* **Drivers**:

Drivers are the software programs that help the operating system communicate with the hardware.

There are so many drivers like USB, network, disk, etc.

* **There are two types of drivers**:

1. User-mode drivers:
2. Kernel-mode drivers:
3. **User-mode Drivers**:

The user application uses user-mode drivers.

Ex. Headphones, USB, etc.

1. **Kernel-mode Drivers**:

Kernel-mode drivers are known as Windows core.

They manage things like network cards, disk drives, etc.

If a Kernel-mode driver breaks or has a bug, the whole system might crash, causing a blue screen error.

* **Examining logs for installing drivers:**

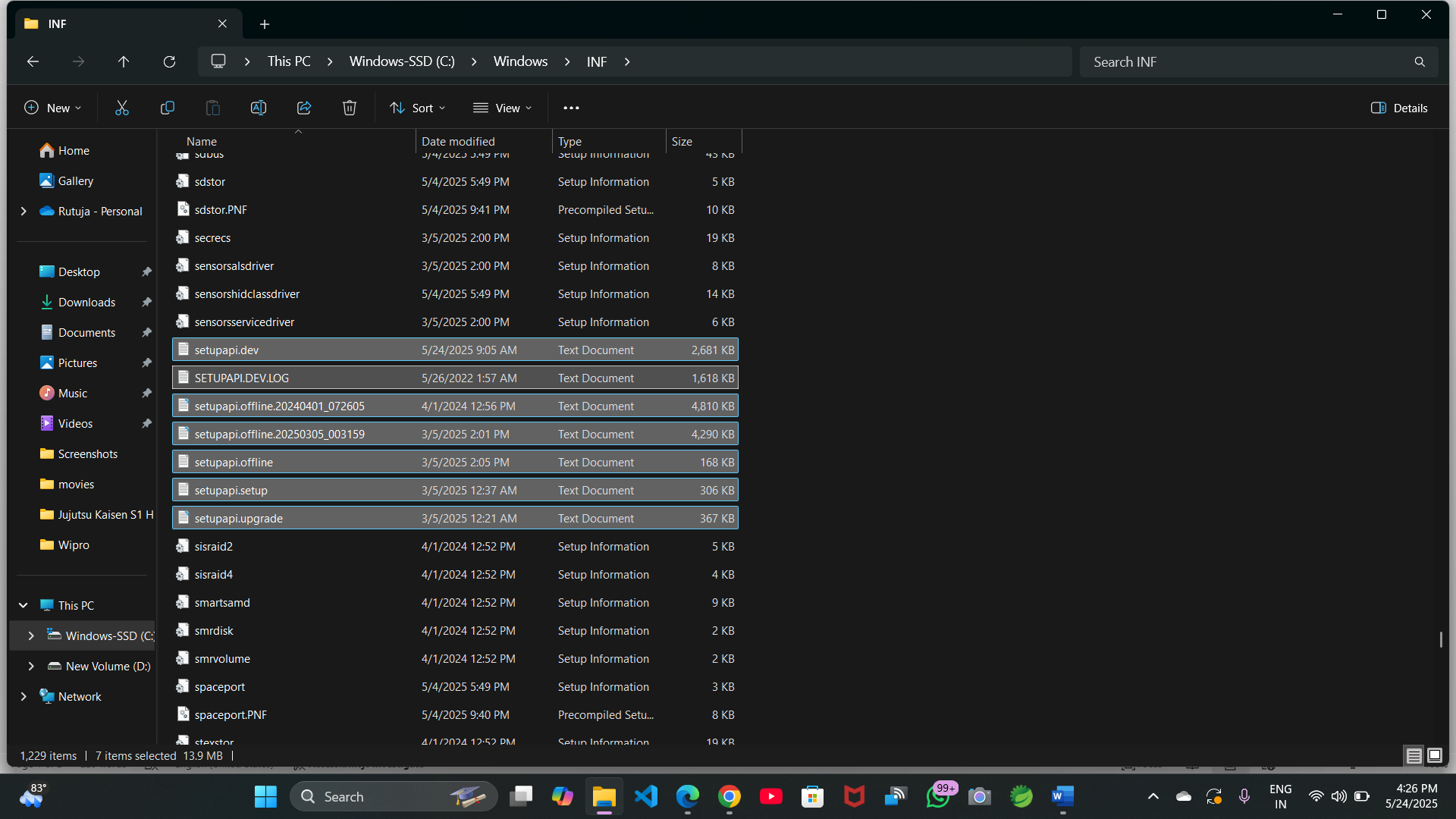
* **To install a driver, we need these files:**  
   .**inf**: It is like a guide for OS. It tells the OS how to install the driver.

.**sys**: It is the actual driver file. Windows copies and registers the driver.

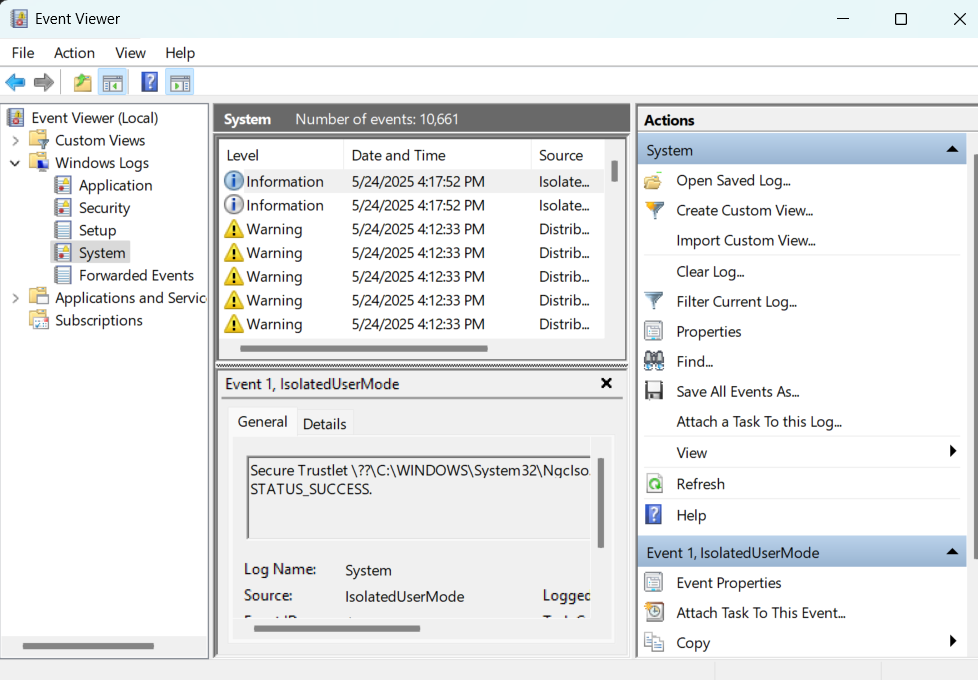
.**cat**: It’s a catalog file. It is like a digital signature for driver files, ensuring the system that it is safe and trusted.

1. **On Windows, we might check the logs in:**

**C:\Windows\inf\setupapi.dev.log (for device installation)**



1. **Event Viewer under "System" logs for Plug and Play (PnP) events**



* **MSIX**

MSIX is the modern app packaging format for Windows.

Traditionally, we have .msi, .app-V, .app-X, now we have MSIX, which contains the features of all mentioned formats.

* **MSI** is the traditional Microsoft Installer.
* **App-V** was used for virtual applications.
* **App-X** was used for a UWP app means the Microsoft Store apps.

We can say MSIX is the combination of MSI, App-V, and App-X.

* **Benefits**:

1. Makes apps easier to install, update, and uninstall.
2. Sandboxes (isolated from the system) apps for better security.
3. Drivers install safely and correctly.

MSIX can’t install a Kernel-mode driver because kernel drivers need full access to the Windows kernel, which can be harmful for MSIX’s sandbox.

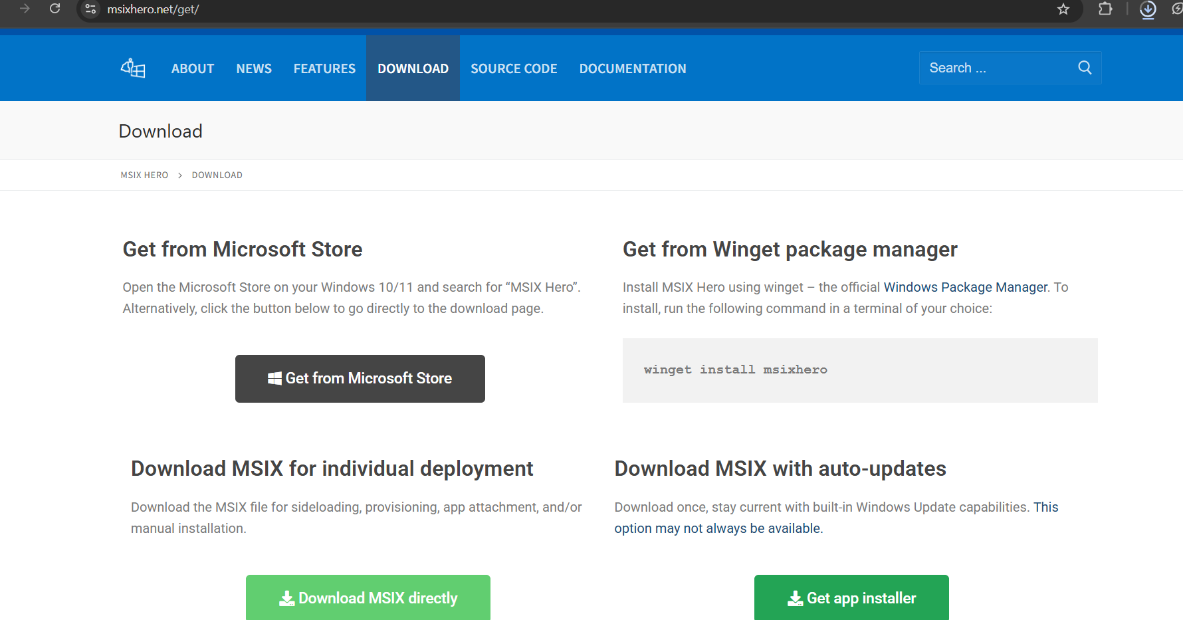
* **How to install the MSIX packaging tool**:

1. Go to the Microsoft store
2. Search for the MSIX packaging tool
3. Click install.

* **Steps to install MSIX:**

**Step 1**: Get the MSIX Package

Download the **.msix** file

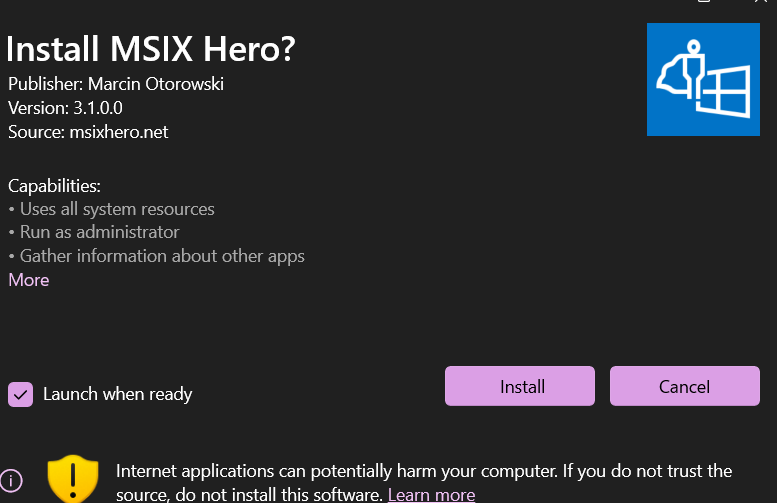


**Step 2**: Double-click the File

On **Windows 10/11**, just **double-click the .msix file**.

You’ll see a simple **installer window** showing:

* App name
* Publisher
* Version

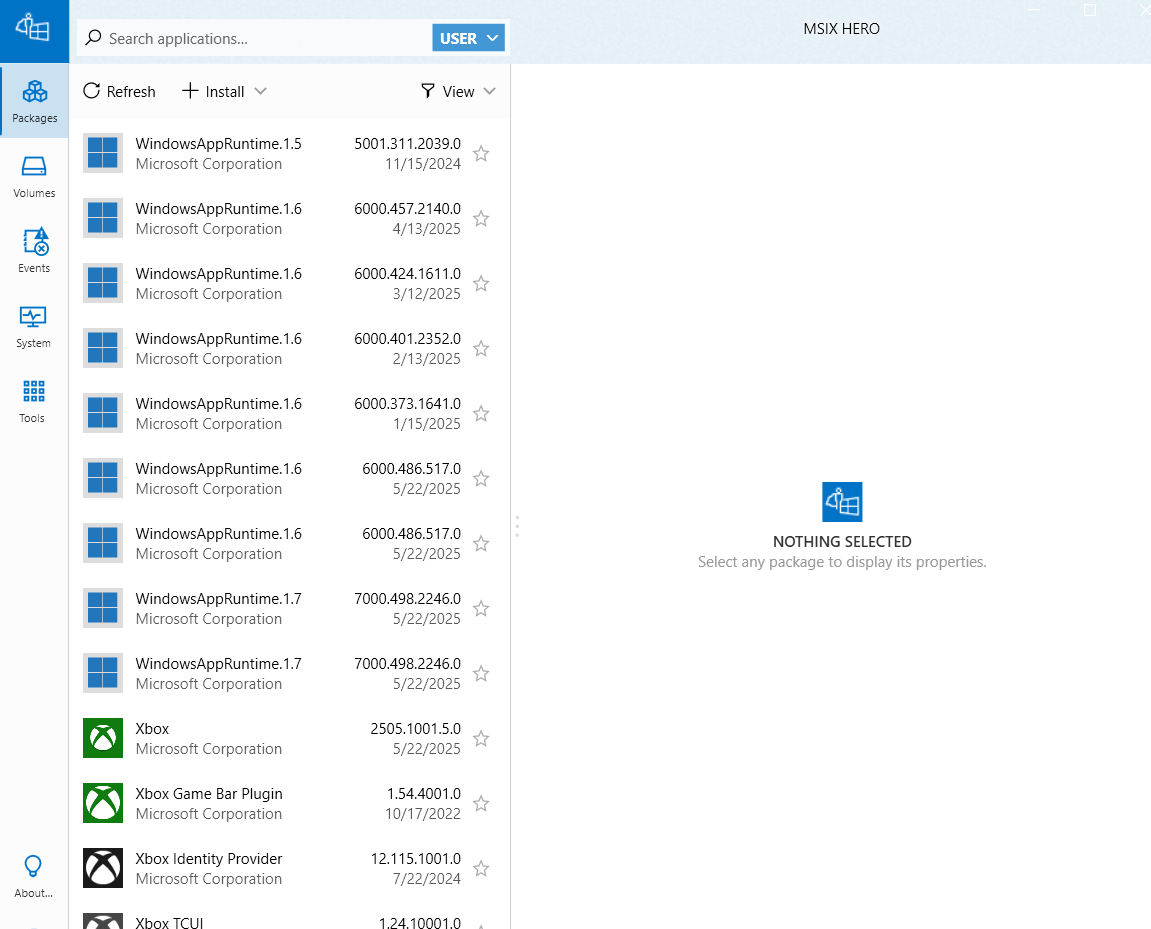


**Step 3: Click Install**

* Click the **Install** button.
* Windows will install the app and place it in the **Start Menu**.
* The app runs in a **sandbox** for better security.
* **Alternative (PowerShell):**
* If double-click doesn’t work, use PowerShell:

**Command**:

**Add-AppxPackage -Path "C:\Path\To\YourFile.msix"**



* **Handling kernel-mode drivers in MSIX**

MSIX is designed for user-mode applications. MSIX has limited access for kernel-mode drivers as compared to traditional installers, MSI, or INF-based installs.

Kernel-mode drivers interact with the OS kernel; hence, they need to be signed and need administrator permissions.

**Step 1**: Package your app in MSIX

**Step 2**: Install the driver separately

Since the driver is a Kernel-mode driver, we need to:

1. Sign in the driver properly (Windows needs it for security)
2. Create an INF file (tells Windows how to install the driver)
3. Use a simple tool like pnputil to install the driver

Let’s say your driver files are in C:\Drivers\MyDriver. You can use this simple command:

**pnputil /add-driver C:\Drivers\MyDriver\MyDriver.inf /install**

* **Application Packaging:**

Application packaging is a type of formatting our software in such a way that it can be easily installed, managed, and uninstalled.

To deploy and use software, we require various useful files such as a settings file, configuration, permissions, etc. To combine all these files, we do application packaging.

Here are some packaging formats:

* **For Windows**:

1. .exe: common executable installer
2. .**msi** (Microsoft installer): standard for Windows applications
3. .**app-V** (Application virtualization): apps run in an isolated environment
4. .**msix**: modern packaging format replacing .msi and .app-V

* **End-to-end packaging process**:

It consists of three steps:

1. Application discovery
2. Application packaging
3. UAT



* **Add ins**

Add-ins are the little programs that provide special features for apps (like Excel, Word, and PowerPoint).

They help us to do tasks like inserting charts, connecting two databases, sending emails, etc.

* **Types of Add-ins:**

1. **Excel, Word, PowerPoint Add-ins:**

These are office add-ins.

Can be installed from the office store or side-loaded from the manifest file

They usually use JavaScript and HTML to interact with Office apps.

1. **COM Add-ins (Component Object Model):**

Older types of add-ins for Office apps

Written in C++ or .NET, or C # language

Installed on Windows as .dll files

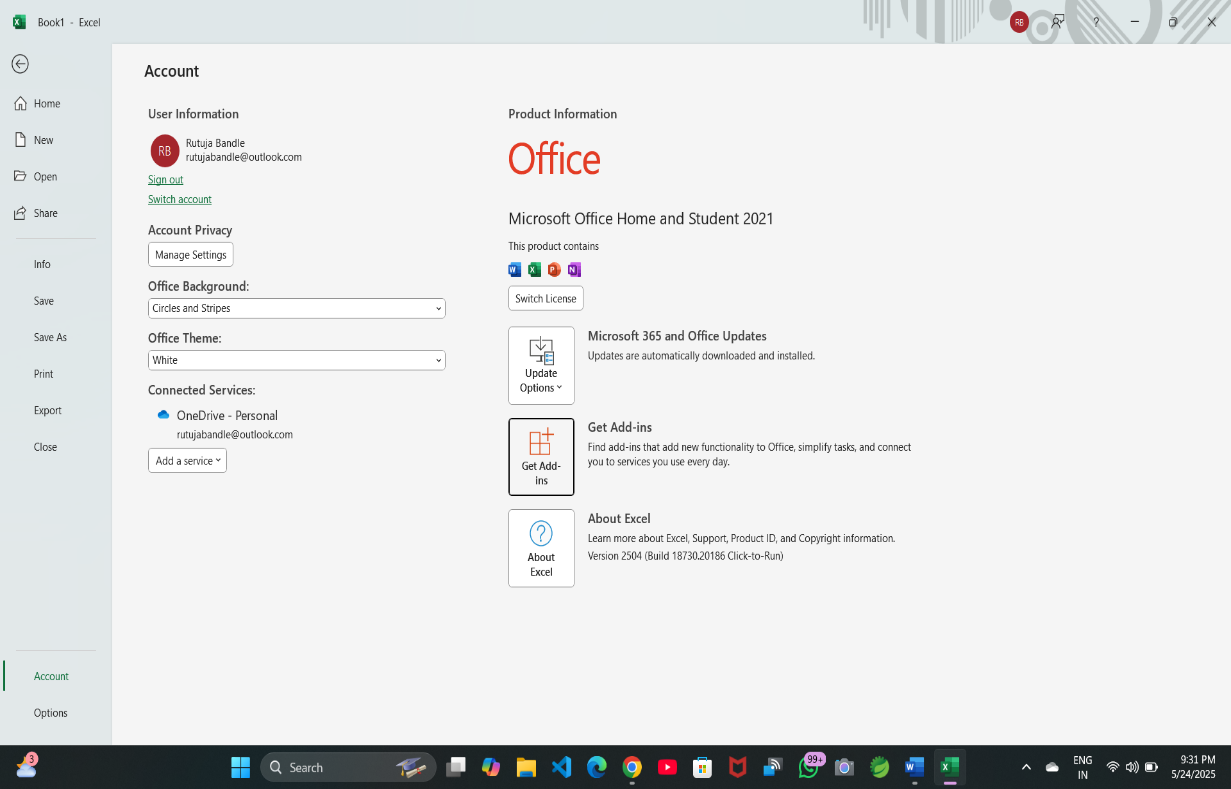
They are more powerful but harder to build as compared to modern add-ins

Used for things like custom buttons, Excel connectors, or Outlook plugins.

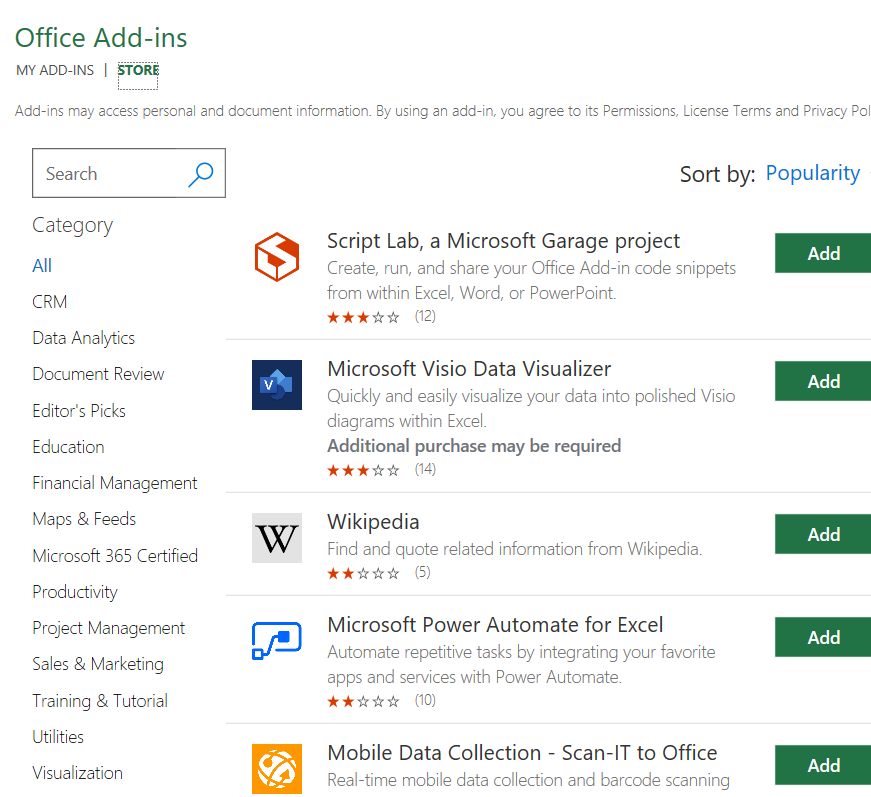
* **How to install add-ins:**

**For Excel, Word, and PowerPoint add-ins:**

1. Open the app (Ex, Excel)
2. Click File > Account > Get add-ins

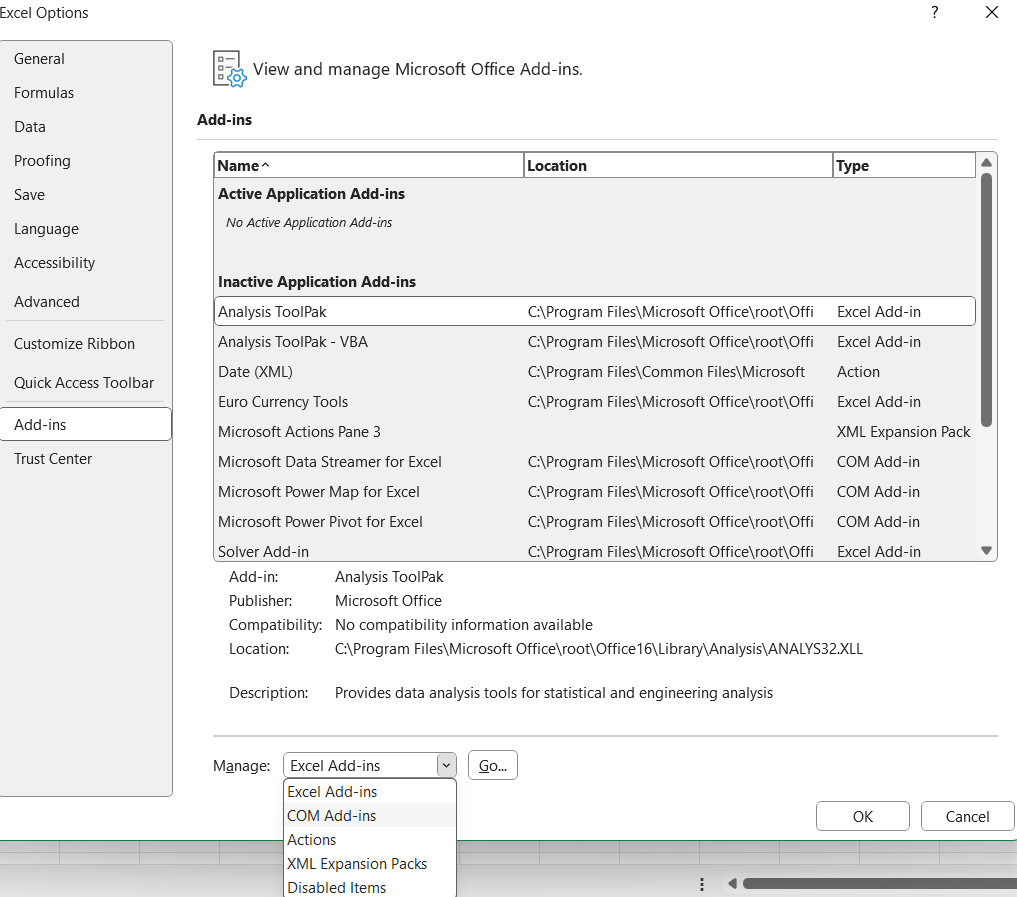


1. Search and install your add-ins

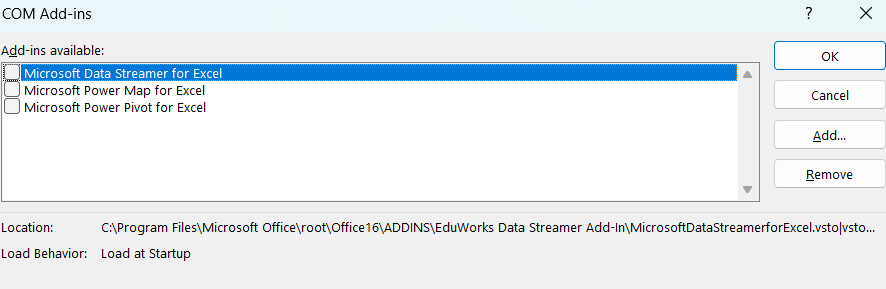


**For COM add-ins:**

1. Open Excel/Word
2. Go to File> Options> Add-ins



1. At the bottom, select COM add-ins and click go



1. Browse and add your COM add-ins (.dll)