Build your first OneNote task pane add-in

Article • 09/17/2024

In this article, you'll walk through the process of building a OneNote task pane add-in.

Prerequisites

- Node.js (the latest LTS version). Visit the [Node.js site](#) ↗ to download and install the right version for your operating system.

- The latest version of Yeoman and the Yeoman generator for Office Add-ins. To install these tools globally, run the following command via the command prompt.

  command line

  ```
  npm install -g yo generator-office
  ```

  > ⓘ **Note**
  >
  > Even if you've previously installed the Yeoman generator, we recommend you update your package to the latest version from npm.

- Office connected to a Microsoft 365 subscription (including Office on the web).

  > ⓘ **Note**
  >
  > If you don't already have Office, you might qualify for a Microsoft 365 E5 developer subscription through the [Microsoft 365 Developer Program](#) ↗; for details, see the [FAQ](#). Alternatively, you can [sign up for a 1-month free trial](#) ↗ or [purchase a Microsoft 365 plan](#) ↗.

Create the add-in project

Run the following command to create an add-in project using the Yeoman generator. A folder that contains the project will be added to the current directory.
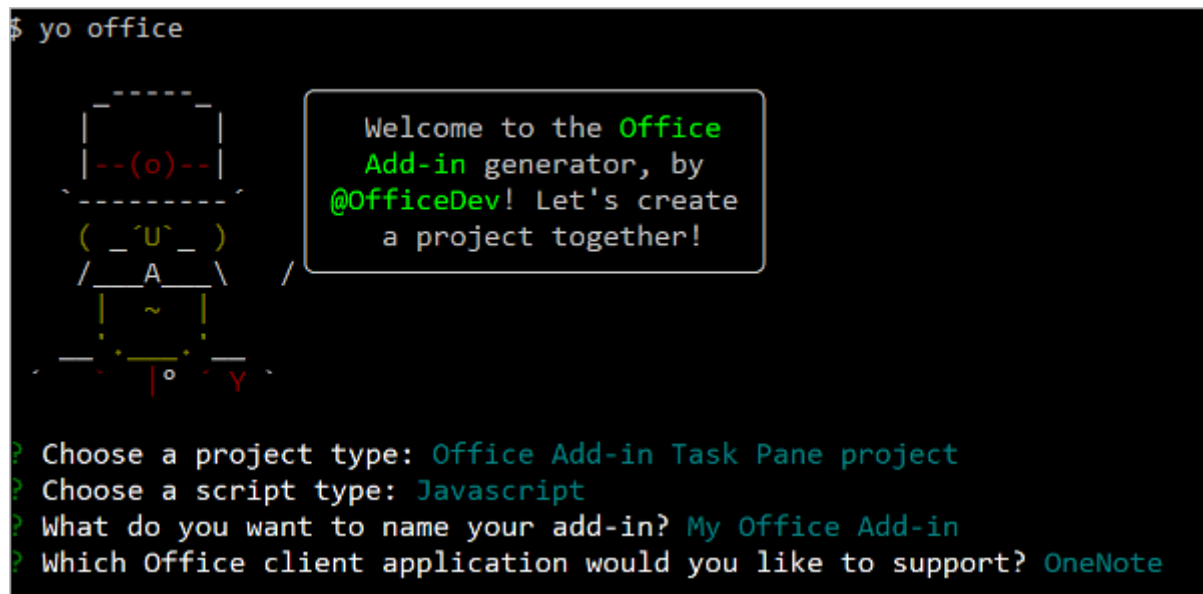
command line

```
yo office
```

When prompted, provide the following information to create your add-in project.

- **Choose a project type:** `Office Add-in Task Pane project`
- **Choose a script type:** `Javascript`
- **What do you want to name your add-in?** `My Office Add-in`
- **Which Office client application would you like to support?** `OneNote`

After you complete the wizard, the generator creates the project and installs supporting Node components.

Explore the project

The add-in project that you've created with the Yeoman generator contains sample code for a very basic task pane add-in.

- The **./manifest.xml** file in the root directory of the project defines the settings and capabilities of the add-in.
- The **./src/taskpane/taskpane.html** file contains the HTML markup for the task pane.

- The **./src/taskpane/taskpane.css** file contains the CSS that's applied to content in the task pane.
- The **./src/taskpane/taskpane.js** file contains the Office JavaScript API code that facilitates interaction between the task pane and the Office client application.

Update the code

In your code editor, open the file **./src/taskpane/taskpane.js** and add the following code within the `run` function. This code uses the OneNote JavaScript API to set the page title and add an outline to the body of the page.

JavaScript

```javascript
try {
    await OneNote.run(async (context) => {

        // Get the current page.
        const page = context.application.getActivePage();

        // Queue a command to set the page title.
        page.title = "Hello World";

        // Queue a command to add an outline to the page.
        const html = "<p><ol><li>Item #1</li><li>Item #2</li></ol></p>";
        page.addOutline(40, 90, html);

        // Run the queued commands.
        await context.sync();
    });
} catch (error) {
    console.log("Error: " + error);
}
```

Try it out

1. Navigate to the root folder of the project.

   command line

   ```
   cd "My Office Add-in"
   ```

2. Start the local web server. Run the following command in the root directory of your project.
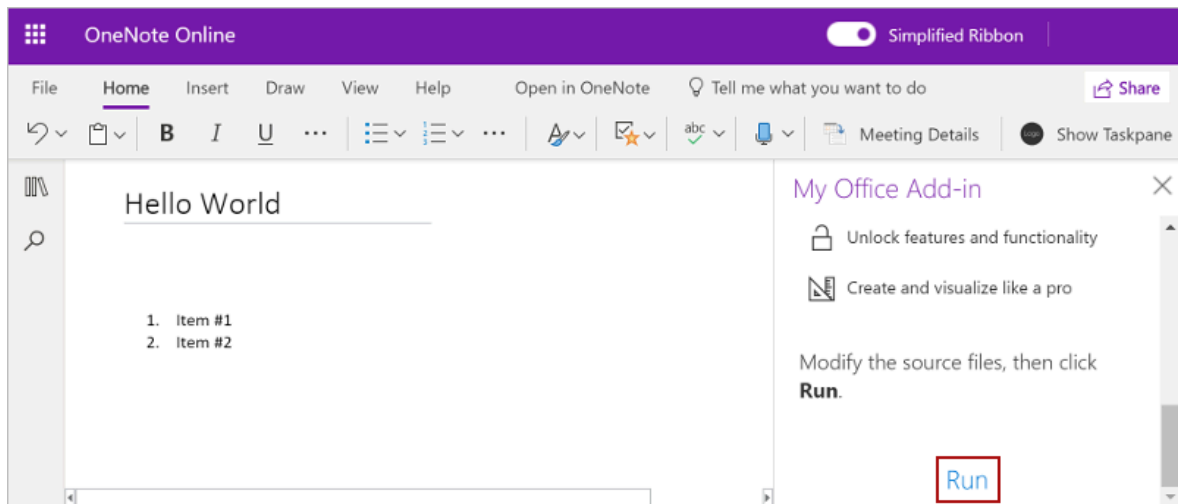
   command line

```
npm run dev-server
```

3. In OneNote on the web ⧉ , open a notebook and create a new page.

4. Choose **Insert** > **Office Add-ins** to open the Office Add-ins dialog.

   - If you're signed in with your consumer account, select the **MY ADD-INS** tab, and then choose **Upload My Add-in**.

   - If you're signed in with your work or education account, select the **MY ORGANIZATION** tab, and then select **Upload My Add-in**.

   The following image shows the **MY ADD-INS** tab for consumer notebooks.

5. In the Upload Add-in dialog, browse to **manifest.xml** in your project folder, and then choose **Upload**.

6. From the **Home** tab, choose the **Show Taskpane** button on the ribbon. The add-in task pane opens in an iFrame next to the OneNote page.

7. At the bottom of the task pane, choose the **Run** link to set the page title and add an outline to the body of the page.

8. When you want to stop the local web server and uninstall the add-in, follow these instructions:

   - To stop the server, run the following command.

command line
`npm stop`

   - To uninstall the sideloaded add-in, see Remove a sideloaded add-in.

Next steps

Congratulations, you've successfully created a OneNote task pane add-in! Next, learn more about the core concepts of building OneNote add-ins.

OneNote JavaScript API programming overview

Troubleshooting

- Ensure your environment is ready for Office development by following the instructions in Set up your development environment.

- Some of the sample code uses ES6 JavaScript. This isn't compatible with older versions of Office that use the Trident (Internet Explorer 11) browser engine. For information on how to support those platforms in your add-in, see Support older Microsoft webviews and Office versions. If you don't already have a Microsoft 365 subscription to use for development, you might qualify for a Microsoft 365 E5 developer subscription through the Microsoft 365 Developer Program ☑; for

details, see the FAQ. Alternatively, you can sign up for a 1-month free trial ⧉ or purchase a Microsoft 365 plan ⧉ .

- The automatic `npm install` step Yo Office performs may fail. If you see errors when trying to run `npm start`, navigate to the newly created project folder in a command prompt and manually run `npm install`. For more information about Yo Office, see Create Office Add-in projects using the Yeoman Generator.

See also

- Office Add-ins platform overview
- Develop Office Add-ins
- OneNote JavaScript API programming overview
- OneNote JavaScript API reference
- Rubric Grader sample ⧉
- Using Visual Studio Code to publish