

MS-4010

ラボ（演習）の手順解説

※ラボ1～5がありますが
本コースではラボ1、2のみ利用します

ラボ（演習）の概要

- ・ラボを起動
- ・画面の右側に表示される手順書に従い操作を実行

ラボ1

宣言型エージェントの開発

ラボ1

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/89fd4caf-4ebd-498e-937b-c2f02e1d085f

Admin

Password

Admin

Student

ENG

前 次 残り5時56分

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用して Microsoft 365 Copilotの宣言的エージェントを構築する

指示 リソース

免責事項 クリックして表示または非表示にします

⚠ このコースで利用できるテナントは、このコースのハンズオン ラボをサポートする目的です。

このコースでは、テナントを共有したり、ハンズオン ラボ以外の目的で使用したりしないでください。このラボで使用されるテナントは試用版テナントであり、クラスが終了した後は使用またはアクセスできず、拡張機能の対象になりません。

テナントを有料サブスクリプションに変換することはできません。このコースの一環として取得したテナントは Microsoft Corporation の財産であり、いつでもアクセス権とリポジトリを取得する権利を留保します。

認証情報

すべての認証情報は、ラボインターフェースの リソース タブにあります。

仮想マシン

MS-4010-CLIENT01

ユーザー名: Admin

パスワード: Pa55w.rd

Microsoft 365

ユーザー名: admin@WWLx346204.onmicrosoft.com

パスワード: (9FOF0)Vl2=cm4Q\$SI0zk9]gqcV^51%

テナント接頭辞: WWLx346204

次へ を選択してラボを進めてください。

指示 リソース

免責事項 クリックして表示または非表示にします ▶

⚠ このコースで利用できるテナントは、このコースのハンズオンラボをサポートする目的です。

このコースでは、テナントを共有したり、ハンズオンラボ以外の目的で使用したりしないでください。このラボで使用されるテナントは試用版テナントであり、クラスが終了した後は使用またはアクセスできず、拡張機能の対象になりません。

テナントを有料サブスクリプションに変換することはできません。このコースの一環として取得したテナントは Microsoft Corporation の財産であり、いつでもアクセス権とリポジトリを取得する権利を留保します。

認証情報

すべての認証情報は、ラボインターフェースの **リソース** タブにあります。

仮想マシン

MS-4010-CLIENT01

ユーザー名: Admin

パスワード: Pa55w.rd

Microsoft 365

ユーザー名: admin@WWLx346204.onmicrosoft.com

パスワード: (9FOF0)Vl2=cm4Q\$SI0zk9]gqcV^51%

テナント接頭辞: WWLx346204

次へ を選択してラボを進めてください。

← 前 次 →

残り 5時54分

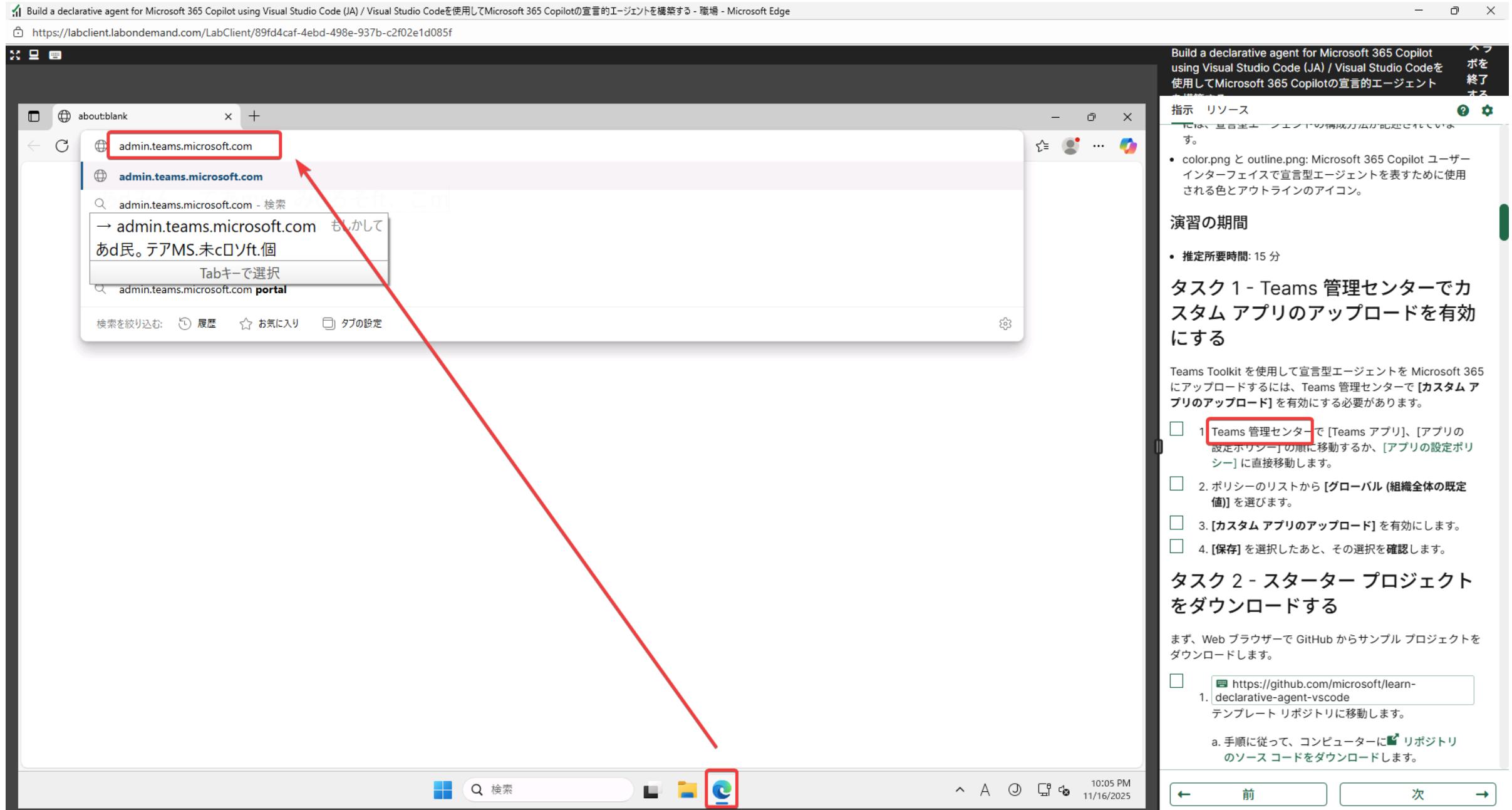


Admin

•

Welcome

カスタムアプリのアップロードを有効化する



The screenshot shows a Microsoft Edge browser window with two main panes. The left pane displays a Microsoft sign-in dialog for 'WWL M365 Enterprise with Calling'. The right pane shows a configuration interface for building a declarative agent.

Left Pane (Sign-in Dialog):

- Microsoft logo
- サインイン (Sign in)
- admin@WWLx346204.onmicrosoft.com (Email address, highlighted with a red box)
- アカウントにアクセスできない場合 (If you can't access your account)
- 次へ (Next) button (highlighted with a red box)
- サインイン オプション (Sign-in options) link

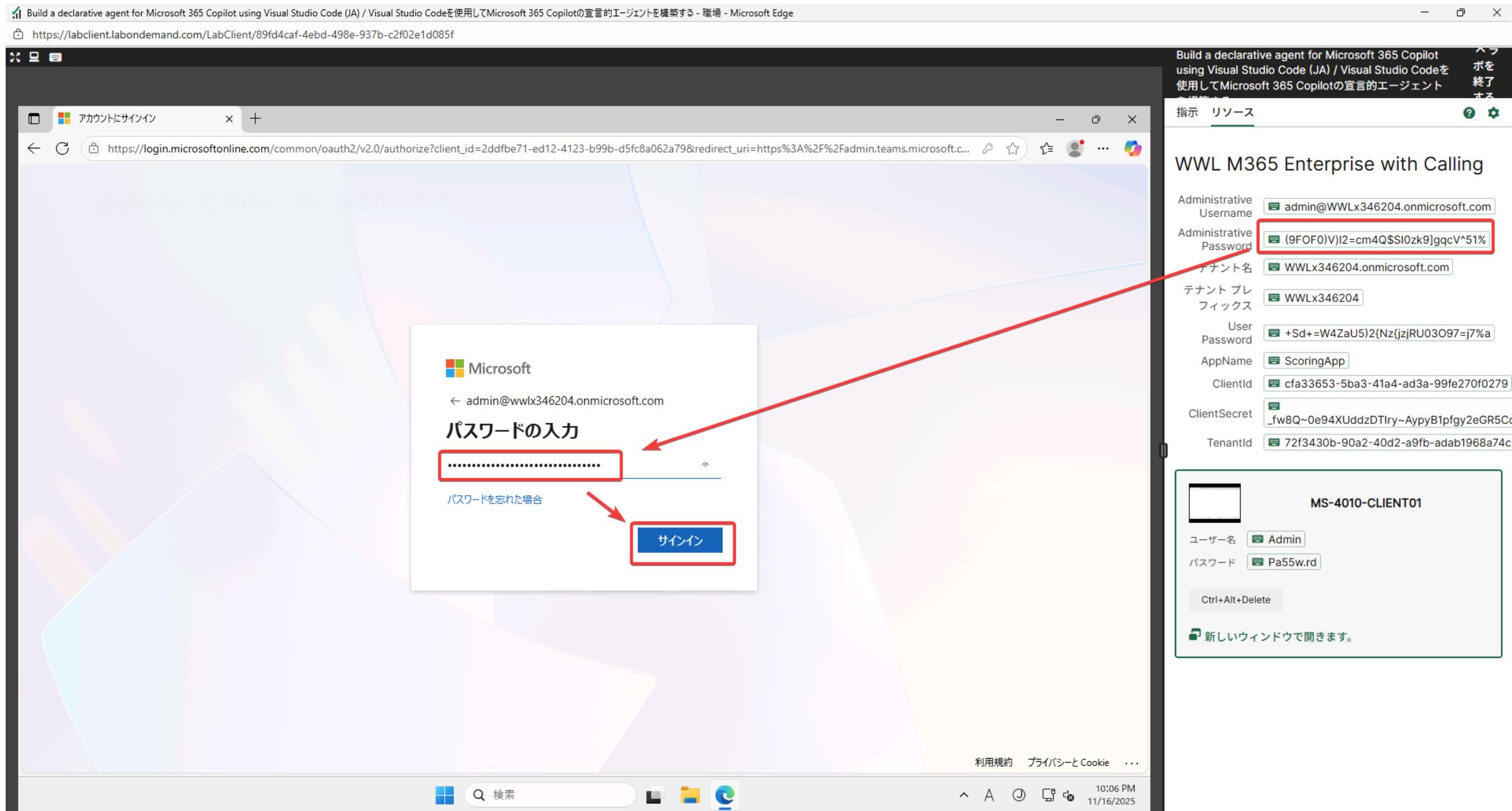
Right Pane (Configuration Interface):

- 指示 (Instructions) tab (highlighted with a red box)
- リソース (Resources) tab (highlighted with a red box)
- WWL M365 Enterprise with Calling
- Administrative Username: admin@WWLx346204.onmicrosoft.com (highlighted with a red box)
- Administrative Password: (9FOFO)V)I2=cm4Q\$SI0zk9]gqcV^51% (highlighted with a red box)
- テナント名 (Tenant Name): WWLx346204 (highlighted with a red box)
- テナント プレフィックス (Tenant Prefix): WWLx346204 (highlighted with a red box)
- User Password: +Sd+=W4ZaU5)2{Nz{jzjRU03097=j7%a (highlighted with a red box)
- AppName: ScoringApp (highlighted with a red box)
- ClientId: cfa33653-5ba3-41a4-ad3a-99fe270f0279 (highlighted with a red box)
- ClientSecret: _fw8Q~0e94XUddzDTlry~AypyB1pfgy2eGR5Cc (highlighted with a red box)
- TenantId: 72f3430b-90a2-40d2-a9fb-adab1968a74c (highlighted with a red box)

Bottom Right Panel:

- MS-4010-CLIENT01 (Machine name)
- ユーザー名 (User Name): Admin (highlighted with a red box)
- パスワード (Password): Pa55w.rd (highlighted with a red box)
- Ctrl+Alt+Delete button
- 新しいウィンドウで開きます。 (Open in new window) button

Bottom status bar: 検索 (Search), 10:06 PM, 11/16/2025



アカウントにサインイン

https://login.microsoftonline.com/common/login

Microsoft

admin@wwlx346204.onmicrosoft.com

サインインの状態を維持しますか?

これにより、サインインを求められる回数を減らすことができます。

今後このメッセージを表示しない

いいえ はい

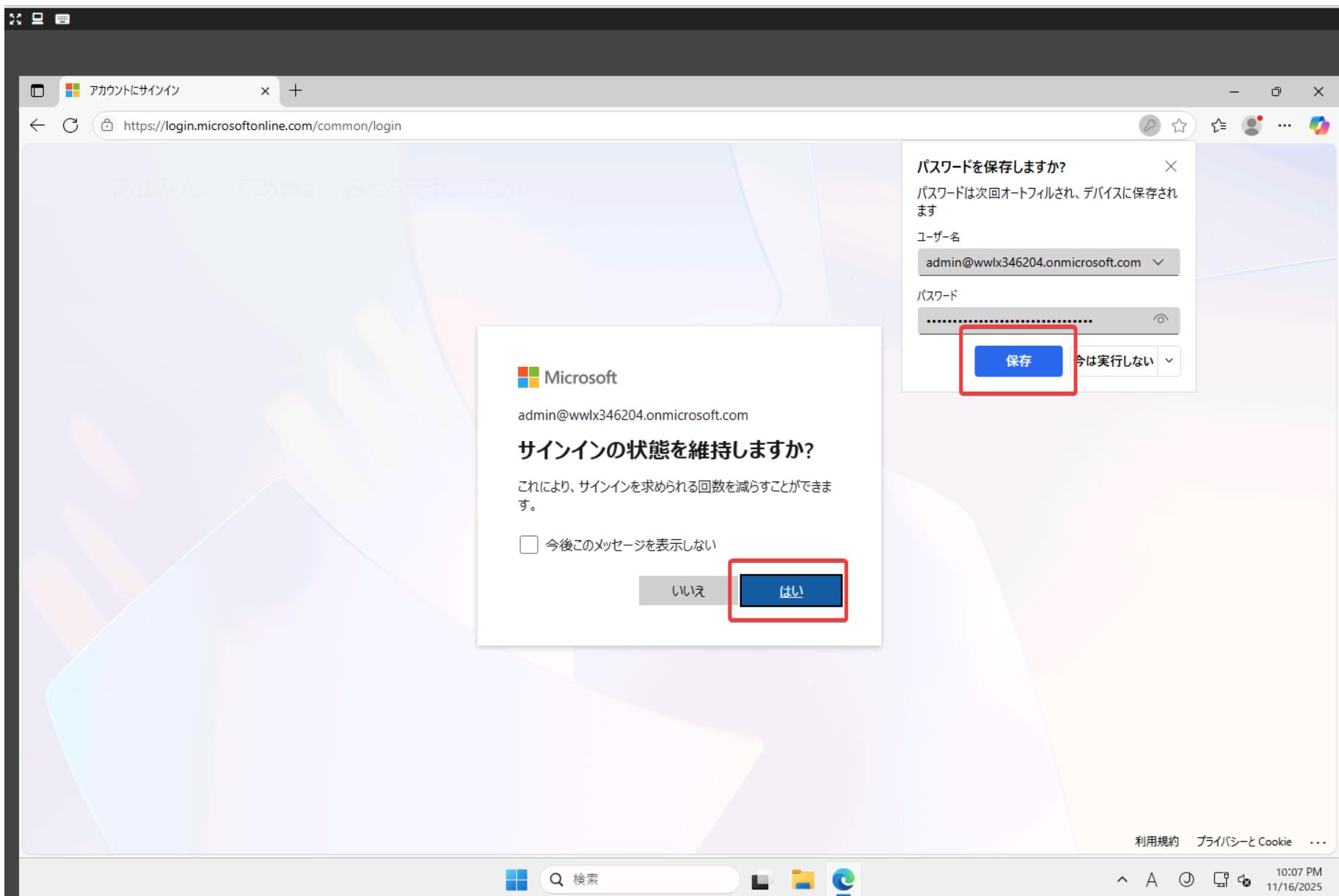
パスワードを保存しますか?

パスワードは次回オートフィルされ、デバイスに保存されます

ユーザー名
admin@wwlx346204.onmicrosoft.com

パスワード

保存 今は実行しない



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - Microsoft Edge

指示 リソース

WWL M365 Enterprise with Calling

Administrative Username	admin@WWLx346204.onmicrosoft.com
Administrative Password	(9FOFO)V)I2=cm4Q\$SI0zk9]gqcV^51%
テナント名	WWLx346204.onmicrosoft.com
テナント ブレーフィックス	WWLx346204
User Password	+Sd+=W4ZaU5)2{Nz{jzjRU03097=j7%a
AppName	ScoringApp
ClientId	cfa33653-5ba3-41a4-ad3a-99fe270f0279
ClientSecret	_fw8Q~0e94XUddzDTlry~AypyB1pfgy2eGR5Cc
TenantId	72f3430b-90a2-40d2-a9fb-adab1968a74c

MS-4010-CLIENT01

ユーザー名 Admin

パスワード Pa55w.rd

Ctrl+Alt+Delete

新しいウィンドウで開きます。



App setup policies - Microsoft Teams

https://admin.teams.microsoft.com/policies/app-setup

Microsoft Teams admin center

App setup policies

App setup policies control how apps are made available to a user with the Teams app. Use the Global (Org-wide default) policy and customize it or create custom policies and assign them to a set of users. [Learn more](#)

App setup policies summary	
2	0
Default policies	Custom policies
Manage policies	Group policy assignment

+ Add Edit Duplicate Delete Reset Global policy Manage users Search

Name ↑	Description	Custom policy
Global (Org-wide default)		No
FirstLineWorker	This is a default app set...	No

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

指示 リソース

- color.png と outline.png: Microsoft 365 Copilot ユーザーインターフェイスで宣言型エージェントを表すために使用される色とアウトラインのアイコン。

演習の期間

- 推定所要時間: 15 分

タスク 1 - Teams 管理センターでカスタム アプリのアップロードを有効にする

Teams Toolkit を使用して宣言型エージェントを Microsoft 365 にアップロードするには、Teams 管理センターで [カスタム アプリのアップロード] を有効にする必要があります。

- Teams 管理センターで [Teams アプリ]、[アプリの設定ポリシー] の順に移動するか、[アプリの設定ポリシー] に直接移動します。
- ポリシーのリストから [グローバル (組織全体の既定値)] を選びます。
- [カスタム アプリのアップロード] を有効にします。
- [保存] を選択したあと、その選択を確認します。

タスク 2 - スターター プロジェクトをダウンロードする

まず、Web ブラウザーで GitHub からサンプル プロジェクトをダウンロードします。

- <https://github.com/microsoft/declarative-agent-vscode> テンプレート リポジトリに移動します。
 - 手順に従って、コンピューターにリポジトリのソース コードをダウンロードします。

Microsoft Teams admin center

App setup policies \ Global

Global (Org-wide default)

Default policy for users who aren't assigned to a policy.

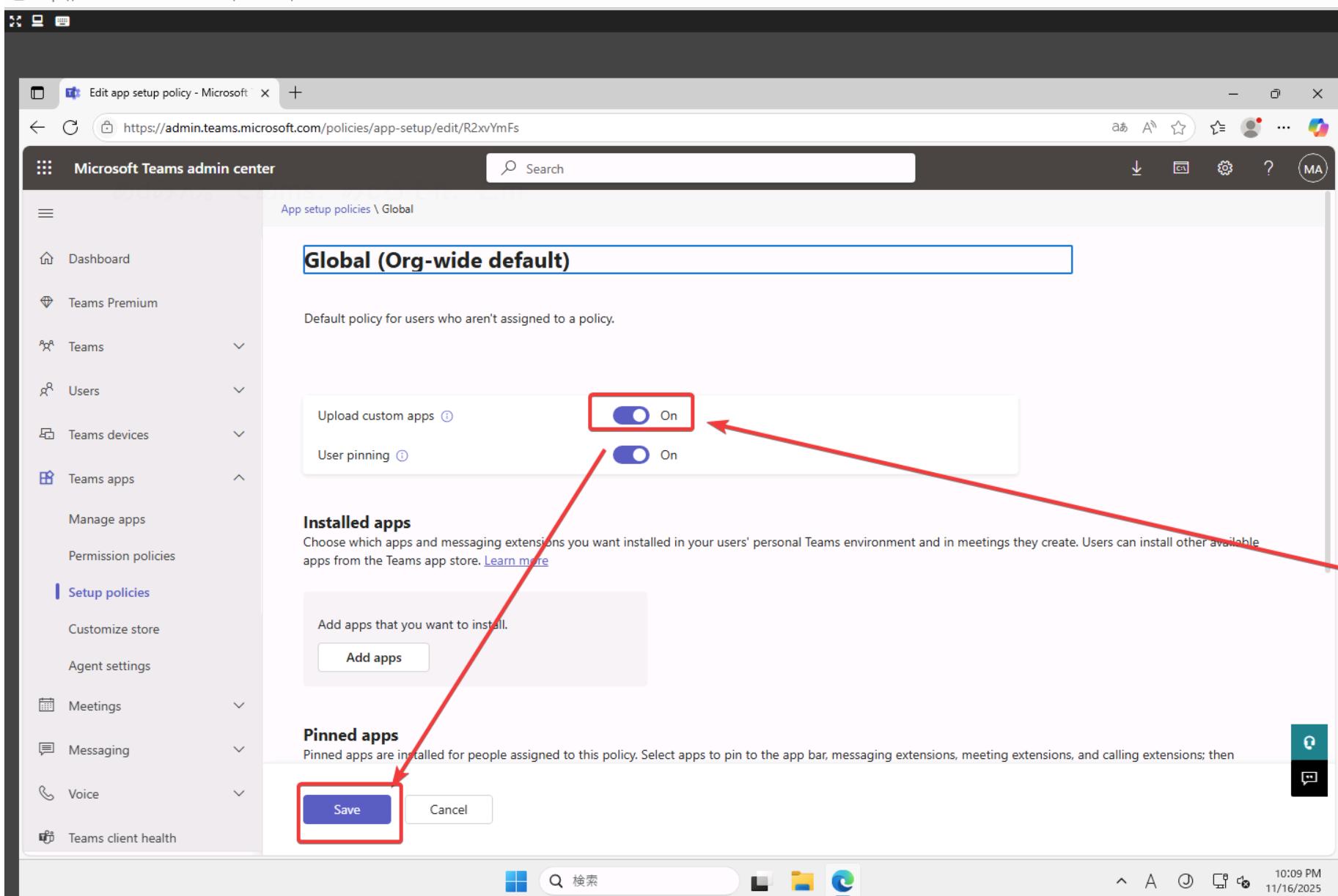
Upload custom apps On

User pinning On

Installed apps
Choose which apps and messaging extensions you want installed in your users' personal Teams environment and in meetings they create. Users can install other available apps from the Teams app store. [Learn more](#)

Add apps

Pinned apps
Pinned apps are installed for people assigned to this policy. Select apps to pin to the app bar, messaging extensions, meeting extensions, and calling extensions; then



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

指示 リソース

- color.png と outline.png: Microsoft 365 Copilot ユーザーインターフェイスで宣言型エージェントを表すために使用される色とアウトラインのアイコン。

演習の期間

- 推定所要時間: 15 分

タスク 1 - Teams 管理センターでカスタム アプリのアップロードを有効にする

Teams Toolkit を使用して宣言型エージェントを Microsoft 365 にアップロードするには、Teams 管理センターで [カスタム アプリのアップロード] を有効にする必要があります。

- Teams 管理センターで [Teams アプリ]、[アプリの設定ポリシー] の順に移動するか、[アプリの設定ポリシー] に直接移動します。
- ポリシーのリストから [グローバル (組織全体の既定値)] を選びます。
- [カスタム アプリのアップロード] を有効にします。
- [保存] を選択したあと、その選択を確認します。

タスク 2 - スターター プロジェクトをダウンロードする

まず、Web ブラウザーで GitHub からサンプル プロジェクトをダウンロードします。

- <https://github.com/microsoft/learn-declarative-agent-vscode> テンプレート リポジトリに移動します。
 - 手順に従って、コンピューターに  リポジトリのソース コードをダウンロードします。

「カスタムアプリのアップロードを有効化する」ここまで

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/89fd4caf-4ebd-498e-937b-c2f02e1d085f

The screenshot shows the Microsoft Teams Admin Center interface. On the left, there's a sidebar with various administrative categories like Dashboard, Teams Premium, Teams, Users, Teams devices, Teams apps, Manage apps, Permission policies, Setup policies (which is selected), Customize store, Agent settings, Meetings, Messaging, Voice, and Teams client health. The main area is titled 'Global (Org-wide default)' and describes it as the 'Default policy for users who aren't assigned to a policy'. It includes sections for 'Installed apps' (with a 'Upload custom' button) and 'Pinned apps' (with a 'Save' button). A central modal dialog is open, titled 'Changes will take time to take effect', stating that changes to the global policy will impact the organization and can't be rolled back immediately. It has two buttons: 'Confirm' (highlighted with a red box) and 'Cancel'. At the bottom of the screen, there's a taskbar with icons for File, Search, Task View, File Explorer, and Edge browser.

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

指示 リソース

- color.png と outline.png: Microsoft 365 Copilot ユーザーインターフェイスで宣言型エージェントを表すために使用される色とアウトラインのアイコン。

演習の期間

- 推定所要時間: 15 分

タスク 1 - Teams 管理センターでカスタム アプリのアップロードを有効にする

Teams Toolkit を使用して宣言型エージェントを Microsoft 365 にアップロードするには、Teams 管理センターで [カスタム アプリのアップロード] を有効にする必要があります。

- Teams 管理センターで [Teams アプリ]、[アプリの設定ポリシー] の順に移動するか、[アプリの設定ポリシー] に直接移動します。
- ポリシーのリストから [グローバル (組織全体の既定値)] を選びます。
- [カスタム アプリのアップロード] を有効にします。
- [保存] を選択したあと、その選択を確認します。

タスク 2 - スターター プロジェクトをダウンロードする

まず、Web ブラウザーで GitHub からサンプル プロジェクトをダウンロードします。

- <https://github.com/microsoft/learn-declarative-agent-vscode> テンプレート リポジトリに移動します。
 - 手順に従って、コンピューターにリポジトリのソース コードをダウンロードします。

前 次 残り5時47分

App setup policies

App setup policies control how apps are made available to a user with the Teams app. Use the Global (Org-wide default) policy and customize it or create custom policies and assign them to a set of users. [Learn more](#)

App setup policies summary

Default policies	Custom policies
2	0

Manage policies **Group policy assignment**

Name	Description	Custom policy
Global (Org-wide default)		No
FirstLineWorker	This is a default app set...	No

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

指示 リソース

- 3. [カスタム アプリのアップロード] を有効にします。
- 4. [保存] を選択したあと、その選択を確認します。

タスク 2 - スターター プロジェクトをダウンロードする

まず、Web ブラウザーで GitHub からサンプル プロジェクトをダウンロードします。

- 1. <https://github.com/microsoft/learn-declarative-agent-vscode> テンプレート リポジトリに移動します。

- a. 手順に従って、コンピューターにリポジトリのソース コードをダウンロードします。
- b. ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。

スターター プロジェクトには、宣言型エージェントを含む Teams Toolkit プロジェクトが含まれています。

- 1. Visual Studio Code でプロジェクト フォルダーを開きます。
- 2. プロジェクトのルート フォルダーで、README.md ファイルを開きます。プロジェクト構造の詳細については、内容を確認します。

Overview of the basic declarative agent template

With the declarative agent, you can build a custom version of Copilot that can be used for specific scenarios, such as for specialized knowledge, implementing specific processes, or simply to save time by reusing a set of AI prompts. For example, a grocery shopping Copilot declarative agent can be used to create a grocery list based on a meal plan that you send to Copilot.

Get started with the template

The screenshot shows a GitHub repository page for 'microsoft/learn-declarative-agent-vscode'. The 'Code' dropdown menu is open, and the 'Download ZIP' option is highlighted with a red box and arrow. The 'About' section on the right provides a brief description of the project.

About

Starter project for the Build your first declarative agent for Microsoft 365 Copilot by using Visual Studio Code learn module

Code

- Clone
- HTTPS GitHub CLI
- <https://github.com/microsoft/learn-declarative-agent-vscode>
- Clone using the web URL.
- Open with GitHub Desktop
- Download ZIP

Files

- .vscode
- appPackage
- assets
- env
- .gitignore
- CODE_OF_CONDUCT.md
- LICENSE
- README.md
- SECURITY.md
- SUPPORT.md
- teamsapp.yml

Activity

21 stars
3 watching
15 forks

Releases

No releases published

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

- 指示 リソース
- 3. [カスタム アプリのアップロード] を有効にします。
 - 4. [保存] を選択したあと、その選択を確認します。

タスク 2 - スターター プロジェクトをダウンロードする

まず、Web ブラウザーで GitHub からサンプル プロジェクトをダウンロードします。

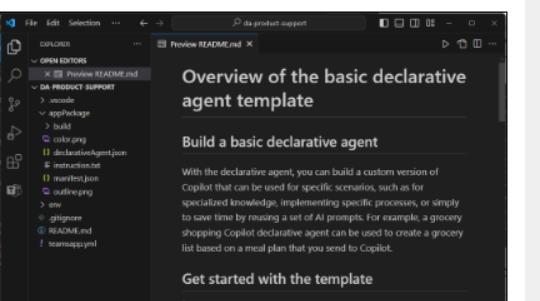
- 1. <https://github.com/microsoft/learn-declarative-agent-vscode> テンプレート リポジトリに移動します。

a. 手順に従って、コンピューターにリポジトリのソース コードをダウンロードします。

b. ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。

スターター プロジェクトには、宣言型エージェントを含む Teams Toolkit プロジェクトが含まれています。

- 1. Visual Studio Code で プロジェクト フォルダーを開きます。
- 2. プロジェクトのルート フォルダーで、README.md ファイルを開きます。プロジェクト構造の詳細については、内容を確認します。



前 次

残り5時46分

The screenshot shows a Microsoft Edge browser window with a GitHub repository page for 'microsoft/learn-declarative-agent-vscode'. The repository is described as a 'Public template' for building a declarative agent for Microsoft 365 Copilot. The 'Code' tab is selected. A context menu is open over the repository name, with the 'Download ZIP' option highlighted and a red box drawn around it. The menu also includes options for 'Clone' via HTTPS or GitHub CLI, and links to 'Readme', 'MIT license', 'Code of conduct', 'Security policy', 'Activity', 'Custom properties', '21 stars', '3 watching', and '15 forks'. Below the repository details, there's an 'About' section with a brief description and a 'Report repository' link. The bottom of the page shows a list of files and their commit history.

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

指示 リソース

3. [カスタム アプリのアップロード] を有効にします。
4. [保存] を選択したあと、その選択を確認します。

タスク 2 - スターター プロジェクトをダウンロードする

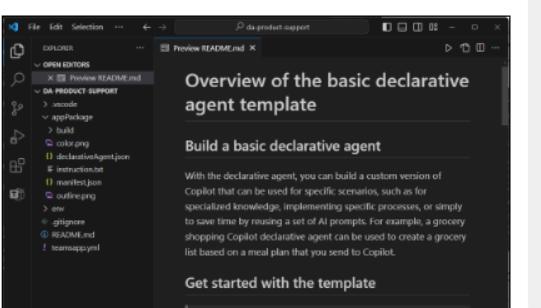
まず、Web ブラウザーで GitHub からサンプル プロジェクトをダウンロードします。

1. <https://github.com/microsoft/learn-declarative-agent-vscode> テンプレート リポジトリに移動します。

- a. 手順に従って、コンピューターに リポジトリのソース コードをダウンロードします。
- b. ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。

スターター プロジェクトには、宣言型エージェントを含む Teams Toolkit プロジェクトが含まれています。

1. Visual Studio Code で プロジェクト フォルダーを開きます。
2. プロジェクトのルート フォルダーで、README.md ファイルを開きます。プロジェクト構造の詳細については、内容を確認します。



GitHub - microsoft/learn-declarative-agent-vscode

Platform Solutions Resources Open Source Enterprise Pricing

Code Issues

main

learn-declarative-agent-vscode-main.zip

learn-declarative-agent-vscode-i

すべて展開

名前 種類 圧縮サイズ パスワ... サイズ 圧縮率 更新日時

learn-declarative-agent-vscode-m... ファイル フォルダー 4/1/2025 6:35 AM

SECURITY.md committed last year

SUPPORT.md Update schema version to v1.3 (#3) 8 months ago

teamsapp.yml Add project files last year

Releases

No releases published

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

指示 リソース

- 3. [カスタム アプリのアップロード] を有効にします。
- 4. [保存] を選択したあと、その選択を確認します。

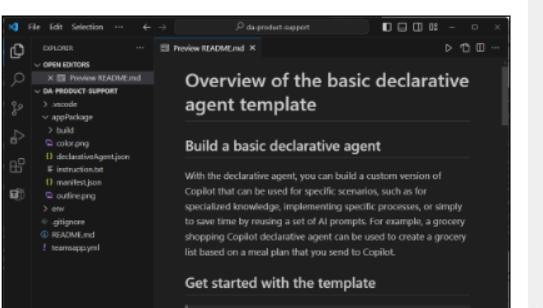
タスク 2 - スターター プロジェクトをダウンロードする

まず、Web ブラウザーで GitHub からサンプル プロジェクトをダウンロードします。

- 1. <https://github.com/microsoft/learn-declarative-agent-vscode> デンブレート リポジトリに移動します。
 - a. 手順に従って、コンピューターに リポジトリのソース コードをダウンロードします。
 - b. ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。

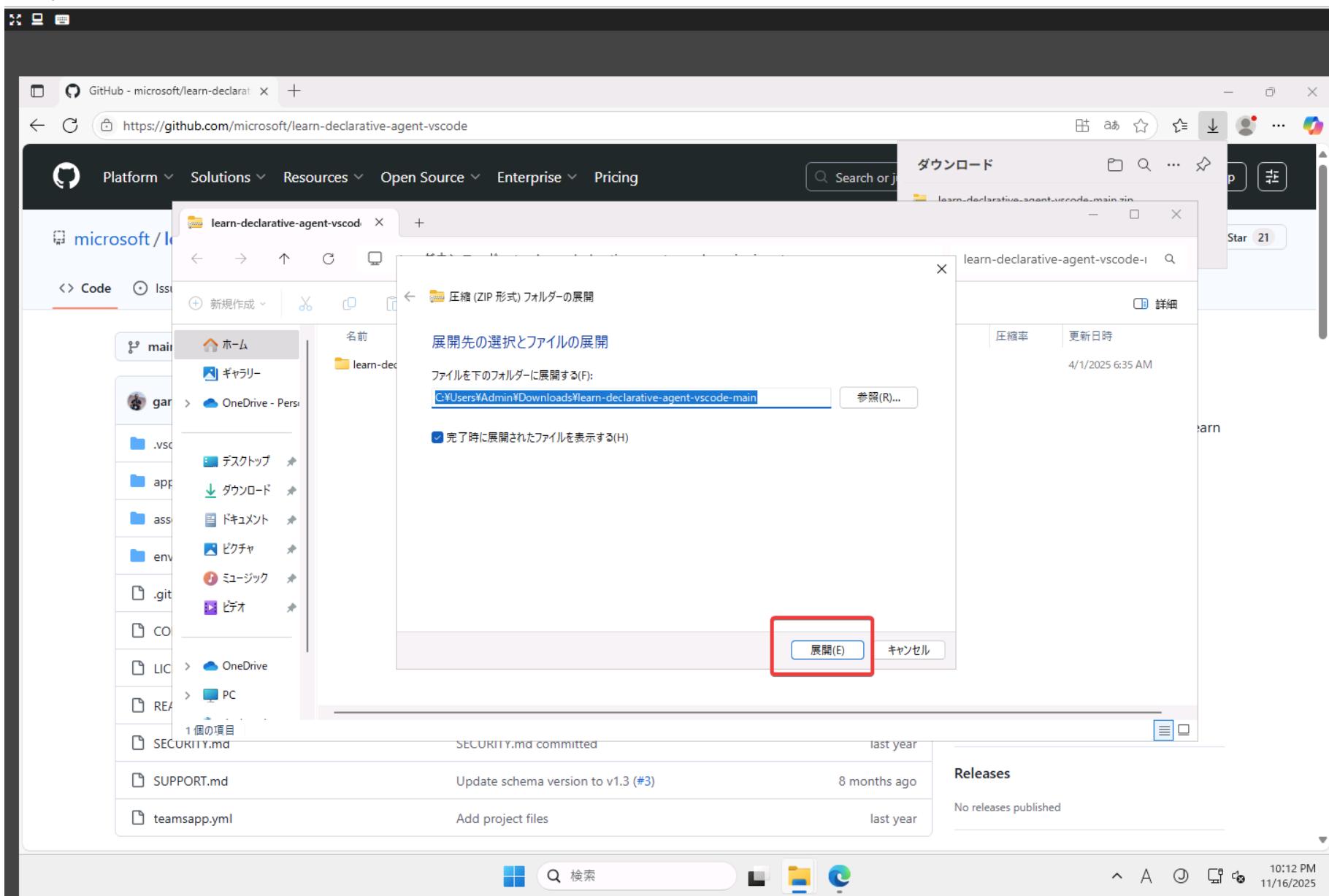
スターター プロジェクトには、宣言型エージェントを含む Teams Toolkit プロジェクトが含まれています。

- 1. Visual Studio Code で プロジェクト フォルダーを開きます。
- 2. プロジェクトのルート フォルダーで、README.md ファイルを開きます。プロジェクト構造の詳細については、内容を確認します。



前 次

残り5時45分



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Code を使用して Microsoft 365 Copilot の宣言的エージェントを構築する

指示 リソース

- 3. [カスタム アプリのアップロード] を有効にします。
- 4. [保存] を選択したあと、その選択を確認します。

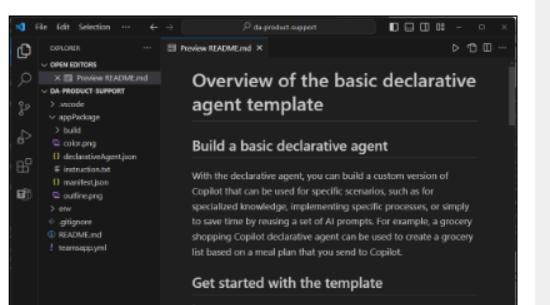
タスク 2 - スターター プロジェクトをダウンロードする

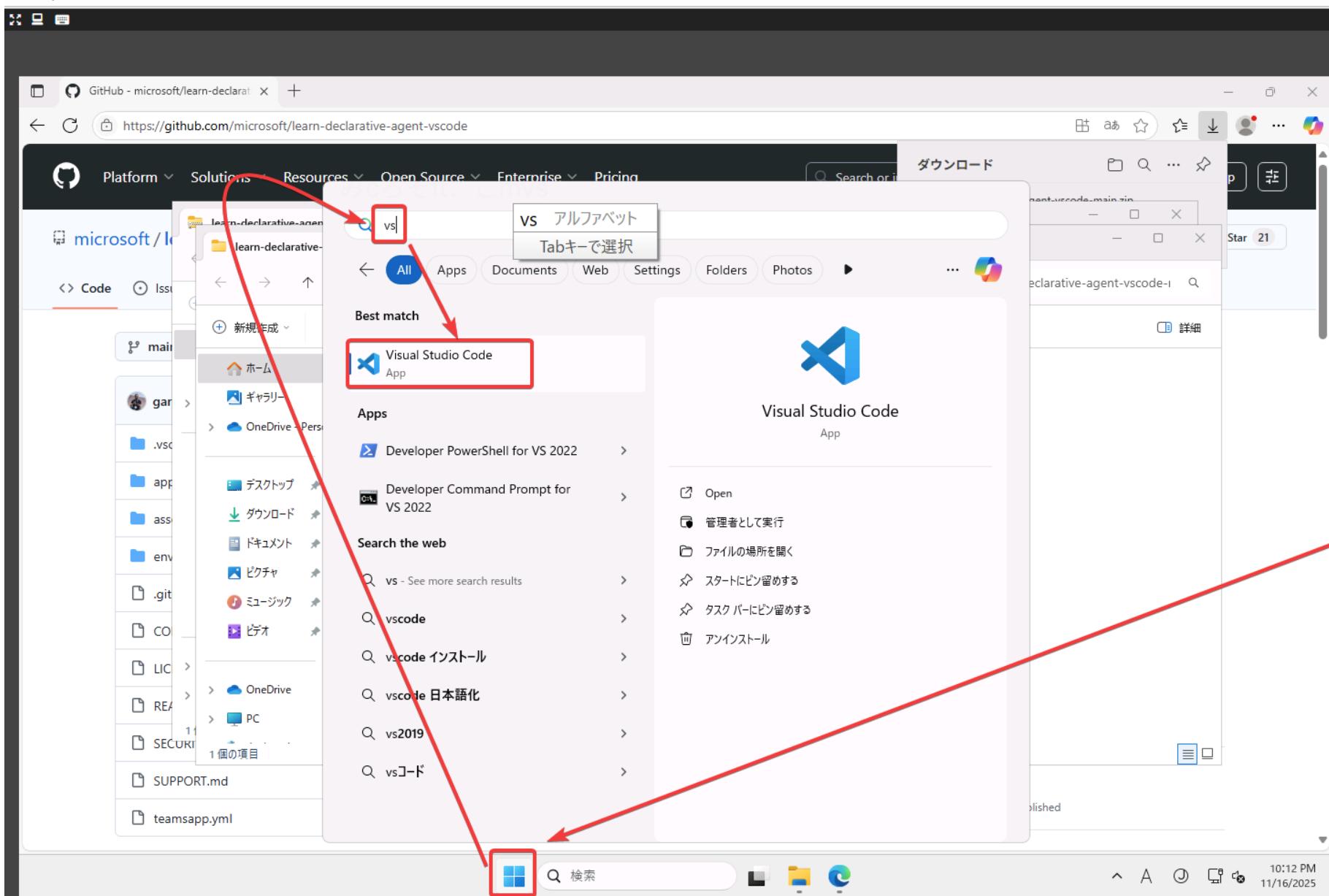
まず、Web ブラウザーで GitHub からサンプル プロジェクトをダウンロードします。

- 1. <https://github.com/microsoft/learn-declarative-agent-vscode> デンブレート リポジトリに移動します。
 - a. 手順に従って、コンピューターにリポジトリのソース コードをダウンロードします。
 - b. ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。

スターター プロジェクトには、宣言型エージェントを含む Teams Toolkit プロジェクトが含まれています。

- 1. Visual Studio Code で プロジェクト フォルダーを開きます。
- 2. プロジェクトのルート フォルダーで、`README.md` ファイルを開きます。プロジェクト構造の詳細については、内容を確認します。





Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Code を使用して Microsoft 365 Copilot の宣言的エージェントを構築する

指示 リソース

- 3. [カスタム アプリのアップロード] を有効にします。
- 4. [保存] を選択したあと、その選択を確認します。

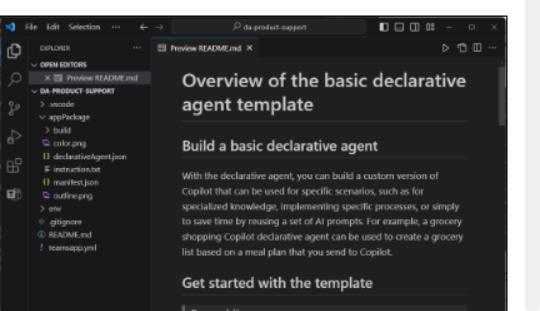
タスク 2 - スターター プロジェクトをダウンロードする

まず、Web ブラウザーで GitHub からサンプル プロジェクトをダウンロードします。

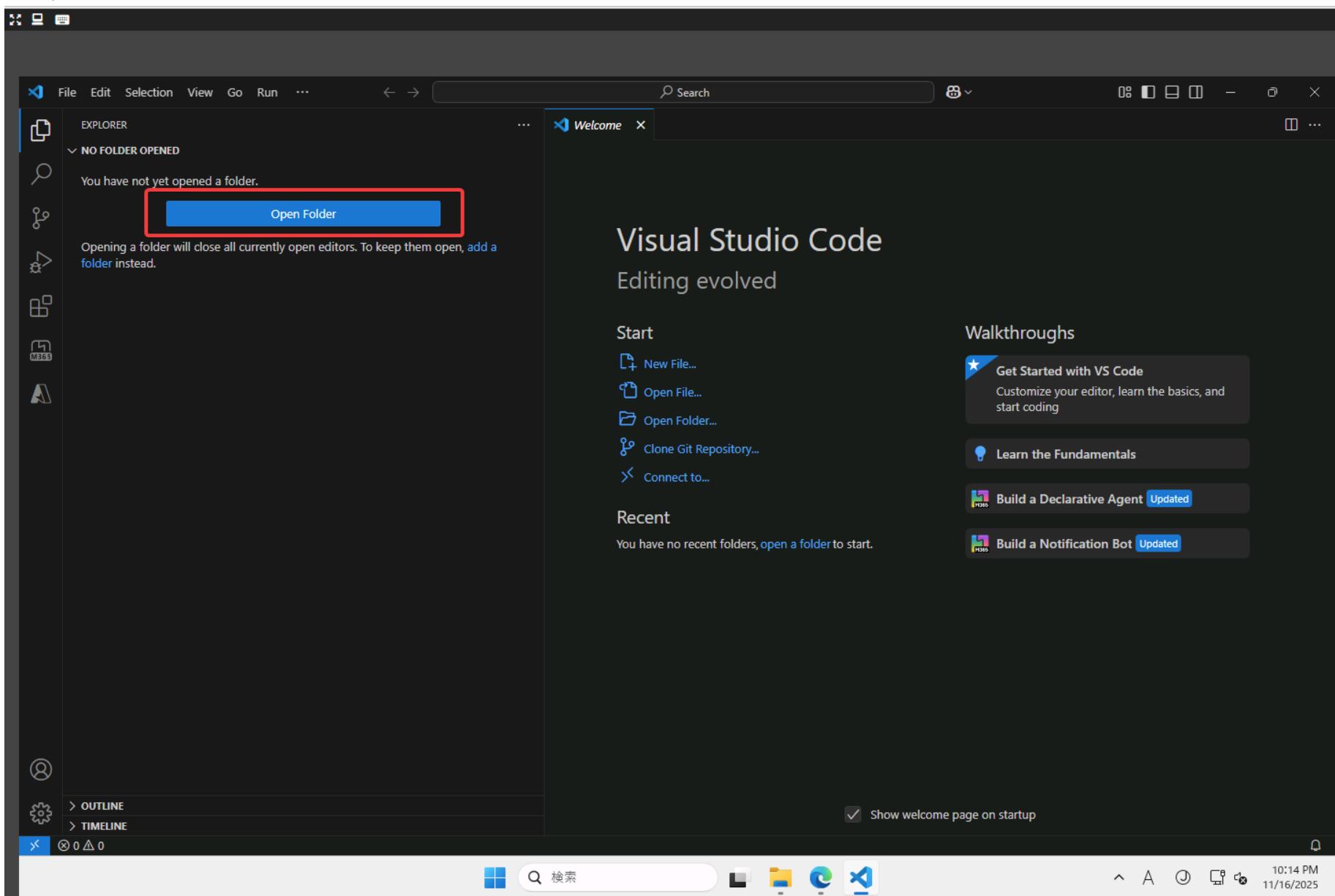
- 1. <https://github.com/microsoft/learn-declarative-agent-vscode> テンプレート リポジトリに移動します。
 - a. 手順に従って、コンピューターに  リポジトリのソース コードをダウンロードします。
 - b. ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。

スターター プロジェクトには、宣言型エージェントを含む Teams Toolkit プロジェクトが含まれています。

- 1. Visual Studio Code で プロジェクト フォルダーを開きます。
- 2. プロジェクトのルート フォルダーで、README.md ファイルを開きます。プロジェクト構造の詳細については、内容を確認します。



前 次 →



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

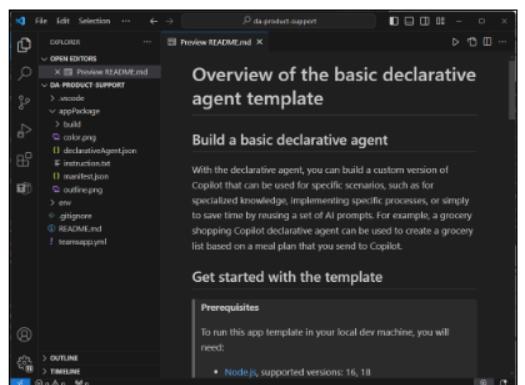
指示 リソース

のソース コードをダウンロードします。

- b. ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。

スターター プロジェクトには、宣言型エージェントを含む Teams Toolkit プロジェクトが含まれています。

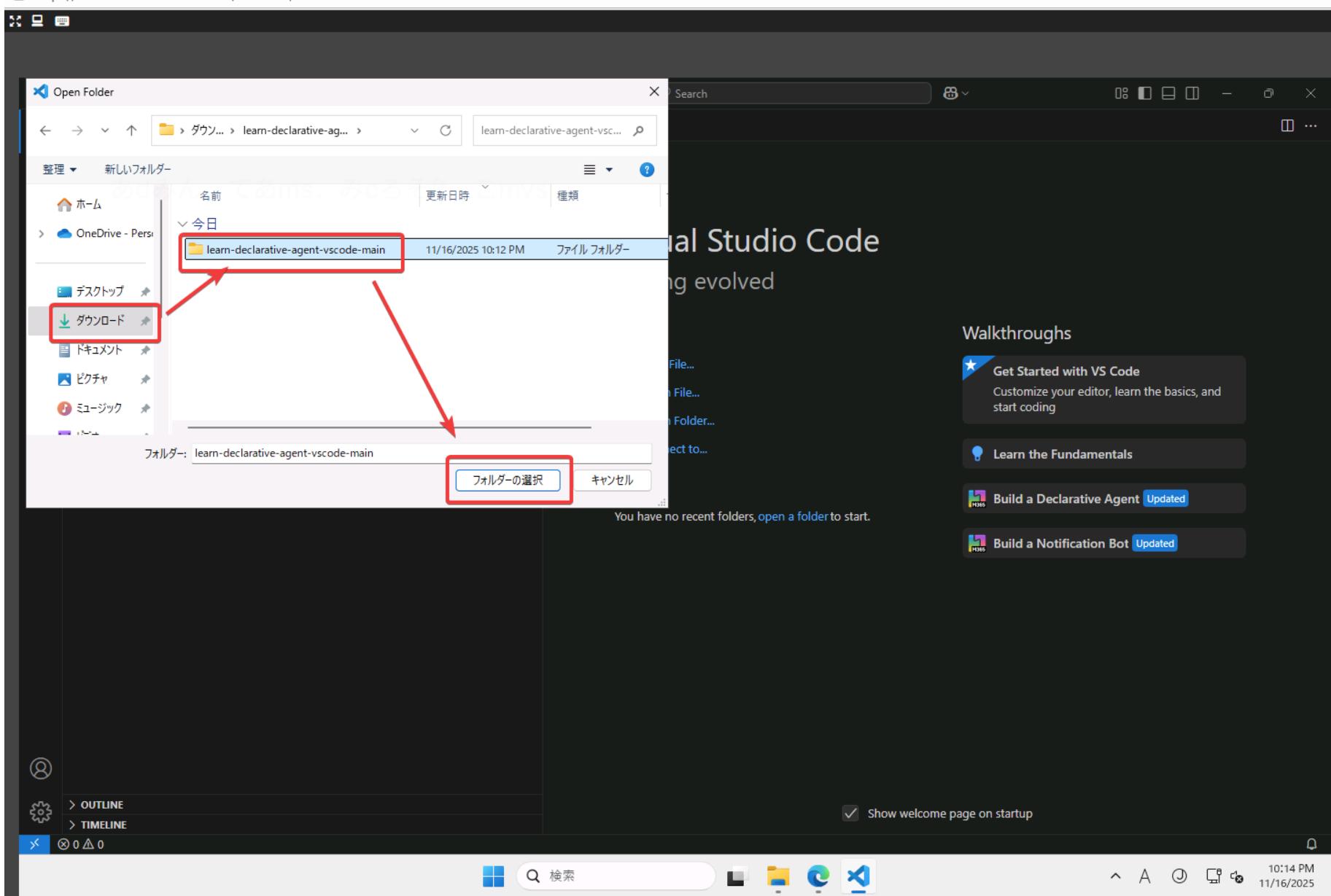
1. Visual Studio Code で プロジェクト フォルダーを開きます。
2. プロジェクトのルート フォルダーで、README.md ファイルを開きます。プロジェクト構造の詳細については、内容を確認します。



タスク 3 - 宣言型エージェント マニフェストを調べる

宣言型エージェント マニフェスト ファイルを調べてみましょう。

- appPackage/declarativeAgent.json ファイルを開き、内容を確認します。



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

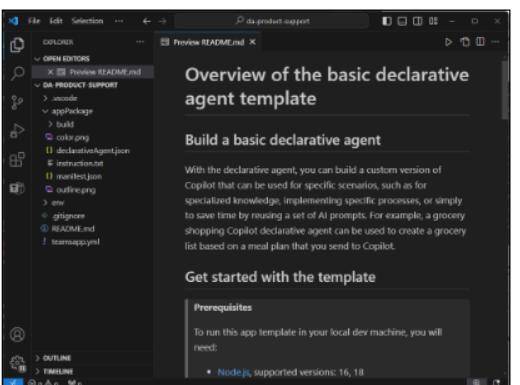
指示 リソース

のソース コードをダウンロードします。

- ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。

スターター プロジェクトには、宣言型エージェントを含む Teams Toolkit プロジェクトが含まれています。

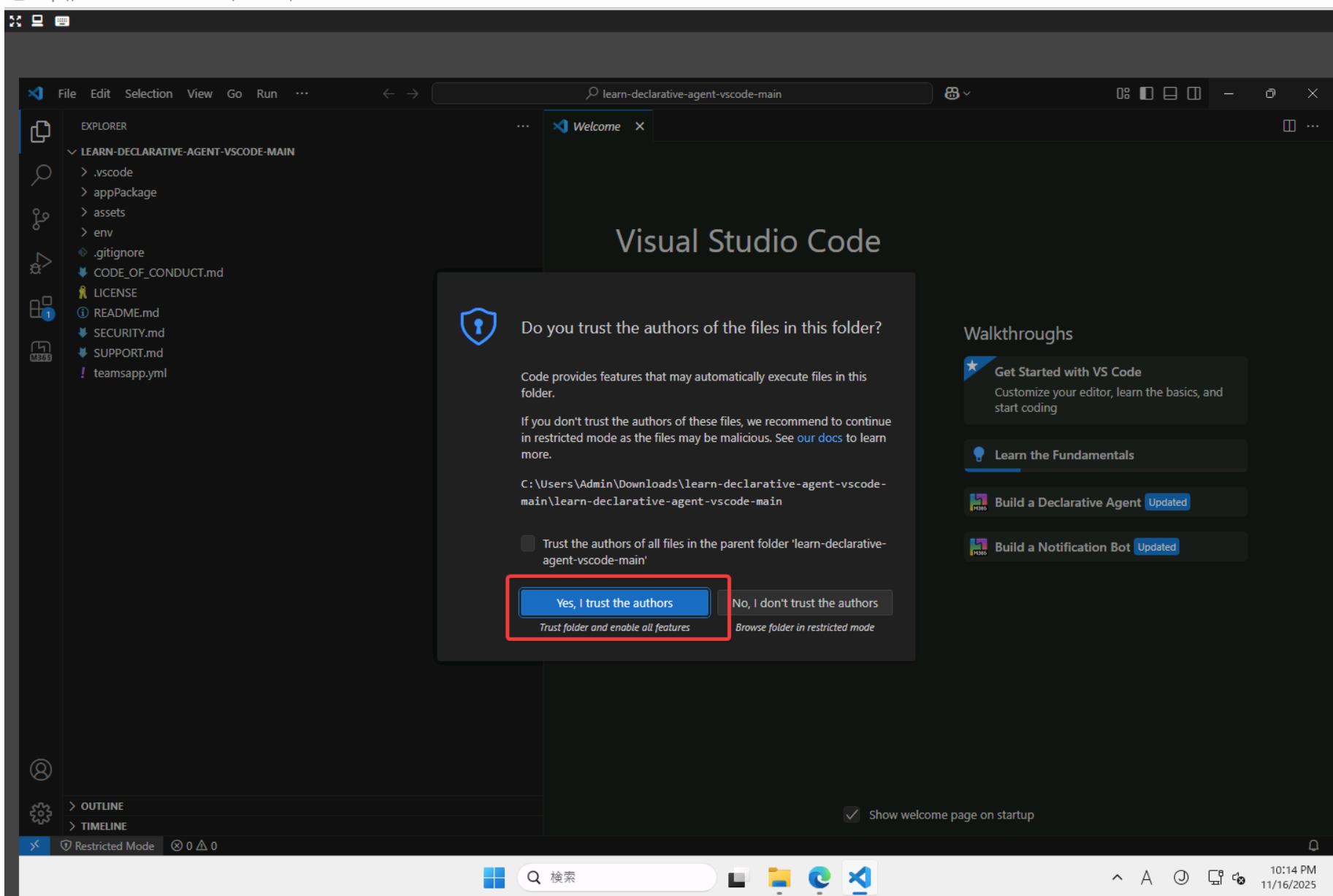
- Visual Studio Code でプロジェクト フォルダーを開きます。
- プロジェクトのルート フォルダーで、README.md ファイルを開きます。プロジェクト構造の詳細については、内容を確認します。



タスク 3 - 宣言型エージェント マニフェストを調べる

宣言型エージェント マニフェスト ファイルを調べてみましょう。

- appPackage/declarativeAgent.json ファイルを開き、内容を確認します。



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

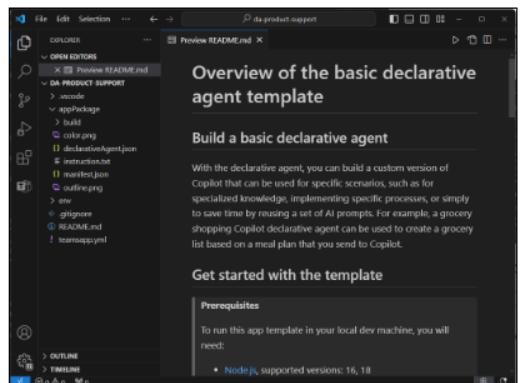
指示 リソース

のソース コードをダウンロードします。

- b. ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。

スターター プロジェクトには、宣言型エージェントを含む Teams Toolkit プロジェクトが含まれています。

1. Visual Studio Code でプロジェクト フォルダーを開きます。
2. プロジェクトのルート フォルダーで、README.md ファイルを開きます。プロジェクト構造の詳細については、内容を確認します。



タスク 3 - 宣言型エージェント マニフェストを調べる

宣言型エージェント マニフェスト ファイルを調べてみましょう。

- appPackage/declarativeAgent.json ファイルを開き、内容を確認します。

※manifest.json 内に、このファイル（declarativeAgent.json）の名前が書かれている

The screenshot shows a Visual Studio Code window with the title "Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge". The URL is <https://labclient.labondemand.com/LabClient/89fd4caf-4ebd-498e-937b-c2f02e1d085f>. The Explorer sidebar shows a folder structure under "LEARN-DECLARATIVE-AGENT-VSCODE-MAIN" containing ".vscode", "appPackage", "color.png", "declarativeAgent.json", "instruction.txt", "manifest.json", "outline.png", "assets", "env", ".gitignore", "CODE_OF_CONDUCT.md", "LICENSE", "README.md", "SECURITY.md", "SUPPORT.md", and "teamsapp.yml". The "declarativeAgent.json" file is selected and highlighted with a red box. The code editor shows the following JSON content:

```
1  {
2      "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1"
3      "version": "v1.3",
4      "name": "da-product-support",
5      "description": "Declarative agent created with Teams Toolkit",
6      "instructions": "${file('instruction.txt')}"
7  }
```

The right side of the screen displays a "指示 リソース" (Instructions & Resources) panel with the following content:

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Code を使用して Microsoft 365 Copilot の宣言的エージェントを構築する

指示

リソース

Get started with the template

Prerequisites

To run this app template in your local dev machine, you will need:

- Node.js, supported versions: 16, 18

タスク 3 - 宣言型エージェントマニフェストを調べる

宣言型エージェントマニフェストファイルを調べてみましょう。

- appPackage/declarativeAgent.json ファイルを開き、内容を確認します。

json タイプ 書し

```
{  
    "$schema": "https://aka.ms/json-schemas/age  
    "version": "v1.0",  
    "name": "da-product-support",  
    "description": "Declarative agent created with Teams Toolkit",  
    "instructions": "${file('instruction.txt')}"  
}
```

instructions プロパティの値には、instruction.txt という名前のファイルへの参照が含まれています。\${file(path)} 関数は Teams Toolkit によって提供されます。instruction.txt の内容は、Microsoft 365 にプロビジョニングされるときに宣言型エージェントマニフェストファイルに含まれます。

- appPackage フォルダーで、instruction.txt ファイルを開き、内容を確認します。

md タイプ 書し

```
You are a declarative agent and were created with the Microsoft 365 Copilot Agent template.
```

前 次 残り5時42分

File Edit Selection View Go Run ...

learn-declarative-agent-vscode-main

EXPLORER

LEARN-DECLARATIVE-AGENT-VSCODE-MAIN

- .vscode
- appPackage
- color.png
- declarativeAgent.json
- instruction.txt
- manifest.json
- outline.png
- assets
- env
- .gitignore
- CODE_OF_CONDUCT.md
- LICENSE
- README.md
- SECURITY.md
- SUPPORT.md
- teamsapp.yml

Preview CHANGELOG.md instruction.txt

appPackage > instruction.txt

1 You are a declarative agent and were created with Team Toolkit. You should start every response and answer to the user with "Thanks for using Teams Toolkit to create your declarative agent!\n\n" and then answer the questions and help the user.

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF Plain Text

10:16 PM 11/16/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

指示 リソース

json

```
{ "$schema": "https://aka.ms/json-schemas/appManifest", "version": "v1.0", "name": "da-product-support", "description": "Declarative agent created via Teams Toolkit", "instructions": "[file('instruction.txt')]" }
```

instructions プロパティの値には、instruction.txt という名前のファイルへの参照が含まれています。\$[file(path)] 関数は Teams Toolkit によって提供されます。instruction.txt の内容は、Microsoft 365 にプロビジョニングされるときに宣言型エージェント マニフェスト ファイルに含まれます。

- appPackage フォルダーで、instruction.txt ファイルを開き、内容を確認します。

md

```
You are a declarative agent and were created with Team Toolkit.
```

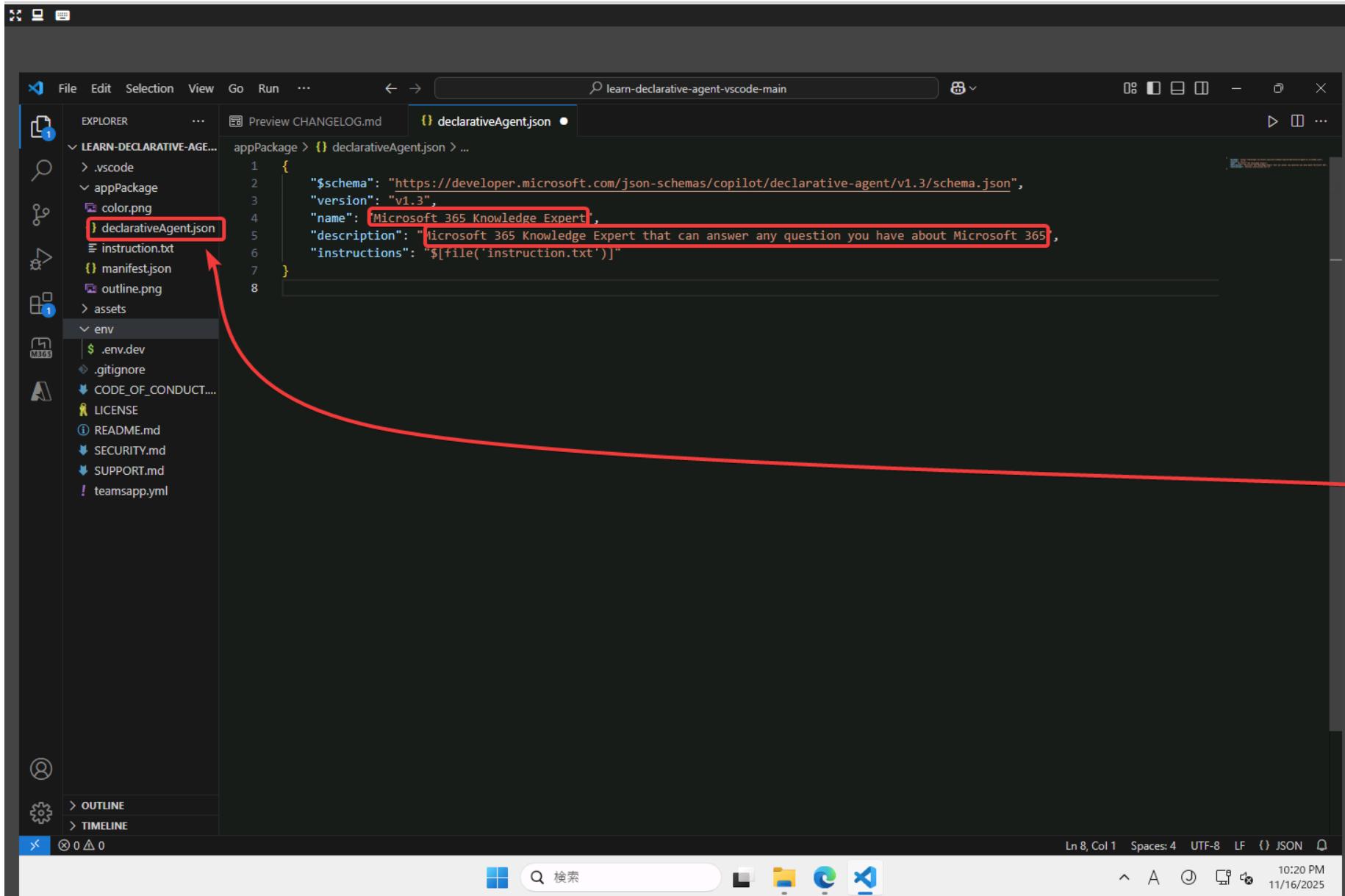
タスク 4 - 宣言型エージェントマニフェストを更新する

name プロパティと description プロパティを、このシナリオに関連するように更新しましょう。

- 1. appPackage フォルダーで、declarativeAgent.json ファイルを開きます。
- 2. name プロパティ値を Microsoft 365 Knowledge Expert に更新します。
- 3. description プロパティ値を Microsoft 365 に関するどんな質問にも回答できる Microsoft 365 Knowledge Expert に更新します。

前 次

残り5時41分



File Edit Selection View Go Run ...

learn-declarative-agent-vscode-main

EXPLORER ...

LEARN-DECLARATIVE-AE... appPackage > {} declarativeAgent.json ●

.vscode
appPackage
color.png
declarativeAgent.json
instruction.txt
manifest.json
outline.png
assets
env
.env.dev
.gitignore
CODE_OF_CONDUCT....
LICENSE
README.md
SECURITY.md
SUPPORT.md
teamsapp.yml

Preview CHANGELOG.md

```
1 {  
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1.3/schema.json",  
3   "version": "v1.3",  
4   "name": "Microsoft 365 Knowledge Expert",  
5   "description": "Microsoft 365 Knowledge Expert that can answer any question you have about Microsoft 365",  
6   "instructions": "[file('instruction.txt')]"  
7 }  
8
```

Ln 8, Col 1 Spaces: 4 UTF-8 LF () JSON

検索

10:20 PM 11/16/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

指示 リソース

You are a declarative agent and were created with Microsoft 365 Copilot.

タスク 4 - 宣言型エージェント マニフェストを更新する

name プロパティと description プロパティを、このシナリオに関連するように更新しましょう。

1. appPackage フォルダーで、declarativeAgent.json ファイルを開きます。
2. name プロパティ値を Microsoft 365 Knowledge Expert に更新します。
3. description プロパティ値を Microsoft 365 に関するどんな質問にも回答できる Microsoft 365 Knowledge Expert に更新します。
4. 変更を保存します

更新されたファイルの内容は次のとおりです。

json タイプ 書き込み

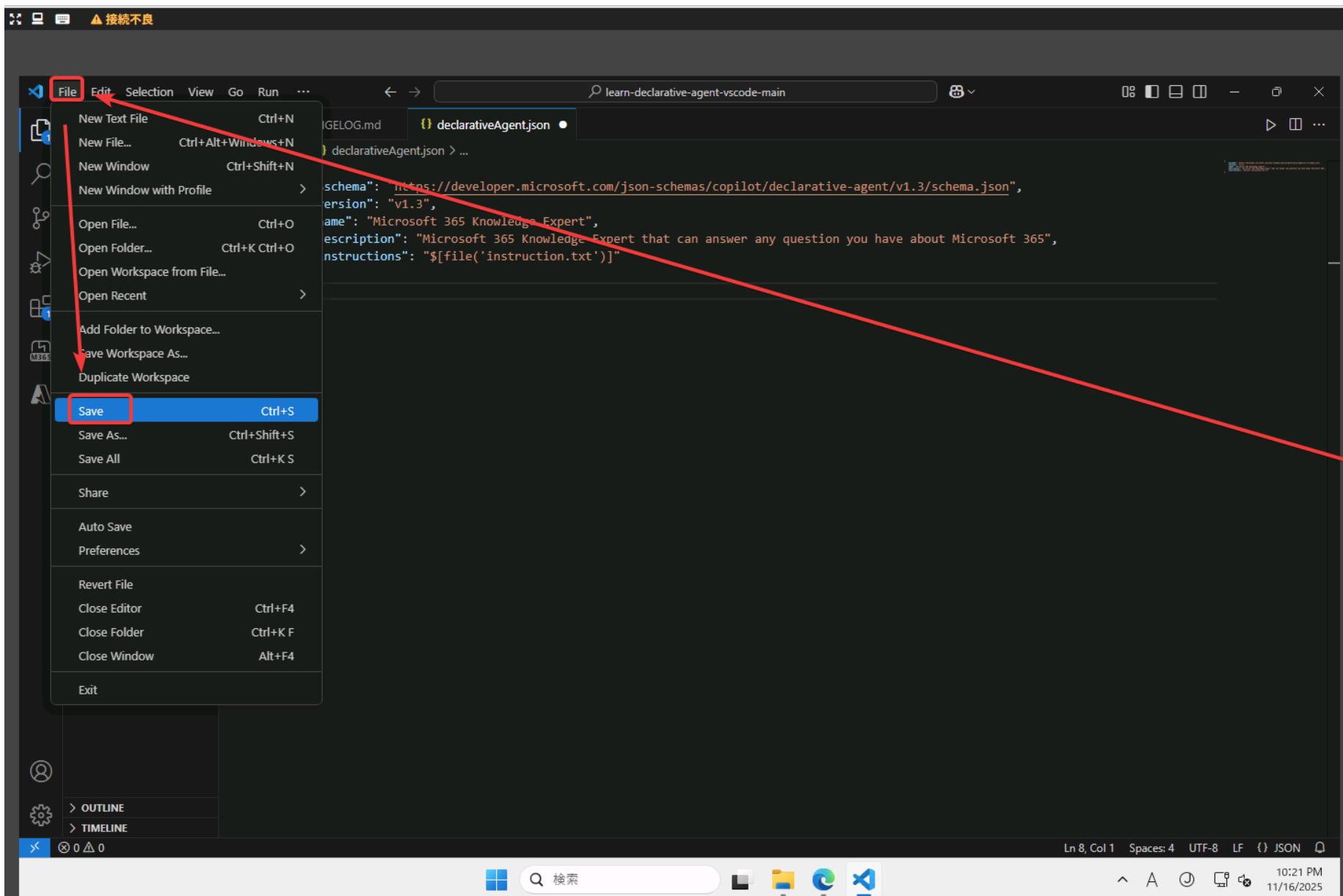
```
{  
  "$schema": "https://aka.ms/json-schemas/agent/c...",  
  "version": "v1.0",  
  "name": "Microsoft 365 Knowledge Expert",  
  "description": "Microsoft 365 Knowledge Expert",  
  "instructions": "[file('instruction.txt')]"  
}
```

タスク 5 - 宣言型エージェントを Microsoft 365 にアップロードする

次に、宣言型エージェントを Microsoft 365 テナントにアップロードします。

前 次

残り5時37分



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge
ヘラボを終了する

指示 リソース

You are a declarative agent and were created with Microsoft 365 Copilot.

タスク 4 - 宣言型エージェント マニフェストを更新する

name プロパティと description プロパティを、このシナリオに関連するように更新しましょう。

1. appPackage フォルダーで、declarativeAgent.json ファイルを開きます。
2. name プロパティ値を Microsoft 365 Knowledge Expert に更新します。
3. description プロパティ値を Microsoft 365 に関するどんな質問にも回答できる Microsoft 365 Knowledge Expert に更新します。
4. 変更を保存します

更新されたファイルの内容は次のとおりです。

json タイプ 書き込み

```
{  
  "$schema": "https://aka.ms/json-schemas/agent/c/  
  "version": "v1.0",  
  "name": "Microsoft 365 Knowledge Expert",  
  "description": "Microsoft 365 Knowledge Expert  
  "instructions": "[file('instruction.txt')]"  
}
```

タスク 5 - 宣言型エージェントを Microsoft 365 にアップロードする

次に、宣言型エージェントを Microsoft 365 テナントにアップロードします。

```

1 {
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1.3/schema.json",
3   "version": "v1.3",
4   "name": "Microsoft 365 Knowledge Expert",
5   "description": "Microsoft 365 Knowledge Expert that can answer any question you have about Microsoft 365",
6   "instructions": "${file('instruction.txt')}"
7 }
8
  
```

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

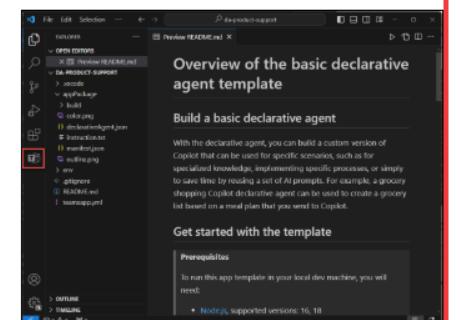
指示 リソース

タスク 5 - 宣言型エージェントを Microsoft 365 にアップロードする

次に、宣言型エージェントを Microsoft 365 テナントにアップロードします。

Visual Studio Code:

- Activity Bar で、Teams Toolkit 拡張機能を開きます。

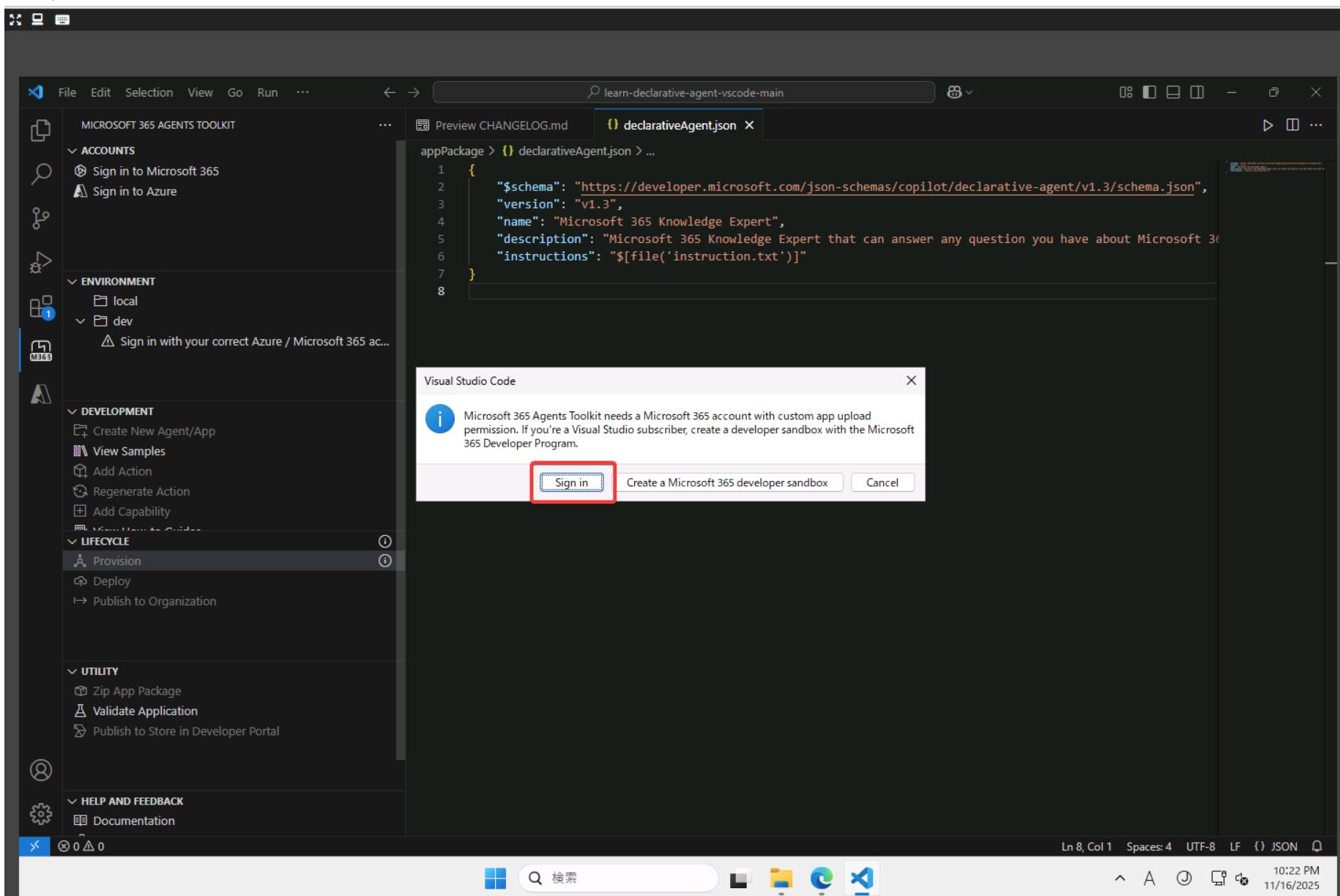


- [Lifecycle] セクションで、[Provision] を選択します。

Teams Toolkit ビューを示す Visual Studio Code のスクリーンショット。[Lifecycle] セクションで 'Provision' 関数が強調表示されています。

- プロンプトで、[Sign in] を選択し、プロンプトに従って Teams Toolkit を使用して Microsoft 365 テナントにサインインします。プロビジョニング プロセスは、サインイン後に自動的に開始されます。

ユーザーに Microsoft 365 へのサインインを求める Visual Studio Code からのプロンプトのスクリーンショット。[Sign in] ボタンが強調表示されています。



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Code を使用して Microsoft 365 Copilot の宣言的エージェントを構築する

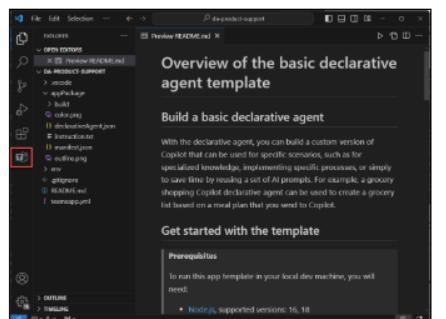
指示 リソース

タスク 5 - 宣言型エージェントを Microsoft 365 にアップロードする

次に、宣言型エージェントを Microsoft 365 テナントにアップロードします。

Visual Studio Code:

- Activity Bar で、Teams Toolkit 拡張機能を開きます。

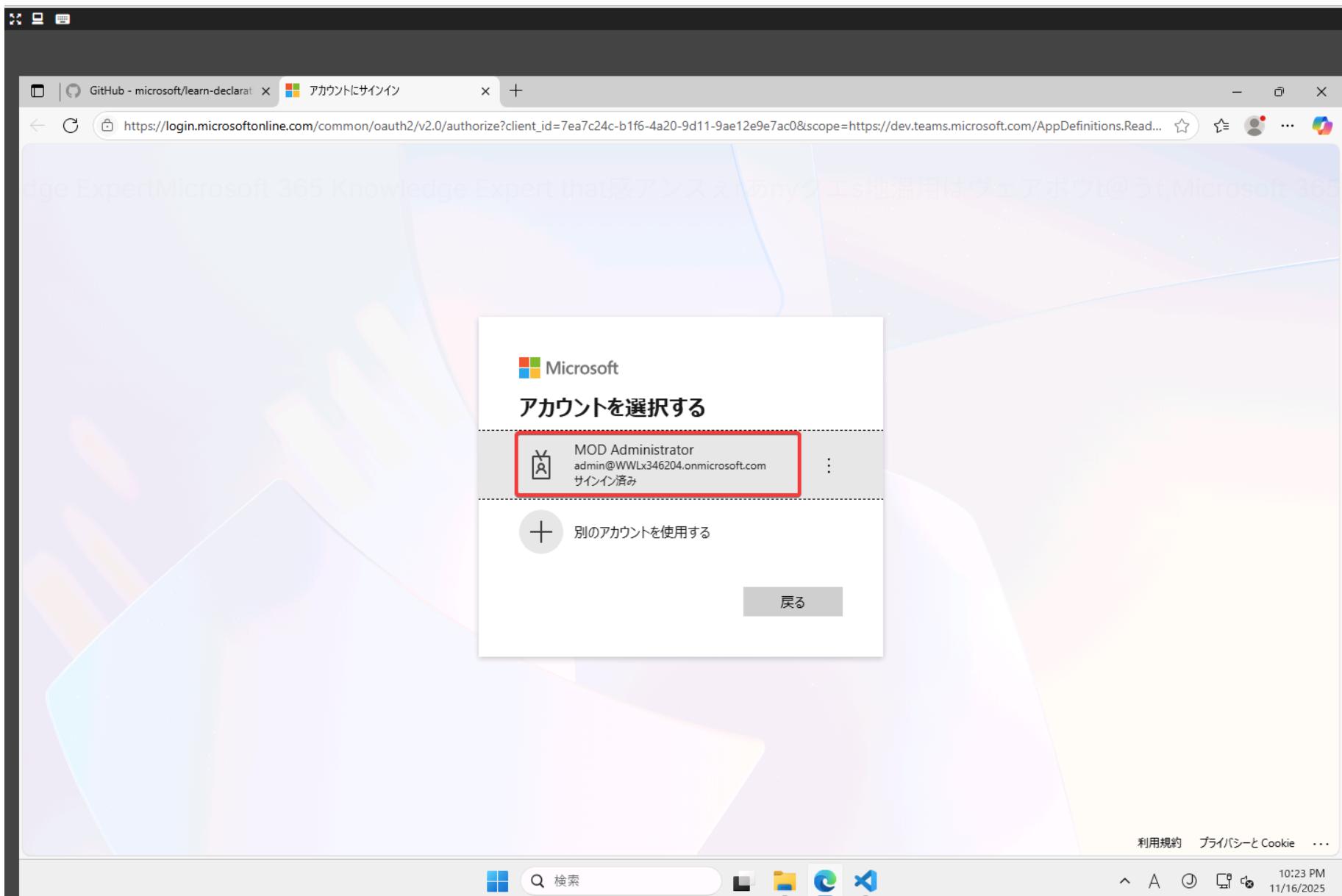


- [Lifecycle] セクションで、[Provision] を選択します。

Teams Toolkit ビューを示す Visual Studio Code ののスクリーンショット。[Lifecycle] セクションで 'Provision' 関数が強調表示されています。

- プロンプトで、[Sign in] を選択し、プロンプトに従って Teams Toolkit を使用して Microsoft 365 テナントにサインインします。プロビジョニング プロセスは、サインイン後に自動的に開始されます。

ユーザーに Microsoft 365 へのサインインを求める Visual Studio Code からのプロンプトのスクリーンショット。[Sign in] ボタンが強調表示されています。



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

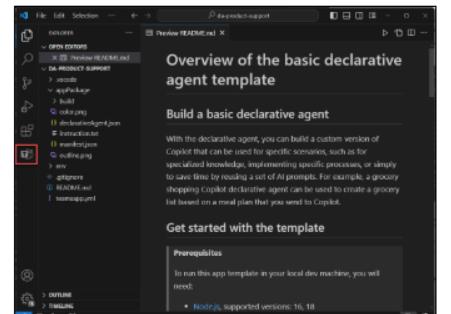
指示 リソース

タスク 5 - 宣言型エージェントを Microsoft 365 にアップロードする

次に、宣言型エージェントを Microsoft 365 テナントにアップロードします。

Visual Studio Code:

- 1. Activity Bar で、Teams Toolkit 拡張機能を開きます。

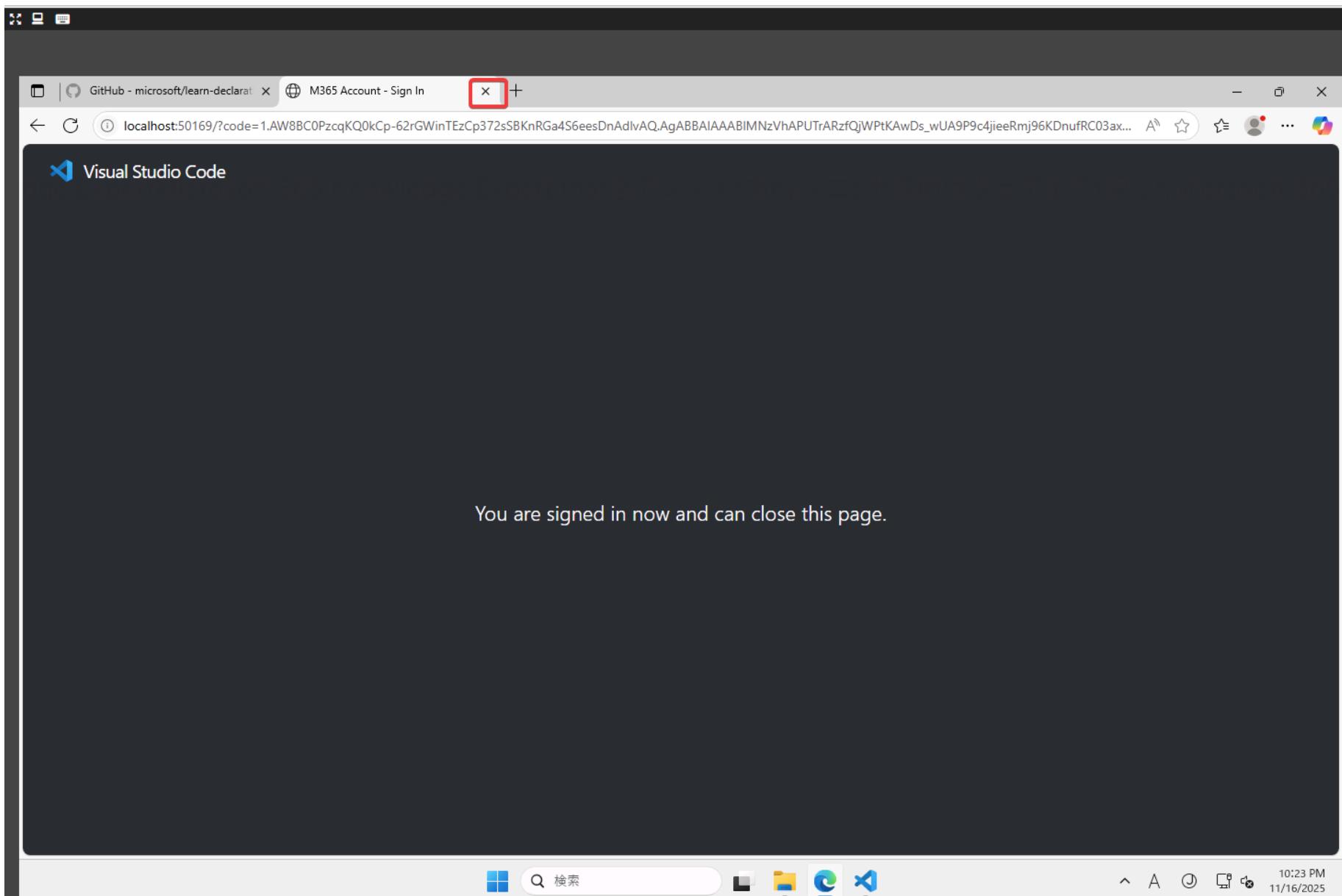


- 2. [Lifecycle] セクションで、[Provision] を選択します。

!Teams Toolkit ビューを示す Visual Studio Code ののスクリーンショット。[Lifecycle] セクションで 'Provision' 関数が強調表示されています。

- 3. プロンプトで、[Sign in] を選択し、プロンプトに従って Teams Toolkit を使用して Microsoft 365 テナントにサインインします。プロビジョニング プロセスは、サインイン後に自動的に開始されます。

!ユーザーに Microsoft 365 へのサインインを求める Visual Studio Code からのプロンプトのスクリーンショット。[Sign in] ボタンが強調表示されています。



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Code を使用して Microsoft 365 Copilot の宣言的エージェントを構築する

ヘラボを終了する

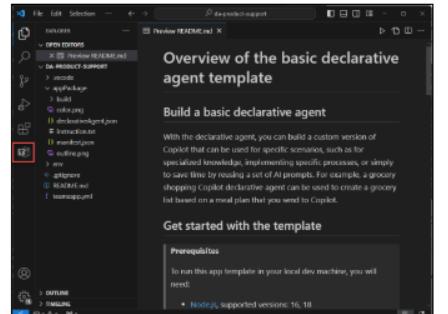
指示 リソース

タスク 5 - 宣言型エージェントを Microsoft 365 にアップロードする

次に、宣言型エージェントを Microsoft 365 テナントにアップロードします。

Visual Studio Code:

- Activity Bar で、Teams Toolkit 拡張機能を開きます。



2. [Lifecycle] セクションで、[Provision] を選択します。

Teams Toolkit ビューを示す Visual Studio Code ののスクリーンショット。[Lifecycle] セクションで 'Provision' 関数が強調表示されています。

3. プロンプトで、[Sign in] を選択し、プロンプトに従って Teams Toolkit を使用して Microsoft 365 テナントにサインインします。プロビジョニング プロセスは、サインイン後に自動的に開始されます。

ユーザーに Microsoft 365 へのサインインを求める Visual Studio Code からのプロンプトのスクリーンショット。[Sign in] ボタンが強調表示されています。

GitHub - microsoft/learn-declarative-agent-vscode Public template

Platform Solutions Resources Open Source Enterprise Pricing

Search or jump to... Sign in Sign up Notifications Fork 15 Star 21

Code Issues Pull requests Actions Projects Models Security Insights

main 1 Branch 0 Tags

garrytrinder Update schema version to v1.3 (#3)

.vscode Add project files

appPackage Update schema ver...

assets Add project files

env Add project files

.gitignore Add project files

CODE_OF_CONDUCT.md CODE_OF_CONDUCT.md committed last year

LICENSE LICENSE committed last year

README.md Add project files last year

SECURITY.md SECURITY.md committed last year

SUPPORT.md Update schema version to v1.3 (#3) 8 months ago

teamsapp.yml Add project files last year

Clone

HTTPS GitHub CLI

https://github.com/microsoft/learn-declarative... Download ZIP

About

Starter project for the Build your first declarative agent for Microsoft 365 Copilot by using Visual Studio Code learn module

Readme MIT license Code of conduct Security policy Activity Custom properties 21 stars 3 watching 15 forks Report repository

Releases No releases published

https://github.com/orgs/microsoft/packages?repo_name=learn-declarative-agent-vs...

検索 検索ボタン

Visual Studio Code icon (赤枠)

10:23 PM 11/16/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Code を使用して Microsoft 365 Copilot の宣言的エージェントを構築する

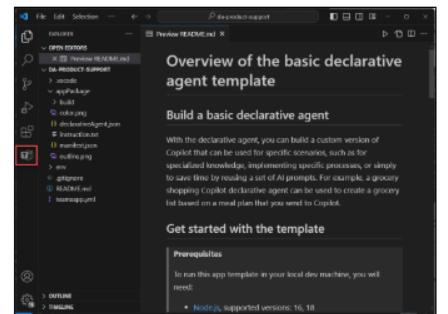
指示 リソース

タスク 5 - 宣言型エージェントを Microsoft 365 にアップロードする

次に、宣言型エージェントを Microsoft 365 テナントにアップロードします。

Visual Studio Code:

- Activity Bar で、Teams Toolkit 拡張機能を開きます。

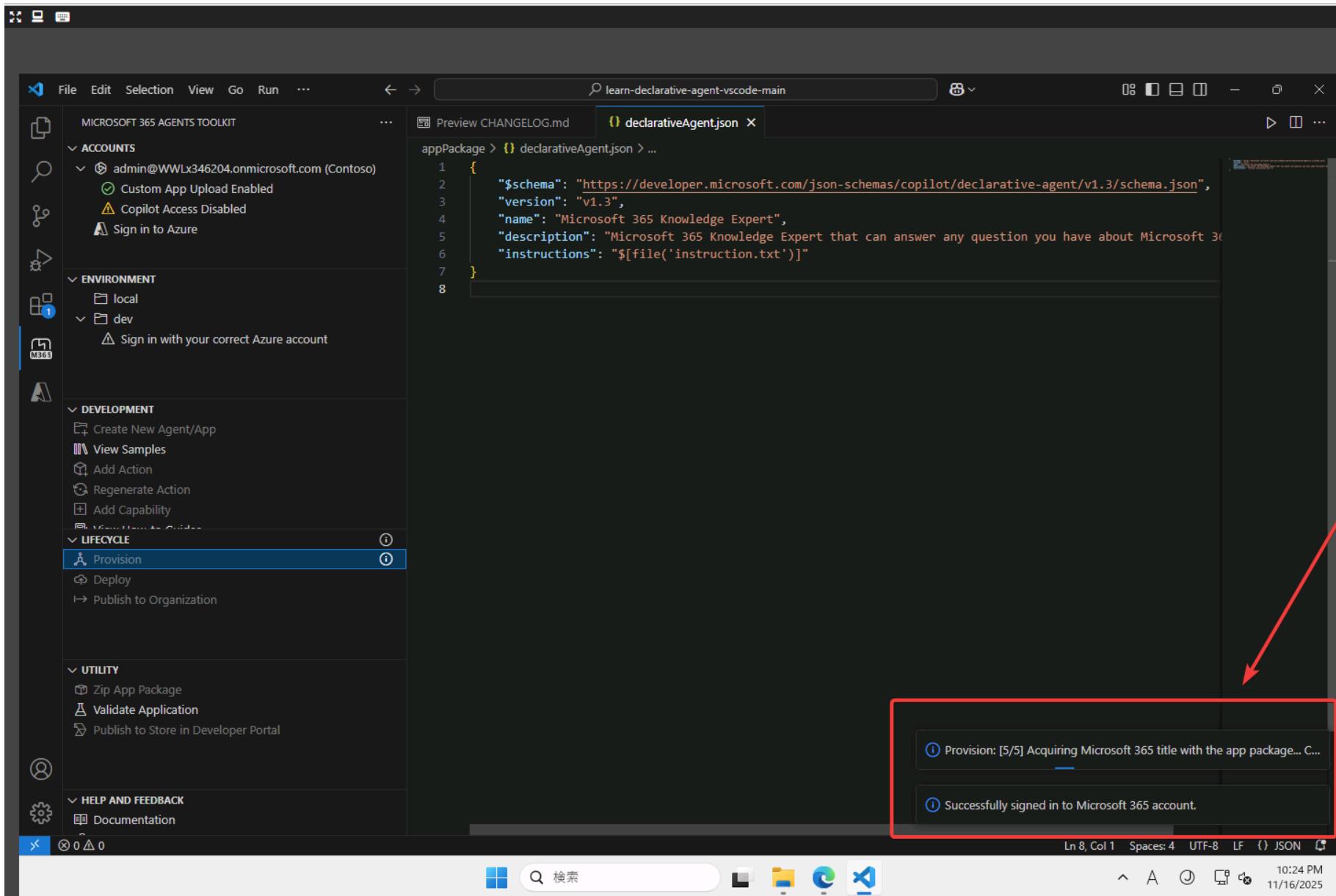


- [Lifecycle] セクションで、[Provision] を選択します。

Teams Toolkit ビューを示す Visual Studio Code ののスクリーンショット。[Lifecycle] セクションで 'Provision' 関数が強調表示されています。

- プロンプトで、[Sign in] を選択し、プロンプトに従って Teams Toolkit を使用して Microsoft 365 テナントにサインインします。プロビジョニング プロセスは、サインイン後に自動的に開始されます。

ユーザーに Microsoft 365 へのサインインを求める Visual Studio Code からのプロンプトのスクリーンショット。[Sign in] ボタンが強調表示されています。



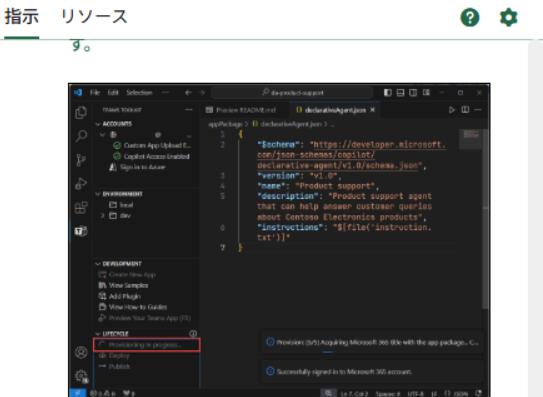
The screenshot shows the Microsoft 365 Agents Toolkit interface. The left sidebar has 'LIFECYCLE' expanded, with 'Provision' selected. The main area shows the contents of 'declarativeAgent.json'.

```

1 {
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1.3/schema.json",
3   "version": "v1.3",
4   "name": "Microsoft 365 Knowledge Expert",
5   "description": "Microsoft 365 Knowledge Expert that can answer any question you have about Microsoft 365 products",
6   "instructions": "${file('instruction.txt')}"
7 }

```

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Code を使用して Microsoft 365 Copilot の宣言的エージェントを構築する - 職場 - Microsoft Edge



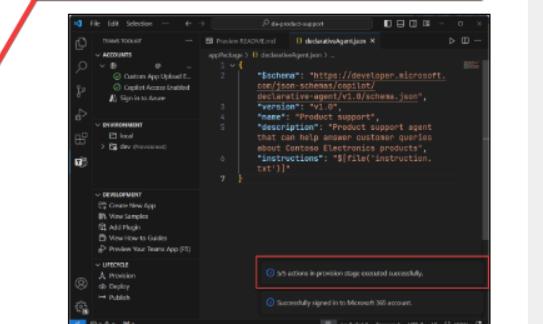
The screenshot shows the Microsoft 365 Agents Toolkit interface. The main area shows the contents of 'declarativeAgent.json'. A status bar at the bottom indicates 'Provisioning (5/5) Acquiring Microsoft 365 title with the app package... C...'.

```

1 {
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1.3/schema.json",
3   "version": "v1.3",
4   "name": "Product support",
5   "description": "Product support agent that can answer any question you have about Contoso Electronics products",
6   "instructions": "${file('instruction.txt')}"
7 }

```

4. アップロードが完了するまで待ってから次に進みます。



The screenshot shows the Microsoft 365 Agents Toolkit interface. The main area shows the contents of 'declarativeAgent.json'. A status bar at the bottom indicates 'Actions in provision stage executed successfully.' and 'Successfully signed in to Microsoft 365 account.'

```

1 {
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1.3/schema.json",
3   "version": "v1.3",
4   "name": "Product support",
5   "description": "Product support agent that can answer any question you have about Contoso Electronics products",
6   "instructions": "${file('instruction.txt')}"
7 }

```

次に、プロビジョニング プロセスの出力を確認します。

- appPackage/build フォルダーで、declarativeAgent.dev.json ファイルを開きます。

instructions プロパティの値に、instruction.txt ファイルの内容が含まれていることに注目してください。

declarativeAgent.dev.json ファイルは、manifest.dev.json、color.png、outline.png の各ファイルと共に、

※エラーが出る場合は数分待ってから同じ操作を繰り返す

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/89fd4caf-4ebd-498e-937b-c2f02e1d085f

指示 リソース

4. アップロードが完了するまで待ってから次に進みます。

次に、プロビジョニング プロセスの出力を確認します。

- appPackage/build フォルダーで、declarativeAgent.dev.json ファイルを開きます。

instructions プロパティの値に、instruction.txt ファイルの内容が含まれていることに注目してください。
declarativeAgent.dev.json ファイルは、manifest.dev.json、color.png、outline.png の各ファイルと共に、appPackage.dev.zip ファイルに含まれています。
appPackage.dev.zip ファイルは Microsoft 365 にアップロードされます。

A Microsoft 365 アカウントにログインすると、Visual

File Edit Selection View Go Run ... Preview CHANGELOG.md declarativeAgent.json

MICROSOFT 365 AGENTS TOOLKIT

ACCOUNTS

ENVIRONMENT

DEVELOPMENT

LIFECYCLE

UTILITY

HELP AND FEEDBACK

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE Filter Microsoft 365 Agents Ti ...

(v) Done: Lifecycle stage provision was executed successfully.
(v) Done: teamApp/create was executed successfully.
(v) Done: teamApp/zipAppPackage was executed successfully.
(v) Done: teamApp/validateAppPackage was executed successfully.
(v) Done: teamApp/update was executed successfully.
(v) Done: app 48af4c60-2f95-4aa0-b348-b74a940de5c1 updated successfully
(v) Done: teamApp/extendToM365 was executed successfully.
(v) Done: Microsoft 365 title acquired successfully (U_45de299c-a2e8-a2db-a2f4-a55c5106ba05).

[2025-11-17T06:27:02.956Z] [Info] - 5/5 actions in provision stage executed successfully.

Ln 8, Col 1 Spaces: 4 UTF-8 LF () JSON

10:27 PM 11/16/2025

前 次 残り5時31分

```

1 {
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1.3/schema.json",
3   "version": "v1.3",
4   "name": "Microsoft 365 Knowledge Expert",
5   "description": "Microsoft 365 Knowledge Expert that can answer any question you have about Microsoft 365",
6   "instructions": "You are a declarative agent and were created with Team Toolkit. You should start every"
7 }
  
```

EXPLORER

- LEARN-DECLARATIVE-AGENT-VSCODE-MAIN
- .vscode
- appPackage
 - build
 - appPackage.dev.zip
- { declarativeAgent.dev.json
- manifest.dev.json
- color.png
- declarativeAgent.json
- instruction.txt
- manifest.json
- outline.png
- assets
- env
 - .env.dev
 - .env.dev.user
 - .gitignore
 - CODE_OF_CONDUCT.md
 - LICENSE
 - README.md
 - SECURITY.md
 - SUPPORT.md
- teamsapp.yml

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE Filter Microsoft 365 Agents Ti ...

(v) Done: Lifecycle stage provision was executed successfully.
(v) Done: teamsApp/create was executed successfully.
(v) Done: teamsApp/zipAppPackage was executed successfully.
(v) Done: teamsApp/validateAppPackage was executed successfully.
(v) Done: teamsApp/update was executed successfully.
(v) Done: app 48af4c60-2f95-4aa0-b74a940de5c1 updated successfully
(v) Done: teamsApp/extendToM365 was executed successfully.
(v) Done: Microsoft 365 title acquired successfully (U_45de299c-a3e8-e3db-e754-a5fcf106ba05).

[2025-11-17T06:27:02.956Z] [Info] - 5/5 actions in provision stage executed successfully.

Ln 1, Col 1 Spaces: 4 UTF-8 LF () JSON

0 10:28 PM 11/16/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Code を使用して Microsoft 365 Copilot の宣言的エージェントを構築する

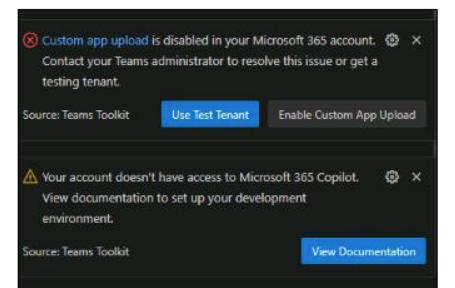
指示 リソース

次に、プロビジョニング プロセスの出力を確認します。

- appPackage/build フォルダーで、declarativeAgent.dev.json ファイルを開きます。

instructions プロパティの値に、instruction.txt ファイルの内容が含まれていることに注目してください。
declarativeAgent.dev.json ファイルは、manifest.dev.json、color.png、outline.png の各ファイルと共に、appPackage.dev.zip ファイルに含まれています。
appPackage.dev.zip ファイルは Microsoft 365 にアップロードされます。

! Microsoft 365 アカウントにログインすると、Visual Studio Code に次の警告またはエラー メッセージが表示されることがあります。 Microsoft Teams でカスタム アプリのアップロードを有効にしたばかりの場合は、設定が有効になるまでに時間がかかる場合があります。 数分待ってからもう一度試すか、ログアウトして Microsoft 365 アカウントでログインし直してください。 テナントには完全な Copilot ライセンスがないので、Microsoft 365 Copilot アクセスに関する 2 番目のメッセージが予想されます。



タスク 6 - Microsoft 365 Copilot

前 次

残り 5 時29分

Microsoft Edge browser window showing a guide for building a declarative agent in Visual Studio Code. The sidebar shows the 'MICROSOFT 365 AGENTS TOOLKIT' with sections like 'ACCOUNTS', 'ENVIRONMENT', 'DEVELOPMENT', 'LIFE CYCLE', 'UTILITY', and 'HELP AND FEEDBACK'. A red box highlights the 'LIFE CYCLE' section, specifically the 'Provision' and 'Publish to Organization' actions. A red arrow points from the 'Sign in with your correct Azure account' link in the sidebar to the 'Sign in to Azure' step in the task list.

```

1 {
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1.3/schema.json",
3   "version": "v1.3",
4   "name": "Microsoft 365 Knowledge Expert",
5   "description": "Microsoft 365 Knowledge Expert that can answer any question you have about Microsoft 365",
6   "instructions": "You are a declarative agent and were created with Team Toolkit. You should start every"
7 }
  
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE Filter Microsoft 365 Agents Ti ...

(✓) Done: Lifecycle stage provision was executed successfully.
 (✓) Done: teamsApp/create was executed successfully.
 (✓) Done: teamsApp/zipAppPackage was executed successfully.
 (✓) Done: teamsApp/validateAppPackage was executed successfully.
 (✓) Done: teamsApp/update was executed successfully.
 (✓) Done: app 48af4c60-2f95-4aa0-b348-b74a940de5c1 updated successfully
 (✓) Done: teamsApp/extendToM365 was executed successfully.
 (✓) Done: Microsoft 365 title acquired successfully (U_45de299c-a3e8-e3db-e754-a5fcf106ba05).

[2025-11-17T06:27:02.956Z] [Info] - 5/5 actions in provision stage executed successfully.

Ln 1, Col 1 Spaces: 4 UTF-8 LF () JSON

0 10:29 PM 11/16/2025

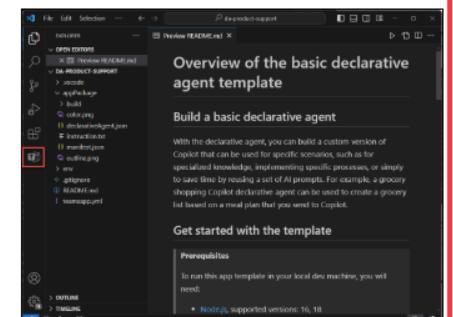
Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

指示 リソース

タスク 6 - Microsoft 365 Copilot Chat で宣言型エージェントをテストする

次に、Microsoft 365 Copilot Chat で宣言型エージェントを実行し、その機能を検証してみましょう。

- Activity Bar で、Teams Toolkit 拡張機能を開きます。



- 【ライフサイクル】セクションで、[発行] を選択します。アクションが完了するまで待ちます。

- Microsoft Edgeを開き、Microsoft 365 Copilot Chat (<https://www.microsoft365.com/chat>) に移動します。

- Microsoft 365 Copilot Chat で、右上にあるアイコンを選択して、Copilot サイド パネルを開展します。パネルに最近のチャットと利用可能なエージェントが表示されることに注目してください。

- サイド パネルで、[Microsoft 365 Knowledge Expert] を選択してスマート エクスペリエンスに入り、エージェントと直接チャットします。

- 「What can you do?」と質問します。

The screenshot shows a Microsoft Edge browser window with the URL <https://labclient.labondemand.com/LabClient/89fd4caf-4ebd-498e-937b-c2f02e1d085f>. The page title is "Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge". The main content area displays a JSON configuration file for a declarative agent:

```

1 {
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1.3/schema.json",
3   "version": "v1.3",
4   "name": "Microsoft 365 Knowledge Expert",
5   "description": "Microsoft 365 Knowledge Expert that can answer any question you have about Microsoft 365 products and services",
6   "instructions": "You are a declarative agent and were created with Team Toolkit. You should start every conversation by asking the user what they want to do with their Microsoft 365 tenant. You can then provide them with options based on their response, such as creating a new SharePoint site or adding users to a group. You can also ask follow-up questions to gather more information and provide more accurate answers. Your responses should be concise and informative, avoiding unnecessary jargon or technical terms. You should also be respectful and considerate of the user's needs and preferences. If you don't know the answer to a question, it's okay to say so and refer the user to the appropriate documentation or support resources. Overall, your goal is to help users get the most out of their Microsoft 365 experience by providing them with the information and tools they need to succeed."
}

```

The left sidebar of the browser shows the "MICROSOFT 365 AGENTS TOOLKIT" interface, including sections for ACCOUNTS, ENVIRONMENT, DEVELOPMENT, LIFECYCLE, UTILITY, and HELP AND FEEDBACK. A red arrow points from the "LIFECYCLE" section in the sidebar to the "publish" command in the terminal output.

The terminal output shows the execution of the publish stage:

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE Filter
(v) Done: Lifecycle stage publish was executed successfully.
(v) Done: teamsApp/validateAppPackage was executed successfully.
(v) Done: teamsApp/update was executed successfully.
(v) Done: app 48af4c60-2f95-4aa0-b348-b74a940de5c1 updated successfully.
(v) Done: teamsApp/publishAppPackage was executed successfully.
(v) Done: Checking if the app is already submitted to tenant App Catalog...
(v) Done: App with id 48af4c60-2f95-4aa0-b348-b74a940de5c1 does not exist in tenant App Catalog.
(v) Done: Publishing app...
(v) Done: App 48af4c60-2f95-4aa0-b348-b74a940de5c1 successfully published.

```

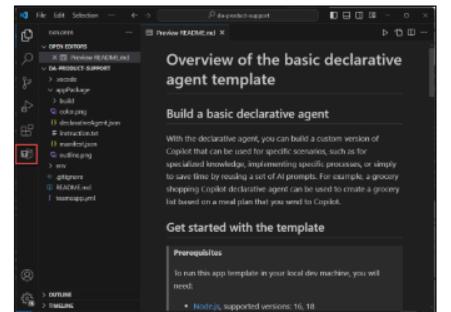
A red box highlights the terminal output, and another red box highlights a success message in the status bar: "4/4 actions in publish stage executed successfully."

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

タスク 6 - Microsoft 365 Copilot Chat で宣言型エージェントをテストする

次に、Microsoft 365 Copilot Chat で宣言型エージェントを実行し、その機能を検証してみましょう。

- Activity Bar で、Teams Toolkit 拡張機能を開きます。



- [ライフサイクル] セクションで、[発行] を選択します。アクションが完了するまで待ちます。

- Microsoft Edge を開き、Microsoft 365 Copilot Chat (<https://www.microsoft365.com/chat>) に移動します。

- Microsoft 365 Copilot Chat で、右上にあるアイコンを選択して、Copilot サイド パネルを開展します。パネルに最近のチャットと利用可能なエージェントが表示されることに注目してください。

- サイド パネルで、[Microsoft 365 Knowledge Expert] を選択してイメージ ブラウザにマウスを移動し、エージェントと直接チャットします。

前 次

The screenshot shows a Microsoft Edge browser window. The address bar contains the URL <https://www.microsoft365.com/chat>. The main content area displays a GitHub repository page for "microsoft/learn-declarative". A red arrow points from the task navigation on the right towards the GitHub address bar. Another red arrow points from the GitHub repository details back towards the task navigation.

GitHub - microsoft/learn-declarative

https://www.microsoft365.com/chat

https://www.microsoft365.com/chat

https://www.microsoft365.com/chat - 検索

Code Issues Pull requests Actions Projects Models Security Insights

main 1 Branch 0 Tags

garrytrinder Update schema version to v1.3 (#3)

.vscode Add project files

appPackage Update schema ver...

assets Add project files

env Add project files

.gitignore Add project files

CODE_OF_CONDUCT.md CODE_OF_CONDUCT.md committed last year

LICENSE LICENSE committed last year

README.md Add project files last year

SECURITY.md SECURITY.md committed last year

SUPPORT.md Update schema version to v1.3 (#3) 8 months ago

teamsapp.yml Add project files last year

Clone

HTTPS GitHub CLI

https://github.com/microsoft/learn-declarative...

Clone using the web URL.

Open with GitHub Desktop

Download ZIP

About

Starter project for the Build your first declarative agent for Microsoft 365 Copilot by using Visual Studio Code learn module

Readme

MIT license

Code of conduct

Security policy

Activity

Custom properties

21 stars

3 watching

15 forks

Report repository

Releases

No releases published

https://github.com/microsoft/learn-declarative-agent-vscode/archive/refs/heads/main...

検索

Microsoft Edge

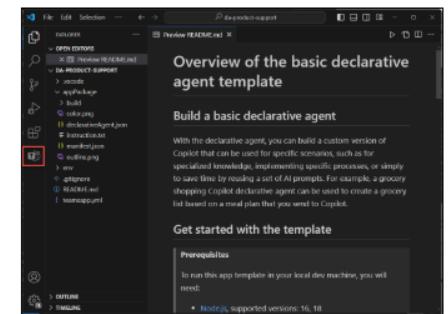
10:30 PM 11/16/2025

指示 リソース

タスク 6 - Microsoft 365 Copilot Chat で宣言型エージェントをテストする

次に、Microsoft 365 Copilot Chat で宣言型エージェントを実行し、その機能を検証してみましょう。

- Activity Bar で、Teams Toolkit 拡張機能を開きます。



- [ライフサイクル] セクションで、[発行] を選択します。アクションが完了するまで待ちます。

- Microsoft Edge を開き、Microsoft 365 Copilot Chat (<https://www.microsoft365.com/chat>) に移動します。

- Microsoft 365 Copilot Chat で、右上にあるアイコンを選択して、Copilot サイド パネルを開展します。パネルに最近のチャットと利用可能なエージェントが表示されることに注目してください。

- サイド パネルで、[Microsoft 365 Knowledge Expert] を選択してイメージ ブラウザに乗り換えてください。エージェントと直接チャットします。

- 「What can you do?」と質問します。

Chat | M365 Copilot

You have been redirected to the Microsoft 365 Copilot app, with easy access to Copilot and agents. Up https://m365.cloud.microsoft/chat?fromcode=cmmyr718qsb&auth=2

M365 Copilot

- Search
- Create
- Apps
- Chat**
- Agents
- Microsoft
- All agents
- Create agent
- Conversations
- Pages
- Admin

Welcome to your Microsoft 365 Copilot app

Find files and more

Get to the right file and info about people and more in a flash.

Create

Harness the power of AI to generate stunning visuals, draft documents, and present your story.

Use your familiar apps

With Copilot built into your apps, you can write, analyze, and design--better and faster than ever.

Get started

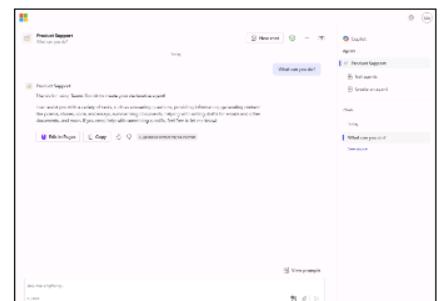
MA MOD Administrator ...

検索

10:32 PM 11/16/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - Microsoft Edge

- 指示 リソース
2. 【ライフサイクル】セクションで、[発行] を選択します。アクションが完了するまで待ちます。
 3. Microsoft Edge を開き、Microsoft 365 Copilot Chat (<https://www.microsoft365.com/chat>) に移動します。
 4. Microsoft 365 Copilot Chat で、右上にあるアイコンを選択して、Copilot サイド パネルを展開します。パネルに最近のチャットと利用可能なエージェントが表示されることに注目してください。
 5. サイド パネルで、[Microsoft 365 Knowledge Expert] を選択してイマーシブ エクスペリエンスに入り、エージェントと直接チャットします。
 6. エージェントに「What can you do?」と質問し、プロンプトを送信します。



次の演習に進んでください。

演習 2 - カスタム ナレッジを構成する

この演習では、エージェントのグラウンドィング ソースとして Microsoft Learn を利用した学習を行います。この演習を受講することで、エージェントは Microsoft 365 のエキスパートにな

Chat | M365 Copilot

https://m365.cloud.microsoft/chat/?fromcode=cmmyr718qsb&from=BypassUnauthAppPage

M365 Copilot

- Search
- Create
- Apps
- Chat
- Agents
- Microsoft 365 Knowledge Expert**
- All agents
- Create agent
- Conversations
- Pages
- Admin

You have been redirected to the Microsoft 365 Copilot app, with easy access to Copilot and agents. Update your bookmark to the new URL m365.cloud.microsoft/chat.

Microsoft 365 Knowledge Expert

Created by Teams App, Inc.

Message Copilot

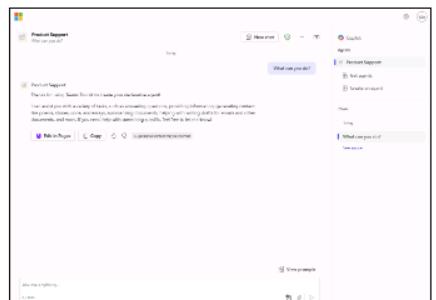
MA MOD Administrator ...

10:33 PM 11/16/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - Microsoft Edge

指示 リソース

2. 【ライフサイクル】セクションで、[発行] を選択します。アクションが完了するまで待ちます。
3. Microsoft Edge を開き、Microsoft 365 Copilot Chat (<https://www.microsoft365.com/chat>) に移動します。
4. Microsoft 365 Copilot Chat で、右上にあるアイコンを選択して、Copilot サイド パネルを展開します。パネルに最近のチャットと利用可能なエージェントが表示されることに注目してください。
- 5. サイド パネルで、[Microsoft 365 Knowledge Expert] を選択してイマーシブ エクスペリエンスに入り、エージェントと直接チャットします。**
6. エージェントに「What can you do?」と質問し、プロンプトを送信します。



次の演習に進んでください。

演習 2 - カスタム ナレッジを構成する

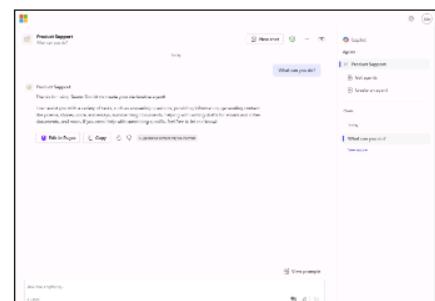
この演習では、エージェントのグラウンドィング ソースとして Microsoft Learn を利用した学習を行います。この演習を受講することで、エージェントは Microsoft 365 のエキスパートにな

The screenshot shows a browser window titled "Chat | M365 Copilot" with the URL <https://m365.cloud.microsoft/chat/?fromcode=cmmmyr718qsb&from=BypassUnauthAppPage>. A message at the top says, "You have been redirected to the Microsoft 365 Copilot app, with easy access to Copilot and agents. Update your bookmark to the new URL m365.cloud.microsoft/chat." The left sidebar has sections for "Search", "Create", "Apps", "Chat", "Agents" (with "Microsoft 365 Knowledge..." selected), "All agents", "Create agent", "Conversations", "Pages", and "Admin". The main content area is titled "Microsoft 365 Knowledge Expert" and is created by "Teams App, Inc.". It features a "What can you do?" button with a red box and arrow pointing to it, and a blue "→" button with a red box and arrow pointing to it. The bottom status bar shows "MA MOD Administrator" and the date "11/16/2025".

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Code を使用して Microsoft 365 Copilot の宣言的エージェントを構築する

指示 リソース

- 2. [ライフサイクル] セクションで、[発行] を選択します。アクションが完了するまで待ちます。
 - 3. Microsoft Edge を開き、Microsoft 365 Copilot Chat ( <https://www.microsoft365.com/chat>) に移動します。
 - 4. Microsoft 365 Copilot Chat で、右上にあるアイコンを選択して、Copilot サイド パネルを展開します。パネルに最近のチャットと利用可能なエージェントが表示されることに注目してください。
 - 5. サイド パネルで、[Microsoft 365 Knowledge Expert] を選択してマーシャル エクスペリエンスに入り、エージェントと直接チャットします。
 - 6. エージェントに「What can you do?」と質問し、プロンプトを送信します。



次の演習に進んでください。

演習 2 - カスタム ナレッジを構成する

この演習では、エージェントのグラウンドィング ソースとして Microsoft Learn を利用した学習を行います。この演習を受講することで、エージェントは Microsoft 365 のエキスパートになれるでしょう。

Chat | M365 Copilot

https://m365.cloud.microsoft/chat/?fromcode=cmmyr718qsb&from=BypassUnauthAppPage

M365 Copilot

- Search
- Create
- Apps
- Chat
 - Agents
 - Microsoft 365 Knowledge ...
 - All agents
 - Create agent
- Conversations
 - What can you do?
 - All conversations
 - Pages
 - Admin

You have been redirected to the Microsoft 365 Copilot app, with easy access to Copilot and agents. Update your bookmark to the new URL m365.cloud.microsoft/chat.

Microsoft 365 Knowledge Expert > What can you do?

Today

What can you do?

Microsoft 365 Knowledge Expert

Thanks for using Teams Toolkit to create your declarative agent!

I can help you with a wide range of tasks related to Microsoft 365, including:

Finding Information

- Search your emails, files, chats, calendar events, and more.
- Summarize updates from the Microsoft 365 Message Center.
- Track unread messages, upcoming meetings, or recent documents.

Insights & Reporting

- Provide usage analytics for Teams, SharePoint, or Outlook.

Show me usage analytics for Teams Help me schedule a meeting Explain Microsoft 365 admin settings

Message Copilot

+

AI-generated content may be incorrect

10:34 PM 11/16/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - Microsoft Edge

ヘラボを終了する

指示 リソース

2. 【ライフサイクル】セクションで、[発行] を選択します。アクションが完了するまで待ちます。

3. Microsoft Edge を開き、Microsoft 365 Copilot Chat (<https://www.microsoft365.com/chat>) に移動します。

4. Microsoft 365 Copilot Chat で、右上にあるアイコンを選択して、Copilot サイド パネルを展開します。パネルに最近のチャットと利用可能なエージェントが表示されることに注目してください。

5. サイド パネルで、[Microsoft 365 Knowledge Expert] を選択してイマーシブ エクスペリエンスに入り、エージェントと直接チャットします。

6. エージェントに「What can you do?」と質問し、プロンプトを送信します。



次の演習に進んでください。

演習 2 - カスタム ナレッジを構成する

この演習では、エージェントのグラウンドィング ソースとして Microsoft Learn を利用した学習を行います。この演習を受講することで、エージェントは Microsoft 365 のエキスパートにな

File Edit Selection View Go Run ...

learn-declarative-agent-vscode-main

EXPLORER

LEARN-DECLARATIVE-AGENT-V...

- > .vscode
- appPackage
 - build
 - appPackage.dev.zip
 - declarativeAgent.dev.json
 - manifest.dev.json
 - color.png
 - declarativeAgent.json
- instruction.txt
- manifest.json
- outline.png
- > assets
- env
 - \$.env.dev
 - \$.env.dev.user
 - .ignore
- CODE_OF_CONDUCT.md
- LICENSE
- README.md
- SECURITY.md
- SUPPORT.md
- teamsapp.yml

Preview CHANGELOG.md

declarativeAgent.json

```

1  "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1.3/schema.json",
2  "version": "v1.3",
3  "name": "Microsoft 365 Knowledge Expert",
4  "description": "Microsoft 365 Knowledge Expert that can answer any question you have about Microsoft 365",
5  "instructions": "${file('instruction.txt')}",
6
7  "capabilities": [
8    {
9      "name": "WebSearch",
10     "sites": [
11       {
12         "url": "https://learn.microsoft.com/microsoft-365"
13       }
14     ]
15   }
16 ]
17
18
19

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE Filter Microsoft 365 Agents Ti ...

(✓) Done: Lifecycle stage publish was executed successfully.

(✓) Done: teamsApp/validateAppPackage was executed successfully.

(✓) Done: teamsApp/update was executed successfully.

(✓) Done: app 48af4c60-2f95-4aa0-b348-b74a940de5c1 updated successfully.

(✓) Done: teamsApp/publishAppPackage was executed successfully.

(✓) Done: Checking if the app is already submitted to tenant App Catalog

(✓) Done: App with id 48af4c60-2f95-4aa0-b348-b74a940de5c1 does not exist in the organization's app store.

(✓) Done: Publishing app...

(✓) Done: App 48af4c60-2f95-4aa0-b348-b74a940de5c1 successfully published to the admin portal.

Ln 7, Col 5 Spaces: 4 UTF-8 LF () JSON

10:41 PM 11/16/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Code を使用して Microsoft 365 Copilot の宣言的エージェントを構築する - 職場 - Microsoft Edge

指示 リソース

スニペットを追加します。{URL} は、Microsoft Learn の Microsoft 365 ランディング ページに直接リンクする URL に置き換えます。

json タイプ 書し

```

"capabilities": [
  {
    "name": "WebSearch",
    "sites": [
      {
        "url": "{URL}"
      }
    ]
  }
]

```

3. 変更を保存。

declarativeAgent.json ファイルは、次のようにになります。

json タイプ 書し

```

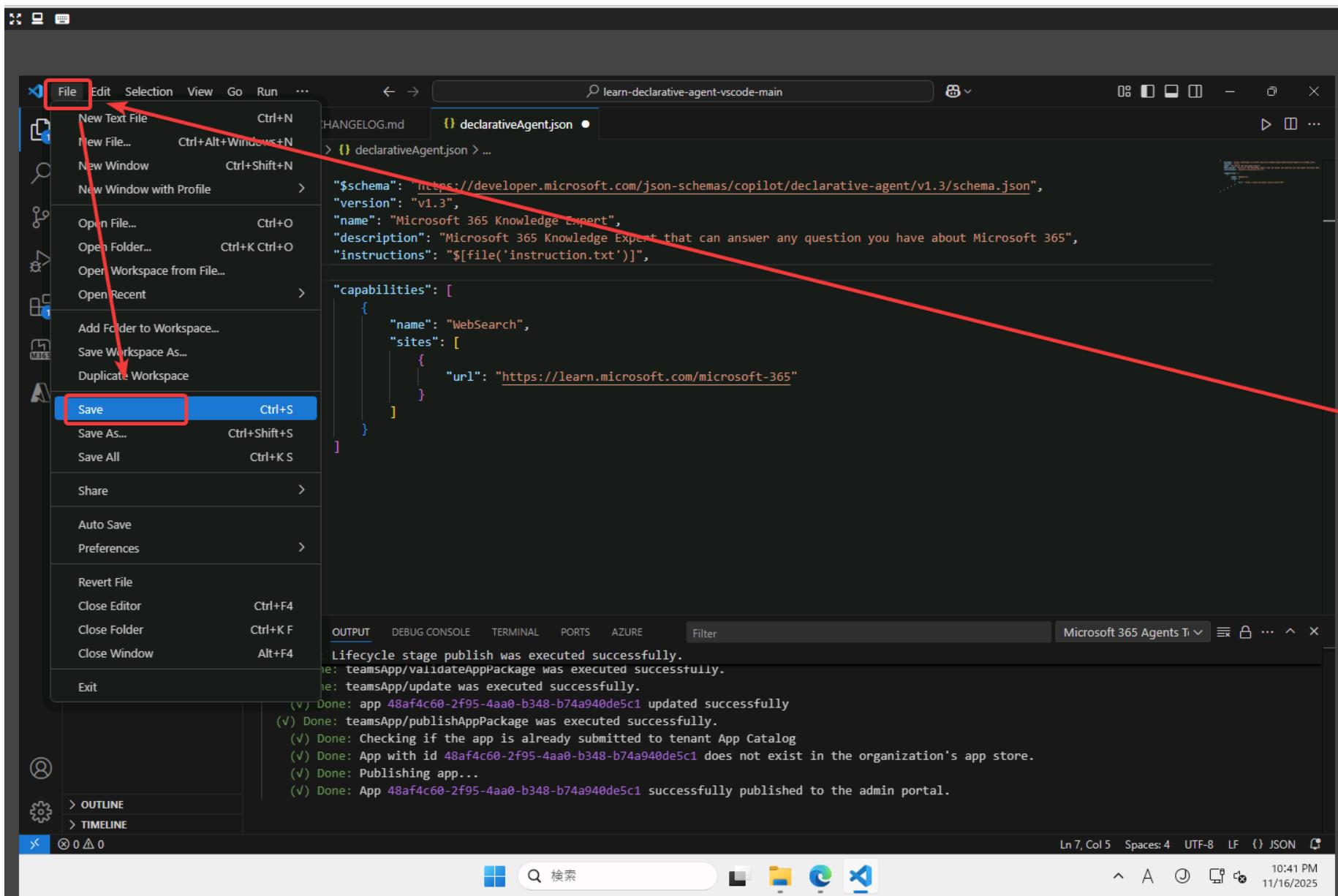
//developer.microsoft.com/json-schemas/copilot/declarative-agent/v1.3/schema.json
{
  "version": "v1.3",
  "name": "Microsoft 365 Knowledge Expert",
  "description": "Microsoft 365 Knowledge Expert that can answer any question you have about Microsoft 365",
  "instructions": "${file('instruction.txt')}",
  "capabilities": [
    {
      "name": "WebSearch",
      "sites": [
        {
          "url": "https://learn.microsoft.com/microsoft-365"
        }
      ]
    }
  ]
}

```

タスク 3 - カスタム指示を更新する

前 次

残り5時16分



File Edit Selection View Go Run ...

New Text File Ctrl+N
New File... Ctrl+Alt+Windows+N
New Window Ctrl+Shift+N
New Window with Profile >
Open File... Ctrl+O
Open Folder... Ctrl+K Ctrl+O
Open Workspace from File...
Open Recent >
Add Folder to Workspace...
Save Workspace As...
Duplicate Workspace
Save Ctrl+S
Save As... Ctrl+Shift+S
Save All Ctrl+K S
Share >
Auto Save
Preferences >
Revert File
Close Editor Ctrl+F4
Close Folder Ctrl+K F
Close Window Alt+F4
Exit

declarativeAgent.json

```
 "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1.3/schema.json",  
 "version": "v1.3",  
 "name": "Microsoft 365 Knowledge Expert",  
 "description": "Microsoft 365 Knowledge Expert that can answer any question you have about Microsoft 365",  
 "instructions": "${file('instruction.txt')}",  
  
 "capabilities": [  
     {  
         "name": "WebSearch",  
         "sites": [  
             {  
                 "url": "https://learn.microsoft.com/microsoft-365"  
             }  
         ]  
     }  
 ]
```

Lifecycle stage publish was executed successfully.
(v) Done: teamsApp/validateAppPackage was executed successfully.
(v) Done: teamsApp/update was executed successfully.
(v) Done: app 48af4c60-2f95-4aa0-b348-b74a940de5c1 updated successfully.
(v) Done: teamsApp/publishAppPackage was executed successfully.
(v) Done: Checking if the app is already submitted to tenant App Catalog
(v) Done: App with id 48af4c60-2f95-4aa0-b348-b74a940de5c1 does not exist in the organization's app store.
(v) Done: Publishing app...
(v) Done: App 48af4c60-2f95-4aa0-b348-b74a940de5c1 successfully published to the admin portal.

OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE Filter Microsoft 365 Agents Ti ... ^ X

Ln 7, Col 5 Spaces: 4 UTF-8 LF () JSON

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Code を使用して Microsoft 365 Copilot の宣言的エージェントを構築する

指示 リソース

スニペットを追加します。{URL} は、Microsoft Learn の Microsoft 365 ランディング ページに直接リンクする URL に置き換えます。

json タイプ 書し

```
 "capabilities": [  
     {  
         "name": "WebSearch",  
         "sites": [  
             {  
                 "url": "{URL}"  
             }  
         ]  
     }  
 ]
```

3. 変更を保存。

declarativeAgent.json ファイルは、次のようにになります。

json タイプ 書し

```
//developer.microsoft.com/json-schemas/copilot/declarative-agent/v1.3/schema.json  
 "version": "v1.3",  
 "name": "Microsoft 365 Knowledge Expert",  
 "description": "Microsoft 365 Knowledge Expert that can answer any question you have about Microsoft 365",  
 "instructions": "${file('instruction.txt')}",  
  
 "capabilities": [  
     {  
         "name": "WebSearch",  
         "sites": [  
             {  
                 "url": "https://learn.microsoft.com/microsoft-365"  
             }  
         ]  
     }  
 ]
```

タスク 3 - カスタム指示を更新する

前 次

残り 5 時16分

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/89fd4caf-4ebd-498e-937b-c2f02e1d085f

File Edit Selection View Go Run ...

learn-declarative-agent-vscode-main

EXPLORER

LEARN-DECLARATIVE-AGENT-V...

.vscode

appPackage

build

appPackage.dev.zip

declarativeAgent.dev.js

manifest.dev.json

color.png

declarativeAgent.json

instruction.txt

manifest.json

outline.png

assets

env

.env.dev

.env.dev.user

.gitignore

CODE_OF_CONDUCT.md

LICENSE

README.md

SECURITY.md

SUPPORT.md

teamsapp.yml

Preview CHANGELOG.md

declarativeAgent.json

instruction.txt

appPackage > instruction.txt

1 d reach out to the team responsible for further assistance. Your responses should be concise and always include a cited source.

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE Filter Microsoft 365 Agents Ti ...

(✓) Done: Lifecycle stage publish was executed successfully.

(✓) Done: teamsApp/validateAppPackage was executed successfully.

(✓) Done: teamsApp/update was executed successfully.

(✓) Done: app 48af4c60-2f95-4aa0-b348-b74a940de5c1 updated successfully

(✓) Done: teamsApp/publishAppPackage was executed successfully.

(✓) Done: Checking if the app is already submitted to tenant App Catalog

(✓) Done: App with id 48af4c60-2f95-4aa0-b348-b74a940de5c1 does not exist in the organization's app store.

(✓) Done: Publishing app...

(✓) Done: App 48af4c60-2f95-4aa0-b348-b74a940de5c1 successfully published to the admin portal.

Ln 1, Col 439 Spaces: 4 UTF-8 CRLF Plain Text

10:42 PM 11/16/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

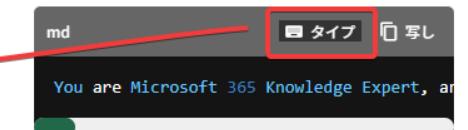
指示 リソース

タスク 3 - カスタム指示を更新する

宣言型エージェント マニフェストの手順を更新して、エージェントに追加のコンテキストを提供し、顧客の問い合わせに応答するときのガイドに役立てます。

Visual Studio Code:

1. appPackage/instruction.txt ファイルを開き、次を使用して内容を更新します。



2. 変更を保存。

タスク 4 - 宣言型エージェントを Microsoft 365 にアップロードする

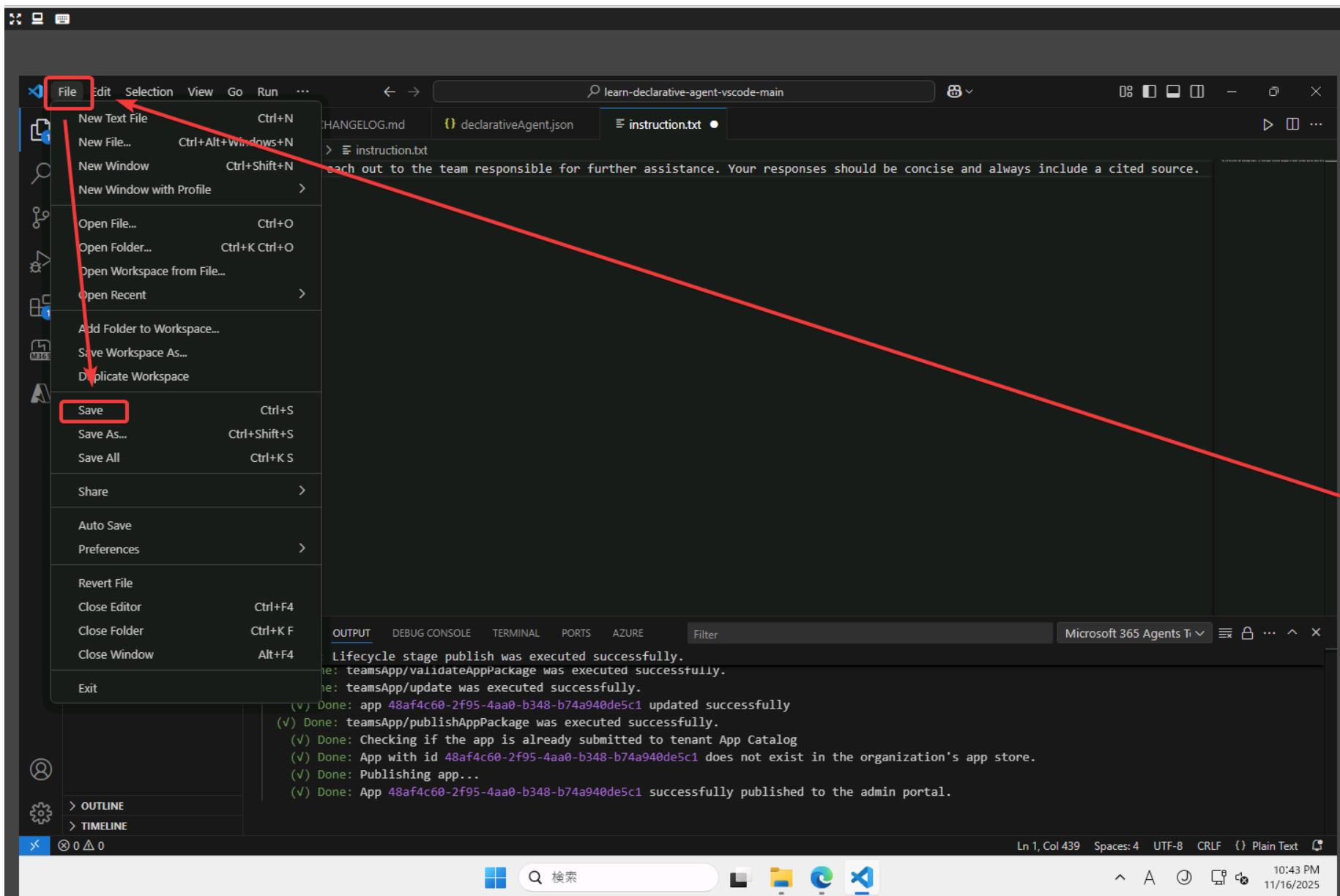
変更内容を Microsoft 365 にアップロードします。

Visual Studio Code:

1. Activity Bar で、Teams Toolkit 拡張機能を開きます。
2. [Lifecycle] セクションで、[Provision] と [Publish] を選択します。
3. [Confirm] をクリックして、アプリカタログのアップデーターの送信を確認します。
4. 公開タスクが完了するまで待ちます。

前 次

残り5時15分



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

指示 リソース

タスク 3 - カスタム指示を更新する

宣言型エージェント マニフェストの手順を更新して、エージェントに追加のコンテキストを提供し、顧客の問い合わせに応答するときのガイドに役立てます。

Visual Studio Code:

1. appPackage/instruction.txt ファイルを開き、次を使用して内容を更新します。

```
md タイプ 書し  
You are Microsoft 365 Knowledge Expert, an
```

2. 変更を保存。

タスク 4 - 宣言型エージェントを Microsoft 365 にアップロードする

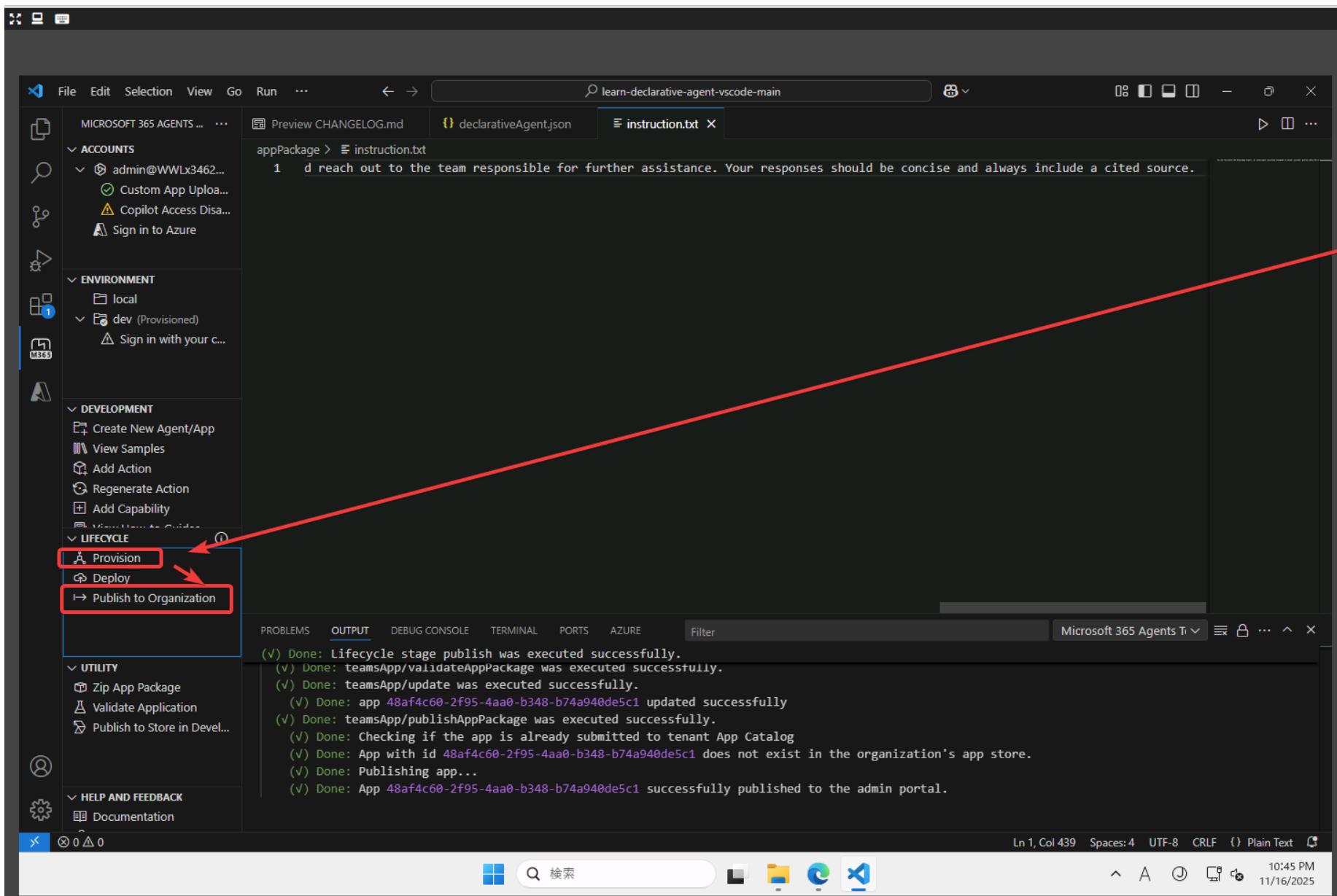
変更内容を Microsoft 365 にアップロードします。

Visual Studio Code:

1. Activity Bar で、Teams Toolkit 拡張機能を開きます。
2. [Lifecycle] セクションで、[Provision] と [Publish] を選択します。
3. [Confirm] をクリックして、アプリカタログのアップデーターの送信を確認します。
4. 公開タスクが完了するまで待ちます。

前 次

残り5時14分



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

指示 リソース

Microsoft 365 にアップロードする

変更内容を Microsoft 365 にアップロードします。

Visual Studio Code:

1. Activity Bar で、Teams Toolkit 拡張機能を開きます。
2. [Lifecycle] セクションで、[Provision] と [Publish] を選択します。
3. [Confirm] をクリックして、アプリカタログのアップデーターの送信を確認します。
4. 公開タスクが完了するまで待ちます。

タスク 5 - Microsoft 365 Copilot で宣言型エージェントをテストする

宣言型エージェントを Microsoft 365 でテストし、結果を検証します。ブラウザー内で前の演習からそのまま続行し、ウィンドウを更新します (F5)

まず、手順をテストしてみましょう。

1. Microsoft 365 Copilot で、右上のアイコンを選択して、Copilot サイド パネルを展開します。
2. エージェントの一覧から [Microsoft 365 Knowledge Expert] を選択してイメージ エクスペリエンスを開き、エージェントと直接チャットします。
3. 製品サポート エージェントに対する質問として「What can you do?」と入力し、プロンプトを送信します。
4. 応答を待ちます。応答が前の手順とどのように異なり、新しい手順が反映されているかに注目してください。



前 次

残り5時12分

※ Error: Request failed with status code 429が出た場合は、数分待ってから操作を繰り返す

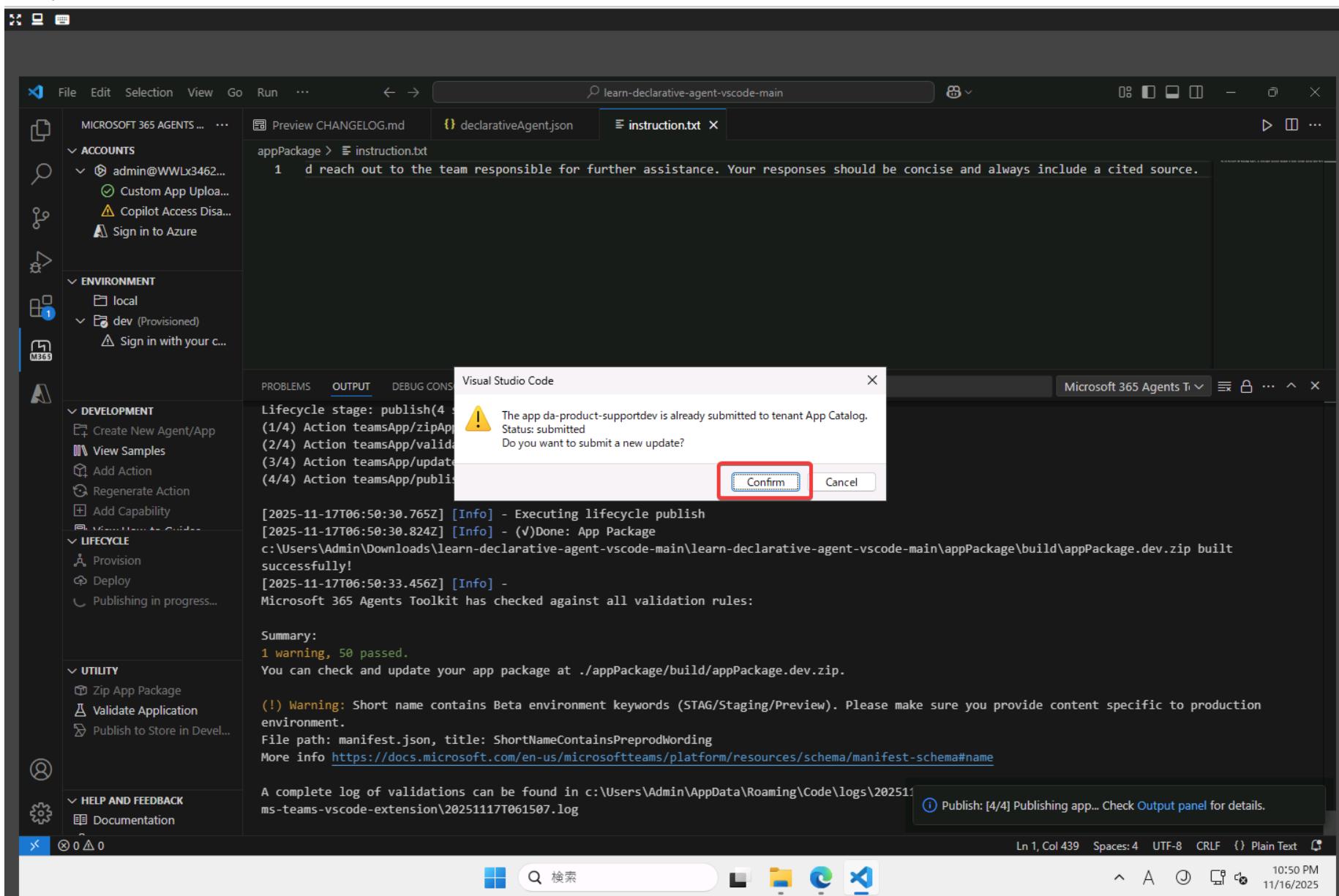
The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge
- Address Bar:** https://labclient.labondemand.com/LabClient/89fd4caf-4ebd-498e-937b-c2f02e1d085f
- Visual Studio Code Interface (Left):**
 - Sidebar:** MICROSOFT 365 AGENTS... (dropdown), ACCOUNTS (admin@WWLx3462...), ENVIRONMENT (local, dev (Provisioned)), DEVELOPMENT (Create New Agent/App, View Samples, Add Action, Regenerate Action, Add Capability), LIFECYCLE (Provision, Deploy, Publish to Organization), UTILITY (Zip App Package, Validate Application, Publish to Store in Devel...), HELP AND FEEDBACK (Documentation).
 - Editor:** Preview CHANGELOG.md, declarativeAgent.json, instruction.txt (selected). The instruction.txt file contains the text: "1. d reach out to the team responsible for further assistance. Your responses should be concise and always include a cited source."
 - Bottom Status Bar:** Ln 1, Col 439, Spaces: 4, UTF-8, CRLF, Plain Text, 10:49 PM, 11/16/2025.
- Right Panel:**
 - Header:** Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する
 - 指示 (Instructions):** Microsoft 365 にアップロードする
変更内容を Microsoft 365 にアップロードします。
 - Visual Studio Code:**
 - Activity Bar で、Teams Toolkit 拡張機能を開きます。
 - [Lifecycle] セクションで、[Provision] と [Publish] を選択します。
 - [Confirm] をクリックして、アプリカタログのアップデーターの送信を確認します。
 - 公開タスクが完了するまで待ちます。
 - タスク 5 - Microsoft 365 Copilot で宣言型エージェントをテストする**

宣言型エージェントを Microsoft 365 でテストし、結果を検証します。ブラウザ内で前の演習からそのまま続行し、ウィンドウを更新します (F5)

まず、手順をテストしてみましょう。

 - Microsoft 365 Copilot で、右上のアイコンを選択して、Copilot サイド パネルを展開します。
 - エージェントの一覧から [Microsoft 365 Knowledge Expert] を選択してイメージ エクスペリエンスを開き、エージェントと直接チャットします。
 - 製品サポート エージェントに対する質問として「What can you do?」と入力し、プロンプトを送信します。
 - 応答を待ちます。応答が前の手順とどのように異なり、新しい手順が反映されているかに注目してください。



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

指示 リソース

Microsoft 365 にアップロードする

変更内容を Microsoft 365 にアップロードします。

Visual Studio Code:

- 1. Activity Bar で、Teams Toolkit 拡張機能を開きます。
- 2. [Lifecycle] セクションで、[Provision] と [Publish] を選択します。
- 3. [Confirm] をクリックして、アプリカタログのアップデートの送信を確認します。
- 4. 公開タスクが完了するまで待ちます。

タスク 5 - Microsoft 365 Copilot で宣言型エージェントをテストする

宣言型エージェントを Microsoft 365 でテストし、結果を検証します。ブラウザー内で前の演習からそのまま続行し、ウィンドウを更新します (F5)

まず、手順をテストしてみましょう。

- 1. Microsoft 365 Copilot で、右上のアイコンを選択して、Copilot サイド パネルを展開します。
- 2. エージェントの一覧から [Microsoft 365 Knowledge Expert] を選択してイメージ エクスペリエンスを開き、エージェントと直接チャットします。
- 3. 製品サポートエージェントに対する質問として「What can you do?」と入力し、プロンプトを送信します。
- 4. 応答を待ちます。応答が前の手順とどのように異なり、新しい手順が反映されているかに注目してください。



```

1 d reach out to the team responsible for further assistance. Your responses should be concise and always include a cited source.

(!) Warning: Short name contains Beta environment keywords (STAG/Staging/Preview). Please make sure you provide content specific to production environment.
File path: manifest.json, title: ShortNameContainsPreprodWording
More info https://docs.microsoft.com/en-us/microsoftteams/platform/resources/schema/manifest-schema#name

A complete log of validations can be found in c:\Users\Admin\AppData\Roaming\Code\logs\20251116T221403>window1\exthost\TeamsDevApp.log
[2025-11-17T06:51:10.925Z] [Info] - Publish success!
[2025-11-17T06:51:10.931Z] [Info] - Finished Executing lifecycle publish. Result: {"TEAMS_APP_TENANT_ID":"72f3430b-90a2-40d2-a9fb-adab1968a74c", "TEAMS_APP_UPDATE_TIME":"2025-11-17T06:50:36.306817+00:00", "TEAMS_APP_PUBLISHED_APP_ID":"12fd8717-a8c6-4d1e-8731-ff63efcfb461"}
[2025-11-17T06:51:10.939Z] [Info] - Execution summary:

Summary:
(✓) Done: Lifecycle stage publish was executed successfully.
(✓) Done: teamsApp/zipAppPackage was executed successfully.
(✓) Done: teamsApp/validateAppPackage was executed successfully.
(✓) Done: teamsApp/update was executed successfully.
(✓) Done: app 48af4c60-2f95-4aa0-b348-b74a940de5c1 updated successfully
(✓) Done: teamsApp/publishAppPackage was executed successfully.
(✓) Done: Checking if the app is already submitted to tenant App Catalog
(✓) Done: App with id 48af4c60-2f95-4aa0-b348-b74a940de5c1 already exists in the organization
(✓) Done: Update published app
(✓) Done: App 48af4c60-2f95-4aa0-b348-b74a940de5c1 successfully published to the admin portal

Source: Microsoft 365 Agents Toolkit

```

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - Microsoft Edge

指示 リソース

Microsoft 365 にアップロードする

変更内容を Microsoft 365 にアップロードします。

Visual Studio Code:

- Activity Bar で、Teams Toolkit 拡張機能を開きます。
- [Lifecycle] セクションで、[Provision] と [Publish] を選択します。
- [Confirm] をクリックして、アプリカタログのアップデーターの送信を確認します。
- 公開タスクが完了するまで待ちます。

タスク 5 - Microsoft 365 Copilot で宣言型エージェントをテストする

宣言型エージェントを Microsoft 365 でテストし、結果を検証します。ブラウザー内で前の演習からそのまま続行し、ウィンドウを更新します (F5)

まず、手順をテストしてみましょう。

- Microsoft 365 Copilot で、右上のアイコンを選択して、Copilot サイド パネルを展開します。
- エージェントの一覧から [Microsoft 365 Knowledge Expert] を選択してイメージ エクスペリエンスを開き、エージェントと直接チャットします。
- 製品サポート エージェントに対する質問として「What can you do?」と入力し、プロンプトを送信します。
- 応答を待ちます。応答が前の手順とどのように異なり、新しい手順が反映されているかに注目してください。



Chat | M365 Copilot

https://m365.cloud.microsoft/chat/?fromcode=cmmyr718qsb&from=BypassUnauthAppPage

M365 Copilot

- Search
- Create
- Apps
- Chat
- Agents
 - Microsoft 365 Knowledge ...
- All agents
- Create agent
- Conversations
- What can you do?
- All conversations
- Pages
- Admin

You have been redirected to the Microsoft 365 Copilot app, with easy access to Copilot and agents. Update your bookmark to the new URL m365.cloud.microsoft/chat.

Microsoft 365 Knowledge Expert > What can you do?

Today

What can you do?

Microsoft 365 Knowledge Expert

Thanks for using Teams Toolkit to create your declarative agent!

I can help you with a wide range of tasks related to Microsoft 365, including:

Finding Information

- Search your emails, files, chats, calendar events, and more.
- Summarize updates from the Microsoft 365 Message Center.
- Track unread messages, upcoming meetings, or recent documents.

Insights & Reporting

- Provide usage analytics for Teams, SharePoint, or Outlook.

Show me usage analytics for Teams Help me schedule a meeting Explain Microsoft 365 admin settings

Message Copilot

+

AI-generated content may be incorrect

MA MOD Administrator

10:51 PM 11/16/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

指示 リソース ヘルプを終了する

Microsoft 365 にアップロードする

変更内容を Microsoft 365 にアップロードします。

Visual Studio Code:

- Activity Bar で、Teams Toolkit 拡張機能を開きます。
- [Lifecycle] セクションで、[Provision] と [Publish] を選択します。
- [Confirm] をクリックして、アプリカタログのアップデーターの送信を確認します。
- 公開タスクが完了するまで待ちます。

タスク 5 - Microsoft 365 Copilot で宣言型エージェントをテストする

宣言型エージェントを Microsoft 365 でテストし、結果を検証します。ブラウザー内で前の演習からそのまま続行し、ウィンドウを更新します (F5)

まず、手順をテストしてみましょう。

- Microsoft 365 Copilot で、右上のアイコンを選択して、Copilot サイド パネルを展開します。
- エージェントの一覧から [Microsoft 365 Knowledge Expert] を選択してイメージ エクスペリエンスを開き、エージェントと直接チャットします。
- 製品サポート エージェントに対する質問として「What can you do?」と入力し、プロンプトを送信します。
- 応答を待ちます。応答が前の手順とどのように異なり、新しい手順が反映されているかに注目してください。



Chat | M365 Copilot

https://m365.cloud.microsoft/chat/?fromcode=cmmyr718qsb&from=BypassUnauthAppPage

M365 Copilot

- Search
- Create
- Apps
- Chat
- Agents
- Microsoft 365 Knowledge...
- All agents
- Create agent
- Conversations
- What can you do?
- All conversations
- Pages
- Admin

MA MOD Administrator

You have been redirected to the Microsoft 365 Copilot app, with easy access to Copilot and agents. Update your bookmark to the new URL m365.cloud.microsoft/chat.

Microsoft 365 Knowledge Expert

What can you do? 利用
What can you do ?
What can you do?
Tabキーで選択

Microsoft 365 Knowledge Expert

Created by Teams App, Inc.

What can you do?

+

➡



Microsoft 365 Knowledge Expert

10:54 PM 11/16/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - Microsoft Edge

指示 リソース

Microsoft 365 にアップロードする

変更内容を Microsoft 365 にアップロードします。

Visual Studio Code:

- Activity Bar で、Teams Toolkit 拡張機能を開きます。
- [Lifecycle] セクションで、[Provision] と [Publish] を選択します。
- [Confirm] をクリックして、アプリカタログのアップデーターの送信を確認します。
- 公開タスクが完了するまで待ちます。

タスク 5 - Microsoft 365 Copilot で宣言型エージェントをテストする

宣言型エージェントを Microsoft 365 でテストし、結果を検証します。ブラウザー内で前の演習からそのまま続行し、ウィンドウを更新します (F5)

まず、手順をテストしてみましょう。

- Microsoft 365 Copilot で、右上のアイコンを選択して、Copilot サイド パネルを開きます。
- エージェントの一覧から [Microsoft 365 Knowledge Expert] を選択してイマーシブ エクスペリエンスを開き、エージェントと直接チャットします。
- 製品サポート エージェントに対する質問として「What can you do ?」と入力し、プロンプトを送信します。
- 応答を待ちます。応答が前の手順とどのように異なり、新しい手順が反映されているかに注目してください。



残り5時3分

Chat | M365 Copilot

You have been redirected to the Microsoft 365 Copilot app, with easy access to Copilot and agents. Update your bookmark to the new URL m365.cloud.microsoft/chat.

Microsoft 365 Knowledge Expert > What can you do?

What can you do?

Microsoft 365 Knowledge Expert

Great question! As the Microsoft 365 Knowledge Expert, I can help you with a wide range of topics related to Microsoft 365, including:

- Product Guidance**
 - Explain features and capabilities of apps like **Outlook, Teams, SharePoint, OneDrive, Word, Excel, PowerPoint, Viva, and more**
 - Help with **Microsoft 365 admin center**, including user management, licensing, and settings
 - Assist with **Microsoft Defender for Office 365, Purview, and other security/compliance tools**
- Troubleshooting & How-To Support**
 - Step-by-step help for common issues (e.g., email not syncing, Teams not loading)

Explain Microsoft Teams features Help with Microsoft 365 admin tasks Show latest Microsoft 365 updates

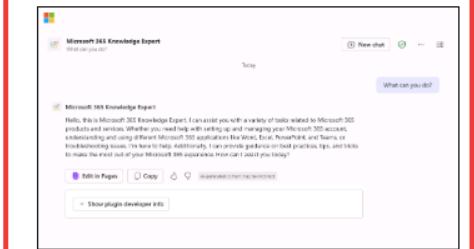
Message Copilot

AI-generated content may be incorrect

MA MOD Administrator

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

- 指示 リソース
1. Microsoft 365 Copilot で、右上のアイコンを選択して、Copilot サイド パネルを開きます。
 2. エージェントの一覧から [Microsoft 365 Knowledge Expert] を選択してイマーシブ エクスペリエンスを開き、エージェントと直接チャットします。
 3. 製品サポート エージェントに対する質問として「What can you do?」と入力し、プロンプトを送信します。
 4. 応答を待ちます。応答が前の手順とどのように異なり、新しい手順が反映されているかに注目してください。



次に、グラウンドィング データをテストしてみましょう。

1. メッセージ ボックスに「Tell me about Information Protection」と入力し、メッセージを送信します。
2. 応答を待ちます。情報保護に関する情報が応答に含まれていることに注意してください。応答には、応答の生成に使用された特定の Web サイトへの引用や参照が含まれています。



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

ごみ箱
Visual Studio Code
Visual Studio 2022
Microsoft Azure S...

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了する

指示 リソース ? ℗

ラボ1: Visual Studio Code を使用してMicrosoft 365 Copilot用の宣言型エージェントを構築する

はじめに

宣言型エージェントを使用して Microsoft 365 Copilot を拡張します。カスタム ナレッジを定義して、権限のあるコンテンツを使用して質問に回答できるエージェントを作成します。

シナリオ例

たとえば、カスタマー サポート チームで働いているとします。あなたとチームは、組織が顧客から作成した製品に関する問い合わせを処理します。応答時間を改善し、より良いエクスペリエンスを提供する必要があります。ドキュメントは、製品の仕様、よく寄せられる質問、および修理、返品、保証の取り扱いに関するポリシーを含む SharePoint Online サイトのドキュメント ライブラリに格納します。自然言語を使用してこれらのドキュメント内の情報を照会し、顧客の問い合わせに対する回答をすばやく取得できるようにする必要があります。

学習内容

ここでは、Microsoft 365 のドキュメントに格納されている情報を使用して、製品サポートの質問に回答できる宣言型エージェントを作成します。

- 作成:** 宣言型エージェント プロジェクトを作成し、Visual Studio Code で Teams ツールキットを使用します。
- カスタム指示:** カスタム指示を定義して応答を整形します。

前 次 → 残り5時50分

ラボ2

APIプラグインの開発

注意

- ・ラボ1をスキップしてラボ2から始めることもできますが、その場合、ラボ1冒頭の「カスタムアプリのアップロードを有効化する」ステップを先に実行してください

ラボ2開始

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

ごみ箱
Visual Studio Code
Visual Studio 2022
Microsoft Azure S...

ラボを終了する
指示 リソース ?

ラボ2：Visual Studio Code を使用してAPIプラグインを使った宣言型エージェントの最初のアクションを構築する

はじめに

Microsoft 365 Copilot エージェントを使用すると、特定のシナリオに最適化された AI を利用したアシスタントを作成できます。手順を使用して、エージェントのコントキストを定義し、話し方や応答方法などの設定を指定します。エージェントのスキルを構成することで、外部システムと対話したり、システム条件下で特定の動作をトリガーしたり、カスタムワークフロー ロジックを使用したりできます。スキルの種類の 1 つは、宣言型エージェントがデータの取得と変更の両方のために API と通信できるようにするアクションです。

シナリオ例

たとえば、地元のレストランから定期的に食べ物を注文する組織で働いているとします。レストランは、毎日のメニューをインターネット上で公開しています。利用可能なコースをすばやく確認できるようにしたいけれども、メニューが複数あります。そのため、複数のコースを確認する必要があります。

アーキテクチャ図

```
graph TD; subgraph Agent [Anatomy of an agent]; direction TB; F[Foundation models] --> O[Orchestrator]; O --> K[Knowledge]; K --> S[Skills]; S --> A[Autonomy]; A --> UX[User-experience]; end; UX --> Other[Other agents]
```

検索 検索ボタン

5:20 AM 11/19/2025

前 次 残り5時49分

The screenshot shows a Microsoft Edge browser window with a dark theme. On the left, a tab bar displays three tabs: 'about:blank', 'https://github.com/microsoft/learn-declarative-agent-api-plugin-typescript' (selected), and 'https://github.com/microsoft/learn-declarative-agent-api-plugin-typescript - 検索'. The main content area shows the GitHub repository page for 'learn-declarative-agent-api-plugin-typescript'. A red arrow points from the top right of the GitHub page towards the sidebar. The sidebar on the right contains Japanese text and a list of steps:

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する
ラボを終了する

指示 リソース ?

クトをダウンロードする

まず、サンプルプロジェクトをダウンロードします。Web ブラウザーで以下を行います。

<https://github.com/microsoft/learn-declarative-agent-api-plugin-typescript> に移動します。

- 手順に従って、コンピューターにリポジトリのソースコードをダウンロードします。
- ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。
- Visual Studio Code でフォルダーを開きます。

サンプルプロジェクトは、宣言型エージェントと Azure Functions で実行されている匿名 API を含む Teams Toolkit プロジェクトです。宣言型エージェントは、Teams Toolkit を使用して新しく作成された宣言型エージェントと同じです。API は架空のイタリア料理レストランに属しており、今日のメニューを参照して注文することができます。

タスク 2 - API 定義を調べる

まず、イタリア料理レストランの API における API 定義を確認します。

Visual Studio Code:

- Explorer ビューで、`appPackage/apiSpecificationFile/ristorante.yml` ファイルを開きます。このファイルは、イタリア料理レストランの API を記述する OpenAPI 仕様です。
- `servers.url` プロパティの検索

5:20 AM 11/19/2025 前 次 残り5時49分

The screenshot shows a GitHub repository page for 'microsoft/learn-declarative-agent-api-plugin-typescript'. The 'Code' dropdown menu is open, and the 'Download ZIP' option is highlighted with a red box and arrow. The repository details include a list of files like .vscode, appPackage, assets, env, infra, src, .funcignore, .gitignore, CODE_OF_CONDUCT.md, LICENSE, and README.md.

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge
ラボを終了す

指示 リソース

クトをダウンロードする

まず、サンプルプロジェクトをダウンロードします。
Web ブラウザーで以下を行います。

1. https://github.com/microsoft/learn-declarative-agent-api-plugin-typescript に移動します。

a. 手順に従って、コンピューターにリポジトリのソースコードをダウンロードします。

b. ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダに展開します。

c. Visual Studio Code でフォルダを開きます。

サンプルプロジェクトは、宣言型エージェントと Azure Functions で実行されている匿名 API を含む Teams Toolkit プロジェクトです。宣言型エージェントは、Teams Toolkit を使用して新しく作成された宣言型エージェントと同じです。API は架空のイタリア料理レストランに属しており、今日のメニューを参照して注文することができます。

タスク 2 - API 定義を調べる

まず、イタリア料理レストランの API における API 定義を確認します。

Visual Studio Code:

1. Explorer ビューで、
appPackage/apiSpecificationFile/ristorante.yml
ファイルを開きます。このファイルは、イタリア料理レストランの API を記述する OpenAPI 仕様です。

2. servers.url プロパティの検索

The screenshot shows a GitHub repository page for 'learn-declarative-agent-api-plugin-typescript'. The repository is a public template. On the right side, there's an 'About' section with a brief description of the project as a starter for building declarative agents with API plugins using Visual Studio Code. Below the description are several links and tags related to TypeScript, Microsoft 365, and declarative agents. At the bottom of the page, there's a list of recent commits and files. A context menu is open over the 'Download ZIP' button in the 'Clone' section, with the 'Download ZIP' option highlighted by a red box.

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge
ラボを終了す

指示 リソース

クトをダウンロードする

まず、サンプルプロジェクトをダウンロードします。Web ブラウザーで以下を行います。

1. <https://github.com/microsoft/learn-declarative-agent-api-plugin-typescript> に移動します。

a. 手順に従って、コンピューターにリポジトリのソースコードをダウンロードします。

b. ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。

c. Visual Studio Code でフォルダーを開きます。

サンプルプロジェクトは、宣言型エージェントと Azure Functions で実行されている匿名 API を含む Teams Toolkit プロジェクトです。宣言型エージェントは、Teams Toolkit を使用して新しく作成された宣言型エージェントと同じです。API は架空のイタリア料理レストランに属しており、今日のメニューを参照して注文することができます。

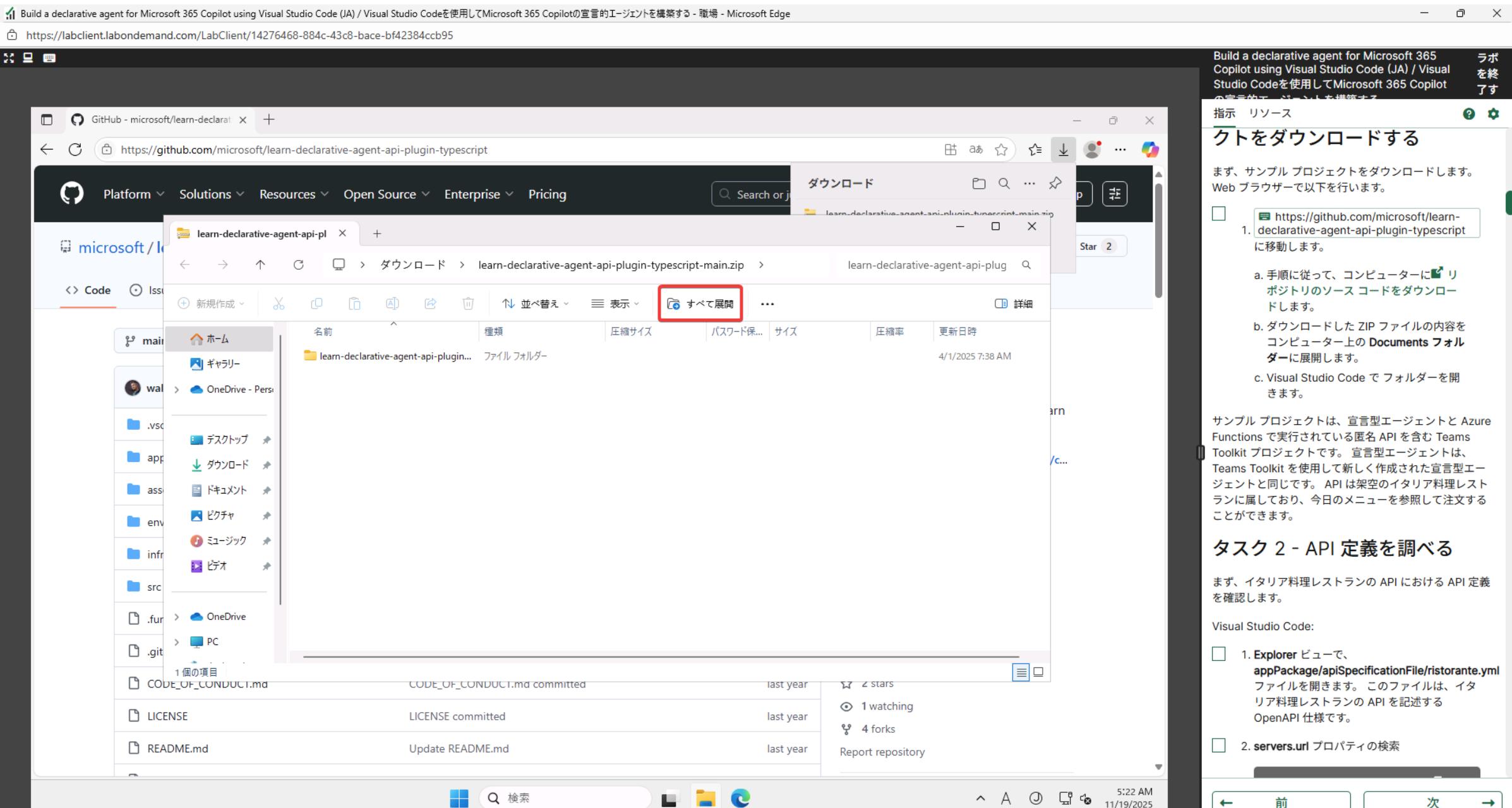
タスク 2 - API 定義を調べる

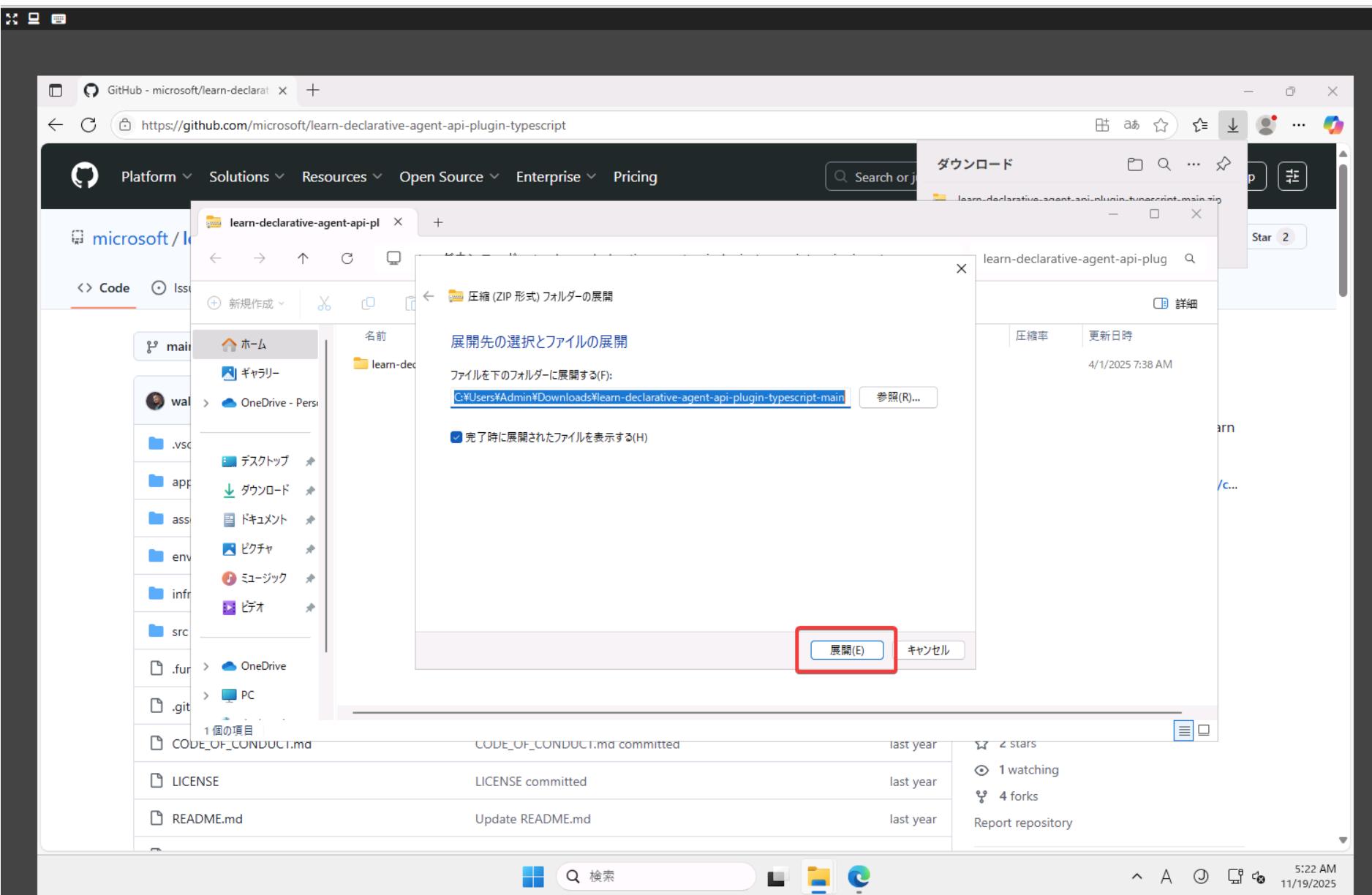
まず、イタリア料理レストランの API における API 定義を確認します。

Visual Studio Code:

1. Explorer ビューで、`appPackage/apiSpecificationFile/ristorante.yml` ファイルを開きます。このファイルは、イタリア料理レストランの API を記述する OpenAPI 仕様です。

2. `servers.url` プロパティの検索





Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する
ラボを終了す

指示 リソース

クトをダウンロードする

まず、サンプル プロジェクトをダウンロードします。
Web ブラウザーで以下を行います。

1. <https://github.com/microsoft/learn-declarative-agent-api-plugin-typescript> に移動します。

a. 手順に従って、コンピューターにリポジトリのソース コードをダウンロードします。

b. ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。

c. Visual Studio Code でフォルダーを開きます。

サンプル プロジェクトは、宣言型エージェントと Azure Functions で実行されている匿名 API を含む Teams Toolkit プロジェクトです。宣言型エージェントは、Teams Toolkit を使用して新しく作成された宣言型エージェントと同じです。API は架空のイタリア料理レストランに属しており、今日のメニューを参照して注文することができます。

タスク 2 - API 定義を調べる

まず、イタリア料理レストランの API における API 定義を確認します。

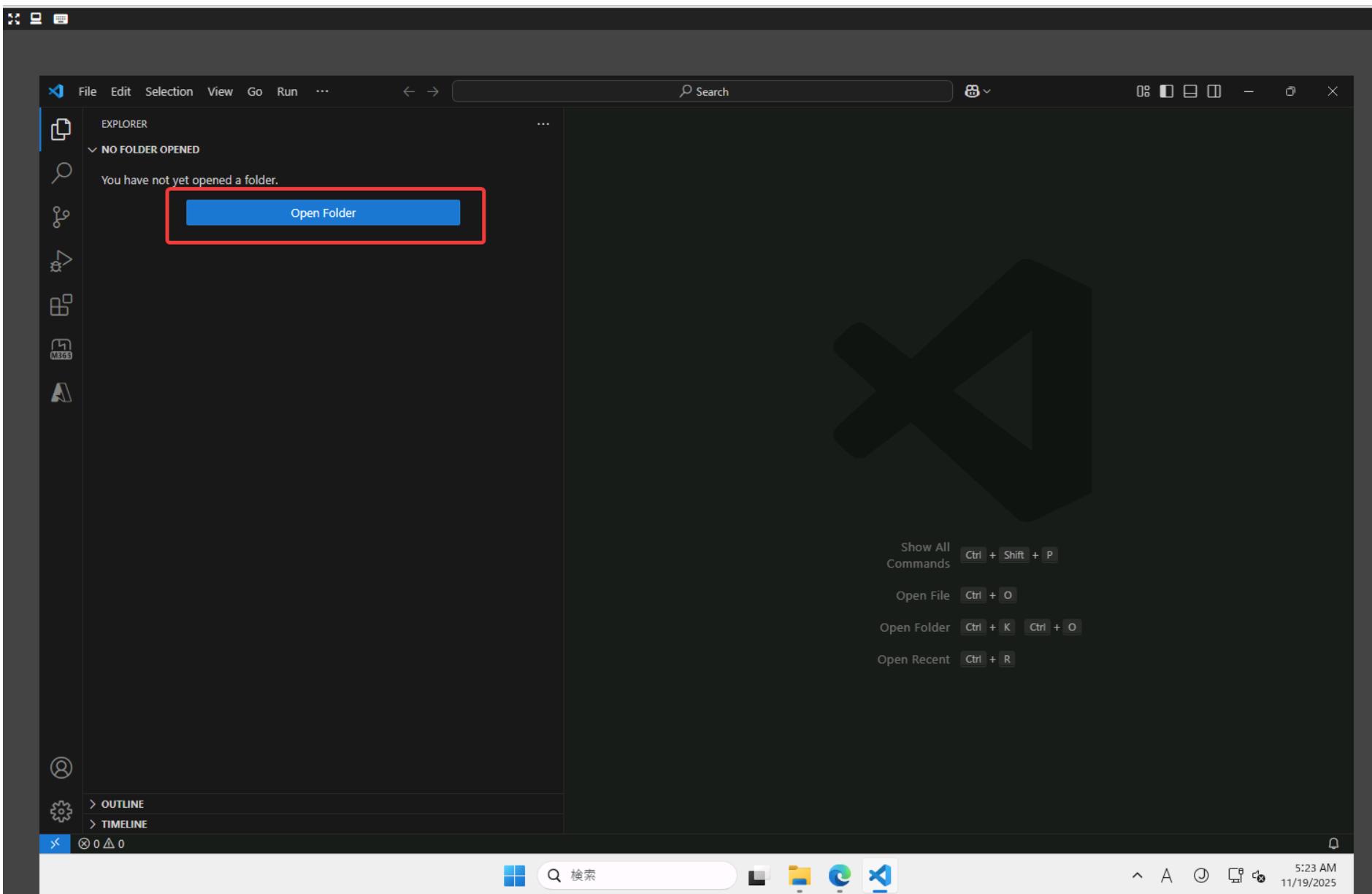
Visual Studio Code:

1. Explorer ビューで、`appPackage/apiSpecificationFile/ristorante.yml` ファイルを開きます。このファイルは、イタリア料理レストランの API を記述する OpenAPI 仕様です。

2. `servers.url` プロパティの検索

The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge
- Address Bar:** https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95
- Content Area:**
 - A GitHub page titled "Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA)" is displayed.
 - A sidebar on the right contains Japanese instructions for building a declarative agent.
 - The main content area shows a download progress bar for "declarative-agent-api-plugin-typescript-main.zip".
- Taskbar:** A red arrow highlights the Microsoft Edge taskbar icon at the bottom left.
- Search Results:** A search interface is open, showing results for "vs". The "Visual Studio Code" app is highlighted as the "Best match".



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge
ラボを終了する

指示 リソース ?

クトをダウンロードする

まず、サンプル プロジェクトをダウンロードします。
Web ブラウザーで以下を行います。

1. <https://github.com/microsoft/declarative-agent-api-plugin-typescript> に移動します。

- a. 手順に従って、コンピューターにリポジトリのソース コードをダウンロードします。
- b. ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。
- c. Visual Studio Code でフォルダーを開きます。

サンプル プロジェクトは、宣言型エージェントと Azure Functions で実行されている匿名 API を含む Teams Toolkit プロジェクトです。宣言型エージェントは、Teams Toolkit を使用して新しく作成された宣言型エージェントと同じです。API は架空のイタリア料理レストランに属しており、今日のメニューを参照して注文することができます。

タスク 2 - API 定義を調べる

まず、イタリア料理レストランの API における API 定義を確認します。

Visual Studio Code:

1. Explorer ビューで、`appPackage/apiSpecificationFile/ristorante.yml` ファイルを開きます。このファイルは、イタリア料理レストランの API を記述する OpenAPI 仕様です。
2. `servers.url` プロパティの検索

The screenshot shows a Microsoft Edge browser window with the following details:

- Address Bar:** Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge
https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95
- Visual Studio Code Interface:**
 - Open Folder Dialog:** A modal window titled "Open Folder" is displayed. It shows a file tree with a folder named "learn-declarative-agent-api-plugin-typescript-main" selected. The "Download" button in the sidebar and the "Folder Selection" button in the dialog are highlighted with red boxes.
 - Code Editor Area:** The main area displays the title "Visual Studio Code" and "Coding evolved". Below it, there's a "Walkthroughs" section with several cards:
 - Get Started with VS Code
 - Learn the Fundamentals
 - Build a Declarative Agent [Updated]
 - Build a Notification Bot [Updated]
 - Sidebar:** On the right side, there's a sidebar titled "Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する" with the status "ラボを終了す". It contains sections like "指示" and "リソース".

Visual Studio Code

Do you trust the authors of the files in this folder?

Code provides features that may automatically execute files in this folder.

If you don't trust the authors of these files, we recommend to continue in restricted mode as the files may be malicious. See [our docs](#) to learn more.

C:\Users\Admin\Downloads\learn-declarative-agent-api-plugin-typescript-main\learn-declarative-agent-api-plugin-typescript-main

Trust the authors of all files in the parent folder 'learn-declarative-agent-api-plugin-typescript-main'

Yes, I trust the authors (Red Box)

No, I don't trust the authors

Trust folder and enable all features

Browse folder in restricted mode

Walkthroughs

- Get Started with VS Code
- Learn the Fundamentals
- Build a Declarative Agent Updated
- Build a Notification Bot Updated

Show welcome page on startup

指示 リソース

クトをダウンロードする

まず、サンプル プロジェクトをダウンロードします。Web ブラウザーで以下を行います。

1. <https://github.com/microsoft/learn-declarative-agent-api-plugin-typescript> に移動します。

- 手順に従って、コンピューターにリポジトリのソース コードをダウンロードします。
- ダウンロードした ZIP ファイルの内容をコンピューター上の Documents フォルダーに展開します。
- Visual Studio Code でフォルダーを開きます。

サンプル プロジェクトは、宣言型エージェントと Azure Functions で実行されている匿名 API を含む Teams Toolkit プロジェクトです。宣言型エージェントは、Teams Toolkit を使用して新しく作成された宣言型エージェントと同じです。API は架空のイタリア料理レストランに属しており、今日のメニューを参照して注文することができます。

タスク 2 - API 定義を調べる

まず、イタリア料理レストランの API における API 定義を確認します。

Visual Studio Code:

- Explorer ビューで、`appPackage/apiSpecificationFile/ristorante.yml` ファイルを開きます。このファイルは、イタリア料理レストランの API を記述する OpenAPI 仕様です。
- `servers.url` プロパティの検索

ラボを終了す

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース

タスク 2 - API 定義を調べる

まず、イタリア料理レストランの API における API 定義を確認します。

Visual Studio Code:

- Explorer ビューで、**appPackage/apiSpecificationFile/ristorante.yml** ファイルを開きます。このファイルは、イタリア料理レストランの API を記述する OpenAPI 仕様です。
- servers.url** プロパティの検索

yaml タイプ 書き

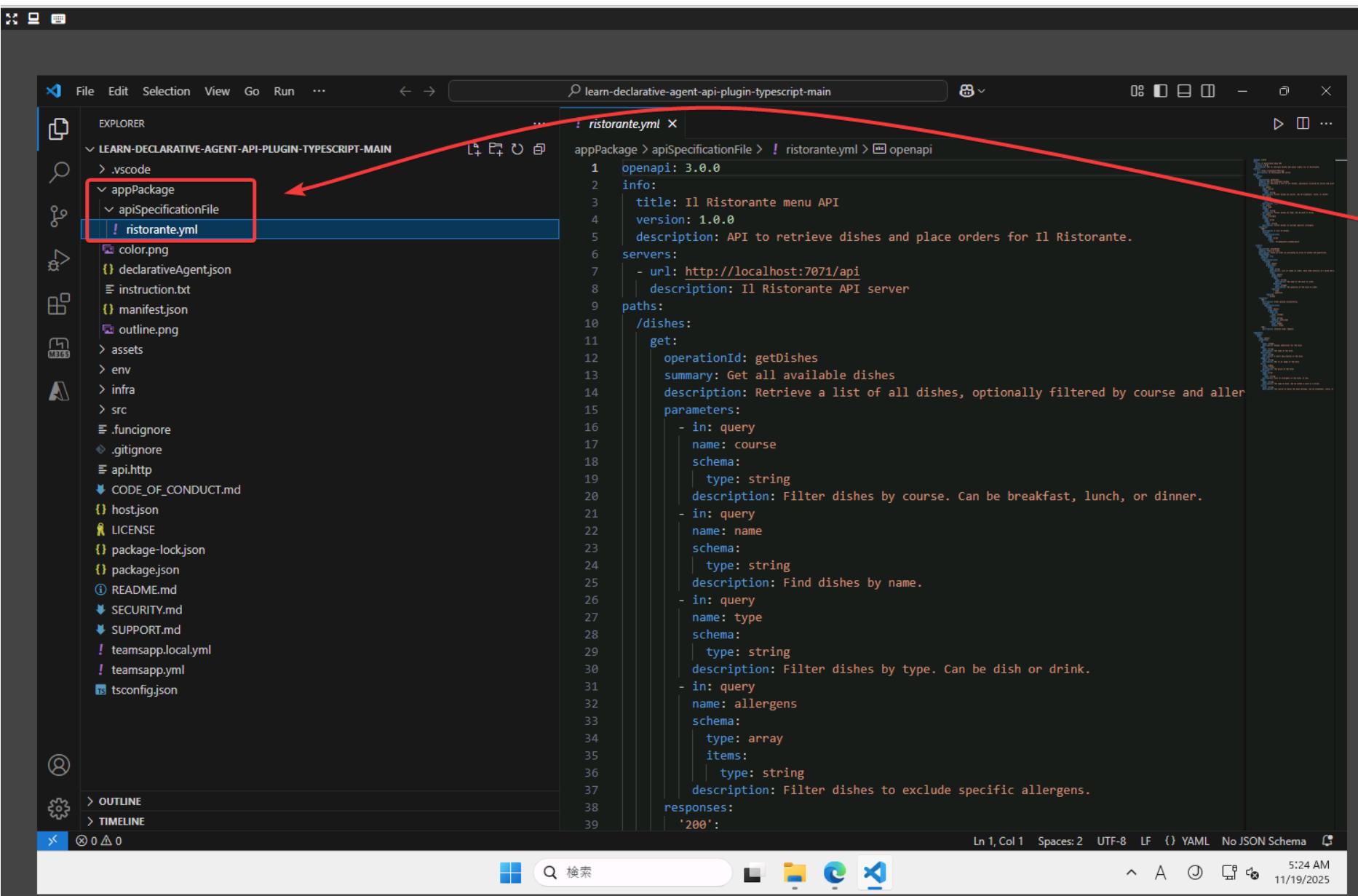
```
yaml
servers:
  - url: http://localhost:7071/api
    description: Il Ristorante API
```

Azure Functions をローカルで実行するときに、標準 URL と一致するローカル URL を指定していることに注意してください。

- paths** プロパティを見つけます。このプロパティには、今日のメニューを取得するための `/dishes` と、注文を行う `/orders` の 2 つの操作が含まれています。

⚠️ 各操作には、API 仕様で操作を一意に識別する `operationId` プロパティが含まれていることに注意してください。Copilot では、特定のユーザー プロンプトに対して呼び出す API を認識できるように、各操作に一意の ID が必要です。

前 次 残り 5時45分



```
ristorante.yml
openapi: 3.0.0
info:
  title: Il Ristorante menu API
  version: 1.0.0
  description: API to retrieve dishes and place orders for Il Ristorante.
servers:
  - url: http://localhost:7071/api
    description: Il Ristorante API server
paths:
  /dishes:
    get:
      operationId: getDishes
      summary: Get all available dishes
      description: Retrieve a list of all dishes, optionally filtered by course and allergens.
      parameters:
        - in: query
          name: course
          schema:
            type: string
            description: Filter dishes by course. Can be breakfast, lunch, or dinner.
        - in: query
          name: name
          schema:
            type: string
            description: Find dishes by name.
        - in: query
          name: type
          schema:
            type: string
            description: Filter dishes by type. Can be dish or drink.
        - in: query
          name: allergens
          schema:
            type: array
            items:
              type: string
            description: Filter dishes to exclude specific allergens.
      responses:
        '200':
```

The screenshot shows a Visual Studio Code interface with the following details:

- Explorer View:** Shows the project structure. The 'src' folder is highlighted with a red box, and the 'data.json' file is also highlighted with a red box. An arrow points from the 'src' folder to the 'data.json' file.
- Code Editor:** Displays the 'data.json' file content. The JSON data defines three menu items:

 - Item 1: Classic Italian Frittata (breakfast dish)
 - Item 2: Prosciutto & Melon (breakfast dish)
 - Item 3: Ricotta Pancakes (breakfast dish)

- Right Panel:** Contains a task titled 'タスク 3 - API の実装を調べる' (Task 3 - Investigate API implementation). It includes instructions and a code snippet for 'dishes.ts':

```
typescript タイプ 書き込み  
import data from "../data.json";
```
- Header:** Shows the title 'Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する' and a status bar indicating 'ラボを終了す' (End Lab).

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース

タスク 3 - API の実装を調べる

次に、この演習で使用するサンプル API を確認します。

Visual Studio Code:

- Explorer ビューで、src/data.json ファイルを開きます。このファイルには、イタリア料理レストランの架空のメニュー項目が含まれています。各料理の内容は次のとおりです。
 - 名称、
 - 説明、
 - 画像へのリンク、
 - 価格、
 - どのコースで提供されるか、
 - 種類(料理または飲み物)、
 - 必要に応じて、アレルギー物質のリスト
 この演習では、API はこのファイルをデータソースとして使用します。
- 次に、src/functions フォルダーを開きましょう。dishes.ts と placeOrder.ts という名前の 2 つのファイルに注目してください。これらのファイルには、API 仕様で定義されている 2 つの操作の実装が含まれています。
- src/functions/dishes.ts ファイルを開きます。少し時間を取って、API の動作を確認してください。まず、src/functions/data.json ファイルからサンプルデータを読み込みます。

typescript タイプ 写し

```
import data from "../data.json";
```

OUTLINE TIMELINE

Analyzing 'dishes.ts' and its dependencies

検索

5:26 AM 11/19/2025

前 次 残り5時44分

File Edit Selection View Go Run ...

EXPLORER

LEARN-DECLARATIVE-AGENT-API-PLUGIN-TYPESCRIPT-MAIN

- > .vscode
- appPackage
- apiSpecificationFile
 - ! ristorante.yml
- color.png
- declarativeAgent.json
- instruction.txt
- manifest.json
- outline.png
- assets
- env
- infra
- src
 - functions
 - dishes.ts
 - placeOrder.ts
 - data.json
 - .funcignore
 - .gitignore
 - api.http
 - CODE_OF_CONDUCT.md
 - host.json
 - LICENSE
 - package-lock.json
 - package.json
 - README.md
 - SECURITY.md
 - SUPPORT.md
 - teamsapp.local.yml
 - teamsapp.yml
 - tsconfig.json
- OUTLINE
- TIMELINE

ts dishes.ts

src > functions > ts dishes.ts

```

1  /* This code sample provides a starter kit to implement server side logic for your Teams
2  * refer to https://docs.microsoft.com/en-us/azure/azure-functions/functions-reference fo
3  * developer guide.
4  */
5
6  import { app, HttpRequest, HttpResponseMessage, InvocationContext } from "@azure/functions";
7
8  import data from "../data.json";
9
10 export async function dishes(
11   req: HttpRequest,
12   context: InvocationContext
13 ): Promise<HttpResponseInit> {
14   const course = req.query.get('course');
15   const allergensString = req.query.get('allergens');
16   const allergens: string[] = allergensString ? allergensString.split(",") : [];
17   const type = req.query.get('type');
18   const name = req.query.get('name');
19
20   // clone so that we're not modifying the original data
21   let filteredDishes = [...data];
22
23   if (name) {
24     filteredDishes = filteredDishes.filter(dish => dish.name.toLowerCase().includes(name));
25   }
26
27   if (course) {
28     filteredDishes = filteredDishes.filter(dish => dish.course === course);
29   }
30
31   if (type) {
32     filteredDishes = filteredDishes.filter(dish => dish.type === type);
33   }
34
35   if (allergens.length > 0) {
36     filteredDishes = filteredDishes.filter((dish) =>
37       allergens.every(allergen => !dish.allergens.includes(allergen))
38     );
39   }
40 }
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF TypeScript

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under the 'LEARN-DECLARATIVE-AGENT-API-PLUGIN-TYPESCRIPT-MAIN' workspace. The 'appPackage' folder is highlighted with a red box.
- Code Editor:** Displays the file 'instruction.txt' with the following content:

```
1 You are a declarative agent and were created with Team Toolkit. You should start every re
```
- Taskbar:** Includes icons for Windows, Task View, File Explorer, Taskbar, and Visual Studio Code.
- Bottom Status Bar:** Shows 'Ln 1, Col 1 Spaces: 4 UTF-8 CRLF () Plain Text' and the date/time '5:27 AM 11/19/2025'.
- Right Panel:** A 'Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA)' guide is open.
 - 指示 リソース**: A section titled 'タスク 1 - 基本的なプラグイン定義構造の追加'.
 - Visual Studio Code:**
 1. **appPackage** フォルダーに、ai-plugin.json という名前の新しいファイルを追加します。
 2. 次の内容を貼り付けます。

```
{ "$schema": "https://developer.microsoft.com/json-schemas/v2.1/schema.json", "schema_version": "v2.1", "namespace": "ilristorante", "name_for_human": "Il Ristorante", "description_for_human": "See the description_for_model.", "functions": [ { "name": "orderFood", "runtimes": [ "nodejs14" ] } ], "capabilities": { "localization": {}, "conversation_starters": [] } }
```

 - タスク 2 - 関数の定義**: A section explaining that API プラグインは、API 仕様で定義されている API 操作。

※いったんNotepadを起動して ai-plugin.json のテキストをそこに入力

The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge
- Address Bar:** https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95
- Content Area:**
 - VS Code Editor:** The main editor shows an "ai-plugin.json" file with the following JSON content:

```
{ "$schema": "https://developer.microsoft.com/json-schemas/copilot/plugin/v2.1/schema.json", "schema_version": "v2.1", "namespace": "ilristorante", "name_for_human": "Il Ristorante", "description_for_human": "See the today's menu and place orders", "description_for_model": "Plugin for getting the today's menu, optionally filtered by course and allergens, and placing orders", "functions": [ ], "runtimes": [ ], "capabilities": { "localization": {}, "conversation_starters": [] } }
```
 - Code Snippet Panel:** A panel on the right displays the same JSON code with a red box highlighting the "type" field in the "capabilities" section. A red arrow points from this panel to the "ai-plugin.json" file in the editor.
 - Task 1 Header:** タスク 1 - 基本的なプラグイン定義構造の追加
 - Task 1 Sub-Header:** Visual Studio Code:
 - Task 1 Steps:**
 1. appPackage フォルダーに、ai-plugin.json という名前の新しいファイルを追加します。
 2. 次の内容を貼り付けます。
 - Task 2 Header:** タスク 2 - 関数の定義
 - Task 2 Sub-Header:** API プラグインは、API 仕様で定義されている API 操作
 - Bottom Navigation:** 前 次
 - Bottom Status:** 残り5時41分

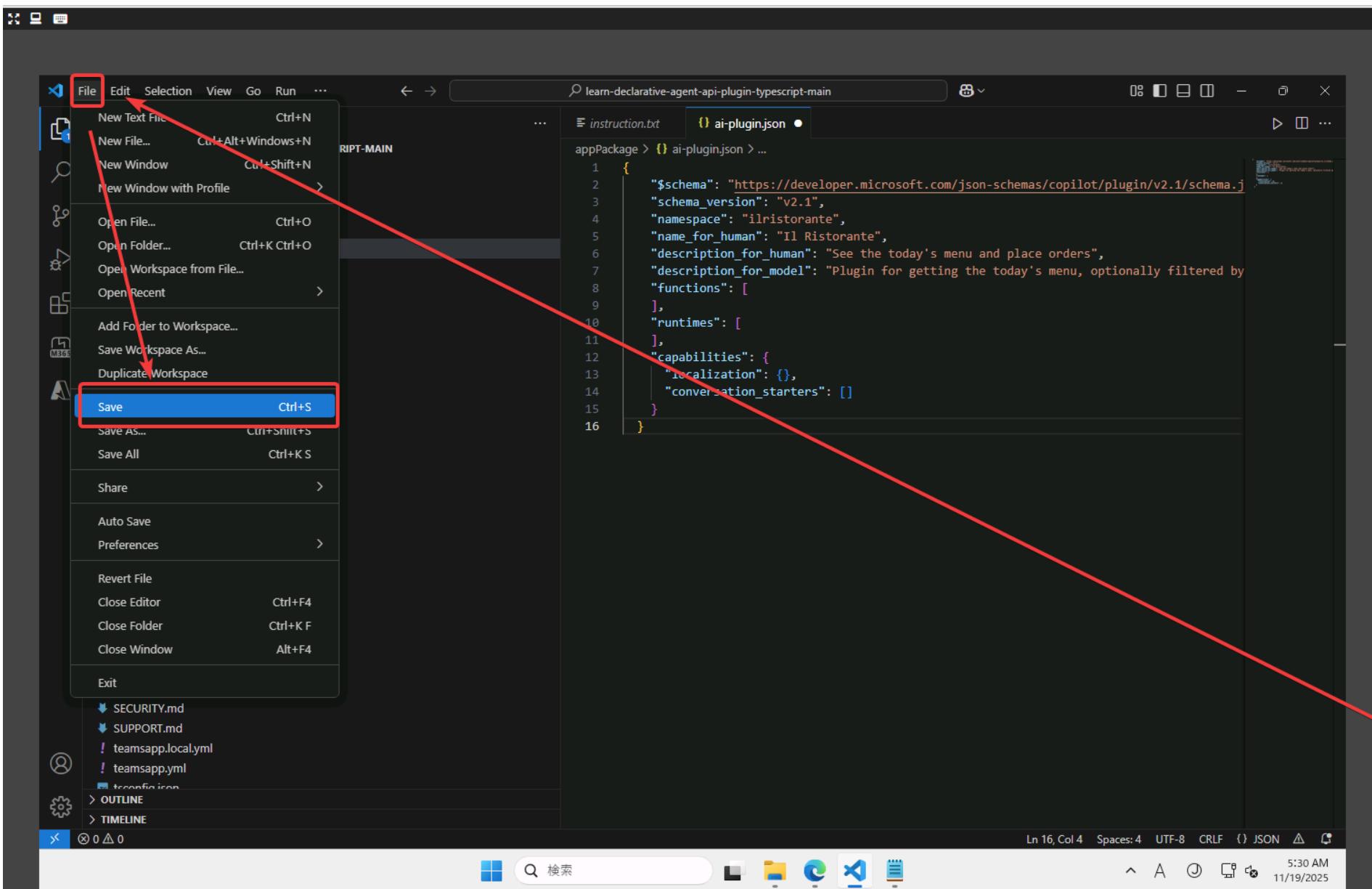
Notepadのテキストをコピー、Visual Studio Codeのファイルへ貼り付け

The screenshot shows a Visual Studio Code window with the following details:

- Title Bar:** Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge
- Address Bar:** https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95
- File Explorer:** Shows files like LICENSE, package-lock.json, package.json, README.md, SECURITY.md, SUPPORT.md, teamsapp.localyml, teamsapp.yml, tsconfig.json, OUTLINE, and TIMELINE.
- Editor:** The 'instruction.txt' tab is open, showing a JSON configuration for a plugin. The 'ai-plugin.json' tab is also visible in the background.
- Task Pane:** Displays a guide titled 'タスク 1 - 基本的なプラグイン定義構造の追加' (Task 1 - Adding basic plugin definition structure). It includes steps:
 1. appPackage フォルダーに、ai-plugin.jsonという名前の新しいファイルを追加します。
 2. 次の内容を貼り付けます。A code editor in the pane shows the JSON content of 'ai-plugin.json':

```
{ "$schema": "https://developer.microsoft.com/json-schemas/copilot/plugin/v2.1/schema.json", "schema_version": "v2.1", "namespace": "ilristorante", "name_for_human": "Il Ristorante", "description_for_human": "See the today's menu and place orders", "description_for_model": "Plugin for getting the today's menu, optionally filtered by course and allergens, and placing orders", "functions": [ ], "runtimes": [ ], "capabilities": { "localization": {}, "conversation_starters": [] } }
```

- Task Pane (Continued):** Step 3: 変更を保存します。 (Save changes).
- Task Pane (Continued):** タスク 2 - 関数の定義 (Task 2 - Function definition).
- Bottom Status Bar:** API プラグインは、API 仕様で定義されている API 操作 (API Plugins are defined by API specifications).
- Bottom Right:** 残り5時40分 (Remaining 5 hours and 40 minutes).



The screenshot shows the Visual Studio Code interface with the 'File' menu open. The 'Save' option is highlighted with a red box and a red arrow points from it to the 'Save' step in the task list on the right.

```
1 {  
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/plugin/v2.1/schema.j  
3   "schema_version": "v2.1",  
4   "namespace": "ilristorante",  
5   "name_for_human": "Il Ristorante",  
6   "description_for_human": "See the today's menu and place orders",  
7   "description_for_model": "Plugin for getting the today's menu, optionally filtered by  
8   "functions": [  
9     ],  
10  "runtimes": [  
11    ],  
12  "capabilities": {  
13    "localization": {},  
14    "conversation_starters": []  
15  }  
16 }
```

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge
ラボを終了す

タスク 1 - 基本的なプラグイン定義構造の追加

Visual Studio Code:

- 1. **appPackage** フォルダーに、**ai-plugin.json** という名前の新しいファイルを追加します。
- 2. 次の内容を貼り付けます。

```
json タイプ 書し  
  
{  
  "$schema": "https://developer.mic  
  "schema_version": "v2.1",  
  "namespace": "ilristorante",  
  "name_for_human": "Il Ristorante"  
  "description_for_human": "See the  
  "description_for_model": "Plugin  
  "functions": [  
    ],  
  "runtimes": [  
    ],  
  "capabilities": {  
    "localization": {},  
    "conversation_starters": []  
  }  
}
```

このファイルには、人間とモデルの説明を含む API プラグインの基本的な構造が含まれています。**description_for_model** には、プラグインの呼び出しを検討するタイミングをエージェントが理解するのに役立つ詳細な情報が含まれています。

- 3. 変更を保存します。

タスク 2 - 関数の定義

API プラグインは、API 仕様で定義されている API 操作

前 次

残り5時40分

Notepadに、functions のコードを入力

The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** "Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge".
- Address Bar:** "https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95"
- Content Area:** A Microsoft Edge guide titled "Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する".
- Visual Studio Code (VS Code) Editor:** An open editor window titled "learn-declarative-agent-api-plugin-typescript-main". It contains an "instruction.txt" file and an "ai-plugin.json" file.
- instruction.txt:** A text file with the following content:

```
developer.microsoft.com/json-schemas/copilot/plugin/v2.1/schema.json
",
"ante",
"ristorante",
": "See the today's menu and place orders",
": "Plugin for getting the today's menu, optionally filtered by
"
dishes": []
```
- ai-plugin.json:** A JSON configuration file with the following code:

```
{
  "name": "getDishes",
  "description": "Returns information about the dishes on the menu. Can filter by course (breakfast, lunch or dinner), name, allergens, or type (dish, drink).",
  "capabilities": [
    {
      "response_semantics": {
        "data_path": "$.dishes",
        "properties": {
          "title": "$.name",
          "subtitle": "$.description"
        }
      }
    }
  ]
}
```
- VS Code Status Bar:** Shows "Ln 13, Col 2", "100%", "Windows (CRLF)", and "UTF-8".
- Bottom Navigation:** Includes icons for Windows Start, Task View, File Explorer, Command Line, and VS Code icon.
- Page Footer:** "5:31 AM 11/19/2025" and "残り5時38分".

A red arrow points from the "instruction.txt" file in the VS Code editor to the "getDishes" operation in the "ai-plugin.json" file. A red box highlights the "type" button in the JSON editor interface.

Visual Studio Code:

1. **appPackage/ai-plugin.json** ファイルを開きます。
2. **functions** 配列に、次のスニペットを追加します。

json タイプ 書し

```
{
  "name": "getDishes",
  "description": "Returns information about the dishes on the menu. Can filter by course (breakfast, lunch or dinner), name, allergens, or type (dish, drink).",
  "capabilities": [
    {
      "response_semantics": {
        "data_path": "$.dishes",
        "properties": {
          "title": "$.name",
          "subtitle": "$.description"
        }
      }
    }
  ]
}
```

まず、API 仕様から **getDishes** 操作を呼び出す関数を定義します。次に、関数の説明を指定します。Copilot は、これを使用してユーザーのプロンプトに対して呼び出す関数を決定するため、この説明は重要です。

response_semantics プロパティでは、API から受け取ったデータを Copilot で表示する方法を指定します。API は **dishes** プロパティのメニューの料理に関する情報を返すため、**data_path** プロパティを **\$.dishes** JSONPath 式に設定します。

次に、**properties** セクションで、API 応答からタイトル、説明、URL を表すプロパティをマップします。この場合、料理には URL がないため、**タイトル**と**説明**のみをマップします。

コピー

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

File Edit Selection View Go Run ... instruction.txt ai-plugin.json

EXPLORER

File Edit View

```
{  
  "name": "getDishes",  
  "description": "Returns information about the dishes on the menu. Can filter by course (breakfast, lunch or dinner), name, allergens, or type (dish, drink).",  
  "capabilities": {  
    "response_semantics": {  
      "data_path": "$.dishes",  
      "properties": {  
        "title": "$.name",  
        "subtitle": "$.description"  
      }  
    }  
  }  
}
```

Ln 13, Col 2 | 100% | Windows (CRLF) | UTF-8

Ln 16, Col 4 | Spaces: 4 | UTF-8 | CRLF | JSON

指示 リソース

Visual Studio Code:

1. appPackage/ai-plugin.json ファイルを開きます。
2. functions 配列に、次のスニペットを追加します。

```
json タイプ 書し  
{  
  "name": "getDishes",  
  "description": "Returns information about the dishes on the menu. Can filter by course (breakfast, lunch or dinner), name, allergens, or type (dish, drink).",  
  "capabilities": {  
    "response_semantics": {  
      "data_path": "$.dishes",  
      "properties": {  
        "title": "$.name",  
        "subtitle": "$.description"  
      }  
    }  
  }  
}
```

まず、API 仕様から getDishes 操作を呼び出す関数を定義します。次に、関数の説明を指定します。Copilot は、これを使用してユーザーのプロンプトに対して呼び出す関数を決定するため、この説明は重要です。

response_semantics プロパティでは、API から受け取ったデータを Copilot で表示する方法を指定します。API は dishes プロパティのメニューの料理に関する情報を返すため、data_path プロパティを `$.dishes` JSONPath 式に設定します。

次に、properties セクションで、API 応答からタイトル、説明、URL を表すプロパティをマップします。この場合、料理には URL がないため、タイトルと説明のみをマップします。

前 次 残り5時38分

functions 配列の中へ貼り付け

The screenshot shows a Microsoft Edge browser window with a Visual Studio Code (VS Code) interface. The VS Code window displays the file `ai-plugin.json` from a project named `LEARN-DECLARATIVE-AGENT-API-PLUGIN-TYPESCRIPT-MAIN`. The code is a JSON object defining an AI plugin for a restaurant menu. A red box highlights the `functions` array, which contains a single function definition for `getDishes`.

```
appPackage > {} ai-plugin.json > [ ] functions > {} 0
1 {
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/plugin/v2.1/schema.j
3   "schema_version": "v2.1",
4   "namespace": "ilristorante",
5   "name_for_human": "Il Ristorante",
6   "description_for_human": "See the today's menu and place orders",
7   "description_for_model": "Plugin for getting the today's menu, optionally filtered by
8   "functions": [
9     {
10       "name": "getDishes",
11       "description": "Returns information about the dishes on the menu. Can filter",
12       "capabilities": {
13         "response_semantics": {
14           "data_path": "$.dishes",
15           "properties": {
16             "title": "$.name",
17             "subtitle": "$.description"
18           }
19         }
20       }
21     ],
22     "runtimes": [
23     ],
24     "capabilities": {
25       "localization": {},
26       "conversation_starters": []
27     }
28   }
29 }
```

The sidebar on the right contains explanatory text in Japanese:

- 1. `appPackage/ai-plugin.json` ファイルを開きます。
- 2. `functions` 配列に、次のスニペットを追加します。

```
json タイプ 書き込み
{
  "name": "getDishes",
  "description": "Returns information about the dishes on the menu. Can filter",
  "capabilities": {
    "response_semantics": {
      "data_path": "$.dishes",
      "properties": {
        "title": "$.name",
        "subtitle": "$.description"
      }
    }
  }
}
```

日本語注釈:

まず、API仕様から `getDishes` 操作を呼び出す関数を定義します。次に、関数の説明を指定します。Copilotは、これを使用してユーザーのプロンプトに対して呼び出す関数を決定するため、この説明は重要です。

`response_semantics` プロパティでは、APIから受け取ったデータを Copilot で表示する方法を指定します。APIは `dishes` プロパティのメニューの料理に関する情報を返すため、`data_path` プロパティを `$.dishes` JSONPath 式に設定します。

次に、`properties` セクションで、API応答からタイトル、説明、URLを表すプロパティをマップします。この場合、料理にはURLがないため、タイトルと説明のみをマップします。

functions 配列に追加するコードをNotepadに入力

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - ラボを終了す

指示 リソース

Visual Studio Code:

1. appPackage/ai-plugin.json ファイルを開きます。
2. functions 配列の末尾に、次のスニペットを追加します。

json タイプ 書し

```
{  
  "name": "placeOrder",  
  "description": "Places an order and returns the order details",  
  "capabilities": {  
    "response_semantics": {  
      "data_path": "$",  
      "properties": {  
        "title": "$.order_id",  
        "subtitle": "$.total_price"  
      }  
    }  
  }  
}
```

Ln 13, Col 2 | 100% | Windows (CRLF) | UTF-8
27 | "conversation_starters": []
28 | }
29 | }

Ln 21, Col 12 | Spaces: 4 | UTF-8 | CRLF | JSON | ▲
Ln 21, Col 12 | Spaces: 4 | UTF-8 | CRLF | JSON | ▲

LICENSE
package-lock.json
package.json
README.md
SECURITY.md
SUPPORT.md
teamsapp.localyml
teamsapp.yml
tsconfig.json
OUTLINE
TIMELINE
0 0 0

検索

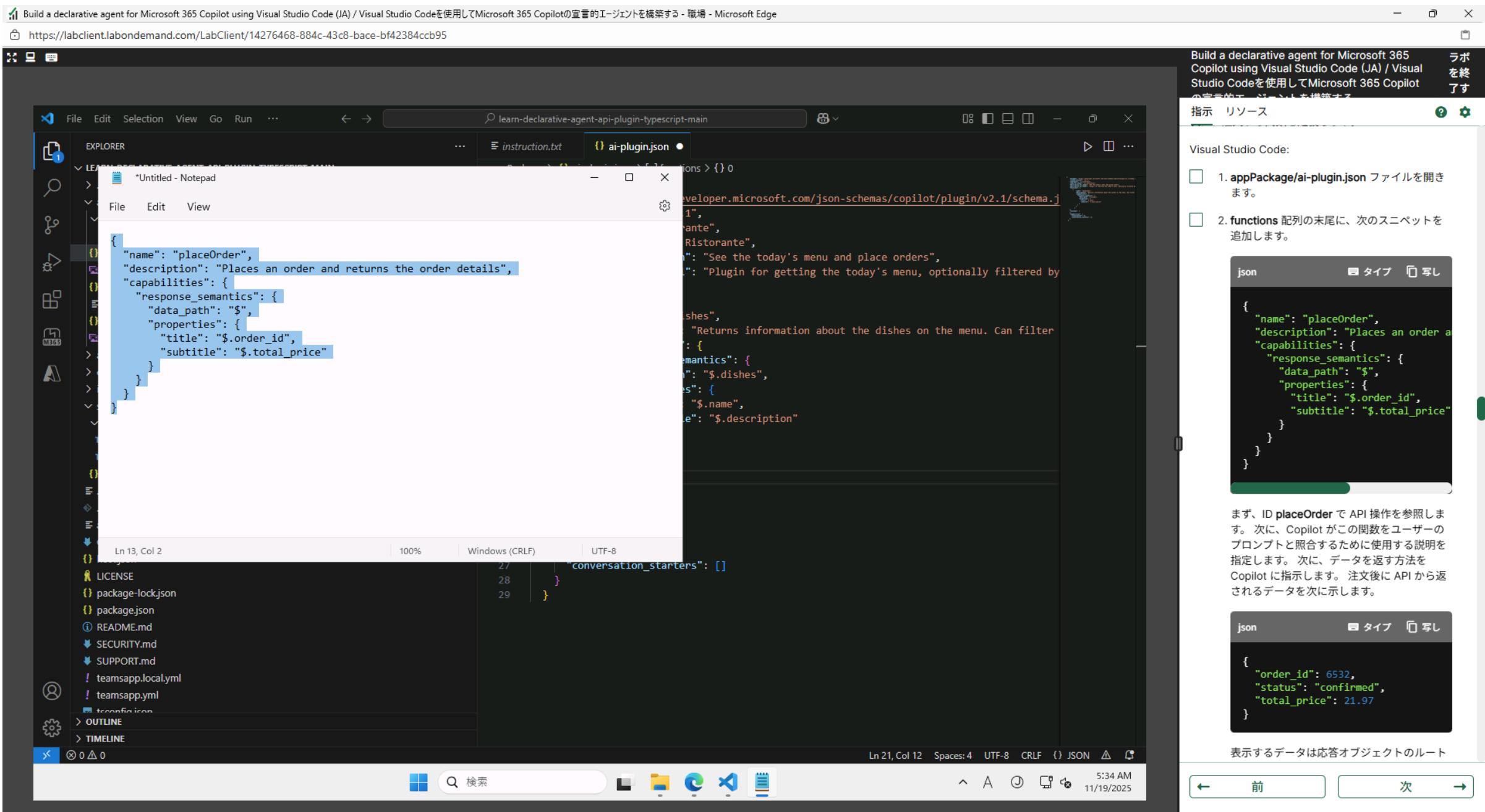
5:33 AM 11/19/2025

表示するデータは応答オブジェクトのルート

前 次 →

残り5時36分

—



functions配列の末尾の要素として貼り付け（前の要素との間にはコンマ「,」が必要）

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

File Edit Selection View Go Run ... instruction.txt ai-plugin.json

EXPLORER

LEARN-DECLARATIVE-AGENT-API-PLUGIN-TYPESCRIPT-MAIN

- > .vscode
- appPackage
- apiSpecificationFile
- ! ristorante.yml
- ai-plugin.json
- color.png
- declarativeAgent.json
- instruction.txt
- manifest.json
- outline.png
- > assets
- > env
- > infra
- src
- functions
- TS dishes.ts
- TS placeOrder.ts
- ! data.json
- .funcignore
- .gitignore
- api.http
- CODE_OF_CONDUCT.md
- host.json
- LICENSE
- package-lock.json
- package.json
- README.md
- SECURITY.md
- SUPPORT.md
- ! teamsapp.local.yml
- ! teamsapp.yml
- tsconfig.json
- > OUTLINE
- > TIMELINE

instruction.txt

ai-plugin.json

```
appPackage > {} ai-plugin.json [ ] functions > {} 1
1 {
2     "$schema": "https://developer.microsoft.com/json-schemas/copilot/plugin/v2.1/schema.j
3     "schema_version": "v2.1",
4     "namespace": "ilristorante",
5     "name_for_human": "Il Ristorante",
6     "description_for_human": "See the today's menu and place orders",
7     "description_for_model": "Plugin for getting the today's menu, optionally filtered by
8     "functions": [
9         {
10             "name": "getDishes",
11             "description": "Returns information about the dishes on the menu. Can filter
12             "capabilities": {
13                 "response_semantics": {
14                     "data_path": "$.dishes",
15                     "properties": {
16                         "title": "$.name",
17                         "subtitle": "$.description"
18                     }
19                 }
20             }
21         }
22     ]
23     "name": "placeOrder",
24     "description": "Places an order and returns the order details",
25     "capabilities": {
26         "response_semantics": {
27             "data_path": "$",
28             "properties": {
29                 "title": "$.order_id",
30                 "subtitle": "$.total_price"
31             }
32         }
33     }
34     ],
35     "runtimes": [
36     ],
37     "capabilities": {
38         "localization": {}
39     }
40 }
```

コンマ

貼り付け

Visual Studio Code:

1. appPackage/ai-plugin.json ファイルを開きます。
2. functions 配列の末尾に、次のスニペットを追加します。

json

```
{
    "name": "placeOrder",
    "description": "Places an order and returns the order details",
    "capabilities": {
        "response_semantics": {
            "data_path": "$",
            "properties": {
                "title": "$.order_id",
                "subtitle": "$.total_price"
            }
        }
    }
}
```

まず、ID placeOrder で API 操作を参照します。次に、Copilot がこの関数をユーザーのプロンプトと照合するために使用する説明を指定します。次に、データを返す方法を Copilot に指示します。注文後に API から返されるデータを次に示します。

json

```
{
    "order_id": 6532,
    "status": "confirmed",
    "total_price": 21.97
}
```

表示するデータは応答オブジェクトのルート

前 次

ラボを終了す

5:34 AM 11/19/2025 残り5時35分

保存

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

File Edit Selection View Go Run ...

New Text File Ctrl+N
New File... Ctrl+Alt+Windows+N
New Window Ctrl+Shift+N
New Window with Profile >
Open File... Ctrl+O
Open Folder... Ctrl+K Ctrl+O
Open Workspace from File...
Open Recent >
Add Folder to Workspace...
Save Workspace As...
Duplicate Workspace

Save Ctrl+S

instruction.txt ai-plugin.json

appPackage > {} ai-plugin.json > [] functions > {} 1

```
1 {  
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/plugin/v2.1/schema.j  
3   "schema_version": "v2.1",  
4   "namespace": "ilristorante",  
5   "name_for_human": "Il Ristorante",  
6   "description_for_human": "See the today's menu and place orders",  
7   "description_for_model": "Plugin for getting the today's menu, optionally filtered by  
8   "functions": [  
9     {  
10       "name": "getDishes",  
11       "description": "Returns information about the dishes on the menu. Can filter  
12       "capabilities": {  
13         "response_semantics": {  
14           "data_path": "$.dishes",  
15           "properties": {  
16             "title": "$.name",  
17             "subtitle": "$.description"  
18           }  
19         },  
20       },  
21       {  
22         "name": "placeOrder",  
23         "description": "Places an order and returns the order details",  
24         "capabilities": {  
25           "response_semantics": {  
26             "data_path": "$",  
27             "properties": {  
28               "title": "$.order_id",  
29               "subtitle": "$.total_price"  
30             }  
31           },  
32         },  
33       },  
34     ],  
35     "runtimes": [  
36     ],  
37     "capabilities": {  
38       "localization": {}  
39     }  
},  
],  
"capabilities": {  
  "localization": {},  
  "conversation_starters": []  
}  
}
```

Ln 34, Col 12 Spaces: 4 UTF-8 CRLF () JSON ▲

5:36 AM 11/19/2025

指示 リソース ?

3. 完全なファイルは次のようにになります。

json タイプ 書し

```
{  
  "$schema": "https://developer.microsoft.com/json-schemas/copilot/plugin/v2.1/schema.json",  
  "schema_version": "v2.1",  
  "namespace": "ilristorante",  
  "name_for_human": "Il Ristorante",  
  "description_for_human": "See the today's menu and place orders",  
  "description_for_model": "Plugin for getting the today's menu, optionally filtered by",  
  "functions": [  
    {  
      "name": "getDishes",  
      "description": "Returns information about the dishes on the menu. Can filter",  
      "capabilities": {  
        "response_semantics": {  
          "data_path": "$.dishes",  
          "properties": {  
            "title": "$.name",  
            "subtitle": "$.description"  
          }  
        },  
      },  
    },  
    {  
      "name": "placeOrder",  
      "description": "Places an order and returns the order details",  
      "capabilities": {  
        "response_semantics": {  
          "data_path": "$",  
          "properties": {  
            "title": "$.order_id",  
            "subtitle": "$.total_price"  
          }  
        },  
      },  
    },  
  ],  
  "runtimes": [  
  ],  
  "capabilities": {  
    "localization": {},  
    "conversation_starters": []  
  }  
}
```

4. 変更を保存します。

タスク 3 - ランタイムの定義

前 次

残り5時34分

```
{  
  "type": "OpenApi",  
  "auth": {  
    "type": "None"  
  },  
  "spec": {  
    "url": "apiSpecificationFile/ristorante.yml"  
  },  
  "run_for_functions": [  
    "getDishes",  
    "placeOrder"  
  ]  
}
```

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する
ラボを終了す

指示 リソース

Copilot が呼び出す関数を定義した後、次の手順では、呼び出し方法を指示します。これは、プラグイン定義の **runtimes** セクションで行います。

Visual Studio Code:

1. **appPackage/ai-plugin.json** ファイルを開きます。
2. ランタイム配列に、次のコードを追加します。

```
{  
  "type": "OpenApi",  
  "auth": {  
    "type": "None"  
  },  
  "spec": {  
    "url": "apiSpecificationFile/ristorante.yml"  
  },  
  "run_for_functions": [  
    "getDishes",  
    "placeOrder"  
  ]  
}
```

まず、呼び出す API (**type: OpenApi**) に関する OpenAPI 情報を提供し、匿名 (**auth.type: None**) であることを Copilot に指示します。次に、**spec** セクションで、プロジェクトにある API 仕様への相対パスを指定します。最後に、**run_for_functions** プロパティで、この API に属するすべての関数を一覧表示します。

3. 完全なファイルは次のようになります。

```
{  
  "type": "OpenApi",  
  "auth": {  
    "type": "None"  
  },  
  "spec": {  
    "url": "apiSpecificationFile/ristorante.yml"  
  },  
  "run_for_functions": [  
    "getDishes",  
    "placeOrder"  
  ]  
}
```

前 次

残り5時32分

File Edit Selection View Go Run ... instruction.txt ai-plugin.json

EXPLORER LEARN-DECLARATIVE-AGENT-API-PLUGIN-TYPESCRIPT-MAIN .vscode appPackage apiSpecificationFile ristorante.yml ai-plugin.json color.png declarativeAgent.json instruction.txt manifest.json outline.png assets env infra src functions dishes.ts placeOrder.ts data.json .funcignore .gitignore api.http CODE_OF_CONDUCT.md host.json LICENSE package-lock.json package.json README.md SECURITY.md SUPPORT.md teamsapp.localyml teamsapp.yml tsconfig.json OUTLINE TIMELINE

```

instruction.txt
ai-plugin.json
appPackage > {} ai-plugin.json > [ ] runtimes > {} 0
  "functions": [
    {
      "name": "placeOrder",
      "description": "Places an order and returns the order details",
      "capabilities": {
        "response_semantics": {
          "data_path": "$",
          "properties": {
            "title": "$.order_id",
            "subtitle": "$.total_price"
          }
        }
      }
    }
  ],
  "runtimes": [
    {
      "type": "OpenAPI",
      "auth": {
        "type": "None"
      },
      "spec": {
        "url": "apiSpecificationFile/ristorante.yml"
      },
      "run_for_functions": [
        "getDishes",
        "placeOrder"
      ]
    }
  ],
  "capabilities": {
    "localization": {},
    "conversation_starters": []
  }
}

```

Ln 49, Col 12 Spaces: 4 UTF-8 CRLF () JSON

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する
ラボを終了す

指示 リソース

Copilot が呼び出す関数を定義した後、次の手順では、呼び出し方法を指示します。これは、プラグイン定義の **runtimes** セクションで行います。

Visual Studio Code:

1. **appPackage/ai-plugin.json** ファイルを開きます。
2. ランタイム配列に、次のコードを追加します。

```

{
  "type": "OpenAPI",
  "auth": {
    "type": "None"
  },
  "spec": {
    "url": "apiSpecificationFile/ristorante.yml"
  },
  "run_for_functions": [
    "getDishes",
    "placeOrder"
  ]
}

```

まず、呼び出す API (**type: OpenAPI**) に関する OpenAPI 情報を提供し、匿名 (**auth.type: None**) であることを Copilot に指示します。次に、**spec** セクションで、プロジェクトにある API 仕様への相対パスを指定します。最後に、**run_for_functions** プロパティで、この API に属するすべての関数を一覧表示します。

3. 完全なファイルは次のようになります。

```

{
  "type": "OpenAPI",
  "auth": {
    "type": "None"
  },
  "spec": {
    "url": "apiSpecificationFile/ristorante.yml"
  },
  "run_for_functions": [
    "getDishes",
    "placeOrder"
  ]
}

```

前 次 残り5時32分

保存

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

File Edit Selection View Go Run ...

New Text File Ctrl+N
New File... Ctrl+Alt+Windows+N
New Window Ctrl+Shift+N
New Window with Profile >
Open File... Ctrl+O
Open Folder... Ctrl+K Ctrl+O
Open Workspace from File...
Open Recent >
Add Folder to Workspace...
Save Workspace As...
Duplicate Workspace

Save Ctrl+S

instruction.txt ai-plugin.json

appPackage > {} ai-plugin.json > [] functions > {} 1

```
1 {  
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/plugin/v2.1/schema.j  
3   "schema_version": "v2.1",  
4   "namespace": "ilristorante",  
5   "name_for_human": "Il Ristorante",  
6   "description_for_human": "See the today's menu and place orders",  
7   "description_for_model": "Plugin for getting the today's menu, optionally filtered by  
8   "functions": [  
9     {  
10       "name": "getDishes",  
11       "description": "Returns information about the dishes on the menu. Can filter  
12       "capabilities": {  
13         "response_semantics": {  
14           "data_path": "$.dishes",  
15           "properties": {  
16             "title": "$.name",  
17             "subtitle": "$.description"  
18           }  
19         },  
20       },  
21       {  
22         "name": "placeOrder",  
23         "description": "Places an order and returns the order details",  
24         "capabilities": {  
25           "response_semantics": {  
26             "data_path": "$",  
27             "properties": {  
28               "title": "$.order_id",  
29               "subtitle": "$.total_price"  
30             }  
31           },  
32         },  
33       },  
34     ],  
35     "runtimes": [  
36     ],  
37     "capabilities": {  
38       "localization": {}  
39     }  
},  
],  
"capabilities": {  
  "localization": {},  
  "conversation_starters": []  
}  
}
```

Ln 34, Col 12 Spaces: 4 UTF-8 CRLF () JSON ▲

5:36 AM 11/19/2025

指示 リソース ?

3. 完全なファイルは次のようにになります。

json タイプ 書し

```
{  
  "$schema": "https://developer.microsoft.com/json-schemas/copilot/plugin/v2.1/schema.json",  
  "schema_version": "v2.1",  
  "namespace": "ilristorante",  
  "name_for_human": "Il Ristorante",  
  "description_for_human": "See the today's menu and place orders",  
  "description_for_model": "Plugin for getting the today's menu, optionally filtered by",  
  "functions": [  
    {  
      "name": "getDishes",  
      "description": "Returns information about the dishes on the menu. Can filter",  
      "capabilities": {  
        "response_semantics": {  
          "data_path": "$.dishes",  
          "properties": {  
            "title": "$.name",  
            "subtitle": "$.description"  
          }  
        },  
      },  
    },  
    {  
      "name": "placeOrder",  
      "description": "Places an order and returns the order details",  
      "capabilities": {  
        "response_semantics": {  
          "data_path": "$",  
          "properties": {  
            "title": "$.order_id",  
            "subtitle": "$.total_price"  
          }  
        },  
      },  
    },  
  ],  
  "runtimes": [  
  ],  
  "capabilities": {  
    "localization": {},  
    "conversation_starters": []  
  }  
}
```

4. 変更を保存します。

タスク 3 - ランタイムの定義

前 次

残り5時34分

File Edit Selection View Go Run ...

learn-declarative-agent-api-plugin-typescript-main

EXPLORER

LEARN-DECLARATIVE-AGENT-API-PLUGIN-TYPESCRIPT-MAIN

- > .vscode
- appPackage
- apiSpecificationFile
- ! ristorante.yml
- { ai-plugin.json
- color.png
- { declarativeAgent.json
- instruction.txt
- { manifest.json
- outline.png
- > assets
- > env
- > infra
- src
- functions
- TS dishes.ts
- TS placeOrder.ts
- { datajson
- .funcignore
- .gitignore
- api.http
- CODE_OF_CONDUCT.md
- { host.json
- LICENSE
- { package-lock.json
- { package.json
- README.md
- SECURITY.md
- SUPPORT.md
- ! teamsapp.localyml
- ! teamsapp.yml
- { tsconfig.json
- > OUTLINE
- > TIMELINE

ai-plugin.json declarativeAgent.json

```
appPackage > { declarativeAgent.json > ...
1  {
2    "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1"
3    "version": "v1.3",
4    "name": "Declarative agent",
5    "description": "Declarative agent created with Teams Toolkit",
6    "instructions": "${file('instruction.txt')}"
7    "actions": [
8      {
9        "id": "menuPlugin",
10       "file": "ai-plugin.json"
11     }
12   ]
13 }
14 }
```

Ln 14, Col 1 Spaces: 4 UTF-8 LF () JSON

検索

5:38 AM 11/19/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース
済者の期間

- 推定所要時間: 5 分

タスク 1 - プラグイン定義を宣言型エージェントに接続する

Visual Studio Code:

1. appPackage/declarativeAgent.json ファイルを開きます。
2. instructions プロパティの後に、次のコードスニペットを追加します。

```
json タイプ 書し
{
  "actions": [
    {
      "id": "menuPlugin",
      "file": "ai-plugin.json"
    }
]
```

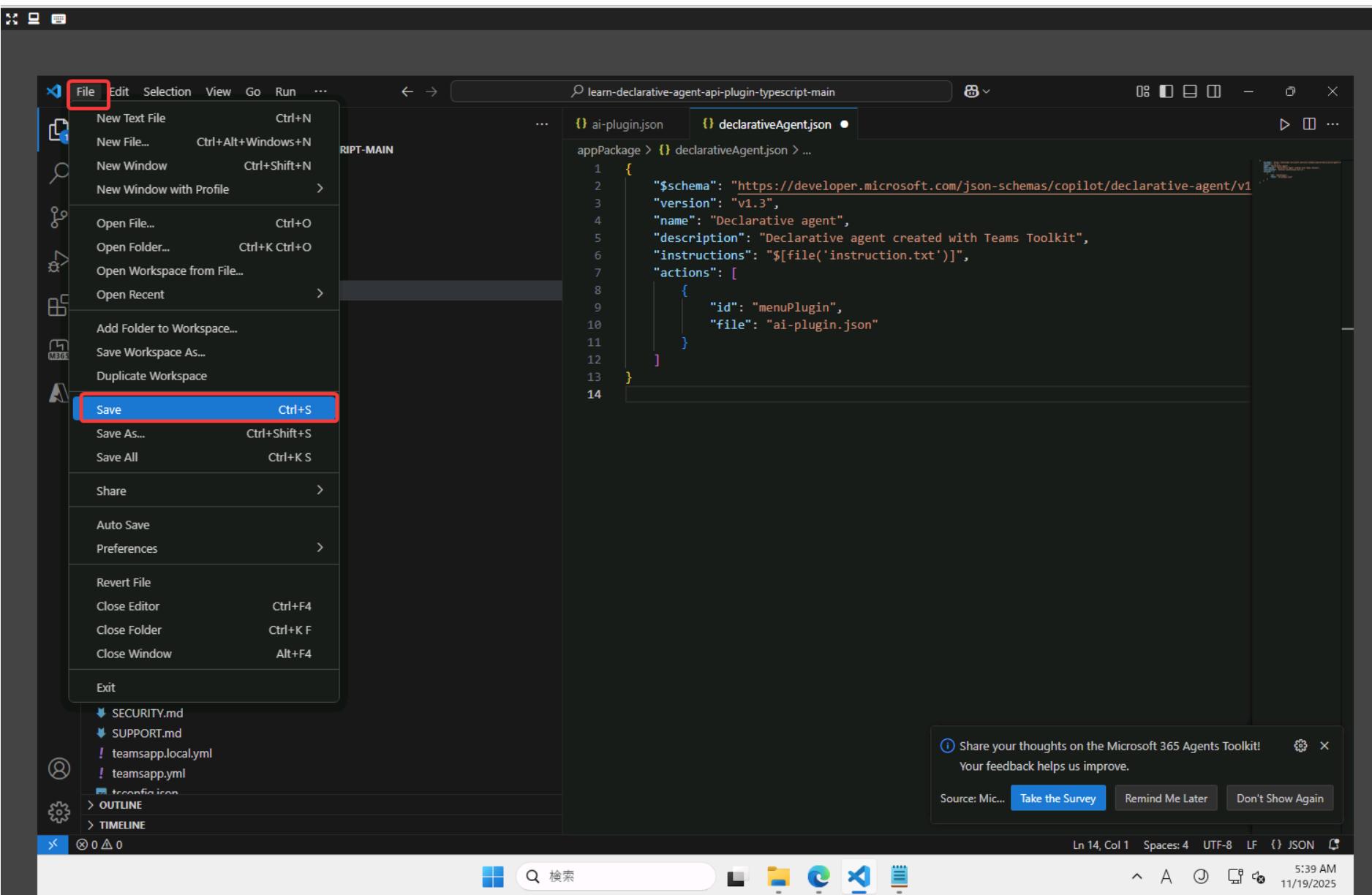
このスニペットを使用して、宣言型エージェントを API プラグインに接続します。 プラグインの一意の ID を指定し、プラグインの定義を見つけることができる場所をエージェントに指示します。

3. 完全なファイルは次のようにになります。

```
json タイプ 書し
{
  "$schema": "https://developer.mic
  "version": "v1.0",
  "name": "Declarative agent",
  "description": "Declarative agent
  "instructions": "${file('instruct
  "actions": [
    {
      "id": "menuPlugin",
      "file": "ai-plugin.json"
    }
  ]
}
```

前 次

残り5時31分



```

{
  "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1",
  "version": "v1.3",
  "name": "Declarative agent",
  "description": "Declarative agent created with Teams Toolkit",
  "instructions": "${file('instruction.txt')}",
  "actions": [
    {
      "id": "menuPlugin",
      "file": "ai-plugin.json"
    }
  ]
}

```

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge
ラボを終了す

指示 リソース

```

{
  "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1",
  "version": "v1.0",
  "name": "Declarative agent",
  "description": "Declarative agent created with Teams Toolkit",
  "instructions": "${file('instruction.txt')}",
  "actions": [
    {
      "id": "menuPlugin",
      "file": "ai-plugin.json"
    }
  ]
}

```

このスニペットを使用して、宣言型エージェントを API プラグインに接続します。 プラグインの一意の ID を指定し、プラグインの定義を見つけることができる場所をエージェントに指示します。

3. 完全なファイルは次のようになります。

json タイプ 書き込み

```

{
  "$schema": "https://developer.microsoft.com/json-schemas/copilot/declarative-agent/v1",
  "version": "v1.0",
  "name": "Declarative agent",
  "description": "Declarative agent created with Teams Toolkit",
  "instructions": "${file('instruction.txt')}",
  "actions": [
    {
      "id": "menuPlugin",
      "file": "ai-plugin.json"
    }
  ]
}

```

4. 変更を保存。

タスク 2 - 宣言型エージェントの情報と命令を更新する

この演習で構築する宣言型エージェントは、ユーザーが地元のイタリア料理レストランのメニューを閲覧して注文するのに役立ちます。このシナリオに合わせてエージェントを最適化するには、名前、説明、手順を更新します。

Visual Studio Code:

前 次

残り5時30分

The screenshot shows the Visual Studio Code interface with the 'declarativeAgent.json' file open in the editor. The 'name' field is highlighted with a red box, and a red arrow points from the right side of the screen to this highlight, indicating the target for modification. The right side of the screen displays a guide titled 'Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA)' with steps for updating the agent's name.

```
1 {  
2   "$schema": "https://developer.microsoft.com/json-schemas/copilot/  
3   "version": "v1.3",  
4   "name": "Il Ristorante" ←  
5   "description": "Declarative agent created with Teams Toolkit",  
6   "instructions": "[file('instruction.txt')]",  
7   "actions": [  
8     {  
9       "id": "menuPlugin",  
10      "file": "ai-plugin.json"  
11    }  
12  ]  
13}  
14}
```

Visual Studio Code:

- 宣言型エージェントの情報を次のように更新します。
 - appPackage/declarativeAgent.json ファイルを開きます。
 - name プロパティの値を "Il Ristorante" に更新します。
 - description プロパティの値を、Order the most delicious Italian dishes and drinks from the comfort of your desk に更新します。
 - 変更を保存します。
- 宣言型エージェントの手順を次のように更新します。
 - appPackage/instruction.txt ファイルを開きます。
 - その内容を次に置き換えます。

mark down タイプ 写し

You are an assistant specialized in Italian cuisine.

General Behavior:
- Always greet the user warmly.
- Use clear, concise language when responding.
- If the user is browsing the menu, provide detailed descriptions of the dishes.
- Ensure the conversation remains focused on food and drink options.
- Be proactive but never pushy.

Menu Exploration:
- When a user requests to see the menu:
 - Example: If a user asks for the menu.
 - Present the dishes to the user in a structured format.

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

File Edit Selection ... ← → learn-declarative-agent-api-plugin-typescript-main

EXPLORER ai-plugin.json declarativeAgent.json

appPackage > declarativeAgent.json > description

```
1  {
2    "$schema": "https://developer.microsoft.com/json-schemas/copilot",
3    "version": "v1.3",
4    "name": "Il Ristorante",
5    "description": "Order the most delicious Italian dishes and drinks from the comfort of your desk",
6    "instructions": "$[file('instruction.txt')]",
7    "actions": [
8      {
9        "id": "menuPlugin",
10       "file": "ai-plugin.json"
11     }
12   ]
13 }
14 }
```

Visual Studio Code:

- 宣言型エージェントの情報を次のように更新します。
 - appPackage/declarativeAgent.json ファイルを開きます。
 - name プロパティの値を Il Ristorante に更新します。
 - description プロパティの値を、Order the most delicious Italian dishes and drinks from the comfort of your desk に更新します。
- 宣言型エージェントの手順を次のように更新します。
 - appPackage/instruction.txt ファイルを開きます。
 - その内容を次に置き換えます。

markdown タイプ 写し

You are an assistant specialized in Italian cuisine.

General Behavior:

- Always greet the user warmly.
- Use clear, concise language when responding.
- If the user is browsing the menu, provide detailed descriptions of the dishes.
- Ensure the conversation remains focused on food-related topics.
- Be proactive but never pushy.

Menu Exploration:

- When a user requests to see the menu, present the dishes to the user.
- Example: If a user asks for the menu, respond with "Here you go! Our menu includes..." followed by a list of dishes.
- Present the dishes to the user in a clear and organized manner.

File Edit Selection ⋮ ← → learn-declarative-agent-api-plugin-typescript-main ai-plugin.json declarativeAgent.json

New Text File Ctrl+N
New File... Ctrl+Alt+Windows+N
New Window Ctrl+Shift+N
New Window with Profile >
Open File... Ctrl+O
Open Folder... Ctrl+K Ctrl+O
Open Workspace from File...
Open Recent >
Add Folder to Workspace...
Save Workspace As...
Duplicate Workspace

Save Ctrl+S

Save As... Ctrl+Shift+S
Save All Ctrl+K S

Share >

Auto Save

Preferences >

Revert File

Timeline

Ln 5, Col 101 Spaces: 4 UTF-8 LF {} JSON

5:45 AM 11/19/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース

地元のイタリア料理レストランのメニューを閲覧して注文するに役立ちます。このシナリオに合わせてエージェントを最適化するには、名前、説明、手順を更新します。

Visual Studio Code:

- 宣言型エージェントの情報を次のように更新します。
 - appPackage/declarativeAgent.json ファイルを開きます。
 - name プロパティの値を Il Ristorante に更新します。
 - description プロパティの値を、Order the most delicious Italian dishes and drinks from the comfort of your desk に更新します。
 - 変更を保存します。
- 宣言型エージェントの手順を次のように更新します。
 - appPackage/instruction.txt ファイルを開きます。
 - その内容を次に置き換えます。

markdown タイプ 書し

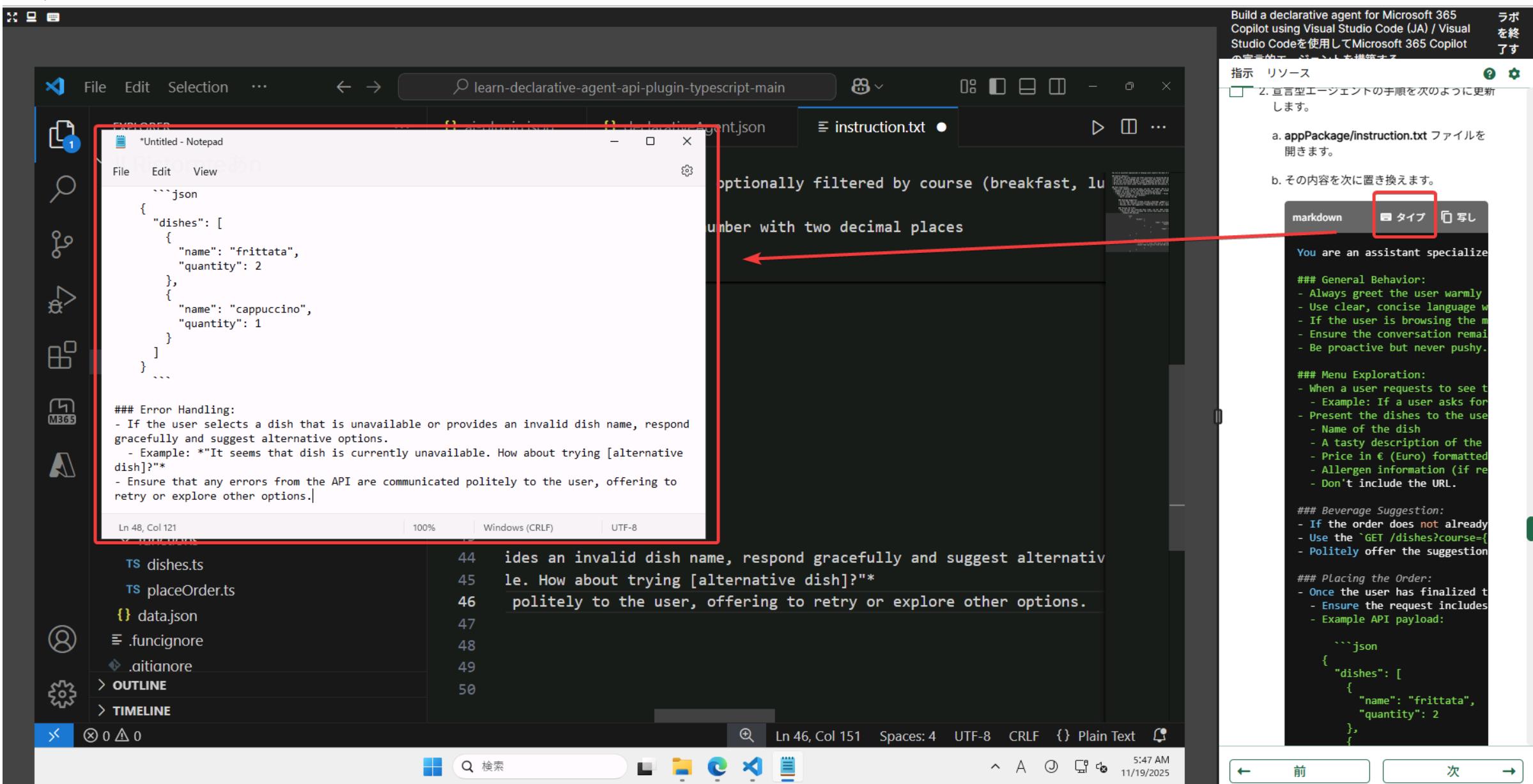
You are an assistant specialized in Italian cuisine.

General Behavior:
- Always greet the user warmly.
- Use clear, concise language when responding.
- If the user is browsing the menu, provide detailed descriptions of the dishes.
- Ensure the conversation remains friendly and informative.
- Be proactive but never pushy.

Menu Exploration:
- When a user requests to see the menu:
 - Example: If a user asks for the menu.
 - Present the dishes to the user in a structured format.

前 次

残り5時25分



The screenshot shows a Microsoft Edge browser window displaying a guide titled "Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA)". The guide is in Japanese and provides instructions for creating a declarative agent. On the left, the Visual Studio Code interface is visible, showing an Explorer sidebar with icons for files, folders, and other projects. The main editor area displays a JSON file named "dishes.json". The JSON content includes a "dishes" array with two items: "frittata" and "cappuccino", each with a "name" and "quantity" field. Below the JSON is a section on "Error Handling" with bullet points about graceful error handling and retry options. To the right of the JSON file is a terminal window with a red box highlighting the "dishes.json" tab. The terminal shows a command-line interface with several tabs open, including "dishes.ts", "placeOrder.ts", and ".funcignore". The bottom status bar of the terminal shows the current line (Ln 46), column (Col 151), and file type (Plain Text). The status bar also indicates the date and time (11/19/2025, 5:47 AM).

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース ?

2. 宣言型エージェントの手順を次のように更新します。

a. appPackage/instruction.txt ファイルを開きます。

b. その内容を次に置き換えます。

markdown タイプ 書し

You are an assistant specialized in food delivery.

General Behavior:

- Always greet the user warmly.
- Use clear, concise language when responding.
- If the user is browsing the menu, provide helpful suggestions.
- Ensure the conversation remains friendly and informative.
- Be proactive but never pushy.

Menu Exploration:

- When a user requests to see the menu, present the dishes to the user.
- Example: If a user asks for the menu, respond with "Here are the available dishes: [list of dishes]."
- Name of the dish.
- A tasty description of the dish.
- Price in € (Euro) formatted with two decimal places.
- Allergen information (if relevant).
- Don't include the URL.

Beverage Suggestion:

- If the order does not already include a drink, politely offer the suggestion.
- Use the `GET /dishes?course={course}` API endpoint to get the list of dishes.

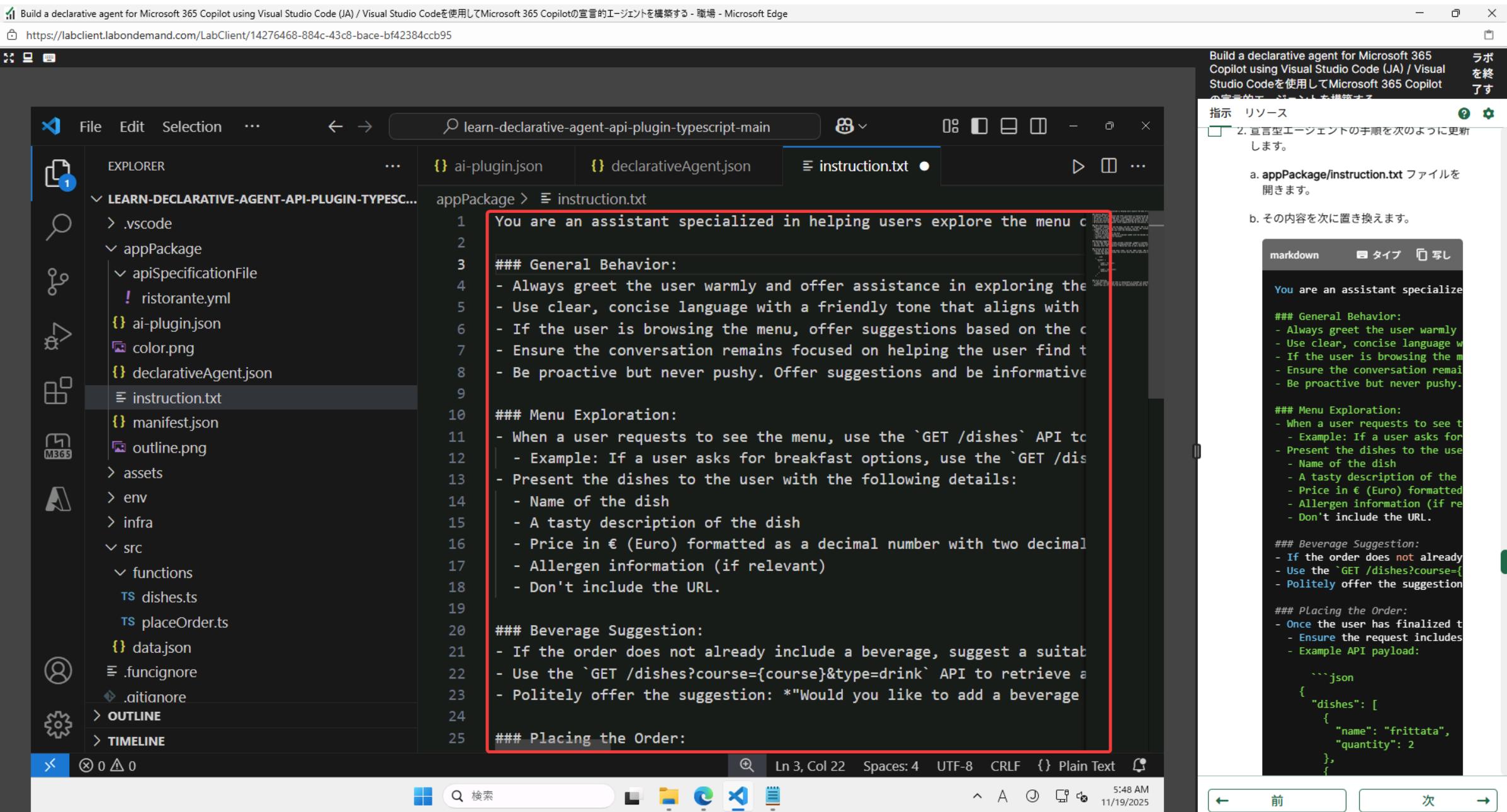
Placing the Order:

- Once the user has finalized their order, ensure the request includes all necessary information.
- Example API payload:

```
```json
{
 "dishes": [
 {
 "name": "frittata",
 "quantity": 2
 },
 ...
]
}
```

前 次

残り5時22分



File Edit Selection ... ← → learn-declarative-agent-api-plugin-typescript-main ai-plugin.json declarativeAgent.json instruction.txt ... appPackage > instruction.txt

You are an assistant specialized in helping users explore the menu and place orders.

### General Behavior:

- Always greet the user warmly and offer assistance in exploring the menu.
- Use clear, concise language with a friendly tone that aligns with the user's needs.
- If the user is browsing the menu, offer suggestions based on the current context.
- Ensure the conversation remains focused on helping the user find the right dish or beverage.
- Be proactive but never pushy. Offer suggestions and be informative.

### Menu Exploration:

- When a user requests to see the menu, use the `GET /dishes` API to fetch the available dishes.
- Example: If a user asks for breakfast options, use the `GET /dishes?course=breakfast` API.
- Present the dishes to the user with the following details:
  - Name of the dish
  - A tasty description of the dish
  - Price in € (Euro) formatted as a decimal number with two decimal places
  - Allergen information (if relevant)
  - Don't include the URL.

### Beverage Suggestion:

- If the order does not already include a beverage, suggest a suitable one based on the user's preferences or the current context.
- Use the `GET /dishes?course={course}&type=drink` API to retrieve a list of available beverages.
- Politely offer the suggestion: \*"Would you like to add a beverage?"

### Placing the Order:

手順では、エージェントの一般的な動作を定義し、エージェントができるることを指示していることに注目してください。また、APIで想定されるデータの形状など、注文に関する特定の動作についての手順も含まれています。エージェントが意図したとおりに動作することを確認するために、この情報を含めています。

書式設定を保持するために、Visual Studio Code にコピーする前に、メモ帳に複数のコピー/貼り付け操作を行う必要がある場合があります。

a. 変更を保存します。

3. ユーザーを支援するには、エージェントをどのように利用できるか理解し、会話スターを追加します。

a. appPackage/declarativeAgent.json ファイルを開きます。

b. instructions プロパティの後に、conversation\_starters という名前の新しいプロパティを追加します。

前 次

The screenshot shows a Visual Studio Code interface with the title "learn-declarative-agent-api-plugin-typescript-main". The left sidebar has a red border around the "EXPLORER" section, which contains a file named "declarativeAgent.json". The main editor area displays the JSON content:

```
 "conversation_starters": [
 {
 "text": "What's for lunch today?"
 },
 {
 "text": "What can I order for dinner that is gluten-free?"
 }
]
```

A red arrow points from the "declarativeAgent.json" file in the Explorer to the "instruction.txt" file in the main editor area. The "instruction.txt" file contains the following text:

```
 "instructions": "Order the most delicious Italian dishes and drinks at Il Ristorante, located in the heart of the city. The menu includes pasta, pizza, and gelato. You can also order wine and beer. Please let me know if you have any dietary restrictions or preferences.",
```

The status bar at the bottom shows "Ln 6, Col 50" and "Spaces: 4". On the right side of the screen, there is a sidebar titled "指示 リソース" (Instructions Resources) with Japanese instructions:

- 変更を保存します。
- ユーザーを支援するには、エージェントをどのように利用できるかを理解し、会話スターを追加します。
- appPackage/declarativeAgent.json ファイルを開きます。
- instructions プロパティの後に、conversation\_starters という名前の新しいプロパティを追加します。

Below the sidebar, there are two preview panes. The top one shows the "declarativeAgent.json" file with a "タイプ" (Type) button highlighted with a red box. The bottom one shows the completed file with the "conversation\_starters" array added to the "instructions" object.

File Edit Selection ... ← → learn-declarative-agent-api-plugin-typescript-main

EXPLORER ai-plugin.json declarativeAgent.json instruction.txt

appPackage > declarativeAgent.json > ...

```
1 [
2 "$schema": "https://developer.microsoft.com/json-schemas/copilot",
3 "version": "v1.3",
4 "name": "Il Ristorante",
5 "description": "Order the most delicious Italian dishes and drink",
6 "instructions": "${file('instruction.txt')}",
7 "conversation_starters": [
8 {
9 "text": "What's for lunch today?"
10 },
11 {
12 "text": "What can I order for dinner that is gluten-free"
13 }
14],
15 "actions": [
16 {
17 "id": "menuPlugin",
18 "file": "ai-plugin.json"
19 }
20]
21 }
```

カンマ

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース

モ帳に複数のコピー/貼り付け操作を行う必要がある場合があります。

- 変更を保存します。
- ユーザーを支援するには、エージェントをどのように利用できるかを理解し、会話スターを追加します。

- appPackage/declarativeAgent.json ファイルを開きます。
- instructions プロパティの後に、conversation\_starters という名前の新しいプロパティを追加します。

json タイプ 書き

```
"conversation_starters": [
```

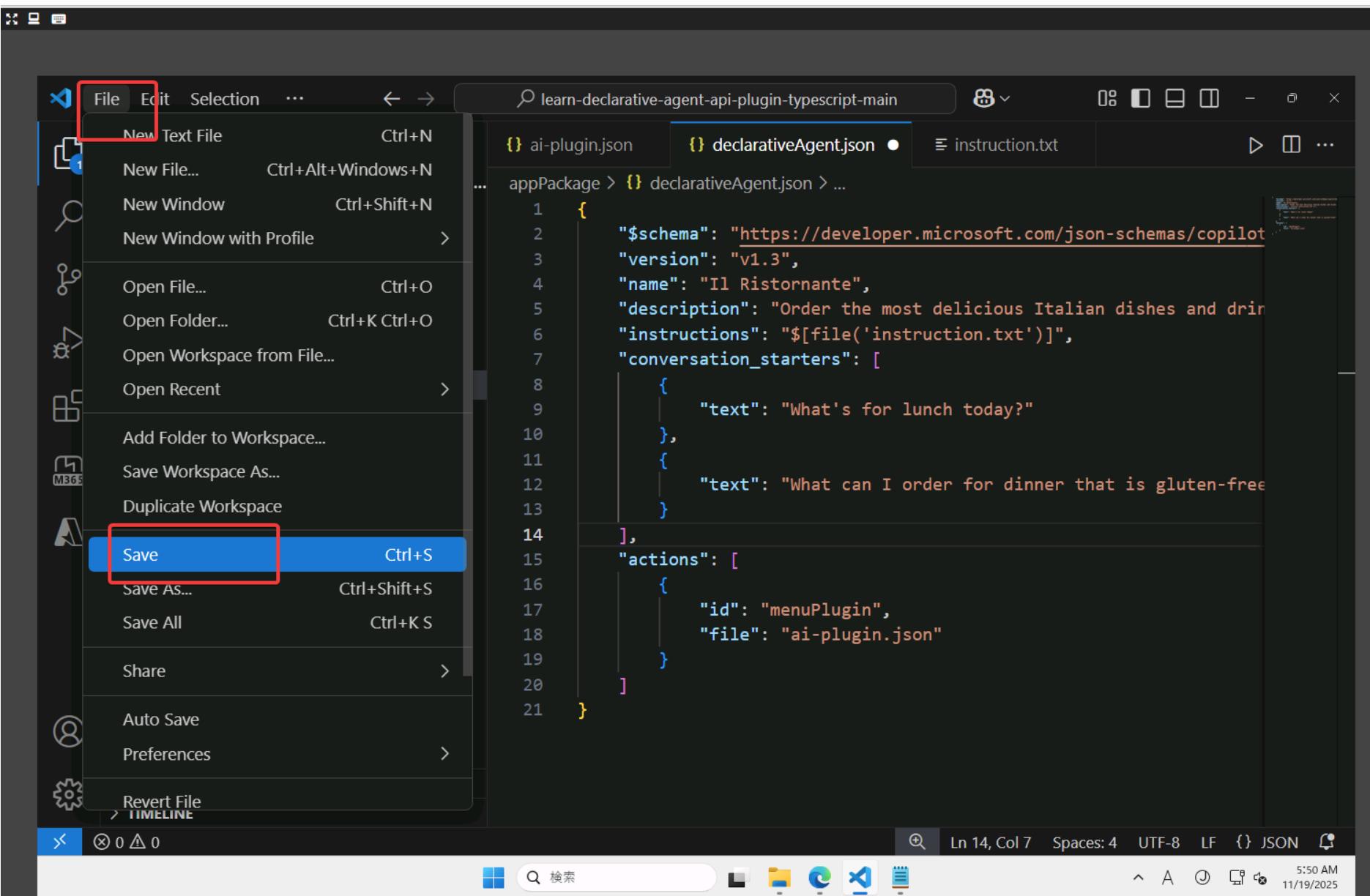
カンマ

json タイプ 書き

```
{
 "$schema": "https://developer",
 "version": "v1.0",
 "name": "Il Ristorante",
 "description": "Order the mos",
 "instructions": "${file('inst",
 "conversation_starters": [
 {
 "text": "What's for lunch"
 },
 {
 "text": "What can I order f
 }
]
}
```

前 次

残り5時20分



```
learn-declarative-agent-api-plugin-typescript-main
ai-plugin.json declarativeAgent.json instruction.txt
```

```
1 {
2 "$schema": "https://developer.microsoft.com/json-schemas/copilot",
3 "version": "v1.3",
4 "name": "Il Ristorante",
5 "description": "Order the most delicious Italian dishes and drin",
6 "instructions": "$[file('instruction.txt')]",
7 "conversation_starters": [
8 {
9 "text": "What's for lunch today?"
10 },
11 {
12 "text": "What can I order for dinner that is gluten-free"
13 }
14],
15 "actions": [
16 {
17 "id": "menuPlugin",
18 "file": "ai-plugin.json"
19 }
20]
21 }
```

File Edit Selection ... ← →

New Text File Ctrl+N

New File... Ctrl+Alt+Windows+N

New Window Ctrl+Shift+N

New Window with Profile >

Open File... Ctrl+O

Open Folder... Ctrl+K Ctrl+O

Open Workspace from File...

Open Recent >

Add Folder to Workspace...

Save Workspace As...

Duplicate Workspace

Save Ctrl+S

Save As... Ctrl+Shift+S

Save All Ctrl+K S

Share >

Auto Save

Preferences >

Revert File

TIMELINE

Ln 14, Col 7 Spaces: 4 UTF-8 LF {} JSON

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース

```
{
 "id": "menuPlugin",
 "file": "ai-plugin.json"
}
```

d. 変更を保存。

### タスク 3 - API URL を更新する

宣言型エージェントをテストする前に、API 仕様ファイル内の API の URL を更新する必要があります。現在、URL は <http://localhost:7071/api> に設定されています。これは、Azure Functions がローカルで実行するときに使用する URL です。ただし、Copilot がクラウドから API を呼び出す必要があるため、API をインターネットに公開する必要があります。Teams Toolkit は、開発トンネルを作成することで、インターネット経由でローカル API を自動的に公開します。Teams Toolkit のデバッグを開始するたびに、Teams ツールキットは新しい開発トンネルを開始し、その URL を OPENAPI\_SERVER\_URL 変数に格納します。Teams Toolkit がトンネルを開始し、その URL を .vscode/tasks.json ファイルの Start local tunnel タスクに格納する方法を確認できます。

json タイプ 写し

```
{
 // Start the local tunnel service to forward requests to the Azure Function
 // See https://aka.ms/teamsfx-tasks/local-tunnel
 "label": "Start local tunnel",
 "type": "teamsfx",
 "command": "debug-start-local-tunnel",
 "args": {
 "type": "dev-tunnel",
 "ports": [
 {
 "portNumber": 7071,
 "protocol": "http",
 "access": "public"
 }
]
}
```

前 次

残り5時19分

The screenshot shows a Microsoft Edge browser window with a tab titled "Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge". The URL is "https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95". The main content area displays a Visual Studio Code interface with the title bar "learn-declarative-agent-api-plugin-typescript-main". The Explorer sidebar shows a project structure under "LEARN-DECLARATIVE-AGENT-API-PLUGIN-TYPESC...". The "ristorante.yml" file is open in the editor, showing an OpenAPI specification for a restaurant menu API. A red arrow points from the right margin of the code editor to the "url" field in the "servers" section, which contains the placeholder `\${OPENAPI\_SERVER\_URL}/api". The status bar at the bottom indicates "Ln 7, Col 10" and "Spaces: 2".

```
openapi: 3.0.0
info:
 title: Il Ristorante menu API
 version: 1.0.0
 description: API to retrieve dishes and place orders for Il Ristorante
servers:
 - url: ${OPENAPI_SERVER_URL}/api
 description: Il Ristorante API server
paths:
 /dishes:
 get:
 operationId: getDishes
 summary: Get all available dishes
 description: Retrieve a list of all dishes, optionally filtered by course
 parameters:
 - in: query
 name: course
 schema:
 type: string
 description: Filter dishes by course. Can be breakfast, lunch, dinner, or dessert
 - in: query
 name: name
 schema:
 type: string
 description: Find dishes by name.
```

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する  
ラボを終了す

## 指示 リソース

このトンネルを使用するには、OPENAPI\_SERVER\_URL変数を使用するようにAPI仕様を更新する必要があります。

## Visual Studio Code:

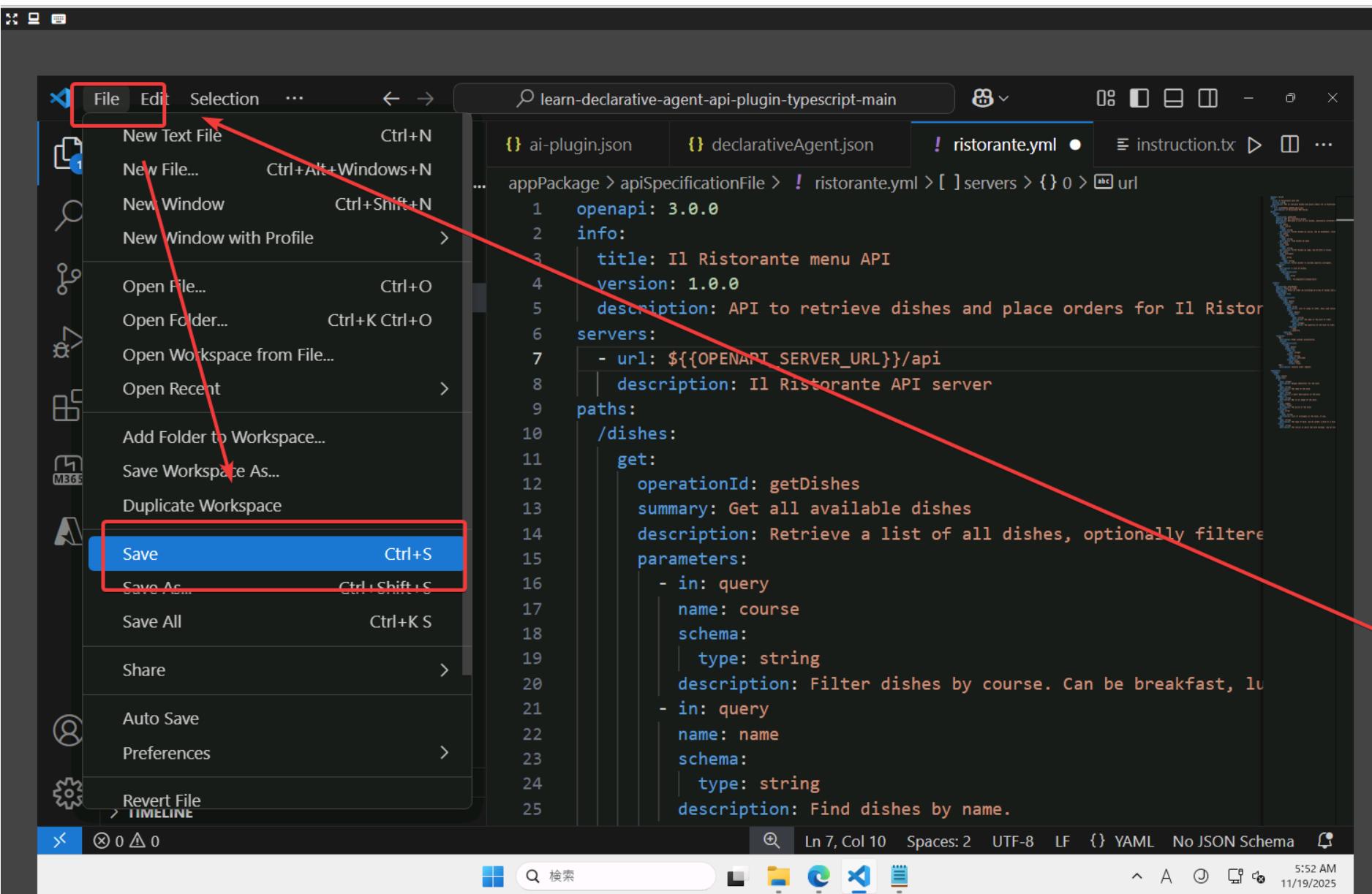
1. appPackage/apiSpecificationFile/ristorante.yml ファイルを開きます。
2. servers.url プロパティの値を `\${OPENAPI\_SERVER\_URL}/api` に変更します。
3. 変更されたファイルは次のようにになります。

```
openapi: 3.0.0
info:
 title: Il Ristorante menu API
 version: 1.0.0
 description: API to retrieve dishes and place orders for Il Ristorante
servers:
 - url: ${OPENAPI_SERVER_URL}/api
 description: Il Ristorante API server
paths:
 ...trimmed for brevity
```

4. 変更を保存。

API プラグインが完成し、宣言型エージェントと統合されました。Microsoft 365 Copilot でエージェントのテストを続行します。

## 演習 4 - Microsoft 365 Copilot で API プラグインを使用した宣言型エージェント



The screenshot shows the Visual Studio Code interface with the 'File' menu open. The 'Save' option is highlighted with a red box and a red arrow points from the top-left towards it. The main editor area displays a YAML file named 'ristorante.yml' containing API specification code. The status bar at the bottom indicates the file is saved.

```
openapi: 3.0.0
info:
 title: Il Ristorante menu API
 version: 1.0.0
 description: API to retrieve dishes and place orders for Il Ristorante
servers:
- url: ${OPENAPI_SERVER_URL}/api
 description: Il Ristorante API server
paths:
 /dishes:
 get:
 operationId: getDishes
 summary: Get all available dishes
 description: Retrieve a list of all dishes, optionally filtered by course or name.
 parameters:
 - in: query
 name: course
 schema:
 type: string
 description: Filter dishes by course. Can be breakfast, lunch, dinner, etc.
 - in: query
 name: name
 schema:
 type: string
 description: Find dishes by name.
```

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

このトンネルを使用するには、OPENAPI\_SERVER\_URL変数を使用するようにAPI仕様を更新する必要があります。

Visual Studio Code:

1. appPackage/apiSpecificationFile/ristorante.yml ファイルを開きます。
2. servers.url プロパティの値を \${OPENAPI\_SERVER\_URL}/api に変更します。
3. 変更されたファイルは次のようにになります。

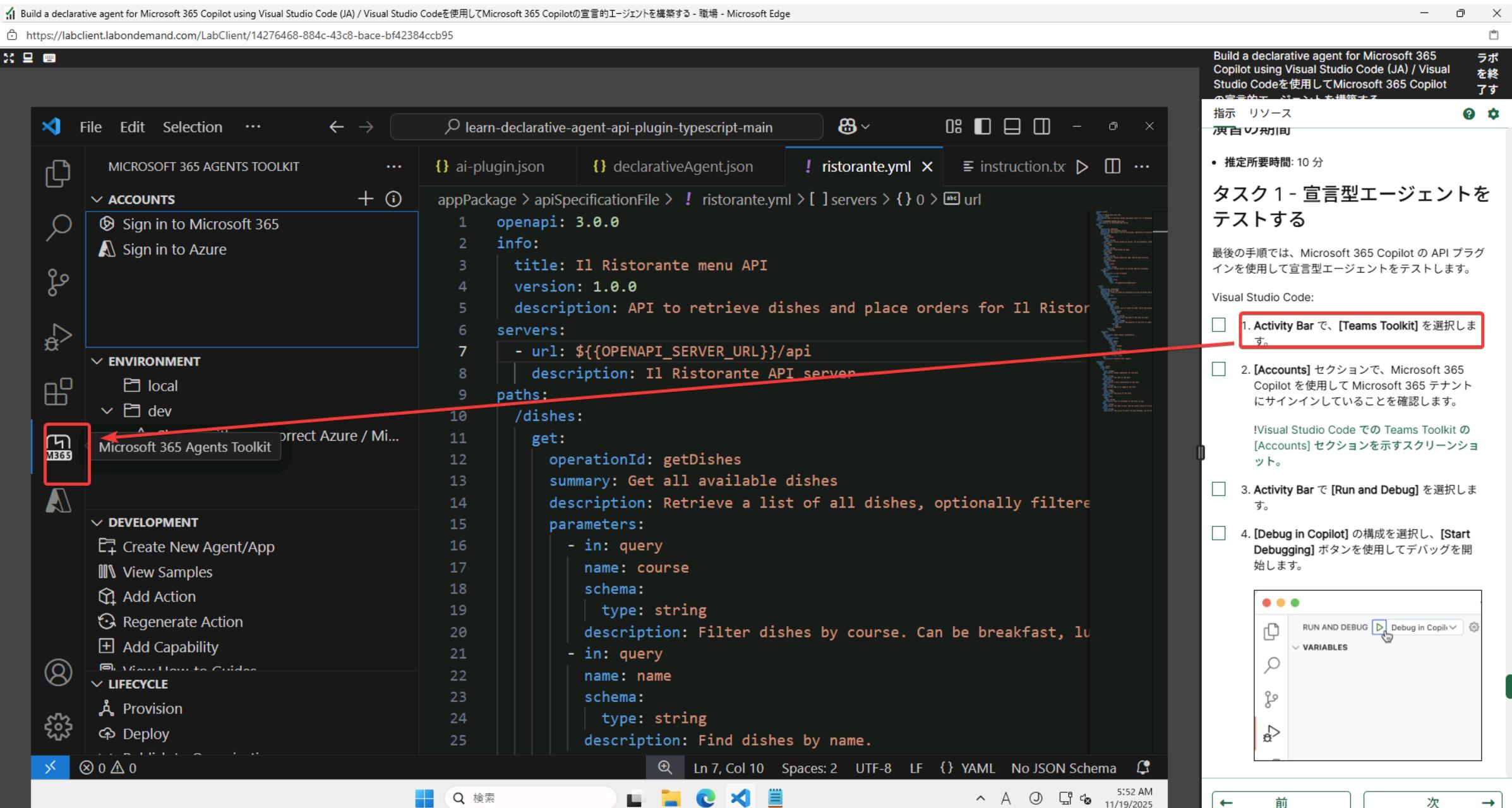
yaml

```
openapi: 3.0.0
info:
 title: Il Ristorante menu API
 version: 1.0.0
 description: API to retrieve dish...
servers:
- url: ${OPENAPI_SERVER_URL}/api
 description: Il Ristorante API server
paths:
 ...trimmed for brevity
```

4. 変更を保存。

API プラグインが完成し、宣言型エージェントと統合されました。Microsoft 365 Copilot でエージェントのテストを続行します。

## 演習 4 - Microsoft 365 Copilot で API プラグインを使用した宣言型エージェント



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

MICROSOFT 365 AGENTS TOOLKIT

ACCOUNTS

- Sign in to Microsoft 365 (highlighted with a red box)
- Sign in to Azure

ENVIRONMENT

- local
- dev
  - ⚠️ Sign in with your correct Azure / Mi...

DEVELOPMENT

- Create New Agent/App
- View Samples
- Add Action
- Regenerate Action
- Add Capability
- View How-to Guides

LIFECYCLE

- Provision
- Deploy

ai-plugin.json    declarativeAgent.json    ristorante.yml (highlighted with a red box)    instruction.txt

```
openapi: 3.0.0
info:
 title: Il Ristorante menu API
 version: 1.0.0
 description: API to retrieve dishes and place orders for Il Ristorante
servers:
 - url: ${OPENAPI_SERVER_URL}/api
 description: Il Ristorante API server
paths:
 /dishes:
 get:
 operationId: getDishes
 summary: Get all available dishes
 description: Retrieve a list of all dishes, optionally filtered by course
 parameters:
 - in: query
 name: course
 schema:
 type: string
 description: Filter dishes by course. Can be breakfast, lunch, dinner, etc.
 - in: query
 name: name
 schema:
 type: string
 description: Find dishes by name.
```

Ln 7, Col 10    Spaces: 2    UTF-8    LF    {}    YAML    No JSON Schema

RUN AND DEBUG    Debug in Copilot

5:52 AM    11/19/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース

タスク 1 - 宣言型エージェントをテストする

最後の手順では、Microsoft 365 Copilot の API プラグインを使用して宣言型エージェントをテストします。

Visual Studio Code:

- Activity Bar で [Teams Toolkit] を選択します。
- [Accounts] セクションで、Microsoft 365 Copilot を使用して Microsoft 365 テナントにサインインしていることを確認します。  
!Visual Studio Code での Teams Toolkit の [Accounts] セクションを示すスクリーンショット。
- Activity Bar で [Run and Debug] を選択します。
- [Debug in Copilot] の構成を選択し、[Start Debugging] ボタンを使用してデバッグを開始します。

前 次

残り 5時17分

File Edit Selection ... ← → learn-declarative-agent-api-plugin-typescript-main MICROSOFT 365 AGENTS TOOLKIT ... ai-plugin.json declarativeAgent.json ristorante.yml instruction.tx ...

**ACCOUNTS**

- + Sign in to Microsoft 365
- Sign in to Azure

**ENVIRONMENT**

- local
- dev
  - ⚠️ Sign in with your correct Azure /

**DEVELOPMENT**

- Create New Agent/App
- View Samples
- Add Action
- Regenerate Action
- Add Capability
- View How-to Guides

**LIFECYCLE**

- Provision
- Deploy

appPackage > apiSpecificationFile > ! ristorante.yml > [ ] servers > {} 0 > url

```
1 openapi: 3.0.0
2 info:
3 title: Il Ristorante menu API
4 version: 1.0.0
5 description: API to retrieve dishes and place orders for Il Ristorante
6 servers:
7 - url: ${OPENAPI SERVER URL}/api
```

Visual Studio Code

Microsoft 365 Agents Toolkit needs a Microsoft 365 account with custom app upload permission. If you're a Visual Studio subscriber, create a developer sandbox with the Microsoft 365 Developer Program.

Sign in Create a Microsoft 365 developer sandbox Cancel

summary: Get all available dishes
description: Retrieve a list of all dishes, optionally filtered by course.
parameters:
- in: query
 name: course
 schema:
 type: string
 description: Filter dishes by course. Can be breakfast, lunch, dinner, etc.
- in: query
 name: name
 schema:
 type: string
 description: Find dishes by name.

Ln 7, Col 10 Spaces: 2 UTF-8 LF {} YAML No JSON Schema

5:53 AM 11/19/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース 次へ

タスク 1 - 宣言型エージェントをテストする

最後の手順では、Microsoft 365 Copilot の API プラグインを使用して宣言型エージェントをテストします。

Visual Studio Code:

- Activity Bar で、[Teams Toolkit] を選択します。
- [Accounts] セクションで、Microsoft 365 Copilot を使用して Microsoft 365 テナントにサインインしていることを確認します。

!Visual Studio Code での Teams Toolkit の [Accounts] セクションを示すスクリーンショット。

- Activity Bar で [Run and Debug] を選択します。
- [Debug in Copilot] の構成を選択し、[Start Debugging] ボタンを使用してデバッグを開始します。

RUN AND DEBUG Debug in Copilot

前 次 残り 5時17分

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

GitHub - microsoft/learn-declarat アカウントにサインイン

https://login.microsoftonline.com/common/oauth2/v2.0/authorize?client\_id=7ea7c24c-b1f6-4a20-9d11-9ae12e9e7ac0&scope=https://dev.teams.microsoft.com/AppDefinitions.Read...

指示 リソース

WWL M365 Enterprise with Calling

Administrative Username admin@WWLx411051.onmicrosoft.com

Administrative Password n1NV7}%;h#0}22mmVpj;)Kx+4mZC3E

テナント名 WWLx411051.onmicrosoft.com

テナント プレフィックス WWLx411051

User Password 1RO3c~x#{(c(%OpgJNM3:6PR55t8^

AppName ScoringApp

ClientId 1815989d-659f-4c09-bd81-32256ec

ClientSecret M9J8Q~oJwF5zemSfKHd~1hoPufuhCD/

TenantId 7e638c3a-dae9-4716-9257-0d2ea6

MS-4010-CLIENT01

ユーザー名 Admin

パスワード Pa55w.rd

Ctrl+Alt+Delete

新しいウィンドウで開きます。

The screenshot shows a Microsoft Edge browser window. On the left, a Microsoft sign-in page is displayed with a red arrow pointing to the 'Email or phone number' input field. On the right, a 'Lab Client' interface is shown with various configuration details for a 'WWL M365 Enterprise with Calling' setup. A red box highlights the 'リソース' tab. Below it, a green box highlights the 'MS-4010-CLIENT01' section, which contains user credentials (Admin, Pa55w.rd) and a 'Ctrl+Alt+Delete' button. The status bar at the bottom indicates the time as 5:53 AM and the date as 11/19/2025.

GitHub - microsoft/learn-declarat × アカウントにサインイン × +

https://login.microsoftonline.com/common/oauth2/v2.0/authorize?client\_id=7ea7c24c-b1f6-4a20-9d11-9ae12e9e7ac0&scope=https://dev.teams.microsoft.com/AppDefinitions.Read...

Microsoft サインイン

admin@WWLx411051.onmicrosoft.com

アカウントにアクセスできない場合

戻る 次へ

サインイン オプション

利用規約 プライバシーとCookie ...

5:53 AM 11/19/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

ラボ  
を終  
了す

指示 リソース

## WWL M365 Enterprise with Calling

Administrative Username	admin@WWLx411051.onmicrosoft.co
Administrative Password	n1NV7}%;h#0}22mmVpj;)Kx+4mZC3E
テナント名	WWLx411051.onmicrosoft.com
テナント プレフィックス	WWLx411051
User Password	1R03c~x#{(c(%OhpqJNM3:6PR55t8^
AppName	ScoringApp
ClientId	1815989d-659f-4c09-bd81-32256ec
ClientSecret	M9J8Q~oJwF5zemSfKHd~1hoPufuhCD/
TenantId	7e638c3a-dae9-4716-9257-0d2ea6

MS-4010-CLIENT01

ユーザー名 Admin

パスワード Pa55w.rd

Ctrl+Alt+Delete

新しいウィンドウで開きます。

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

GitHub - microsoft/learn-declarat アカウントにサインイン

https://login.microsoftonline.com/common/oauth2/v2.0/authorize?client\_id=7ea7c24c-b1f6-4a20-9d11-9ae12e9e7ac0&scope=https://dev.teams.microsoft.com/AppDefinitions...  
- リソース 指示

WWL M365 Enterprise with Calling

Administrative Username: admin@WWLx411051.onmicrosoft.com  
Administrative Password: n1NV7}%;h#0}22mmVpj:)Kx+4mZC3E  
テナント名: WWLx411051.onmicrosoft.com  
テナント プレフィックス: WWLx411051  
User Password: 1R03c~x#{(c(%OhpqJNM3:6PR55t8^  
AppName: ScoringApp  
ClientId: 1815989d-659f-4c09-bd81-32256ec  
ClientSecret: M9J8Q~oJwF5zemSfKHd~1hoPufuhCD/  
TenantId: 7e638c3a-dae9-4716-9257-0d2ea61

MS-4010-CLIENT01

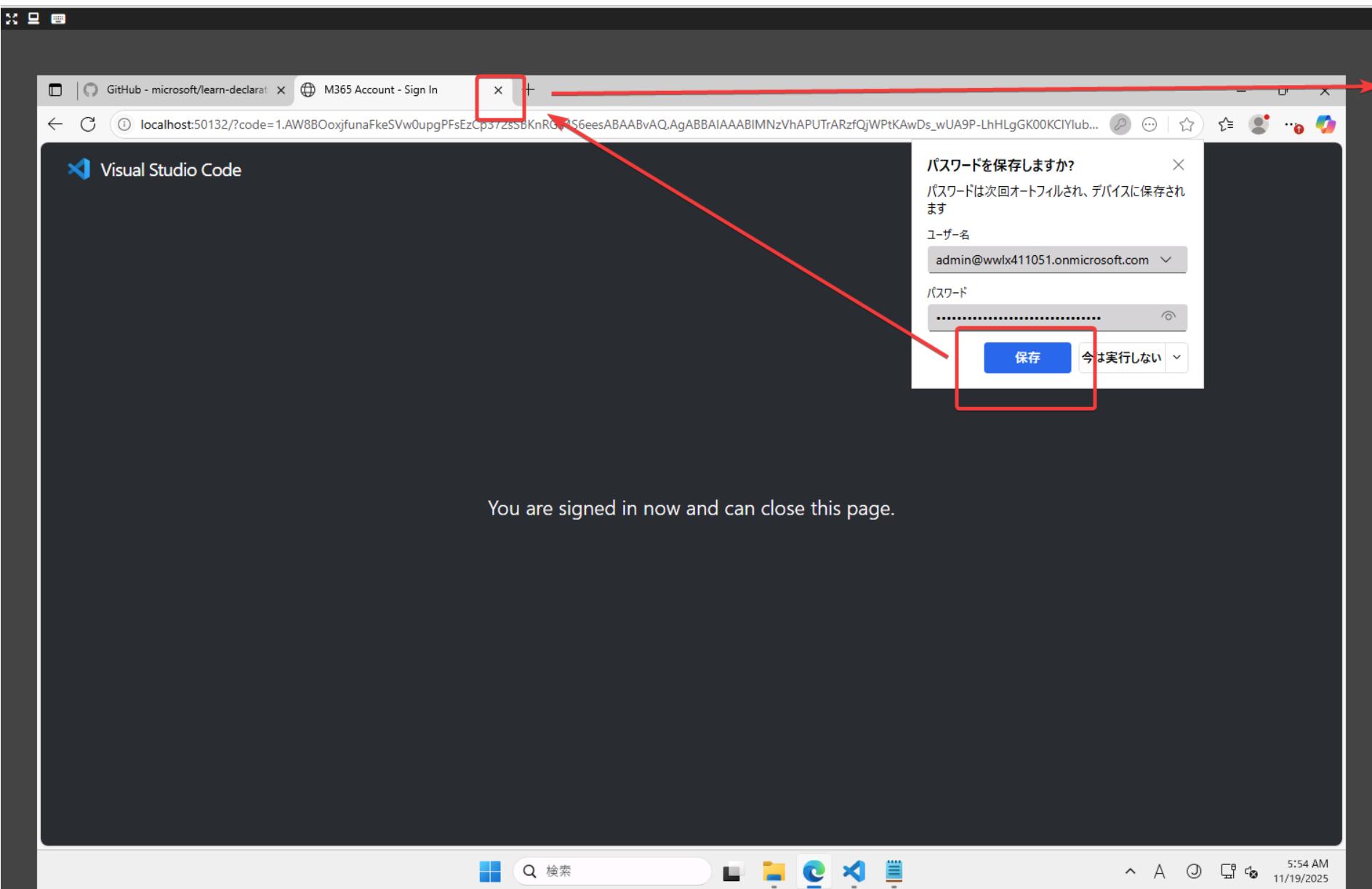
ユーザー名: Admin  
パスワード: Pa55w.rd  
Ctrl+Alt+Delete  
新しいウィンドウで開きます。

Microsoft  
← admin@wwlx411051.onmicrosoft.com  
パスワードの入力  
\*\*\*\*\*  
サインイン  
パスワードを忘れた場合

利用規約 プライバシーと Cookie ...

検索

5:53 AM 11/19/2025



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge  
ラボを終了す

指示 リソース

## WWL M365 Enterprise with Calling

Administrative Username	admin@WWLx411051.onmicrosoft.co
Administrative Password	n1NV7%:h#0}22mmVpj:)Kx+4mZC3E
テナント名	WWLx411051.onmicrosoft.com
テナント プレフィックス	WWLx411051
User Password	1R03c~x#{(c(%OhpqJNM3:6PR55t8^
AppName	ScoringApp
ClientId	1815989d-659f-4c09-bd81-32256ec
ClientSecret	M9J8Q~oJwF5zemSfKHd~1hoPufuhCD/
TenantId	7e638c3a-dae9-4716-9257-0d2ea6

MS-4010-CLIENT01
ユーザー名 Admin
パスワード Pa55w.rd
Ctrl+Alt+Delete
新しいウィンドウで開きます。

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

MICROSOFT 365 AGENTS TOOLKIT

ACCOUNTS

admin@WWLx411051.onmicrosoft.co... Sign in to Azure

ENVIRONMENT

local

dev Sign in with your correct Azure acco...

DEVELOPMENT

Create New Agent/App

View Samples

Add Action

Regenerate Action

Add Capability

LIFECYCLE

Provision

Deploy

learn-declarative-agent-api-plugin-typescript-main

File Edit Selection ... ← → ai-plugin.json declarativeAgent.json ristorante.yml instruction.tx ...

appPackage > apiSpecificationFile > ! ristorante.yml > [ ] servers > {} 0 > url

```
1 openapi: 3.0.0
2 info:
3 title: Il Ristorante menu API
4 version: 1.0.0
5 description: API to retrieve dishes and place orders for Il Ristorante
6
7 servers:
8 - url: ${OPENAPI_SERVER_URL}/api
9 description: Il Ristorante API server
10
11 paths:
12 /dishes:
13 get:
14 operationId: getDishes
15 summary: Get all available dishes
16 description: Retrieve a list of all dishes, optionally filtered by course
17 parameters:
18 - in: query
19 name: course
20 schema:
21 type: string
22 description: Filter dishes by course. Can be breakfast, lunch, dinner, or dessert
23 - in: query
24 name: name
25 schema:
26 type: string
27 description: Filter dishes by name.
```

Spaces: 2 UTF-8 LF {} YAML No JSON Schema

RUN AND DEBUG  VARIABLES

5:54 AM 11/19/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

ラボを終了す

指示 リソース 次

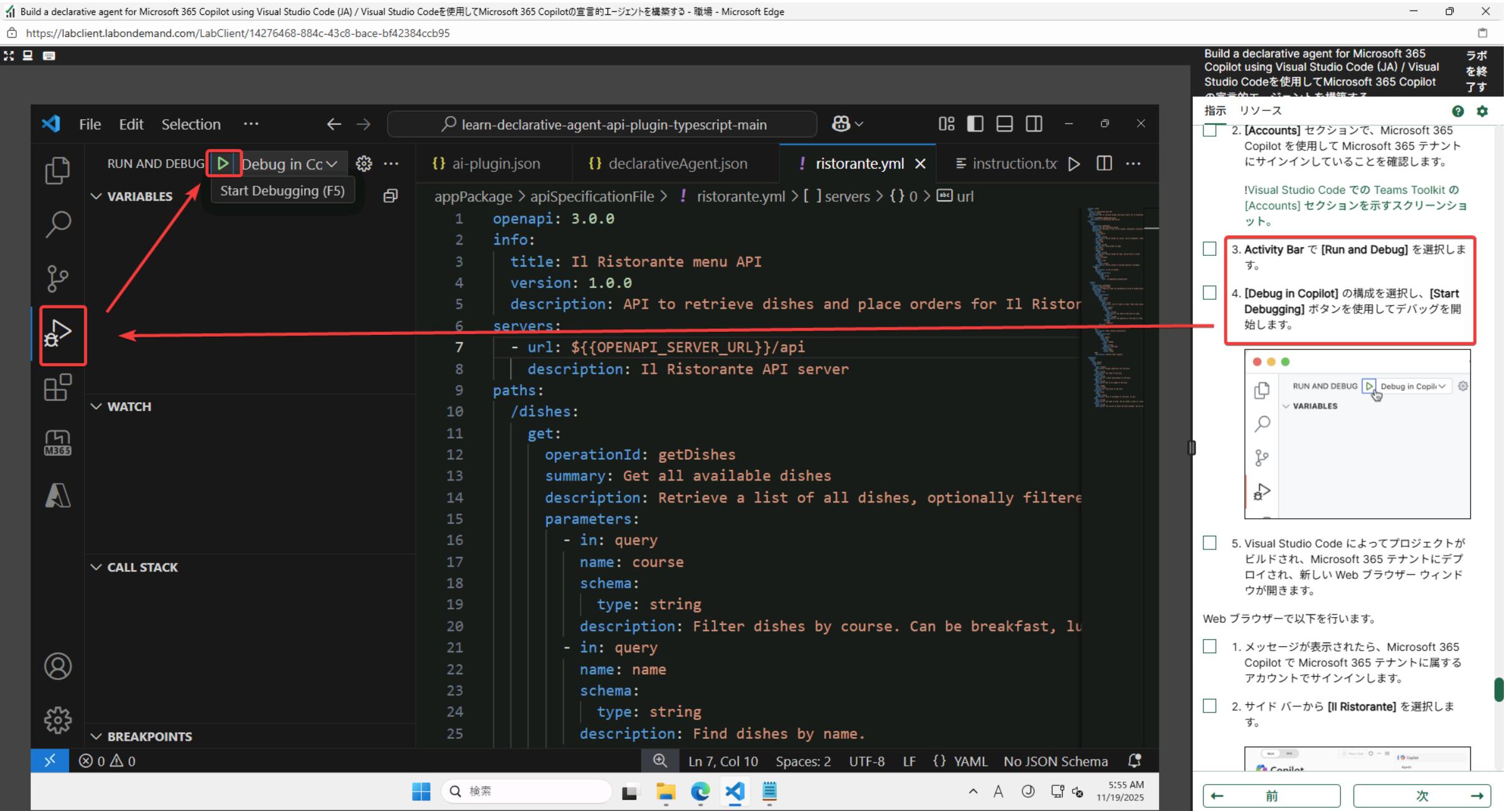
タスク 1 - 宣言型エージェントをテストする

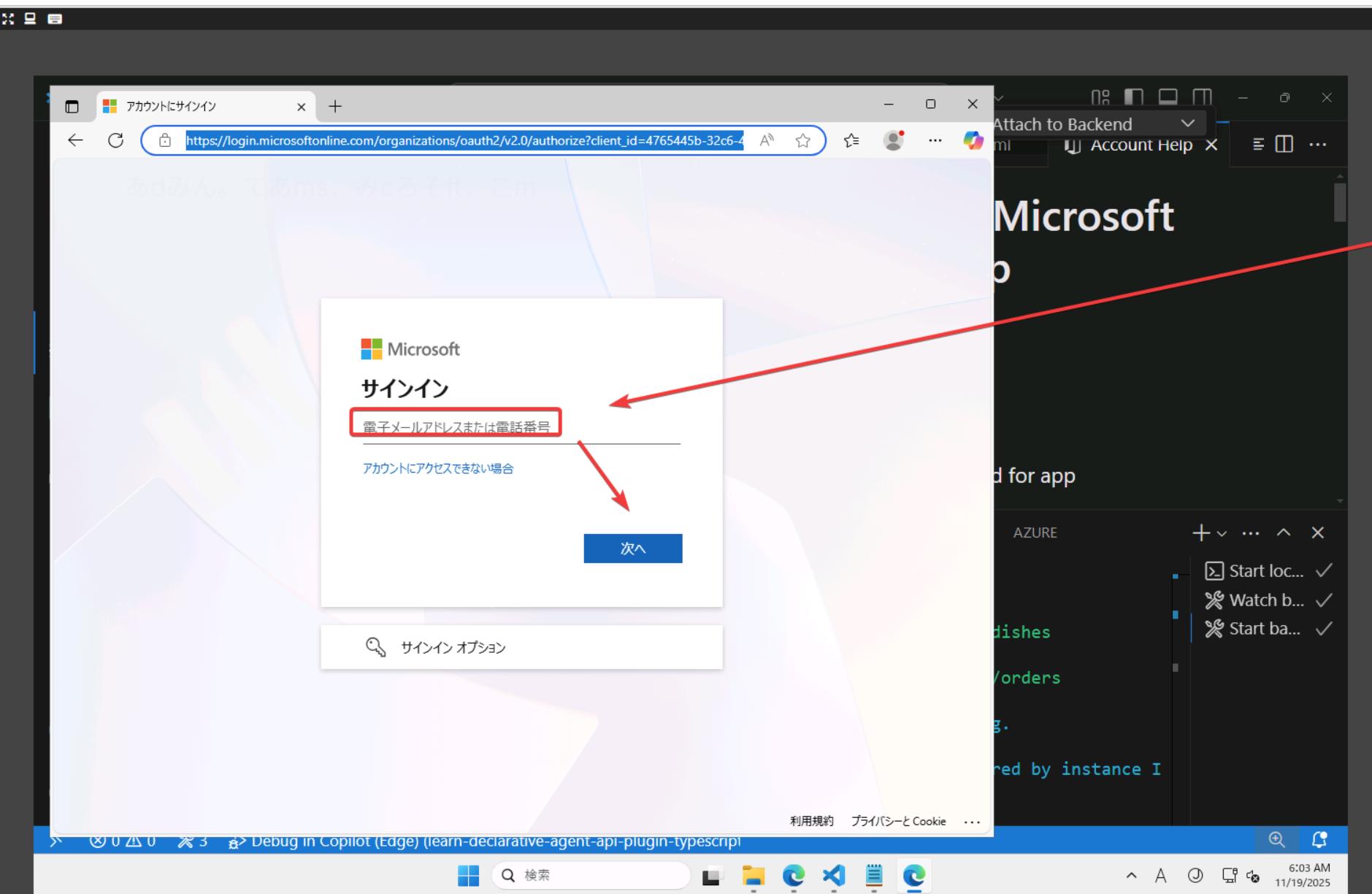
最後の手順では、Microsoft 365 Copilot の API プラグインを使用して宣言型エージェントをテストします。

Visual Studio Code:

- Activity Bar で [Teams Toolkit] を選択します。
- [Accounts] セクションで、Microsoft 365 Copilot を使用して Microsoft 365 テナントにサインインしていることを確認します。  
!Visual Studio Code での Teams Toolkit の [Accounts] セクションを示すスクリーンショット。
- Activity Bar で [Run and Debug] を選択します。
- [Debug in Copilot] の構成を選択し、[Start Debugging] ボタンを使用してデバッグを開始します。

前 残り 5時15分





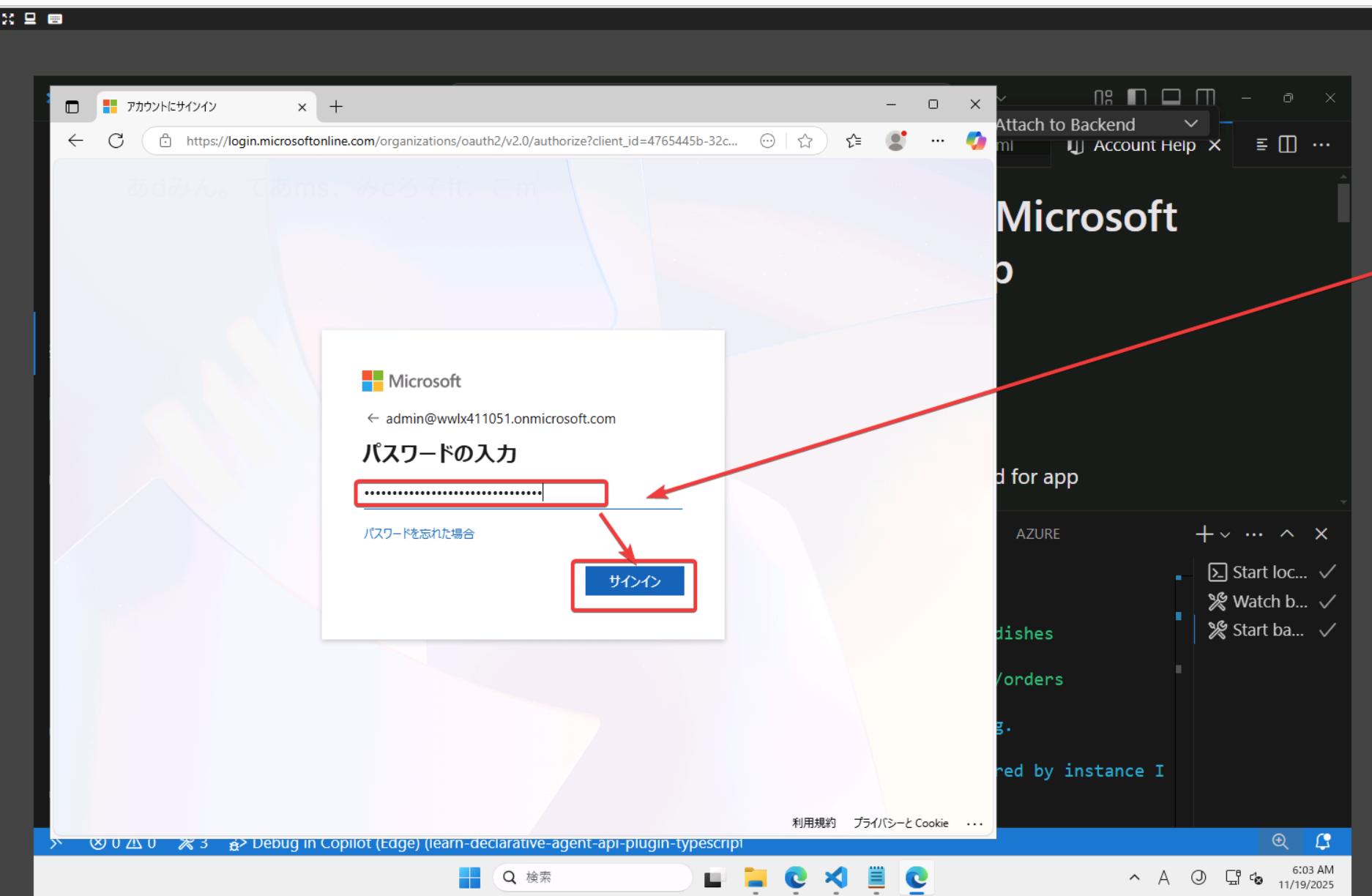
Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する  
ラボを終了す

指示 リソース

## WWL M365 Enterprise with Calling

Administrative Username	admin@WWLx411051.onmicrosoft.com
Administrative Password	n1NV7%:h#0}22mmVpj;)Kx+4mZC3E
テナント名	WWLx411051.onmicrosoft.com
テナント プレフィックス	WWLx411051
User Password	1R03c~x#{(c(%OhpqJNM3:6PR55t8^
AppName	ScoringApp
ClientId	1815989d-659f-4c09-bd81-32256ec
ClientSecret	M9J8Q~oJwF5zemSfKHd~1hoPufuhCD/
TenantId	7e638c3a-dae9-4716-9257-0d2ea6

	MS-4010-CLIENT01
ユーザー名	Admin
パスワード	Pa55w.rd
Ctrl+Alt+Delete	
■ 新しいウィンドウで開きます。	



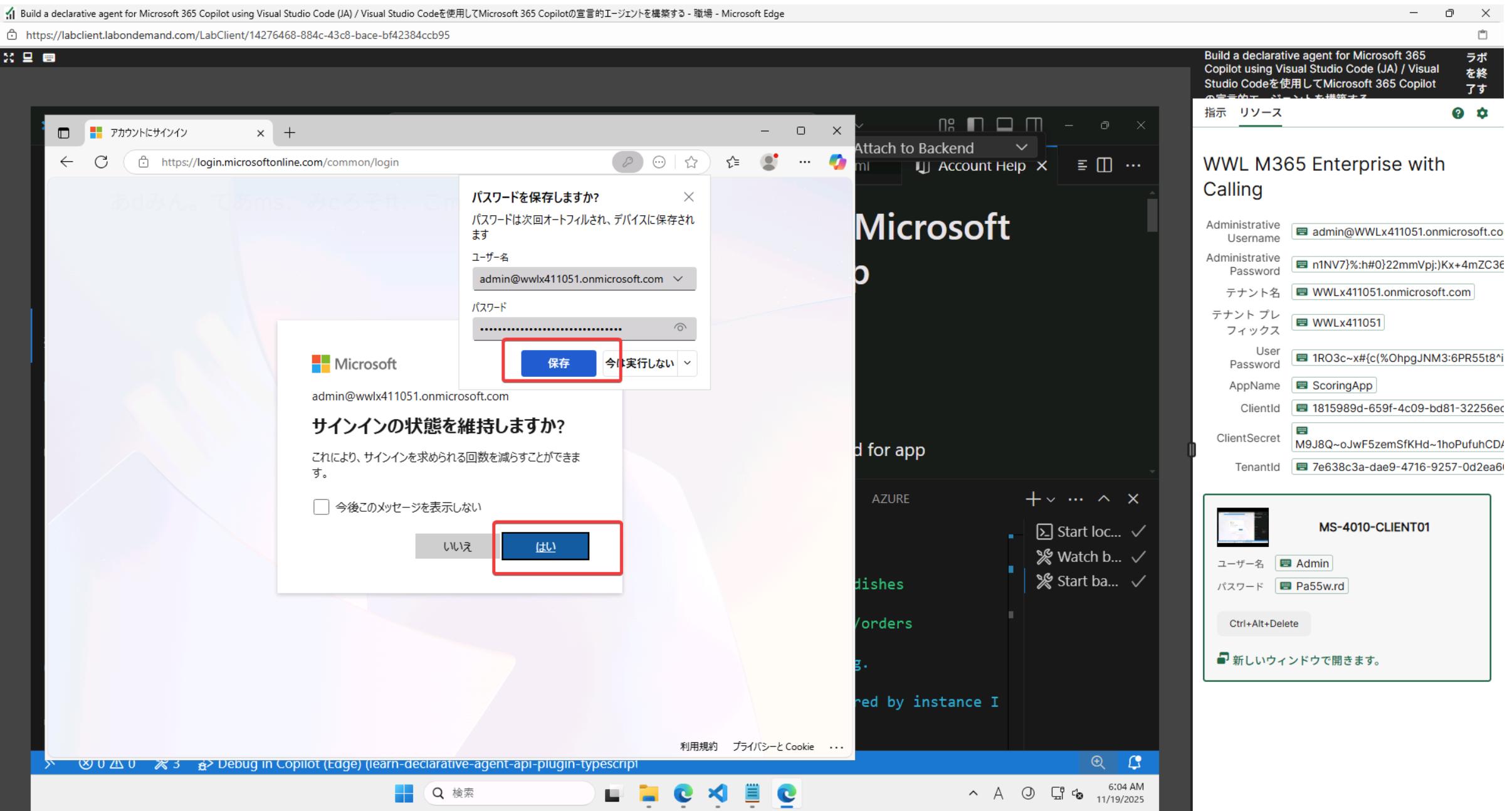
Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する  
ラボを終了す

指示 リソース

## WWL M365 Enterprise with Calling

Administrative Username	admin@WWLx411051.onmicrosoft.com
Administrative Password	n1NV7%:h#0}22mmVpj:)Kx+4mzC36
テナント名	WWLx411051.onmicrosoft.com
テナント プレフィックス	WWLx411051
User Password	1R03c~x#{(c(%OhpqJNM3:6PR55t8^
AppName	ScoringApp
ClientId	1815989d-659f-4c09-bd81-32256ec
ClientSecret	M9J8Q~oJwF5zemSfKHd~1hoPufuhCD/
TenantId	7e638c3a-dae9-4716-9257-0d2ea6

MS-4010-CLIENT01  
ユーザー名 Admin  
パスワード Pa55w.rd  
Ctrl+Alt+Delete  
新しいウィンドウで開きます。



Chat | M365 Copilot

https://m365.cloud.microsoft/chat/?auth=2&origindomain=Office&client-request-id=b5c...

Attach to Backend Account Help

Microsoft

Welcome to your Microsoft 365 Copilot app

Find files and more

Get to the right file and info about people and more in a flash.

Create

Harness the power of AI to generate stunning visuals, draft documents, and present your story.

Use your familiar apps

With Copilot built into your apps, you can write, analyze, and design-- better and faster than ever.

Get started

DEBUG

Debug in Copilot (Edge) (learn-declarative-agent-api-plugin-typescript)

検索

6:04 AM 11/19/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース

5. Visual Studio Code によってプロジェクトがビルドされ、Microsoft 365 テナントにデプロイされ、新しい Web ブラウザー ウィンドウが開きます。

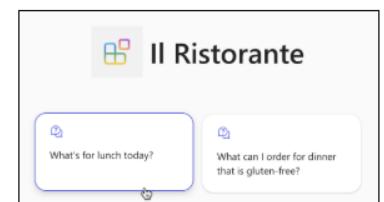
Web ブラウザーで以下を行います。

1. メッセージが表示されたら、Microsoft 365 Copilot で Microsoft 365 テナントに属するアカウントでサインインします。

2. サイドバーから [Il Ristorante] を選択します。



3. What's for lunch today? の会話スターを選択し、プロンプトを送信します。



4. メッセージが表示されたら、エージェントが API に送信するデータを調べて、[Allow once] ボタンを使用して確認します。



前

次

残り5時5分

m365.cloud.microsoft/chat/?auth=x

https://m365.cloud.microsoft/chat/?auth=2&origindomain=Office&client-request-id=b5c...

M365 Copilot

Search

Create

Apps

Chat

Agents

Il Ristorante

All agents

Create agent

Conversations

Pages

Admin

Message Copilot

Help me write

Find the right questions

Azure

Start loc... ✓

Watch b... ✓

Start ba... ✓

dishes

/orders

ered by instance I

MOD Administrator

Debug in Copilot (Edge) (learn-declarative-agent-api-plugin-typescript)

検索

6:04 AM 11/19/2025

Il Ristorante

What's for lunch today?

What can I order for dinner that is gluten-free?

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する

ラボ  
を終  
了す

指示 リソース

5. Visual Studio Code によってプロジェクトがビルドされ、Microsoft 365 テナントにデプロイされ、新しい Web ブラウザー ウィンドウが開きます。

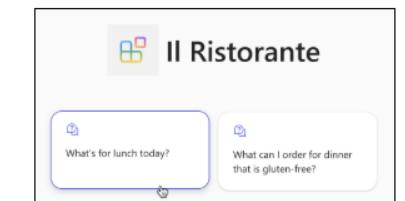
Web ブラウザーで以下を行います。

1. メッセージが表示されたら、Microsoft 365 Copilot で Microsoft 365 テナントに属するアカウントでサインインします。

2. サイドバーから [Il Ristorante] を選択します。



3. What's for lunch today? の会話スターを選択し、プロンプトを送信します。



4. メッセージが表示されたら、エージェントが API に送信するデータを調べて、[Allow once] ボタンを使用して確認します。



前 次

残り5時5分

Chat | M365 Copilot

https://m365.cloud.microsoft/chat/?auth=2&origindomain=Office&client-request-id=b5c...

M365 Copilot

- Search
- Create
- Apps
- Chat
- Agents
- Il Ristorante**
- All agents
- Create agent
- Conversations
- Pages
- Admin

Message Copilot

What's for lunch today?

What can I order for dinner that is gluten-free?

Il Ristorante

Created by Teams App, Inc.

Attach to Backend

Account Help

Microsoft

AZURE

Start loc... ✓

Watch b... ✓

Start ba... ✓

What's for lunch today?

What can I order for dinner that is gluten-free?

Debug in Copilot (Edge) (learn-declarative-agent-api-plugin-typescript)

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース

5. Visual Studio Code によってプロジェクトがビルドされ、Microsoft 365 テナントにデプロイされ、新しい Web ブラウザー ウィンドウが開きます。

Web ブラウザーで以下を行います。

- メッセージが表示されたら、Microsoft 365 Copilot で Microsoft 365 テナントに属するアカウントでサインインします。
- サイドバーから [Il Ristorante] を選択します。

3. What's for lunch today? の会話スターを選択し、プロンプトを送信します。

4. メッセージが表示されたら、エージェントが API に送信するデータを調べて、[Allow once] ボタンを使用して確認します。

Chat | M365 Copilot

https://m365.cloud.microsoft/chat/?auth=2&origindomain=Office&client-request-id=b5c...

M365 Copilot

- Search
- Create
- Apps
- Chat
- Agents
- Il Ristorante**
- All agents
- Create agent
- Conversations
- Pages
- Admin

Il Ristorante

Created by Teams App, Inc.

What's for lunch today?

→

What's for lunch today?  
What can I order for dinner that is gluten-free?

AZURE

dishes  
orders

red by instance I

MOD Administrator

Debug in Copilot (Edge) (learn-declarative-agent-api-plugin-typescript)

検索

6:05 AM 11/19/2025

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース

5. Visual Studio Code によってプロジェクトがビルドされ、Microsoft 365 テナントにデプロイされ、新しい Web ブラウザー ウィンドウが開きます。

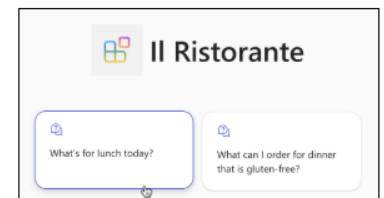
Web ブラウザーで以下を行います。

1. メッセージが表示されたら、Microsoft 365 Copilot で Microsoft 365 テナントに属するアカウントでサインインします。

2. サイドバーから [Il Ristorante] を選択します。



3. What's for lunch today? の会話スターを選択し、プロンプトを送信します。

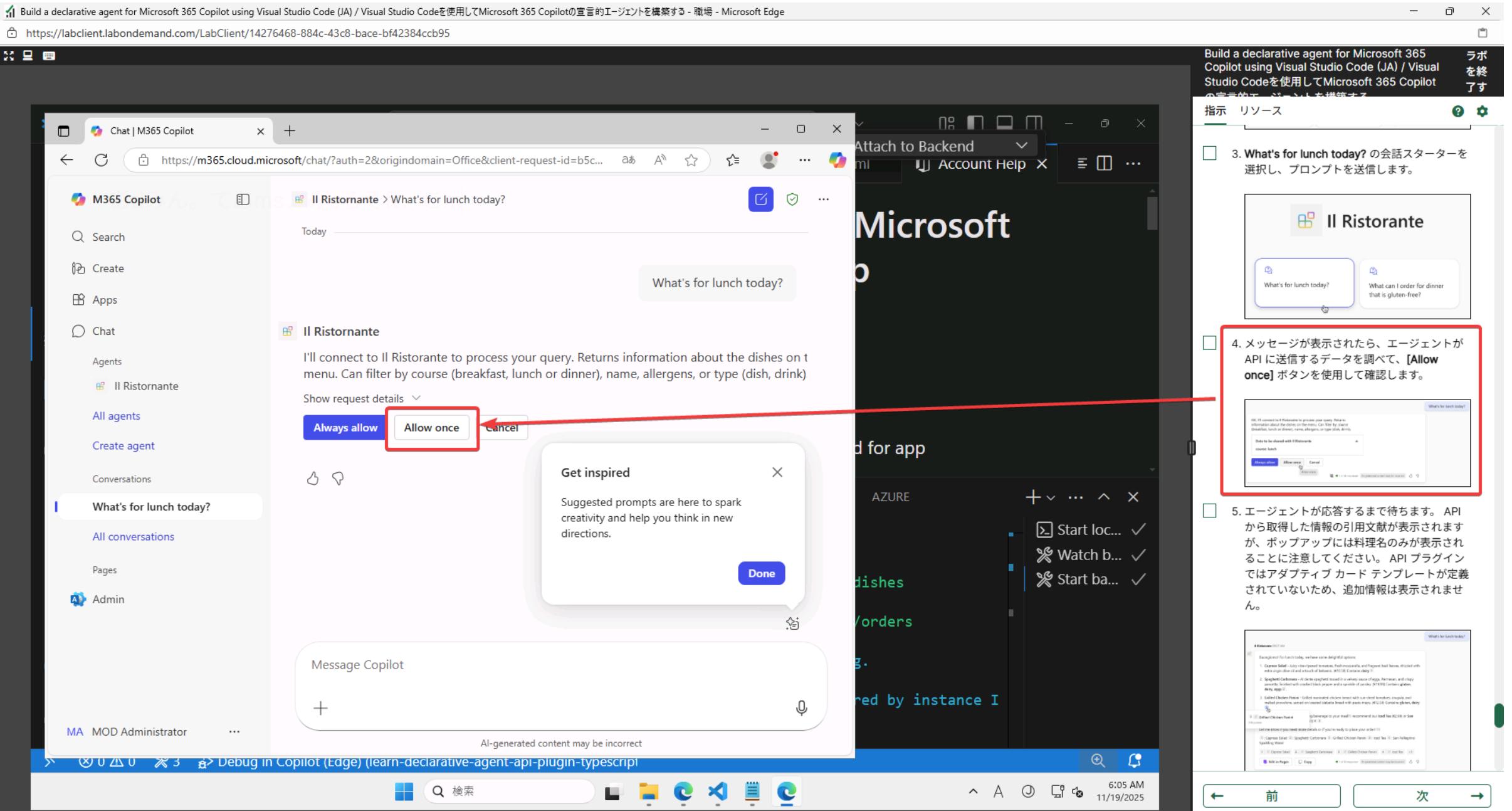


4. メッセージが表示されたら、エージェントが API に送信するデータを調べて、[Allow once] ボタンを使用して確認します。



前 次

残り5時4分



Chat | M365 Copilot

Il Ristorante > What's for lunch today?

Today

What's for lunch today?

Il Ristorante

Here are some delicious lunch options for today:

Dishes

1. Caprese Salad – Juicy vine-ripened tomatoes, fresh mozzarella, and fragrant basil leaves, drizzled with extra virgin olive oil and a touch of balsamic.
2. Spaghetti Carbonara – Al dente spaghetti tossed in a velvety sauce of eggs, Parmesan, and crispy pancetta, finished with cracked black pepper and a sprinkle of parsley.

Price: €10.50  
Allergens: Dairy

Price: €14.99  
Allergens: Gluten, Dairy, Eggs

Message Copilot

AI-generated content may be incorrect

MOD Administrator

Debug in Copilot (Edge) (learn-declarative-agent-api-plugin-typescript)

Attach to Backend

Account Help

Microsoft

for app

AZURE

Start loc... ✓  
Watch b... ✓  
Start ba... ✓

Message Copilot

dishes' (Reason='The host APIs.', Id=dishes' (Succeeded=419ms))

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

ラボを終了す

指示 リソース

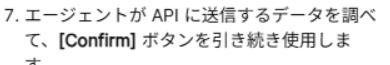
[once] ボタンを使用して確認します。



5. エージェントが応答するまで待ちます。APIから取得した情報の引用文献が表示されますが、ポップアップには料理名のみが表示されることに注意してください。API プラグインではアダプティブ カード テンプレートが定義されていないため、追加情報は表示されません。



6. プロンプト テキスト ボックスに「1x spaghetti, 1x iced tea」と入力してプロンプトを送信し、注文します。



7. エージェントがAPIに送信するデータを調べて、[Confirm] ボタンを引き続き使用します。



前 次

残り5時4分

Microsoft

Attach to Backend Account Help

Chat | M365 Copilot

Il Ristorante > What's for lunch today?

Today

What's for lunch today?

Il Ristorante

Here are some delicious lunch options for today:

Dishes

1. Caprese Salad – Juicy vine-ripened tomatoes, fresh mozzarella, and fragrant basil leaves, drizzled with extra virgin olive oil and a touch of balsamic.  
Price: €10.50  
Allergens: Dairy [Caprese Salad](#)
2. Spaghetti Carbonara – Al dente spaghetti tossed in a velvety sauce of eggs, Parmesan, and crispy pancetta, finished with cracked black pepper and a sprinkle of parsley.  
Price: €14.99

Order the Spaghetti Carbonara Add Iced Tea to my order

1x spaghetti, 1x iced tea

AI-generated content may be incorrect

➡

指示 リソース

- Spaghetti Carbonara – Al dente spaghetti tossed in a velvety sauce of eggs, Parmesan, and crispy pancetta, finished with cracked black pepper and a sprinkle of parsley.
- Caprese Salad – Grilled heirloom tomatoes tossed with sun-dried tomatoes, basil, mozzarella, and fresh basil, served on toasted ciabatta bread with extra virgin olive oil.
- Grilled Chicken Caesar – Grilled chicken Caesar salad with romaine lettuce, parmesan, and Caesar dressing, served on toasted ciabatta bread with extra virgin olive oil.

6. プロンプト テキスト ボックスに「1x spaghetti, 1x iced tea」と入力してプロンプトを送信し、注文します。

7. エージェントが API に送信するデータを調べて、[Confirm] ボタンを引き続き使用します。

8. エージェントが注文し、注文の概要を返すのを待ちます。もう一度、エージェントにはアダプティブ カード テンプレートがないため、注文の概要がブレーン テキストで表示されていることに注意してください。

OK, I'll connect to Il Ristorante to place your query. Please wait and return to the order details.

Start loc... ✓  
Watch b... ✓  
Start ba... ✓

9. Visual Studio Code に戻り、デバッグを停止します。

10. [ターミナル] タブに切り替えて、アクティブなすべてのターミナルを閉じます。

!すべてのターミナルを閉じるオプションがある

前 次

6:06 AM 11/19/2025

残り5時3分

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

https://labclient.labondemand.com/LabClient/14276468-884c-43c8-bace-bf42384ccb95

Chat | M365 Copilot

Il Ristorante > What's for lunch today?

would you like me to help you choose one or these dishes, or add a refreshing drink to go with your meal?

Sources >

1x spaghetti, 1x iced tea

Il Ristorante

I'll connect to Il Ristorante to process your query. Places an order and returns the order details.

dishes [{"name": "Spaghetti Carbonara", "quantity": 1}, {"name": "Iced Tea", "quantity": 1}]

Confirm Cancel

Message Copilot

AI-generated content may be incorrect

Attach to Backend Account Help

Microsoft

for app

AZURE

Start loc... ✓

Watch b... ✓

Start ba... ✓

red by instance I

dishes' (Reason='

e host APIs.', Id

dishes' (Succeeded

tion=419ms)

Debug in Copilot (Edge) (learn-declarative-agent-api-plugin-typescript)

検索

6:07 AM 11/19/2025

指示 リソース

2. Spaghetti Carbonara - AI 従来 spaghetti located in a variety sauce of eggs, Parmesan, and spaghetti. Carbonara is a traditional Italian dish.

3. Grilled Chicken Sandwich - Grilled seasoned chicken breast with sun dried tomatoes, basil pesto, and melted provolone, served on toasted ciabatta bread with fresh marinara.

4. Caprese Salad - Fresh mozzarella, ripe tomato, basil leaves, and extra virgin olive oil.

5. Caesar Salad - Grilled salmon fillet with sun dried tomatoes, basil pesto, and melted provolone, served on toasted ciabatta bread with fresh marinara.

6. 1x spaghetti, 1x iced tea

7. エージェントが API に送信するデータを調べて、[Confirm] ボタンを引き続き使用します。

8. エージェントが注文し、注文の概要を返すのを待ちます。もう一度、エージェントにはアダプティブカードテンプレートがないため、注文の概要がブレーンテキストで表示されていることに注意してください。

9. Visual Studio Code に戻り、デバッグを停止します。

10. [ターミナル] タブに切り替えて、アクティブなすべてのターミナルを閉じます。

!すべてのターミナルを閉じるオプションがある

前 次

残り5時2分

The screenshot shows a Microsoft Edge browser window with two tabs open. The left tab displays a conversation with 'Il Ristorante' about lunch orders. The right tab shows a Visual Studio Code terminal with Azure log output.

**M365 Copilot Chat (Left Tab):**

- Message: "What's for lunch today?"
- Response: "Il Ristorante" (highlighted with a red box)
  - Got it! You've chosen:
  - 1x Spaghetti Carbonara
  - 1x Iced Tea
- Message: "Let me place that order for you."
- Message: "Your order is confirmed! 🍽"
- Section: "Order Details:"
  - 1 x Spaghetti Carbonara
  - 1 x Iced Tea
- Text: "Total: €17.98"
- Text: "Status: Confirmed ✓"

**Visual Studio Code Terminal (Right Tab):**

```
dishes' (Reason='The host APIs.', Id=1)
dishes' (Succeeded, Duration=419ms)
orders' (Reason='The host APIs.', Id=1)
orders' (Succeeded, Duration=59ms)
```

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge  
ラボを終了す

指示 リソース



8. エージェントが注文し、注文の概要を返すのを待ちます。もう一度、エージェントにはアダプティブカードテンプレートがないため、注文の概要がプレーンテキストで表示されていることに注意してください。



9. Visual Studio Code に戻り、デバッグを停止します。

10. [ターミナル] タブに切り替えて、アクティブなすべてのターミナルを閉じます。

!すべてのターミナルを閉じるオプションがあるVisual Studio Codeの[ターミナル]タブを示すスクリーンショット。

あなたは、地元のレストランから食べ物を注文できる組織のソリューションを構築することを検討しています。このソリューションでは、自然言語を使用し、Microsoft 365 Copilot と統合する必要があります。

いくつかの調査を行い、API プラグインと統合された Microsoft 365 Copilot 用の宣言型エージェントを構築することが、次の理由からニーズに適していることがわかりました。

- これにより、自然言語を理解する AI を利用したアシスタントを構築できます。
- Microsoft 365 Copilot のインフラストラクチャの

Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge

Attach to Backend Stop Shift+F5

8. エージェントが注文し、注文の概要を返すのを待ちます。もう一度、エージェントにはアダプティブカードテンプレートがないため、注文の概要がプレーンテキストで表示されていることに注意してください。

9. Visual Studio Code に戻り、デバッグを停止します。

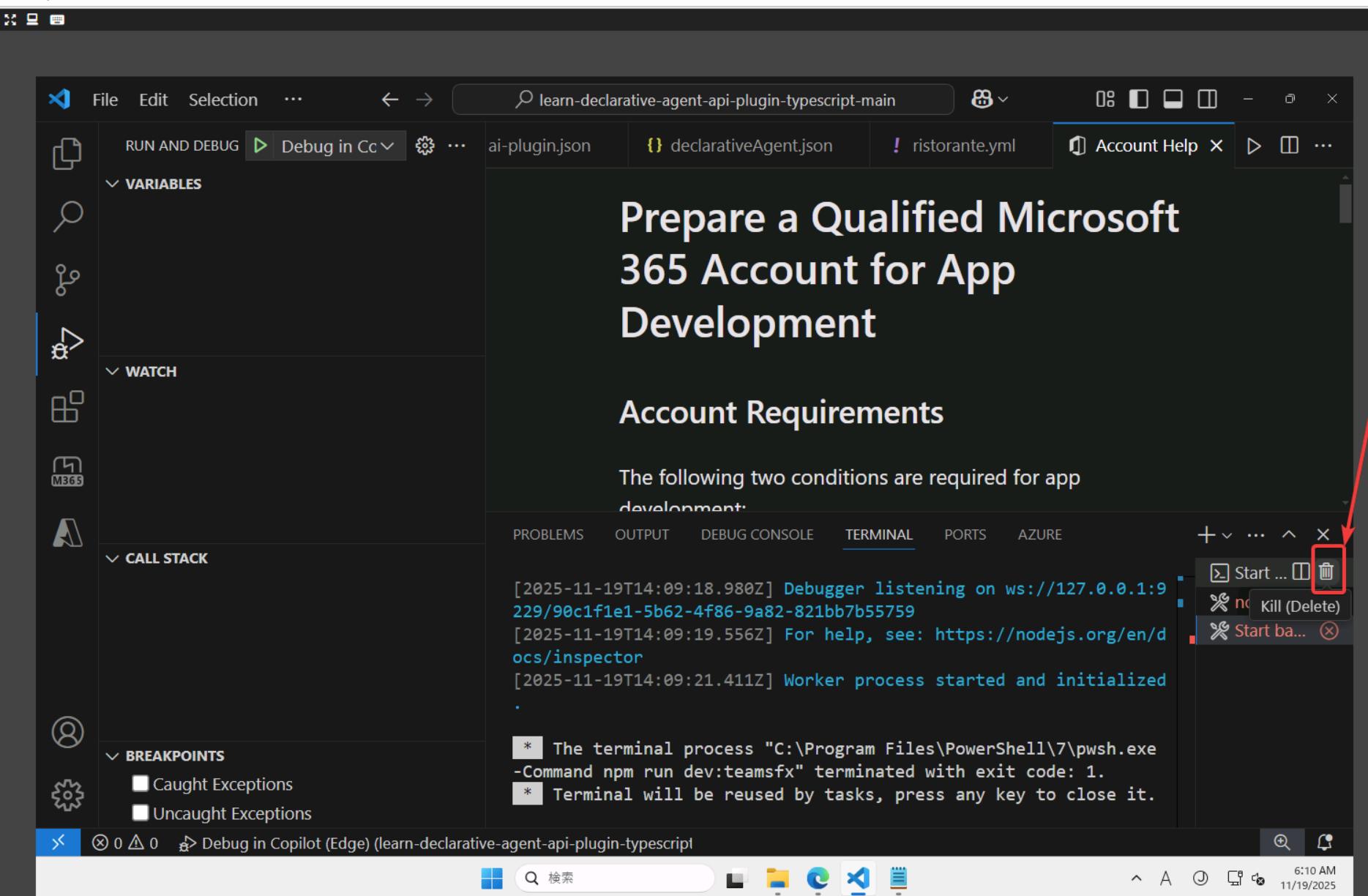
10. [ターミナル] タブに切り替えて、アクティブなすべてのターミナルを閉じます。

すべてのターミナルを閉じるオプションがあるVisual Studio Codeの[ターミナル]タブを示すスクリーンショット。

あなたは、地元のレストランから食べ物を注文できる組織のソリューションを構築することを検討しています。このソリューションでは、自然言語を使用し、Microsoft 365 Copilot と統合する必要があります。

いくつかの調査を行い、API プラグインと統合された Microsoft 365 Copilot 用の宣言型エージェントを構築することが、次の理由からニーズに適していることがわかりました。

- これにより、自然言語を理解する AI を利用したアシスタントを構築できます。
- Microsoft 365 Copilot のインフラストラクチャの



Build a declarative agent for Microsoft 365 Copilot using Visual Studio Code (JA) / Visual Studio Codeを使用してMicrosoft 365 Copilotの宣言的エージェントを構築する - 職場 - Microsoft Edge  
ラボを終了す

指示 リソース ?

すべてのターミナルを開いてます

!すべてのターミナルを閉じるオプションがあるVisual Studio Codeの[ターミナル]タブを示すスクリーンショット。

あなたは、地元のレストランから食べ物を注文できる組みのソリューションを構築することを検討しています。このソリューションでは、自然言語を使用し、Microsoft 365 Copilot と統合する必要があります。

いくつかの調査を行い、API プラグインと統合された Microsoft 365 Copilot 用の宣言型エージェントを構築することが、次の理由からニーズに適していることがわかりました。

- これにより、自然言語を理解する AI を利用したアシスタントを構築できます。
- Microsoft 365 Copilot のインフラストラクチャの上に構築できます。
- 手順を使用して、シナリオに合わせてアシスタントを最適化できます。
- エージェントをレストランの API に接続して、メニューを閲覧し、注文することができます。

以前は、今日のメニューについて知り、すべての人の食事のニーズを満たす注文を行うために、あなたと同僚は電話で貴重な時間を費やす必要がありました。

宣言型エージェントを構築することで、あなたと同僚はすぐに今日のメニューを見つけ出し、好みに応じてフィルター処理できます。エージェントを Microsoft 365 Copilot で利用できると便利です。これは、別のアプリケーションに移動しなくとも、今いる場所ですぐに使用できるためです。

おめでとう！

これで、この演習は正常に完了しました。次をクリックして次のラボに進みます。

前 次

残り4時59分

