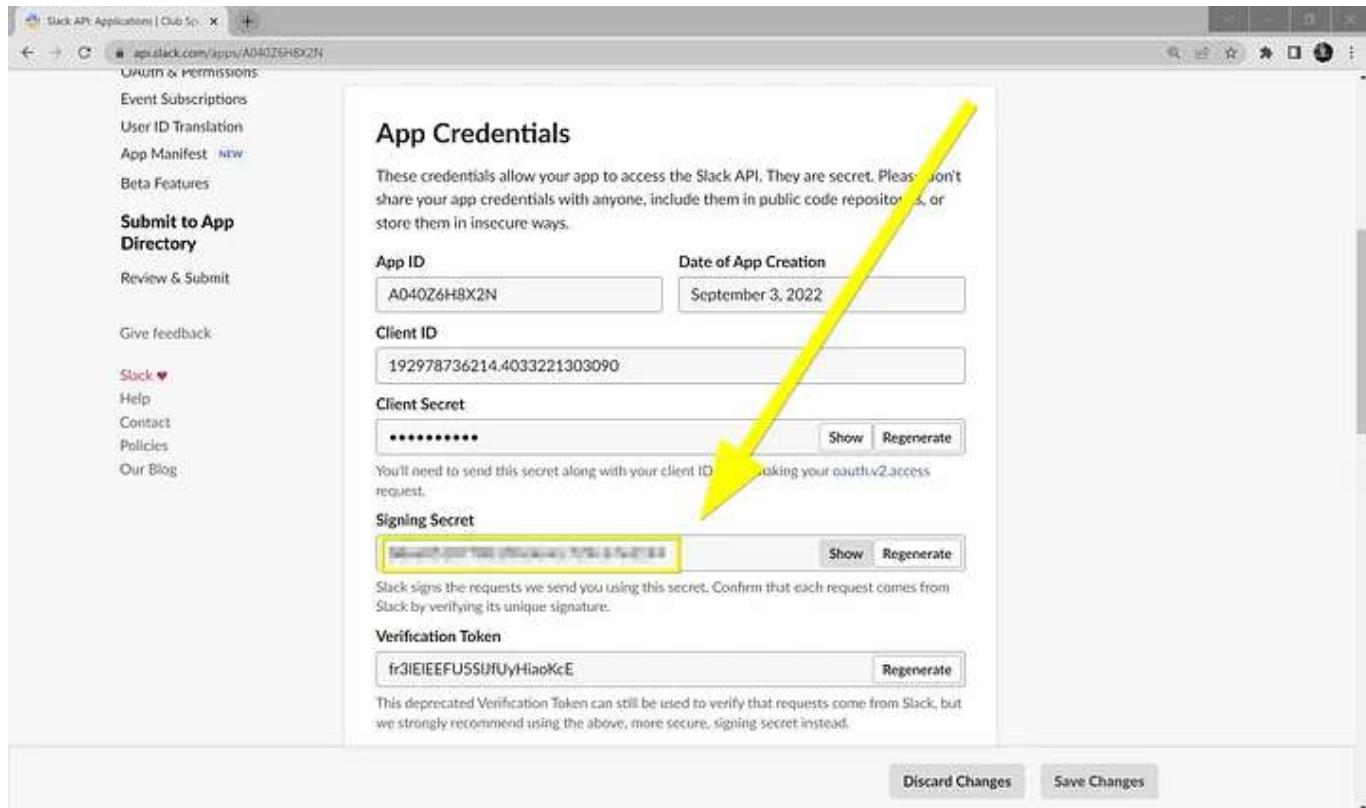
Bot User OAuth Token

`XXXXXXXXXXXXXXXXXXXX-XXXXXXXXXXXXXXXXXXXX-XXXXXXXXXXXXXXXXXXXX`

Redirect URLs
You will need to configure redirect URLs in order to automatically generate the Add to Slack button or to distribute your app. If you pass a URL in an OAuth request, it must (partially) match one of the URLs you enter here. [Learn more](#).

Click on the Basic Information tab on the left. Scroll down to the App Credentials section. Find the Signing Secret and click the Show button. Copy this value into Notepad or something similar. It will be used in a later section.

按一下左側的“基本資訊”標籤。向下捲動至應用程式憑證部分。找到簽名金鑰並點擊顯示按鈕。將此值複製到記事本或類似的東西。它將在後面的部分中使用。



Create New Slack Channel 建立新的 Slack 頻道

In this tutorial the Slackbot will need to read and write to a Slack Channel. A #tennis Slack Channel will be used in this guide.

在本教程中，Slackbot 將需要讀取和寫入 Slack 通道。本指南將使用 #tennis Slack 頻道。

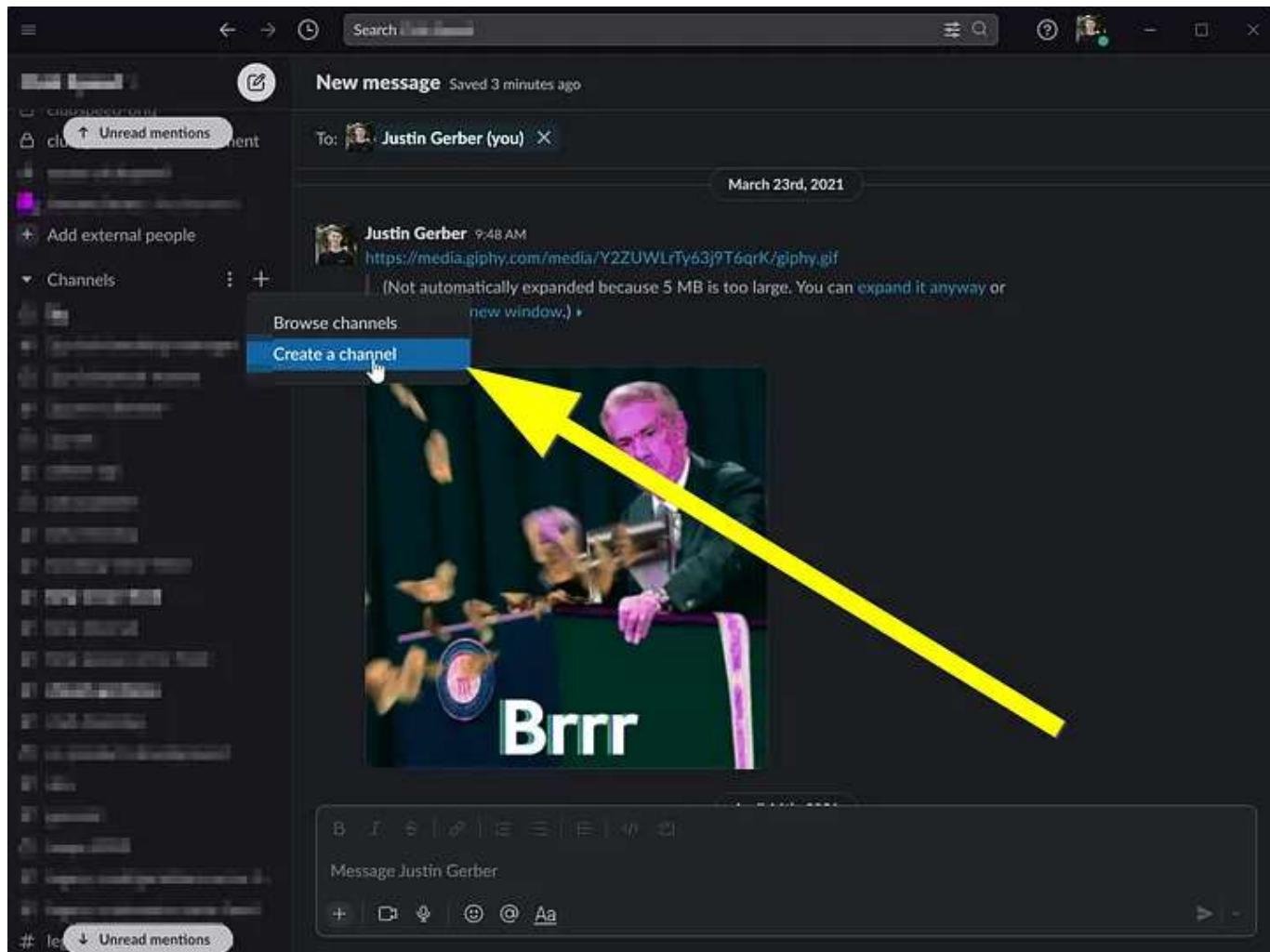
Open the Slack application.

打開 Slack 應用程式。

On the left menu find the Channels menu option and then click the + icon.

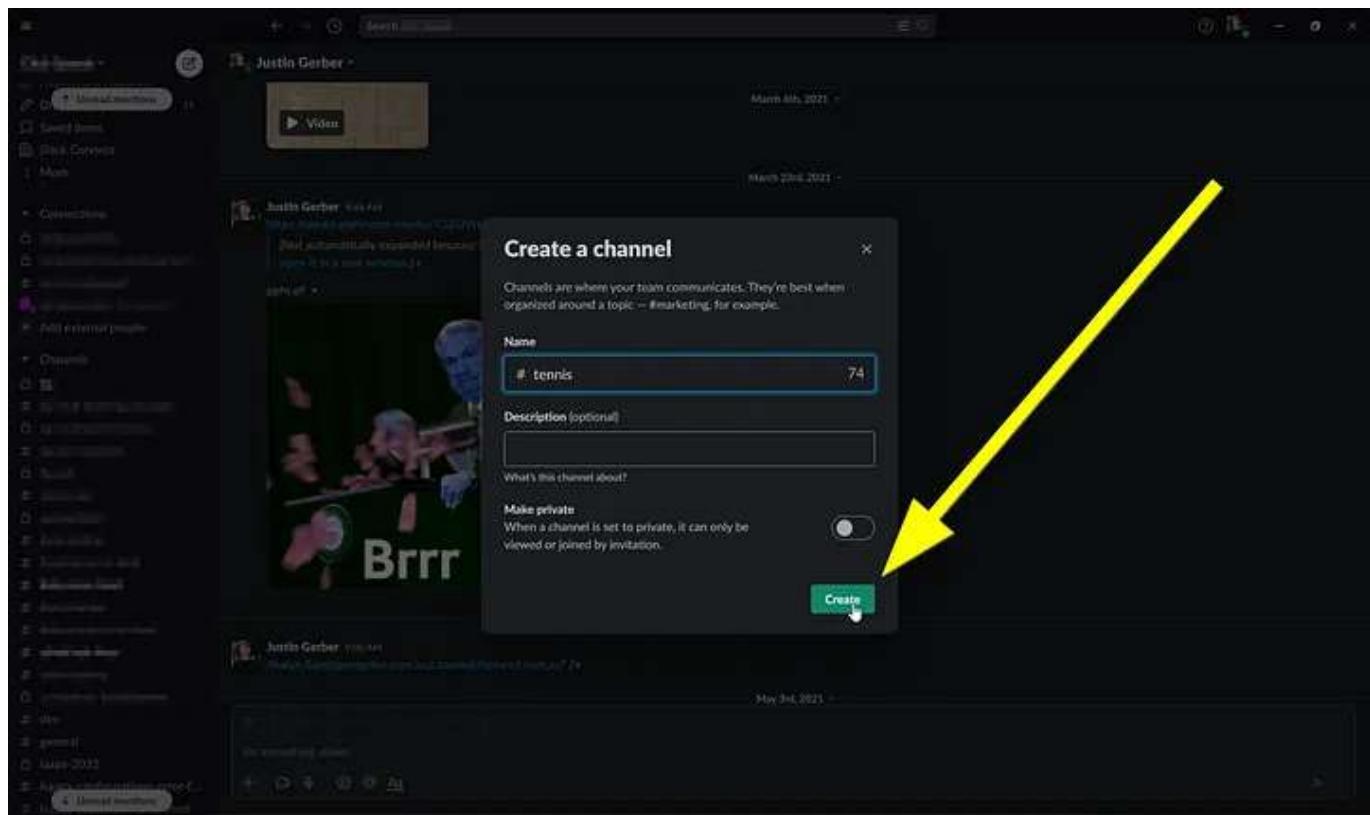
Click the Create a channel menu option.

在左側選單中找到“頻道”選單選項，然後按一下“+”圖示。點選建立頻道選單選項。



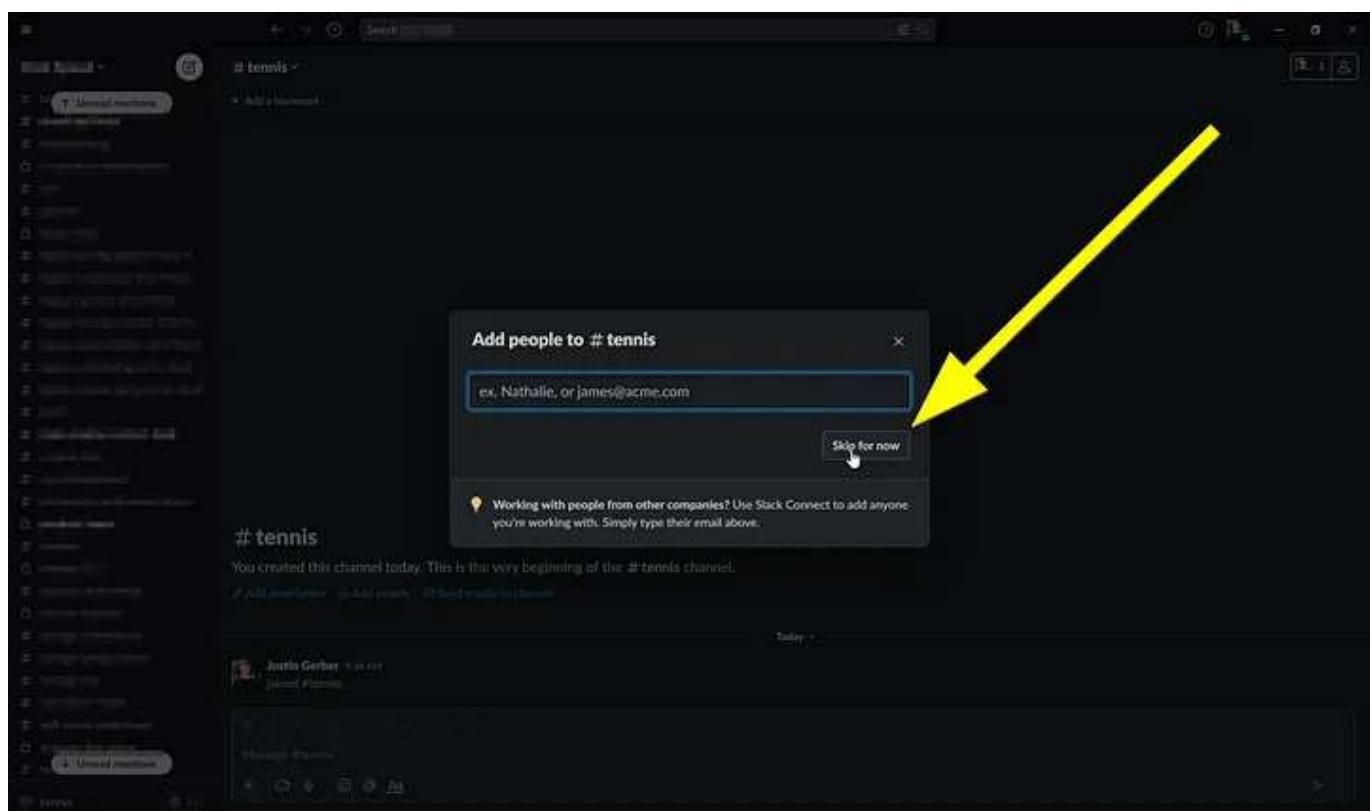
Provide a name of tennis and then click the Create button.

提供網球名稱，然後按一下「建立」按鈕。

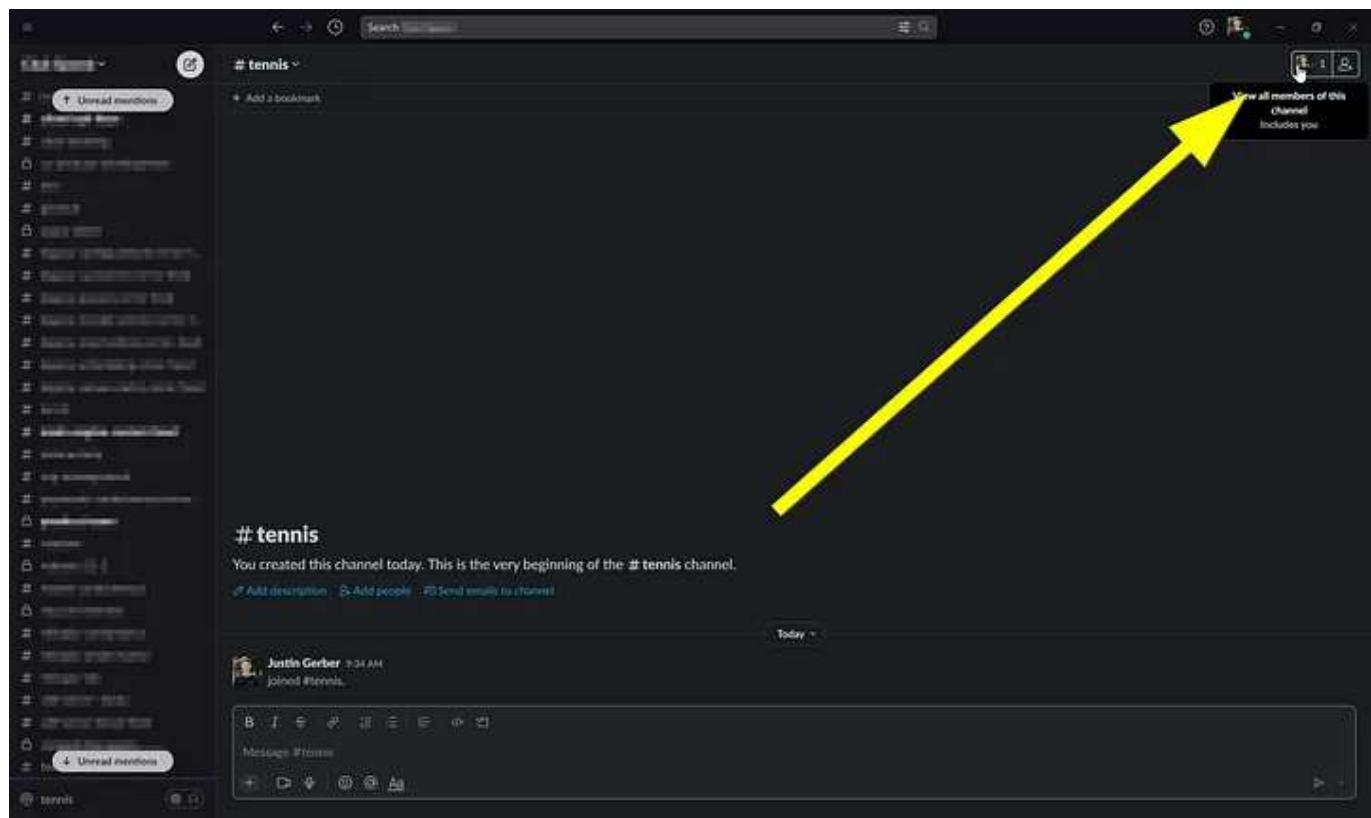


Click the Skip for now button.

按一下“立即跳過”按鈕。



按兩下查看此頻道的所有成員按鈕。



按兩下“集成”選項卡。按兩下「添加應用程式」按鈕。

tennis

Get Notifications for @ Mentions Start a Call

About Members 1 Integrations Settings

Workflows
Automate the tasks and processes unique to your team, no coding required.

Add a Workflow

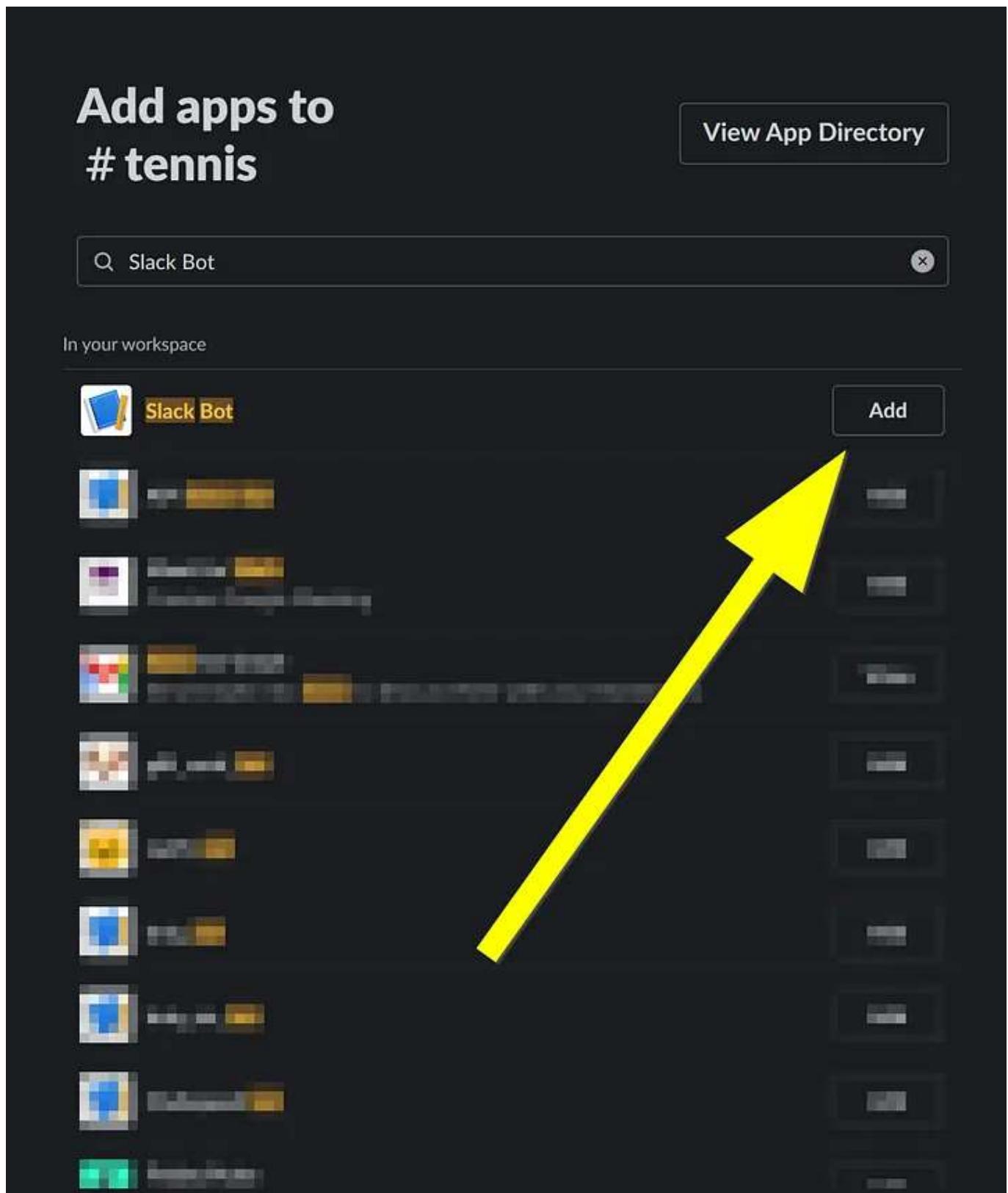
Apps
Bring the tools you need into this channel to pull reports, start calls, file tickets and more.

Add an App

Send email to this channel
Get an email address that posts incoming emails in this channel.

A large yellow arrow points from the 'Send email to this channel' section up towards the 'Add an App' button.

搜索“Slack 機器人”，然後按下“添加”按鈕。



現在，Slackbot 應用程式已連接到 #tennis 通道。

克隆存儲庫

請克隆此處找到的 slack-bot-api 儲存庫。這個存儲庫是由作者編寫的。如果讀者給它一顆星，這對他來說意味著世界。請在 Visual Studio 2022 中打開解決方案。

GitHub - Justbeingjustin/slack-bot-api : 使用 C 語言在 slack 中構建機器人的指南#

您目前無法執行此操作。您使用其他選項卡或視窗登錄。您在另一個分頁中登出，或者...

[github.com](https://github.com/Justbeingjustin/slack-bot-api)

修改 Appsettings.json

在「解決方案資源管理器」中打開appsettings.json 檔。

填寫 SlackAccessToken。此值應保存在記事本或類似內容中。此值是機器人使用者 OAuth 令牌，如上一部分所示。

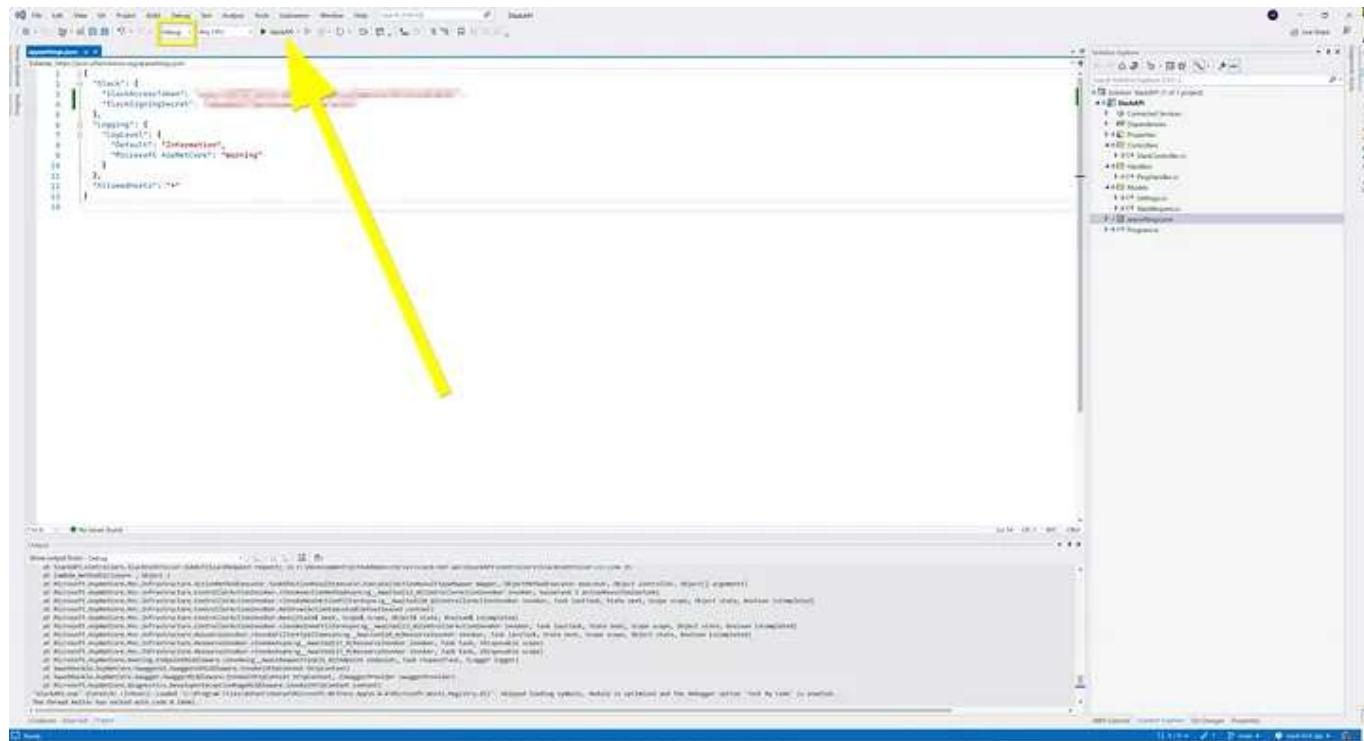
填寫 SlackSigningSecret。此值應保存在記事本或類似內容中。此值是簽名金鑰，如上一節所示。

保存檔案。

```
{
  "Slack": {
    "SlackAccessToken": "xoxb-19[REDACTED]b86[REDACTED]",
    "SlackSigningSecret": "b86[REDACTED]"
  },
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  },
  "AllowedHosts": "*"
}
```

測試提交終點

第一個終點已準備好進行測試。在測試第二個終點之前，還需要執行其他工作。確保 Visual Studio 處於調試模式，然後按兩下「運行」按鈕。



瀏覽器視窗應打開，Swagger 頁面應載入。展開 POST /Slack/Submit 端點。

SlackAPI 1.0 OAS3
https://localhost:7199/swagger/v1/swagger.json

Slack

POST /Slack/Submit

POST /Slack/Event

Schemas

SlackRequest >

按兩下「試用」按鈕。

SlackAPI 1.0 OAS3
https://localhost:7199/swagger/v1/swagger.json

Slack

POST /Slack/Submit

Parameters

No parameters

Request body

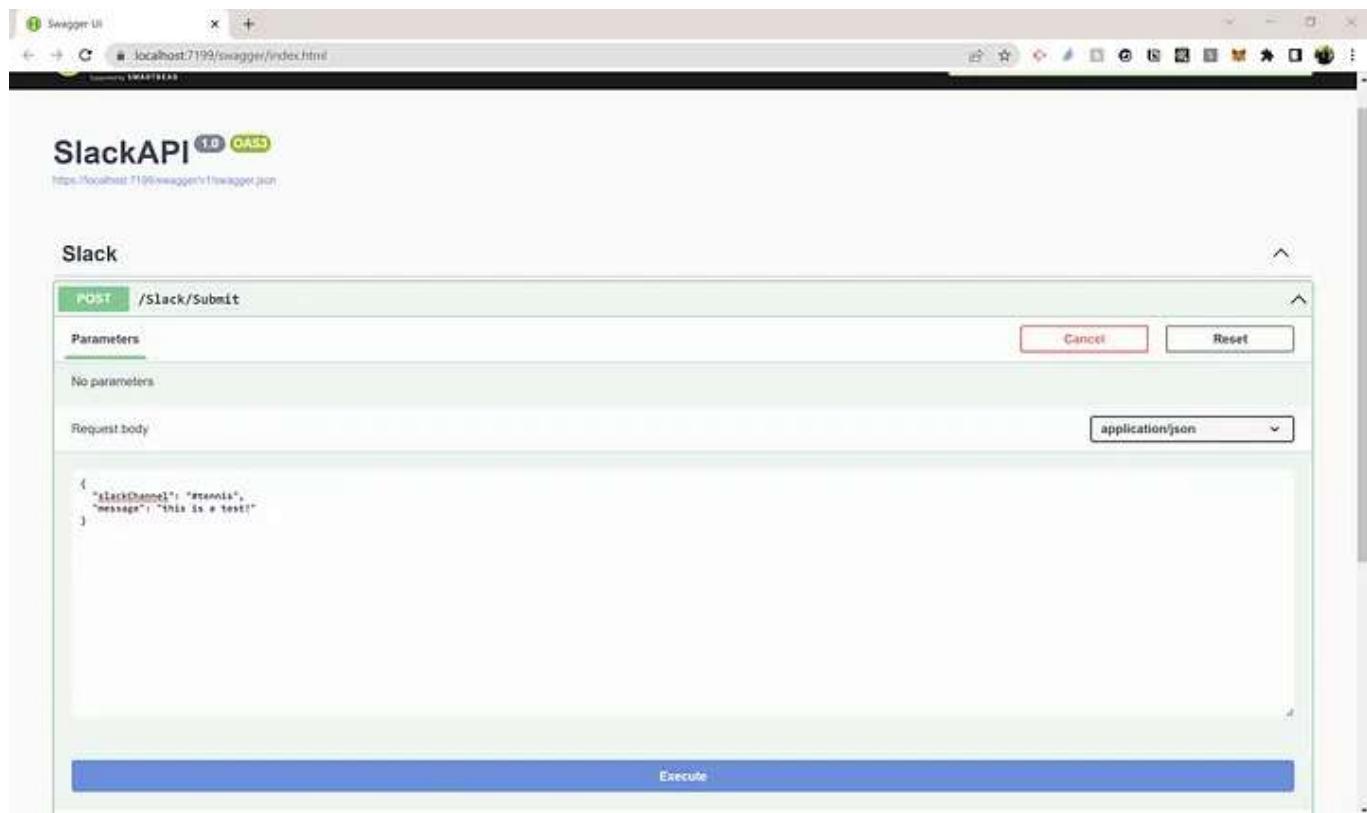
Example Value | Schema

```
{
  "slackChannel": "string",
  "message": "string"
}
```

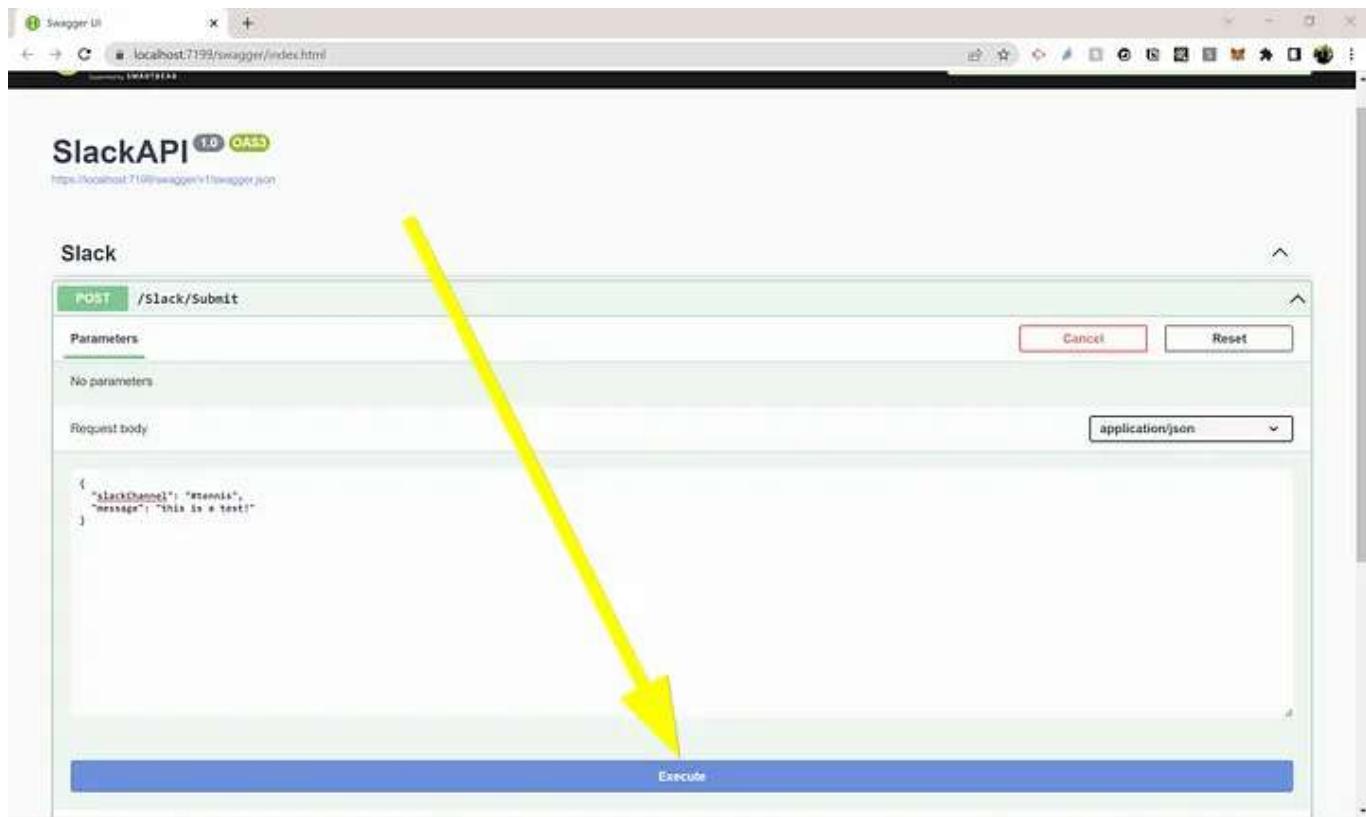
Responses

Code	Description	Links
200	Success	No links

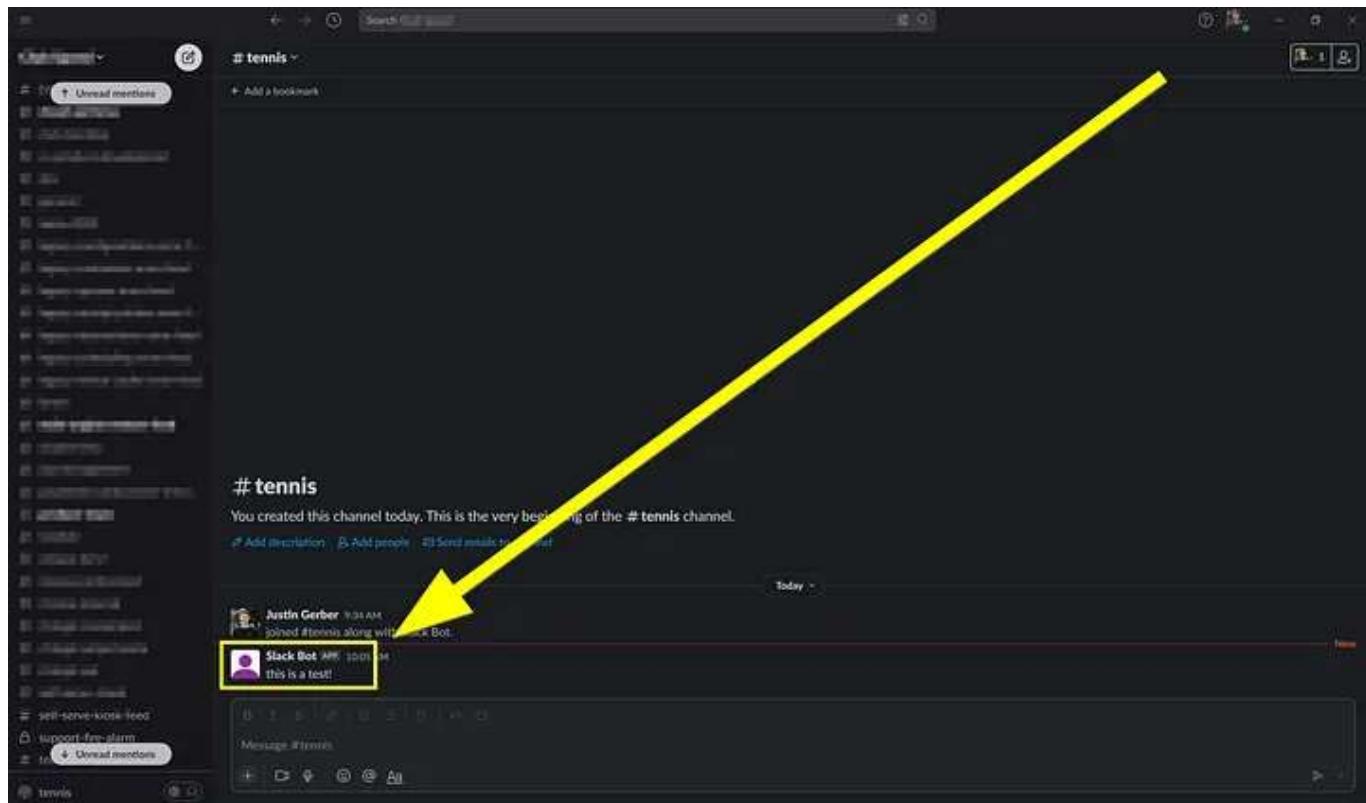
修改文字框中的文字，以便 #tennis slackChannel 值。修改文本，使消息值為“This is a test！”。



按兩下“執行”按鈕。



打開 Slack 應用程式，查看是否在 #tennis Slack 頻道中發佈了消息。恭喜，如果這有效。

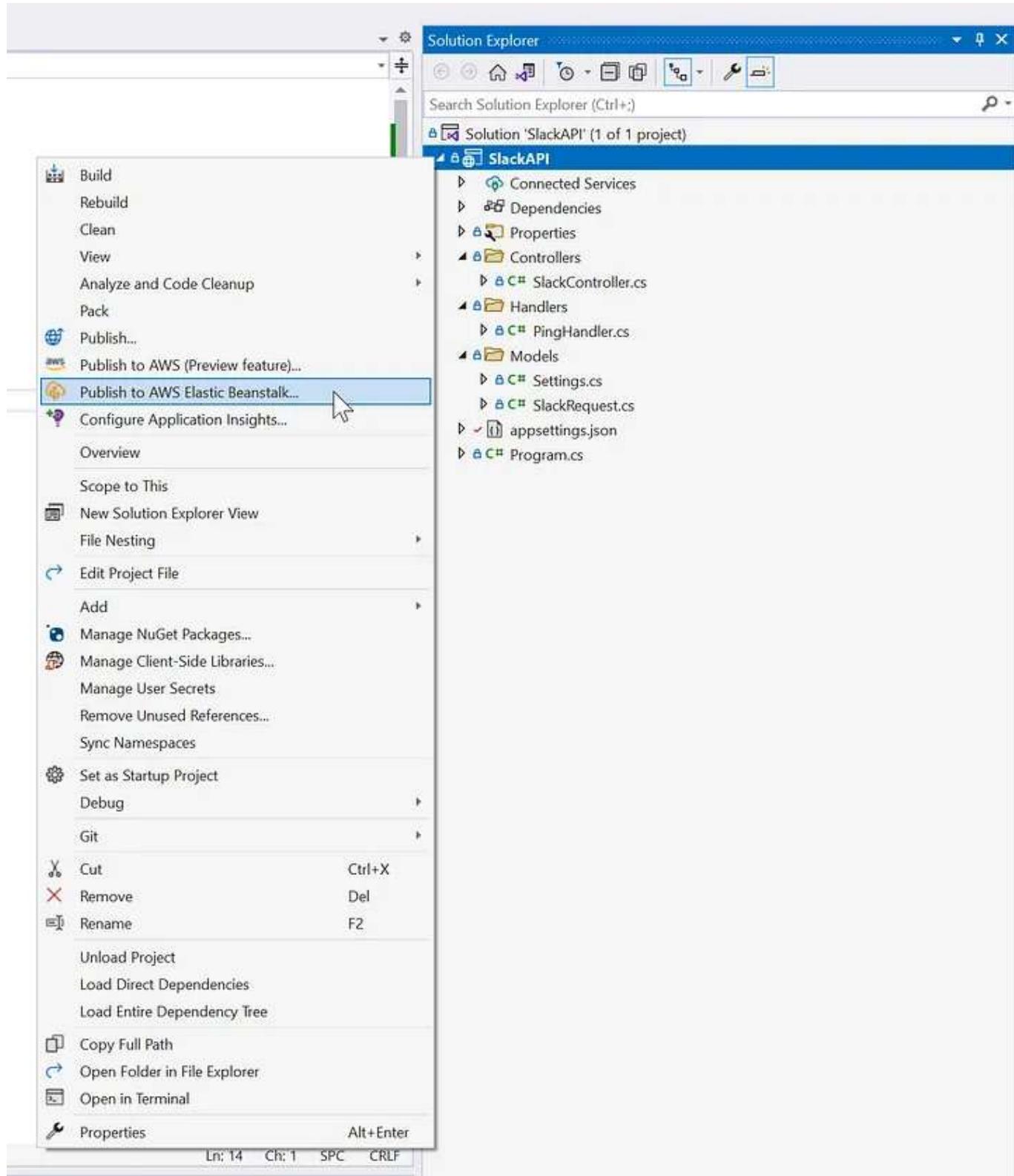


將 API 部署到 Elastic Beanstalk

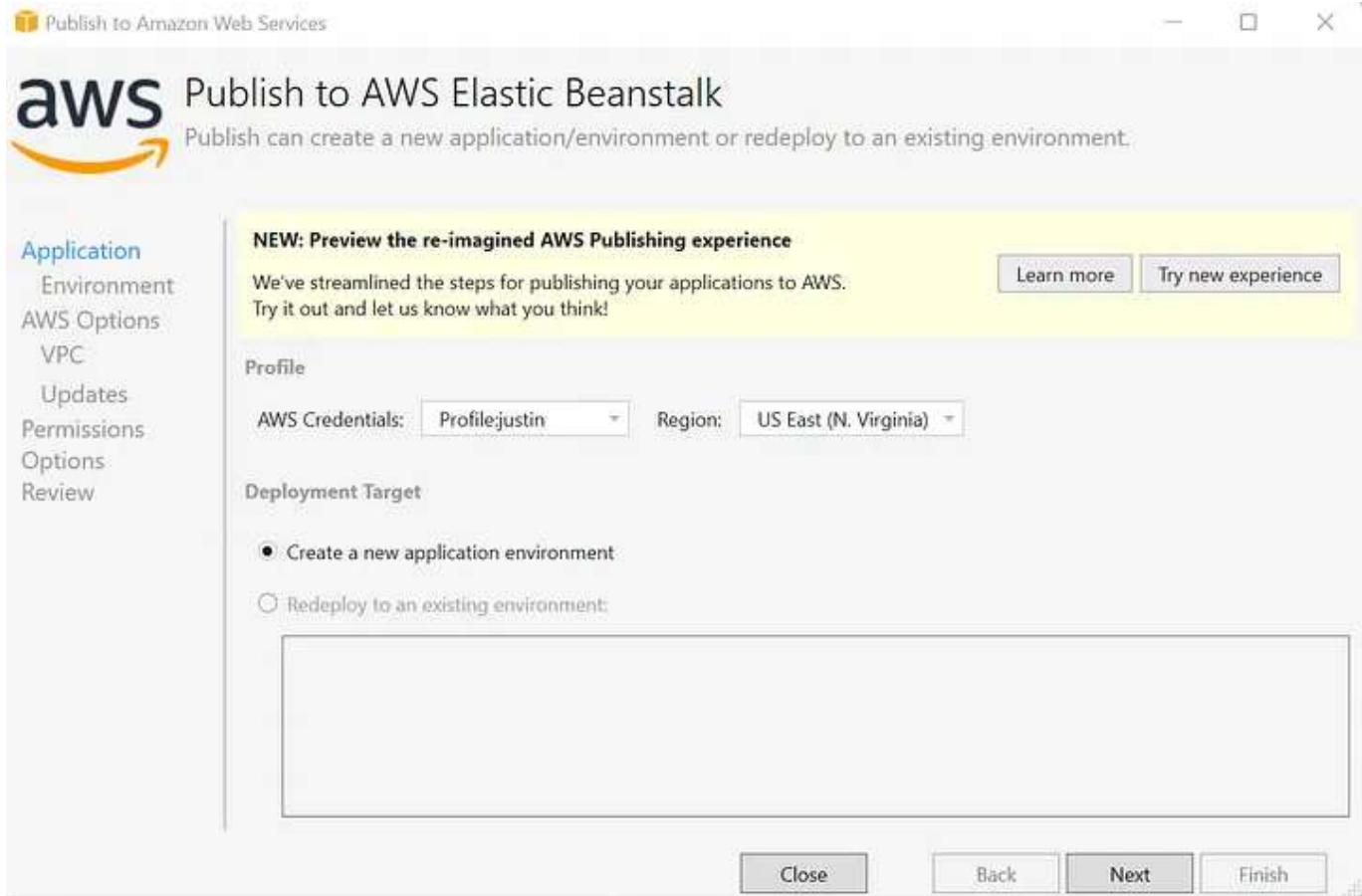
若要使第二個終結點正常工作，需要公開託管 API。有多種方法可以公開部署 API。本部分將使用 Elastic Beanstalk 服務將 API 部署到 AWS。

在 Visual Studio 中，打開「解決方案資源管理器」。

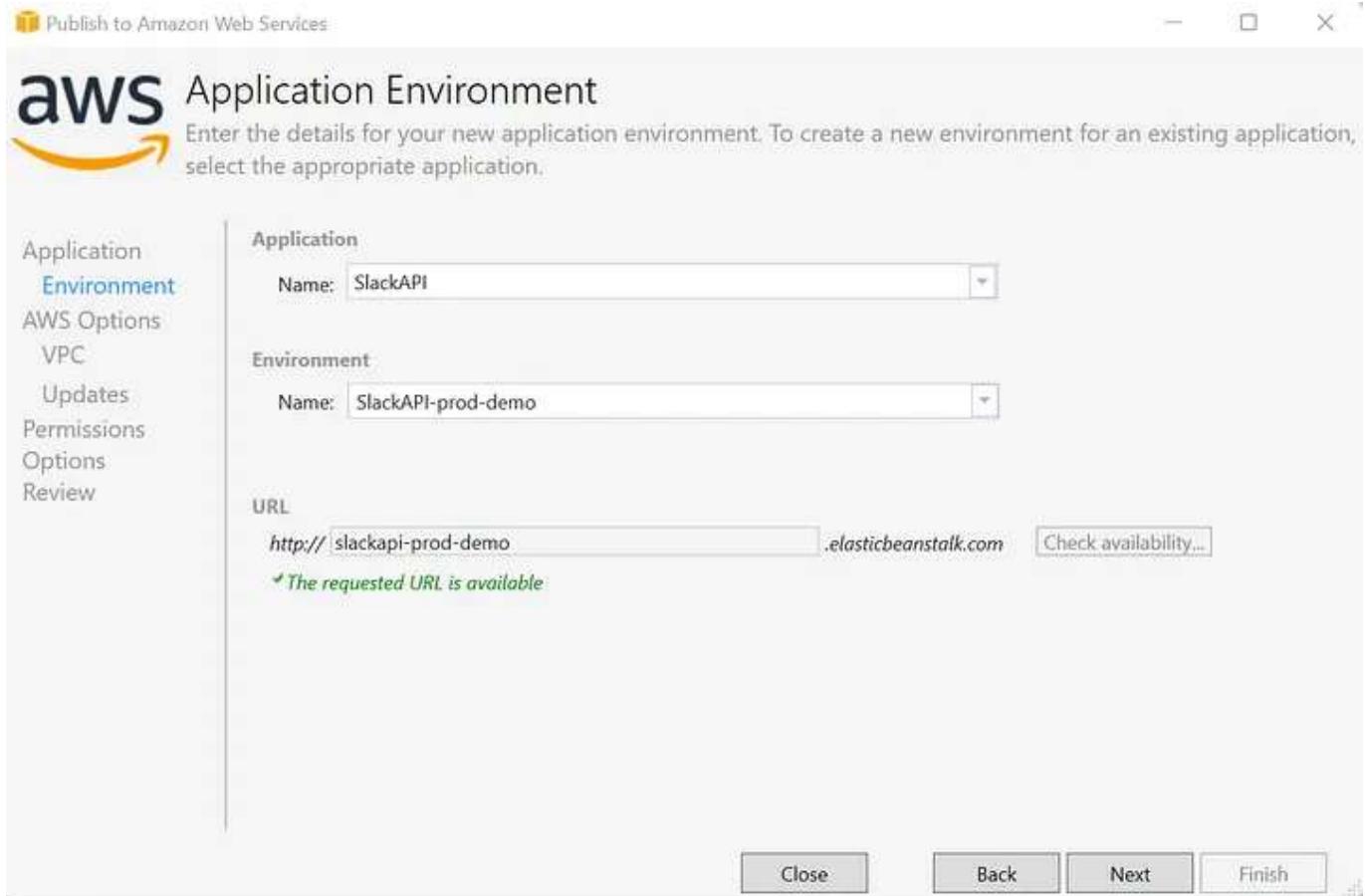
右鍵按下 SlackAPI 專案，然後選擇 Publish to AWS Elastic Beanstalk。如果您沒有看到此選項，請按照本指南進行操作。



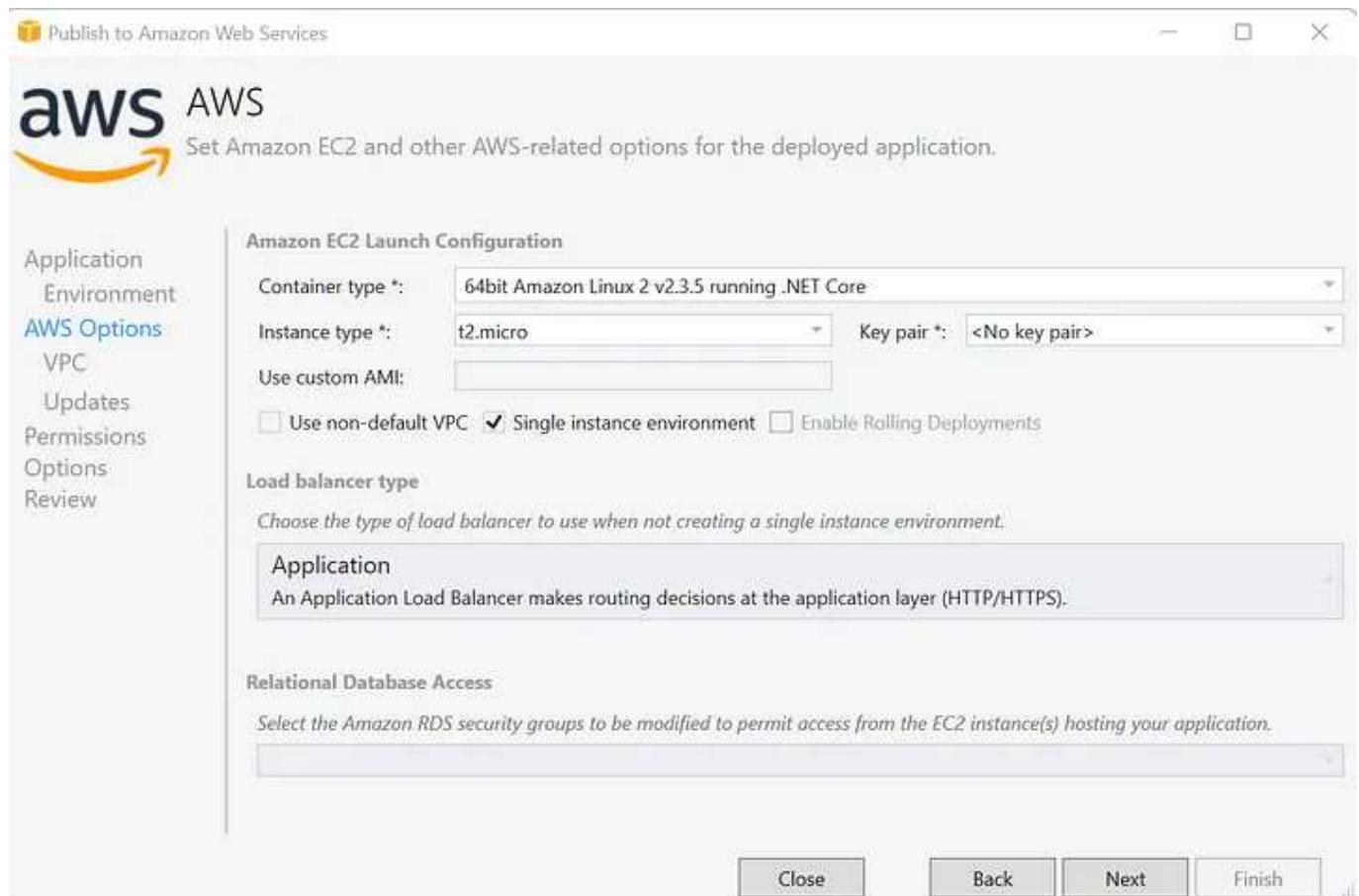
確保選取「創建新的應用程式環境」單選按鈕，然後按下一步按鈕。



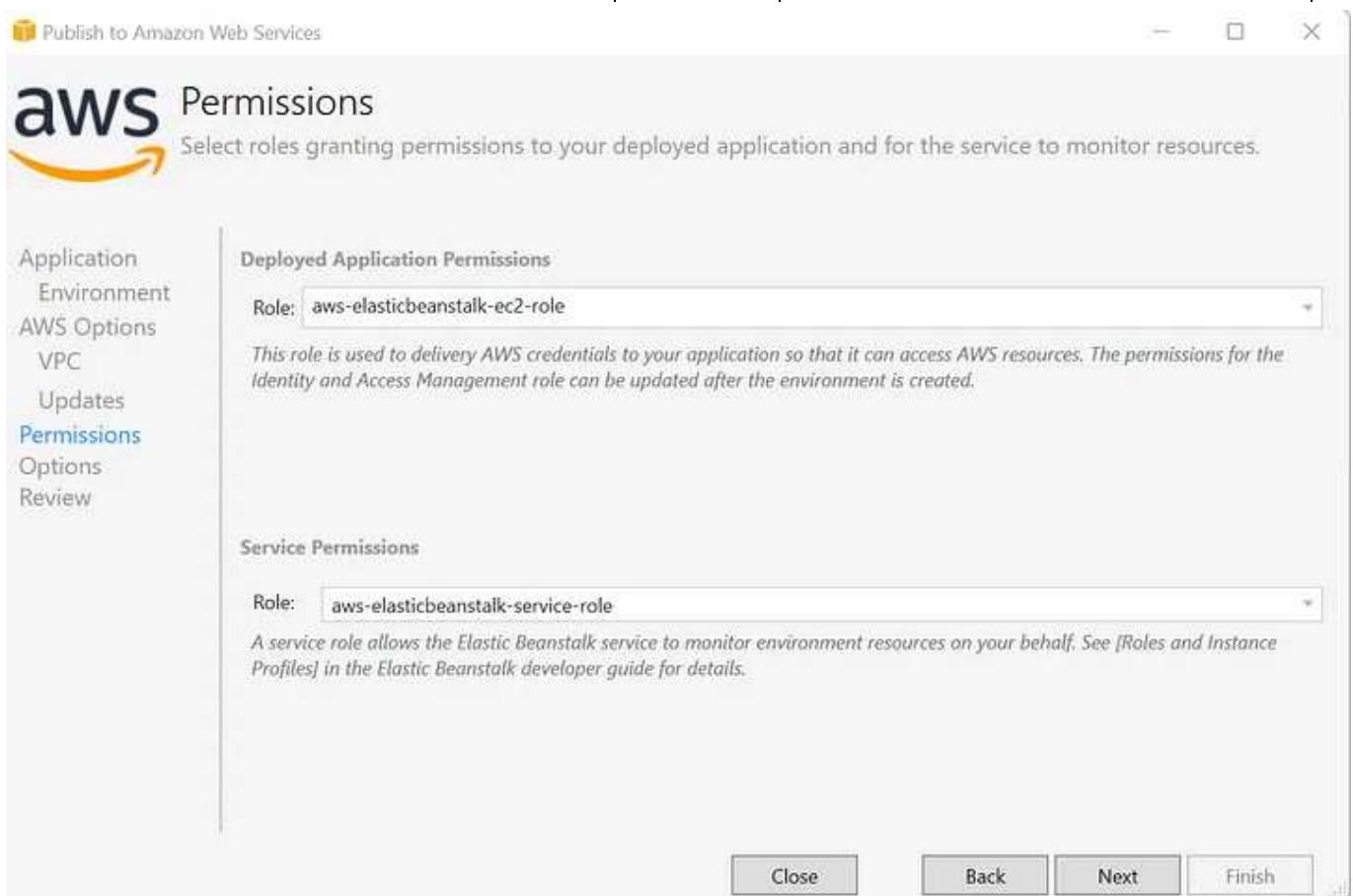
設置環境名稱。作者選擇了 SlackAPI-prod-demo，但讀者可以將其命名為任何名稱。按兩下一步按鈕。



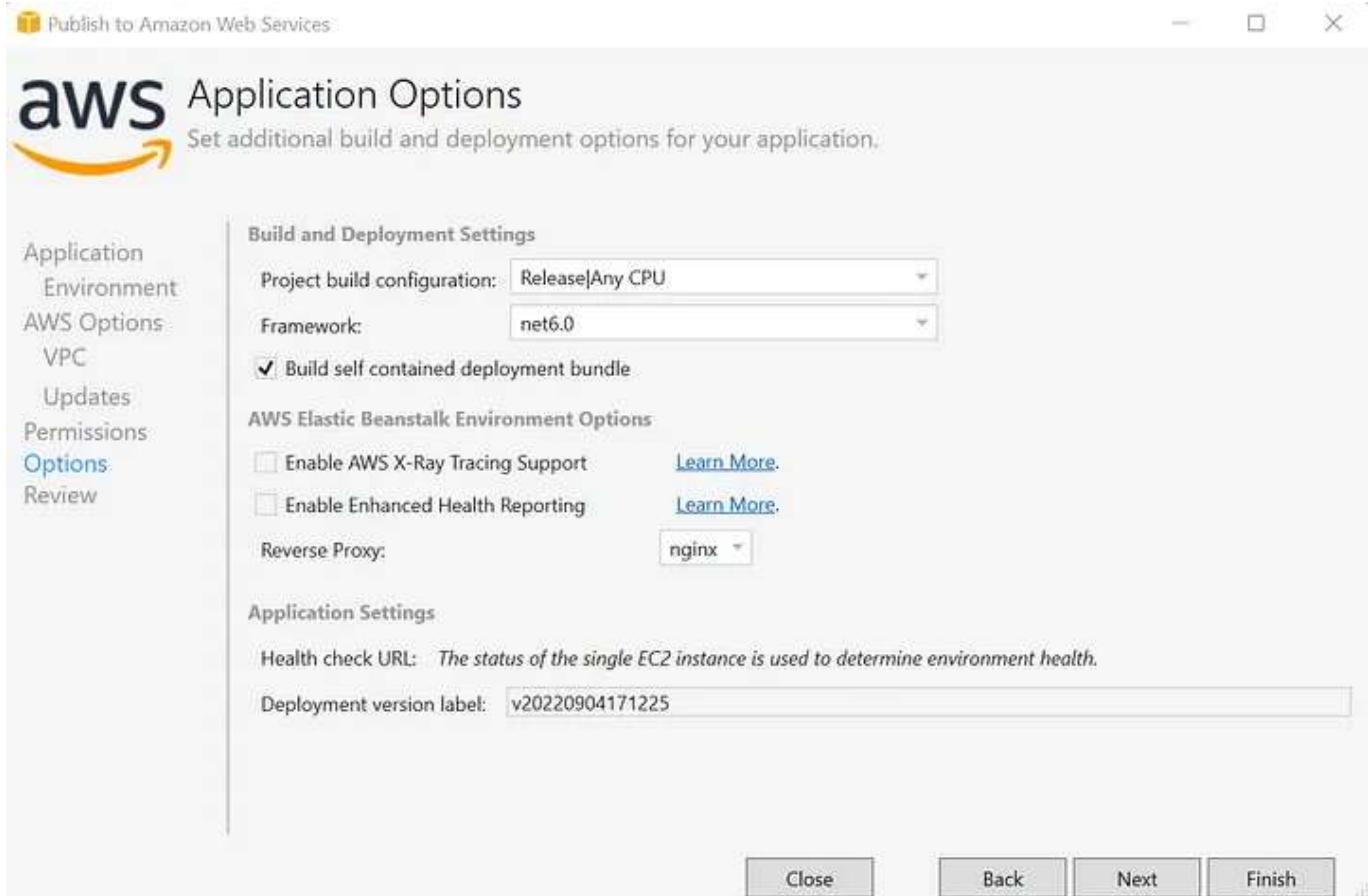
將實例類型設置為 t2.micro。這是最便宜的選擇，應該在一定程度上是免費的。按兩下一步按鈕。



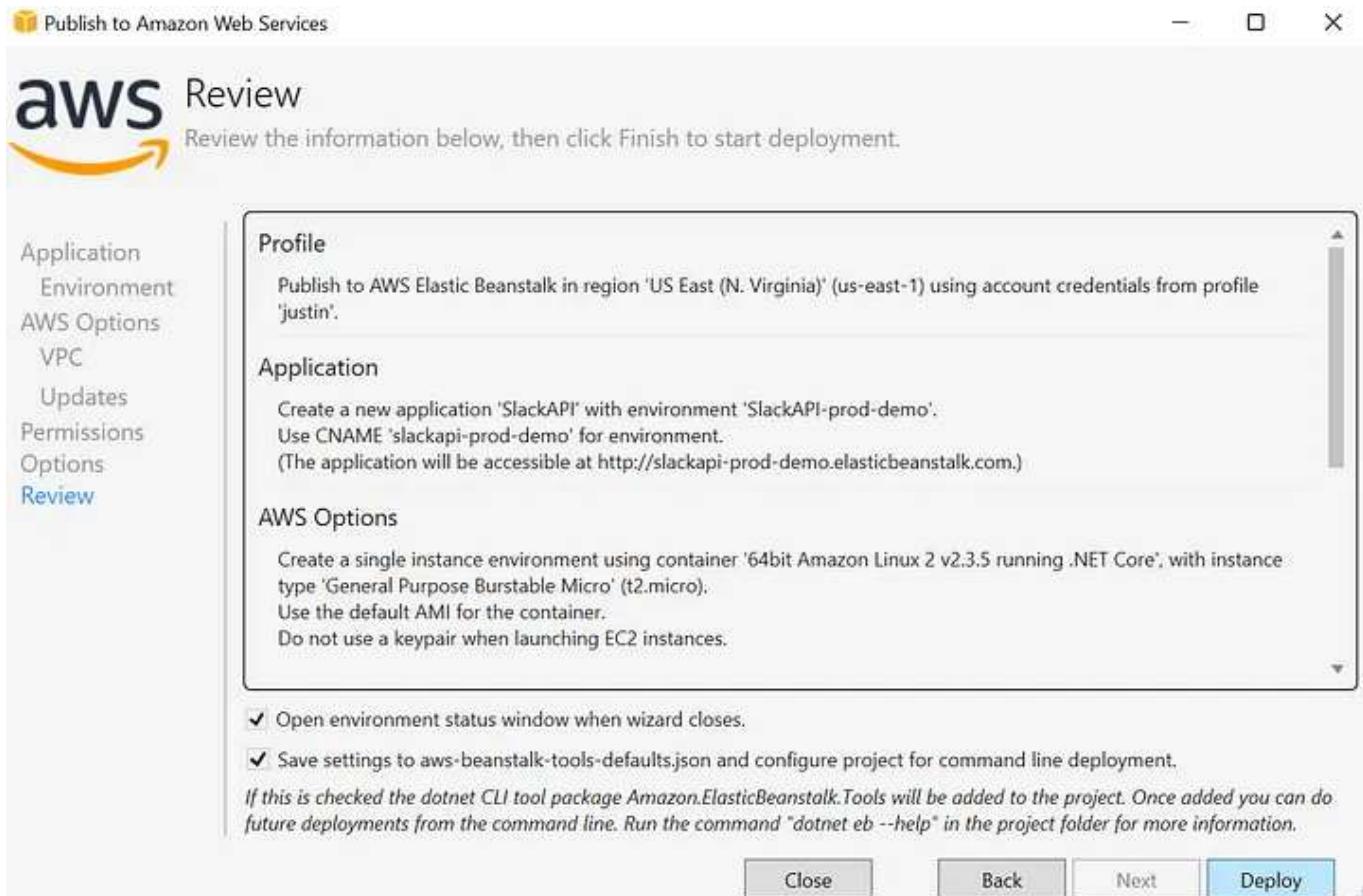
按兩下一步按鈕。



將“專案生成配置”設置為“發佈”|“任何 CPU”。按兩下“完成”按鈕。



按兩下“部署”按鈕。



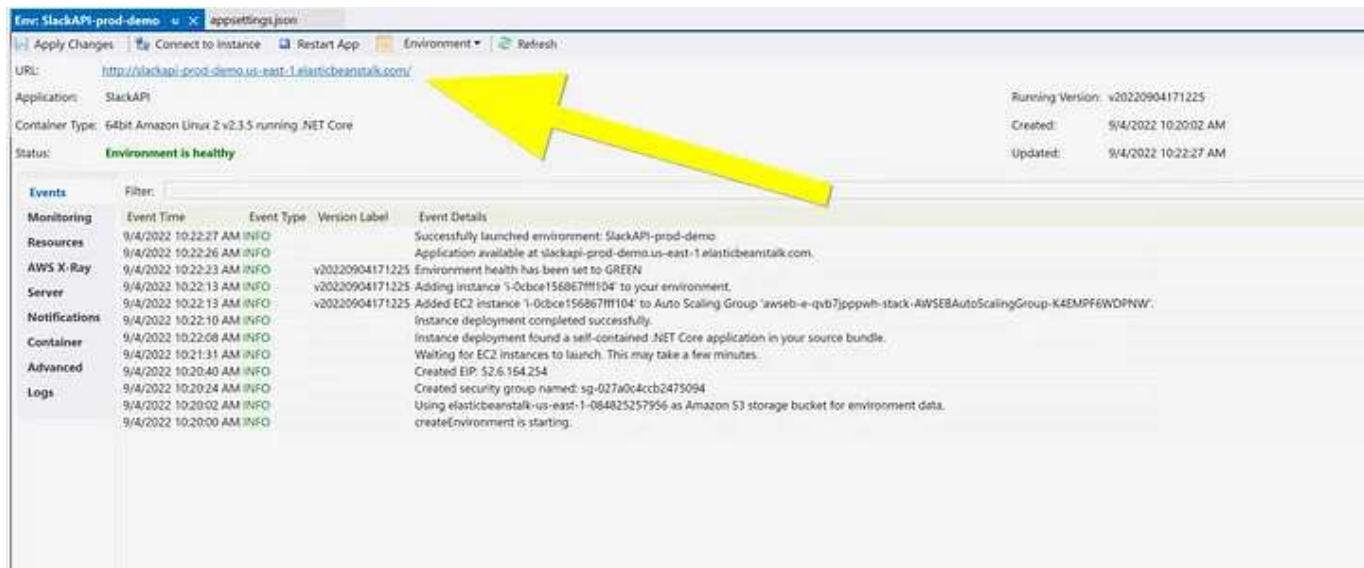
應出現一個新視窗。坐下來繼續觀看「狀態」欄位。幾分鐘後，它最終會說環境是健康的。

Event Type	Event Time	Version Label	Event Details
Monitoring	9/4/2022 10:20:24 AM	INFO	Created security group named: sg-027a0c4ccb2475094
Resources	9/4/2022 10:20:02 AM	INFO	Using elasticbeanstalk-us-east-1-084825257956 as Amazon S3 storage bucket for environment data.
AWS X-Ray	9/4/2022 10:20:00 AM	INFO	createEnvironment is starting.

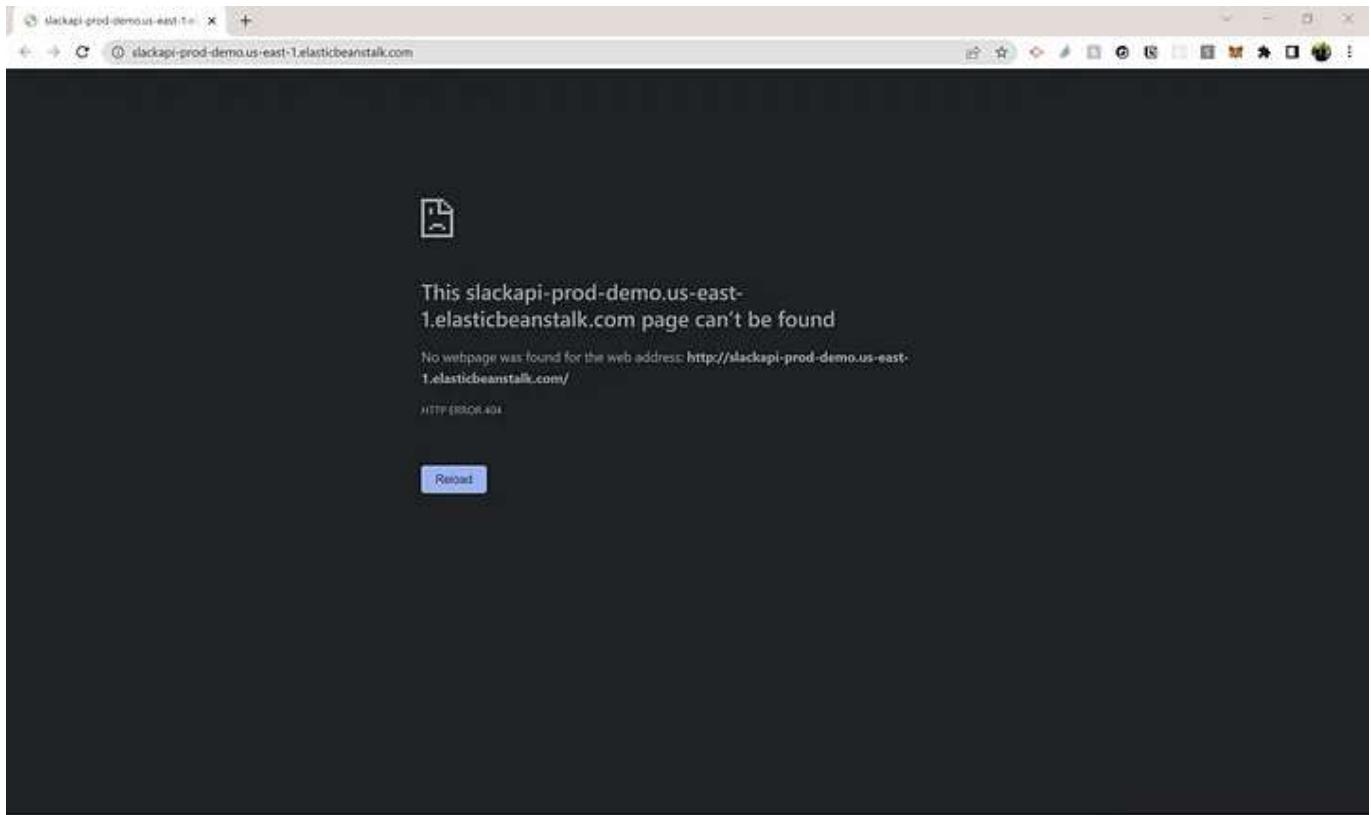
記下 URL。將此值複製到記事本或類似內容中。它將在後面的部分中使用。

在作者的情況下，URL 是 <http://slackapi-prod-demo.us-east-1.elasticbeanstalk.com/>

按兩下 URL。



它應該說找不到該頁面。



將“/Swagger”附加到 URL。

Swagger 頁面應可見。祝賀。該 API 現已公開發佈。現在，讓我們讓第二個端點正常工作。

A screenshot of a web browser displaying the SlackAPI v1 documentation via Swagger UI. The title bar shows the URL as 'Not secure slackapi-prod-demo.us-east-1.elasticbeanstalk.com/swagger/index.html'. The main content area is titled 'SlackAPI 1.0 OAS3' and shows two POST methods under the 'Slack' section: '/Slack/Submit' and '/Slack/Event'. Below these, there is a 'Schemas' section containing a 'SlackRequest' schema definition.

(選擇) 將子域附加到 Elastic Beanstalk

此部分可以完全跳過。讀者可能會注意到他們的URL名稱有點時髦。他們的網站 <http://slackapi-prod-demo.us-east-1.elasticbeanstalk.com/> 並沒有完全從舌頭上滾下來。此外，讀者還應該注意到該網站不支援 HTTPS/SSL。

如果讀者希望將自己的域綁定到其 Elastic Beanstalk API，並添加 HTTPS/SSL，則應遵循本指南。

使用 .NET Core Web API 將子域附加到 Elastic Beanstalk 環境

因此，假設讀取器已將 .NET Core Web API 部署到 AWS 上的 Elastic Beanstalk。不知道我在說什麼.....

medium.com 媒體網站

在指南結束時，他們的 API 應該在

<https://slack.justingerber.com/swagger>

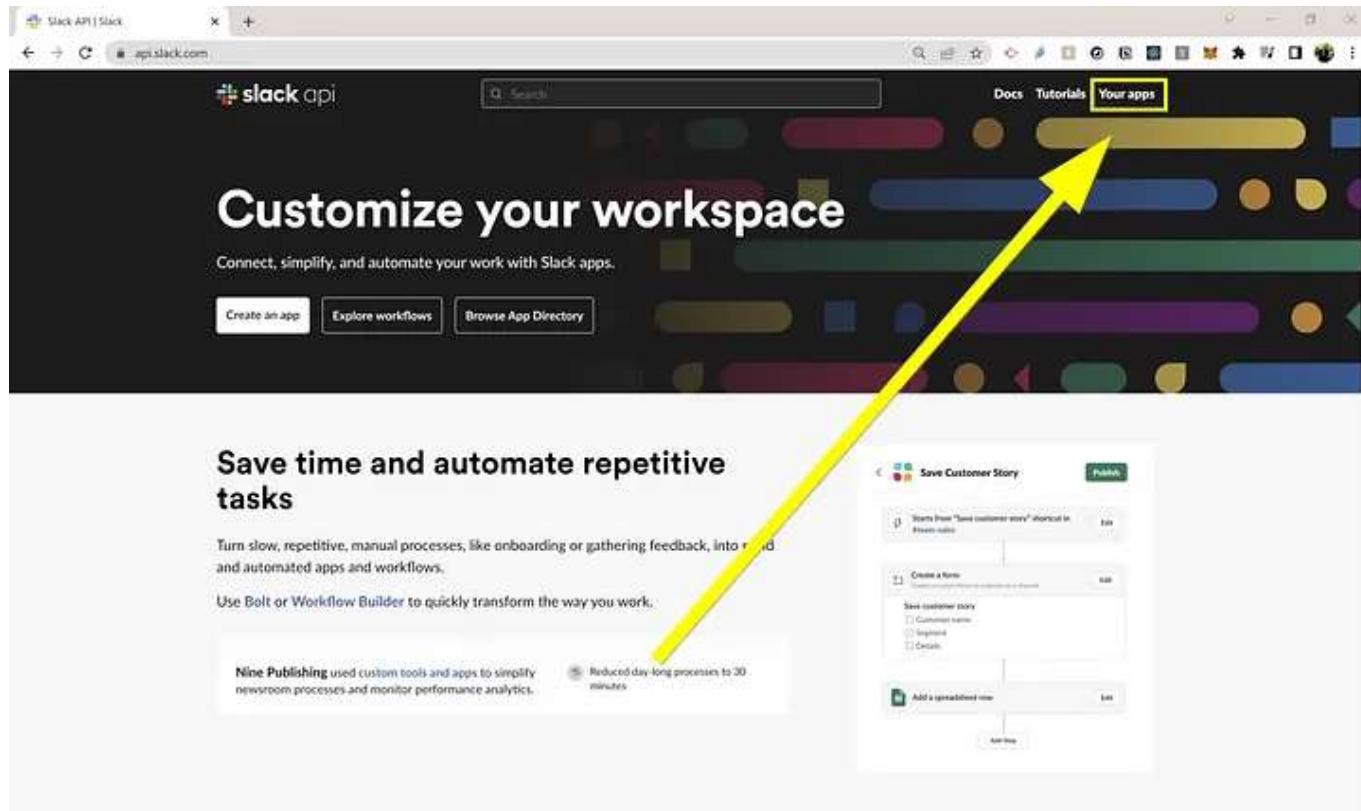
當然，這是假設讀者購買了 justingerber.com 域。這也是假設讀者遵循指南並在 GoDaddy 中附加了鬆弛的 NS 記錄。

為 SlackBot 添加請求 URL

在 #tennis Slack 頻道中寫入消息時，Slack 需要知道將消息發送到何處。本節將告訴 Slack 讀取器的 API 所在的位置。它將命中其第二個端點 POST /Slack/Event。

導航回 <https://api.slack.com>

按兩下您的應用程式。



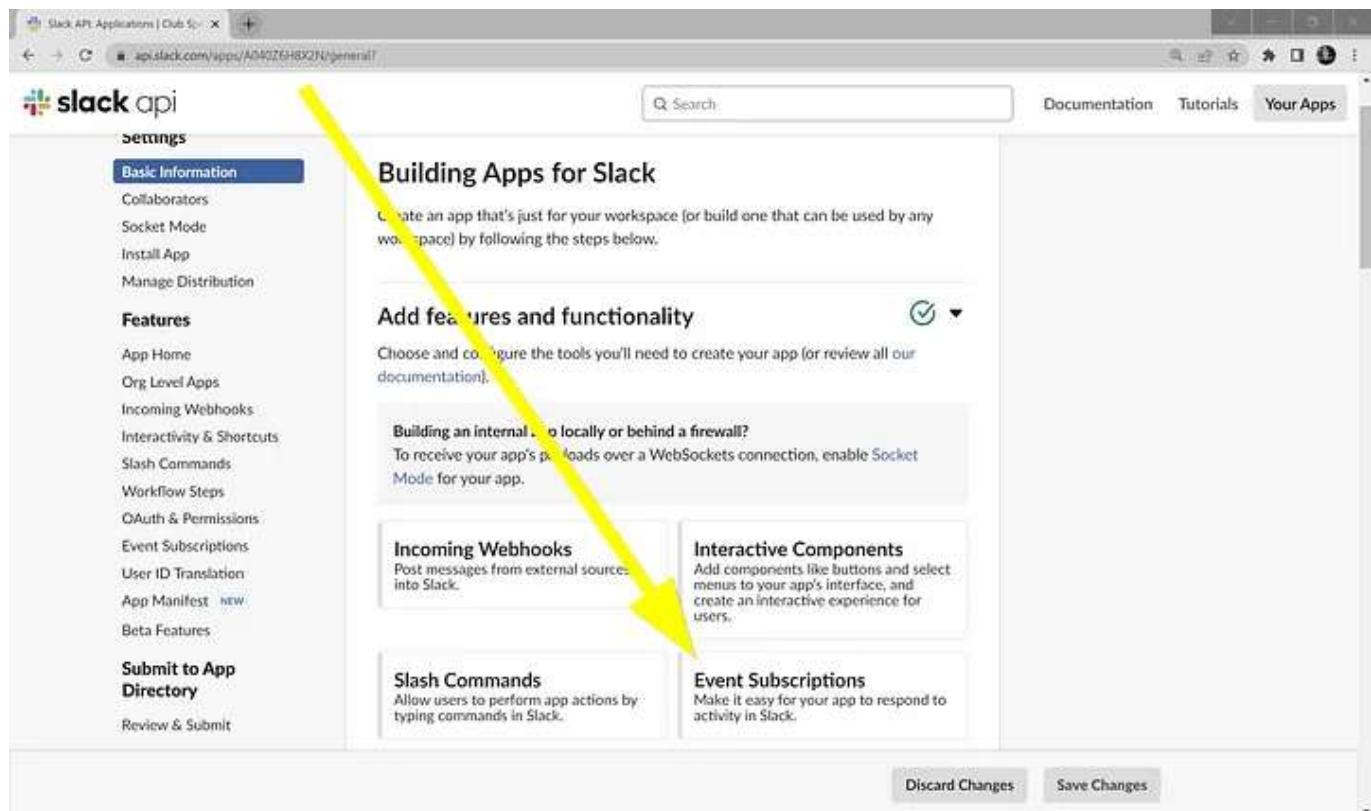
選擇「Slack 機器人應用程式」。

The screenshot shows the Slack API Applications interface. On the left, there's a sidebar with various documentation links. The main area is titled 'Your Apps' and contains a table with columns: App Name, Workspace, and Distribution Status. There are several entries, including 'Slack Bot' which is highlighted with a yellow arrow. The 'Distribution Status' for 'Slack Bot' is listed as 'Not distributed'.

按兩下左側的「基本資訊」選項卡。然後按下「添加特性和功能」。

The screenshot shows the 'Basic Information' page for the 'Slack Bot' application. On the left, there's a sidebar with various settings options. The 'Basic Information' tab is selected and highlighted with a yellow box. The main content area has a heading 'Building Apps for Slack' with a sub-section 'Add features and functionality' which is also highlighted with a yellow arrow. Below it are other sections: 'Install your app' and 'Manage distribution'. At the bottom, there's a section titled 'App Credentials' with a note about keeping them secret. At the very bottom are 'Discard Changes' and 'Save Changes' buttons.

按兩下「Event Subscriptions」（事件訂閱）。



按兩下開關並確保它已打開。

Slack API Applications | Club Slack

slack api

Event Subscriptions

Settings

- Basic Information
- Collaborators
- Socket Mode
- Install App
- Manage Distribution

Features

- App Home
- Org Level Apps
- Incoming Webhooks
- Interactivity & Shortcuts
- Slash Commands
- Workflow Steps
- OAuth & Permissions
- Event Subscriptions** (selected)
- User ID Translation
- App Manifest (NEW)
- Beta Features

Enable Events

Your app can subscribe to be notified of events in Slack (for example, when a user adds a reaction or creates a file) at a URL you choose. [Learn more](#).

On

Request URL

We'll send HTTP POST requests to this URL when events occur. As soon as you enter a URL, we'll send a request with a `challenge` parameter, and your endpoint must respond with the challenge value. [Learn more](#).

New event authorization format

Recent changes to Events API payloads
The Events API now sends information about authorized users and workspaces in a new, compact format. [Learn more](#).

Subscribe to bot events

應出現一個新部分。

Slack API Applications | Club Slack

slack api

Event Subscriptions

Settings

- Basic Information
- Collaborators
- Socket Mode
- Install App
- Manage Distribution

Features

- App Home
- Org Level Apps
- Incoming Webhooks
- Interactivity & Shortcuts
- Slash Commands
- Workflow Steps
- OAuth & Permissions
- Event Subscriptions** (selected)
- User ID Translation
- App Manifest (NEW)
- Beta Features

Enable Events

Your app can subscribe to be notified of events in Slack (for example, when a user adds a reaction or creates a file) at a URL you choose. [Learn more](#).

On

Request URL

We'll send HTTP POST requests to this URL when events occur. As soon as you enter a URL, we'll send a request with a `challenge` parameter, and your endpoint must respond with the challenge value. [Learn more](#).

New event authorization format

Recent changes to Events API payloads
The Events API now sends information about authorized users and workspaces in a new, compact format. [Learn more](#).

Subscribe to bot events

調用公開託管的 API 的 URL。它應該在記事本或類似的東西中。

作者的 URL 已 <http://slackapi-prod-demo.us-east-1.elasticbeanstalk.com>

讀者的 URL 會有所不同。

讀者應將 /Slack/Event 附加到其URL中。

作者提供的請求URL為

<http://slackapi-prod-demo.us-east-1.elasticbeanstalk.com/Slack/Event>

讀者應提供其請求URL。swagger 一詞不應出現在讀者的請求URL中。

如果 URL 已正確提供，則它應該顯示「已驗證」，並帶有綠色複選標記。在完成此操作之前不要繼續。

The screenshot shows the 'Event Subscriptions' settings page for a Slack app. The 'Request URL' field is highlighted with a yellow box, showing the URL <http://slackapi-prod-demo.us-east-1.elasticbeanstalk.com/Slack/Event>. A green 'Verified' badge with a checkmark is visible next to the URL. The 'Enable Events' toggle switch is turned 'On'. The 'New event authorization format' section is expanded, showing a note about recent changes to Events API payloads. At the bottom, there are 'Discard Changes' and 'Save Changes' buttons.

向下滾動並展開「訂閱機器人事件」部分。

The screenshot shows the Slack API Applications settings page. On the left, there's a sidebar with various features like App Home, Org Level Apps, Incoming Webhooks, etc., and the 'Event Subscriptions' option is selected. The main area has a 'Request URL' field set to 'http://slackapi-prod-demo.us-east-1.elasticbeanstalk.com/Slack/Event'. Below it, a note says: 'We'll send HTTP POST requests to this URL when events occur. As soon as you enter a URL, we'll send a request with a challenge parameter, and your endpoint must respond with the challenge value. Learn more.' A yellow arrow points from this note to the 'Subscribe to bot events' section. This section contains a table with one row: 'Event Name' (empty), 'Description' (empty), and 'Required Scope' (empty). A button labeled 'Add Bot User Event' is visible. At the bottom, there's a link 'Subscribe to events on behalf of users' and two buttons: 'Discard Changes' and 'Save Changes'.

按兩下「添加機器人使用者事件」按鈕。

The screenshot shows the Slack API Applications interface for managing event subscriptions. On the left sidebar, under the 'Event Subscriptions' section, the 'Event Subscriptions' tab is selected. In the main content area, there's a 'New event authorization format' notice. Below it, the 'Subscribe to bot events' section is visible. A yellow arrow points to the 'Add Bot User Event' button. At the bottom right of the page are 'Discard Changes' and 'Save Changes' buttons.

選擇 message.channels 選項。

This screenshot shows the same Slack API Applications interface as the previous one, but with a yellow arrow pointing to the 'message.channels' event in the list. The 'message.channels' event is described as 'A message was posted to a channel'. The 'Discard Changes' and 'Save Changes' buttons are also visible at the bottom right.

按下保存更改按鈕。

The screenshot shows the 'Event Subscriptions' section of the Slack API Applications interface. A yellow arrow points from the 'Save Changes' button at the bottom right towards the 'Discard Changes' button, indicating the action to take after saving the configuration.

New event authorization format

Recent changes to Events API payloads
The Events API now sends information about authorized users and workspaces in a new, compact format. [Learn more](#).

Subscribe to bot events

Apps can subscribe to receive events the bot user has access to (like new messages in a channel). If you add an event here, we'll add the necessary OAuth scope for you.

Event Name	Description	Required Scope
message.channels	A message was posted to a channel	channels:history

[Add Bot User Event](#)

Subscribe to events on behalf of users

You may also want your app to receive events related to users who have authorized the app (and conversations they're part of). If you add an event here, we'll add the necessary OAuth scope for you.

[Discard Changes](#) [Save Changes](#)

按兩下重新安裝應用連結。

The screenshot shows the same 'Event Subscriptions' section after changes have been made. A yellow arrow points from the 'Save Changes' button at the bottom right towards the 'Discard Changes' button, indicating the action to take after saving the configuration.

New event authorization format

Recent changes to Events API payloads
The Events API now sends information about authorized users and workspaces in a new, compact format. [Learn more](#).

Subscribe to bot events

Apps can subscribe to receive events the bot user has access to (like new messages in a channel). If you add an event here, we'll add the necessary OAuth scope for you.

Event Name	Description	Required Scope
message.channels	A message was posted to a channel	channels:history

[Add Bot User Event](#)

Subscribe to events on behalf of users

You may also want your app to receive events related to users who have authorized the app (and conversations they're part of). If you add an event here, we'll add the necessary OAuth scope for you.

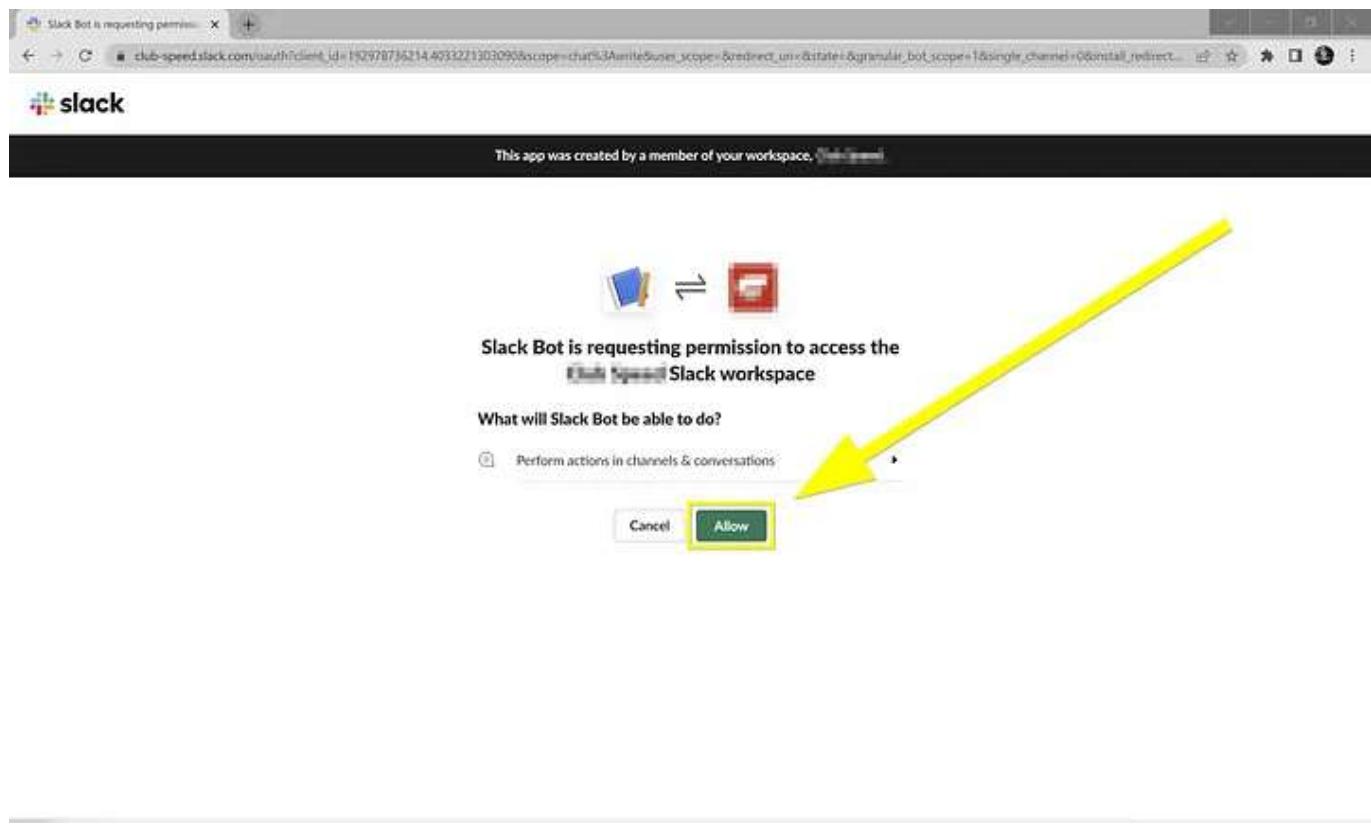
Event Name	Description	Required Scope
(No events added yet)		

[Add Workspace Event](#)

App unfurl domains

[Discard Changes](#) [Save Changes](#)

按兩下允許按鈕。



測試第二個終點

打開 Visual Studio 2022 。

查看 PingHandler.cs 檔。

請注意，MessageEvent 被傳遞到第 16 行的 Handle 方法中。

在第 18 行，代碼正在檢查消息是否包含單詞「ping」。

如果消息確實包含單詞 ping，則在第 20 行，代碼會將帶有 pong 消息寫回通道。

```
1  using SlackNet;
2  using SlackNet.Events;
3  using SlackNet.WebApi;
4
5  namespace SlackAPI.Handlers
6  {
7      public class PingHandler : IEventHandler<MessageEvent>
8      {
9
10         private readonly ISlackApiClient _slack;
11         public PingHandler(ISlackApiClient slack)
12         {
13             _slack = slack;
14         }
15
16         public async Task Handle(MessageEvent slackEvent)
17         {
18             if (slackEvent.Text.Contains("ping"))
19             {
20                 await _slack.Chat.PostMessage(new Message
21                 {
22                     Text = "pong",
23                     Channel = slackEvent.Channel
24                 }).ConfigureAwait(false);
25             }
26         }
27     }
28 }
```

PingHandler hosted with ❤ by GitHub

[view raw](#)

打開 Slack 應用程式。

打開 #tennis Slack 頻道。

鍵入 ping 並按回車鍵。

如果一切順利，那麼 Slackbot 應該回復 pong。

tennis

You created this channel today. This is the very beginning of the # tennis channel.

[Add description](#) [Add people](#) [Send emails to channel](#)

Today ▾

 **Justin Gerber** 10:34 PM

joined #tennis along with Slack Bot.

B I S | ⌂ | ⌂ ⌂ | ⌂ ⌂ | ⌂ ⌂ | ⌂ ⌂

|Message #tennis I

+ | ⌂ ⌂ | ☺ ☺ | ☺ ☺ | Aa

▶ | ▾

最後的思考

我想再次向 soxtoby 的 SlackNet 儲存庫致以崇高的敬意。在評論中讓我知道你們的想法或如何改進。快樂的建築。

鬆弛

鬆弛機器人

夏普

自主技術

Elastic Beanstalk



由賈斯汀·格伯 (Justin Gerber) 撰寫

15 粉絲

跟隨