**NAME:** - ANSHUL SAINI  
**MAIL:** - [sainianshul7565@gmail.com](mailto:sainianshul7565@gmail.com)

**User ID**: - 34732

**Date: -** 11/08/2025

**Batch: -** DWS\_B5\_25VID2550

**Topic Assignment: -**

**PowerShell App Deploy Toolkit**

* Understand basic platform value, concepts and usage
* Overview of the Tool
* Folder Structure
* Tool Configuration
* Template Script
* Predefined Environment Variables defined within Script and its usage
* Commands for Installation/Uninstallation
* Execute-MSI - Installation and Uninstallation
* Execute-Process

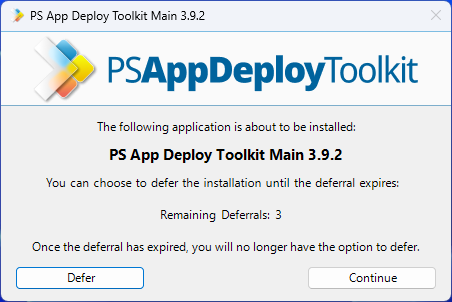
**PowerShell App Deploy Toolkit**

* **Understand basic platform value, concepts and usage: -**

The PowerShell App Deployment Toolkit (PSADT) is a free tool that helps IT teams install, update, and remove software in a standard and automated way. Instead of writing complex scripts from scratch, PSADT provides ready-made commands and a framework to make deployments easier and more reliable. It’s widely used in enterprises to save time and avoid errors.

* **Overview of the Tool Folder Structure: -**

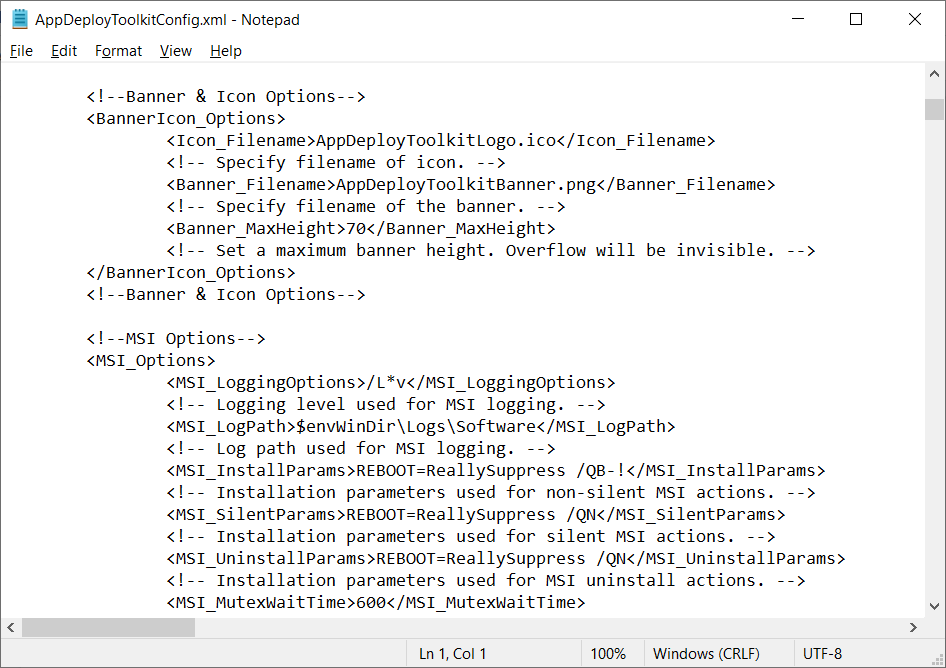
The toolkit is organized into folders so that files are easy to manage. The Files folder stores the application’s setup files, SupportFiles keeps extra resources like icons or configs, Toolkit contains the main PowerShell scripts, and Logs stores records of installations and uninstalls. This clear structure helps keep everything neat and easy to find.



* **Tool Configuration: -**

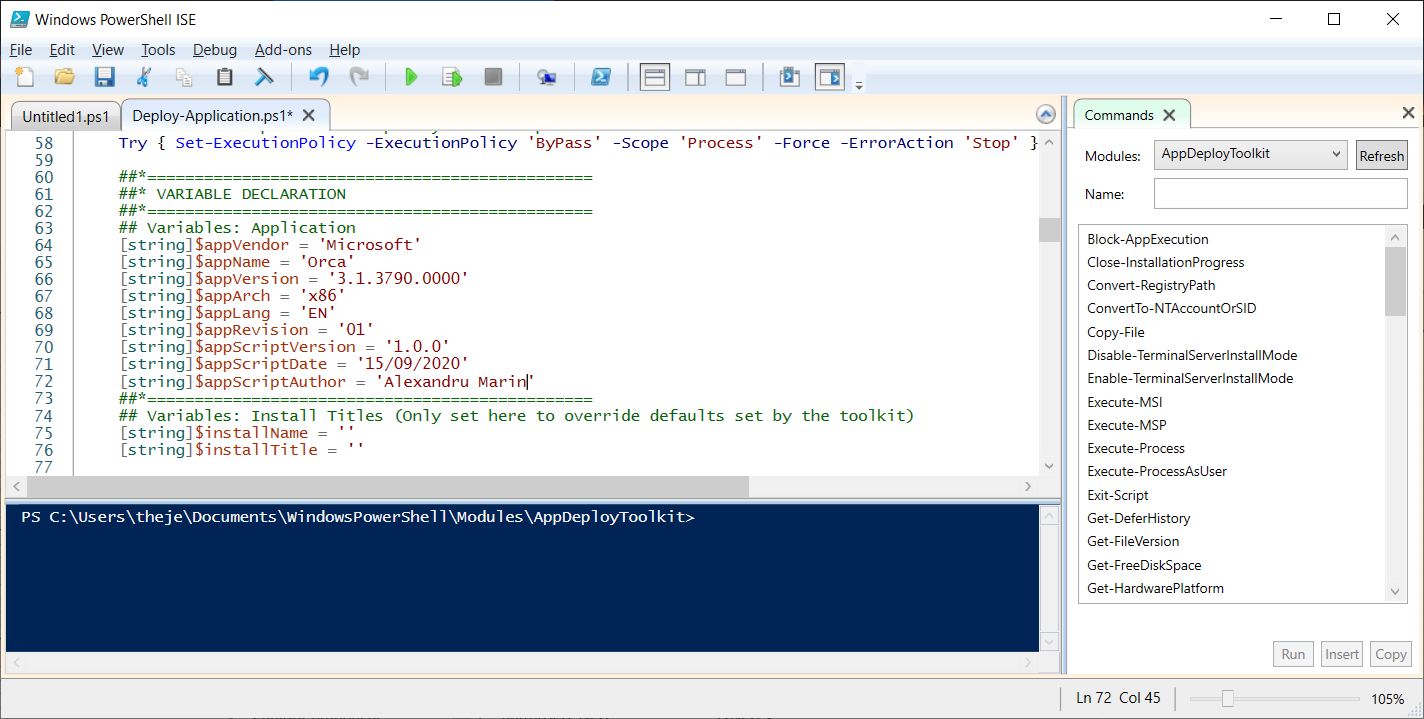
PSADT has a configuration file where you can change default settings such as log locations, whether installation progress is shown to users, and timeouts. This allows you to customize how the toolkit behaves to match your organization’s standards.

**Practice Activity: -**

****

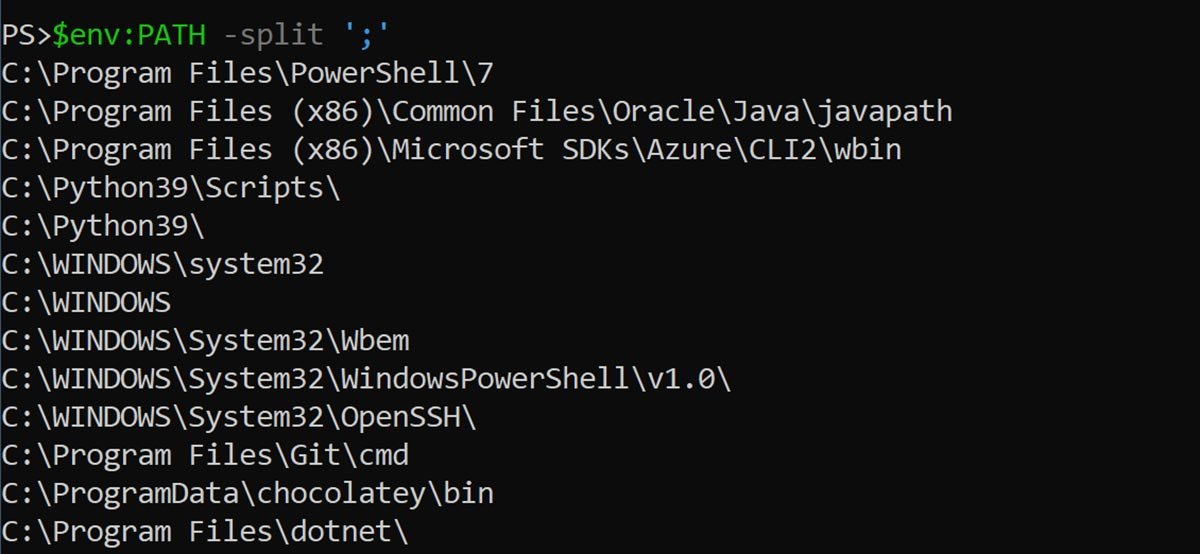
* **Template Script: -**

When you start a new deployment, you use the provided Deploy-Application.ps1 template. It already has sections for installing, uninstalling, or repairing applications. You simply add the commands in the right section, saving time and keeping scripts consistent.

**Practice Activity: -**

* **Predefined Environment Variables and Their Usage: -**

PSADT provides predefined variables that store system and application information, like $env:COMPUTERNAME for the PC’s name or $appDeployAppName for the application name from the config. These variables help make scripts more dynamic and reusable without hardcoding details.

**Practice Activity: -**

* **Commands for Installation/Uninstallation: -**

The toolkit has simple commands for installing or uninstalling applications. Instead of writing long PowerShell commands, you can use these built-in functions to handle software deployment in a cleaner and more reliable way.

# Using the PowerShell script directly

powershell.exe -executionpolicy bypass -file Deploy-Application.ps1

# Using the executable (recommended for suppressing console window)

Deploy-Application.exe

# Using the PowerShell script for uninstallation

powershell.exe -executionpolicy bypass -file Deploy-Application.ps1 -DeploymentType Uninstall

# Using the executable for uninstallation (recommended for suppressing console window)

Deploy-Application.exe -DeploymentType Uninstall

* **Execute-MSI – Installation and Uninstallation: -**

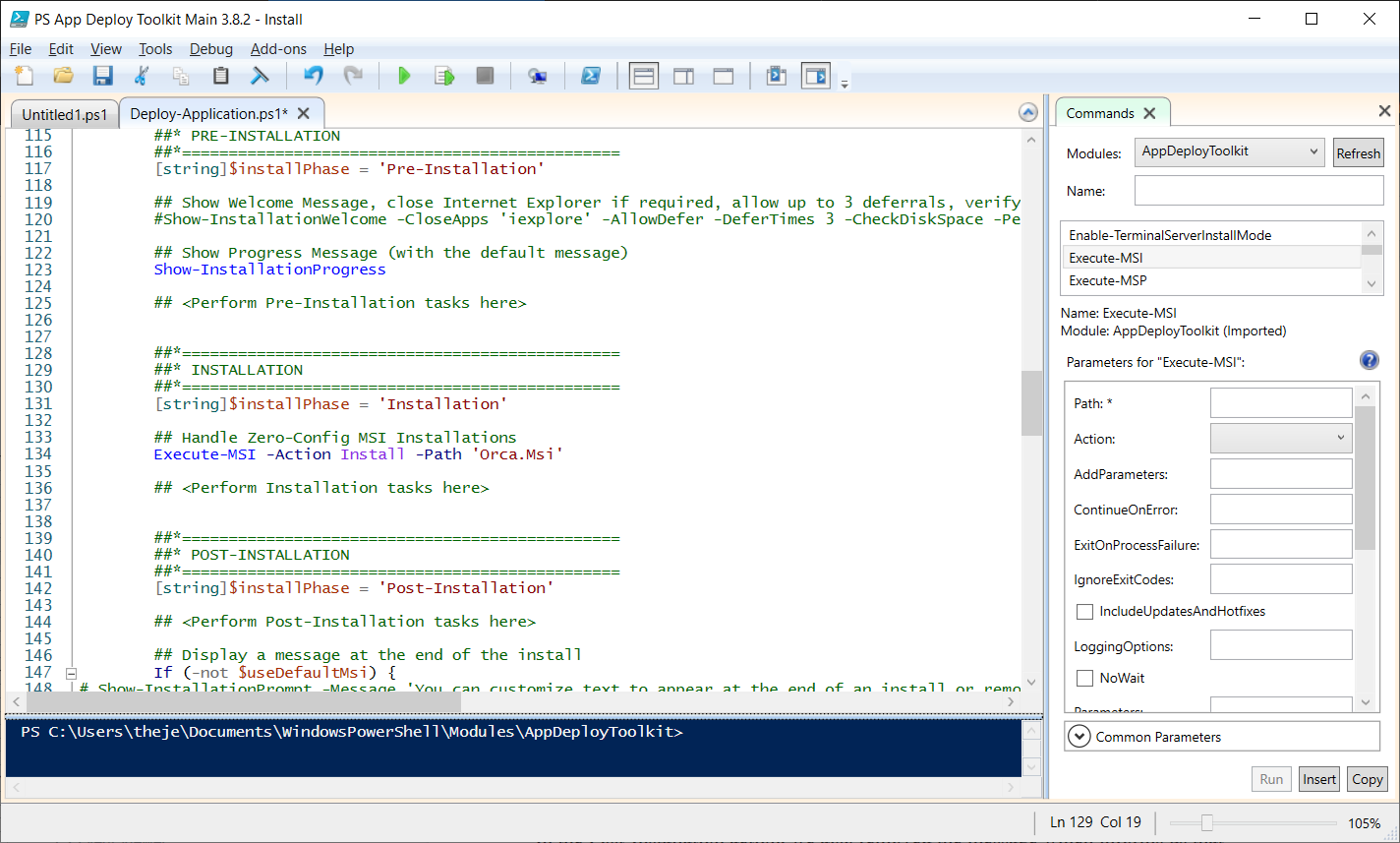
This command is used to install or uninstall MSI-based applications. For example, you can run Execute-MSI -Action Install -Path "app.msi" to install, or change the action to Uninstall to remove it. This makes MSI deployments very straightforward.

**Practice Activity: -**



* **Execute-Process: -**

The Execute-Process function runs EXE files or any program from within the script. This is useful for installing software that doesn’t come as an MSI, or for running helper tools during deployment.

**Practice Activity: -**

**Practice Activity: -**

