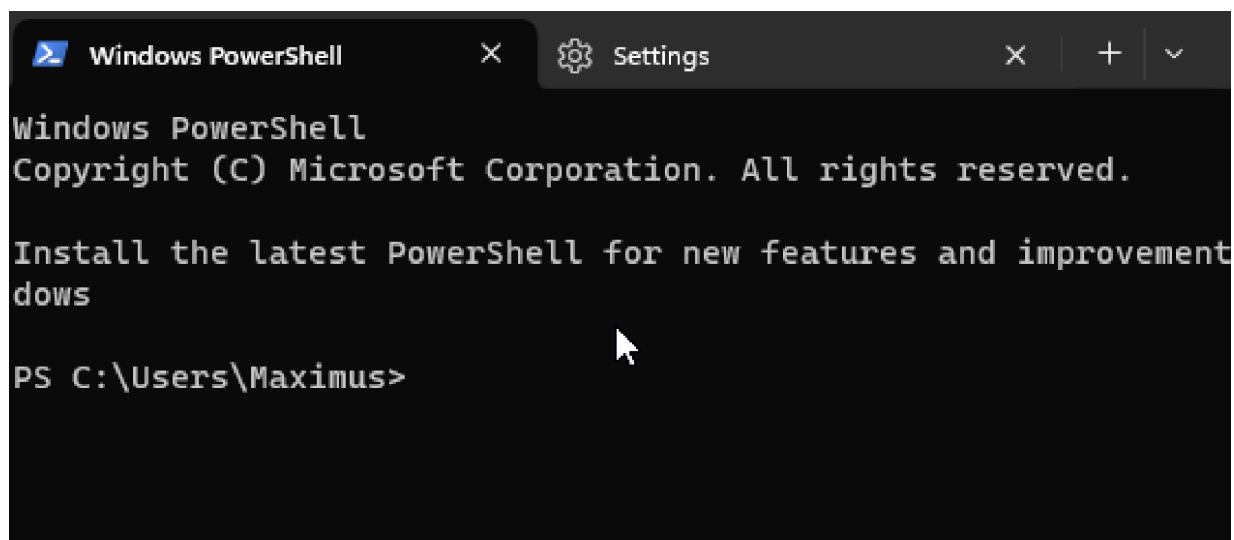
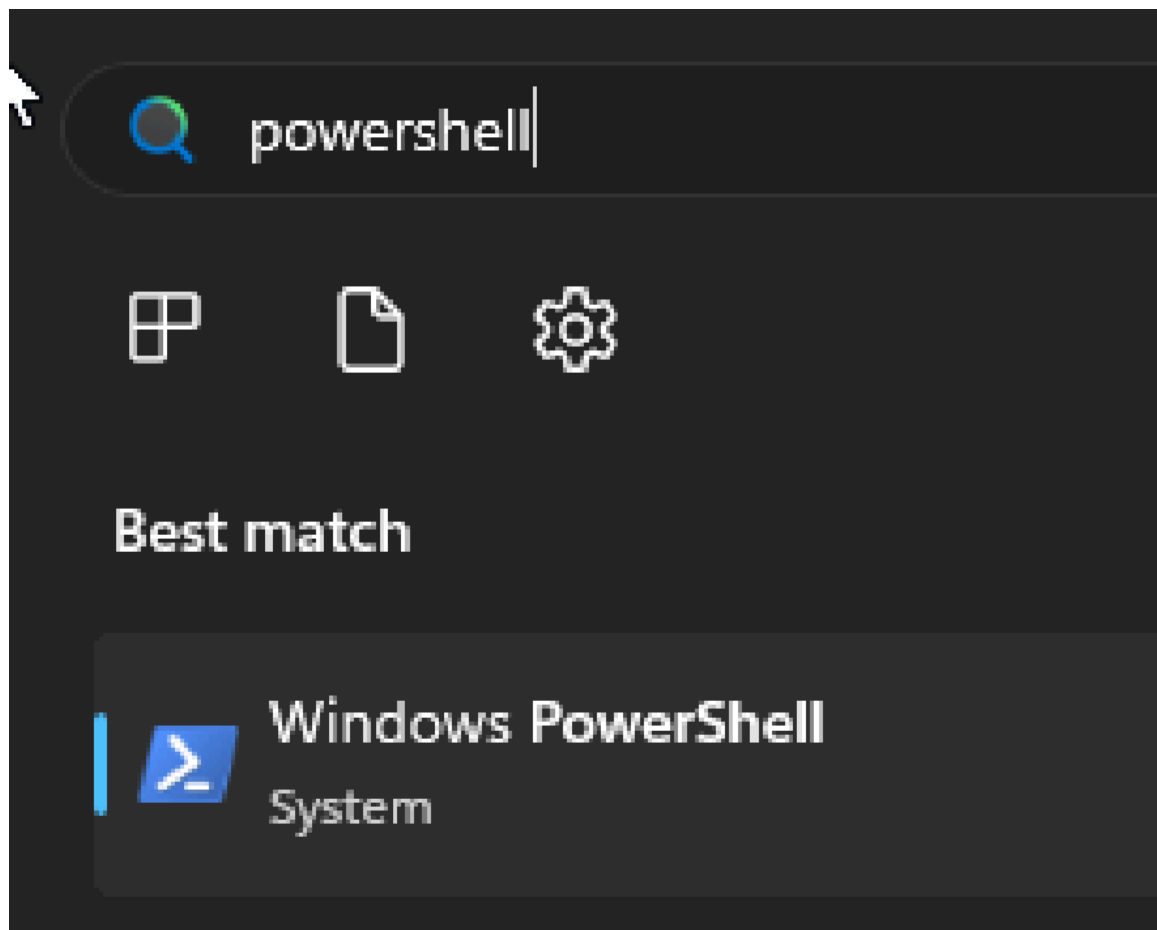


Windows PowerShell - This is similar to the CMD but starts in PowerShell



Windows PowerShell ISE - This is PowerShell With a GUI



powershell ise

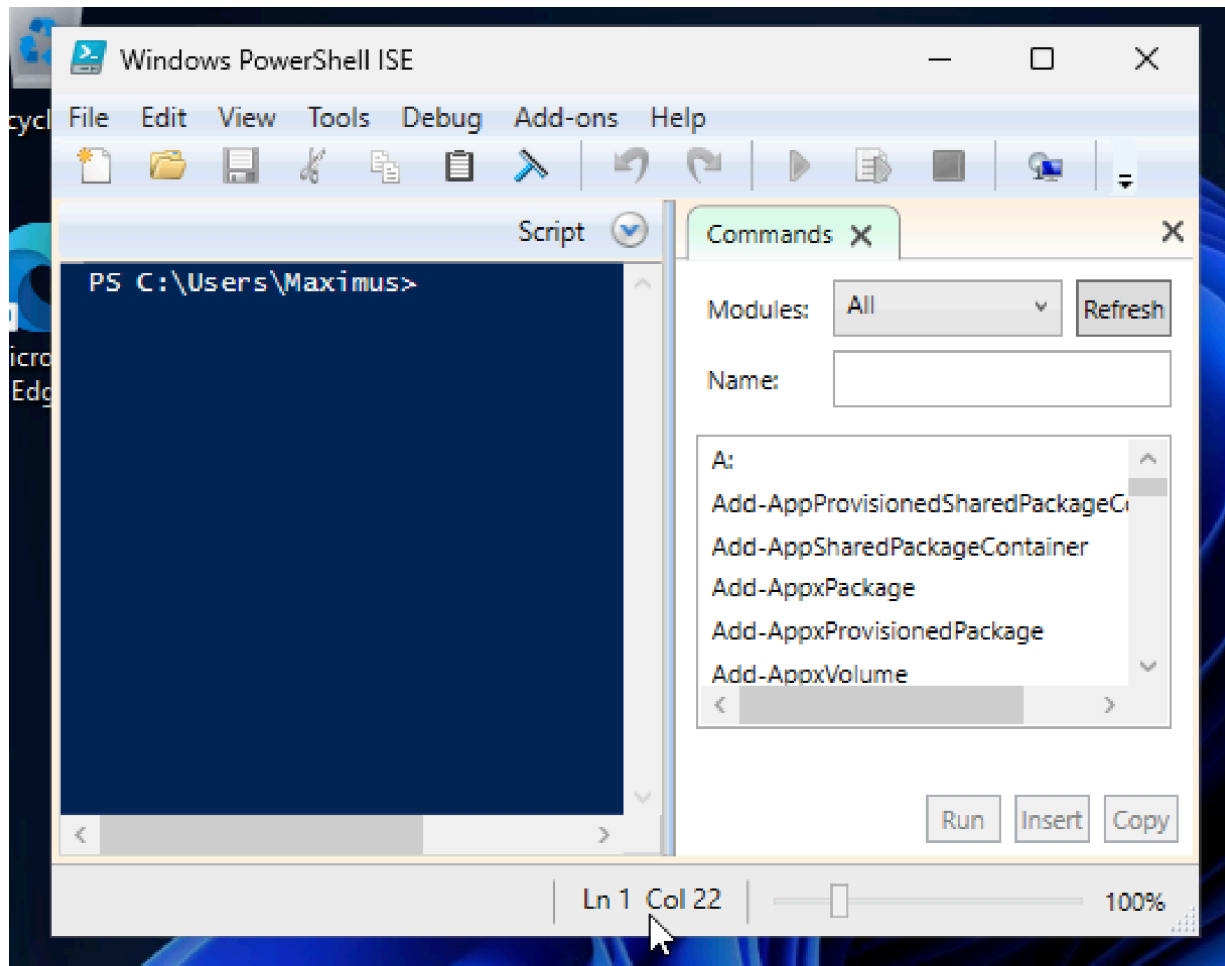


Best match



Windows PowerShell ISE

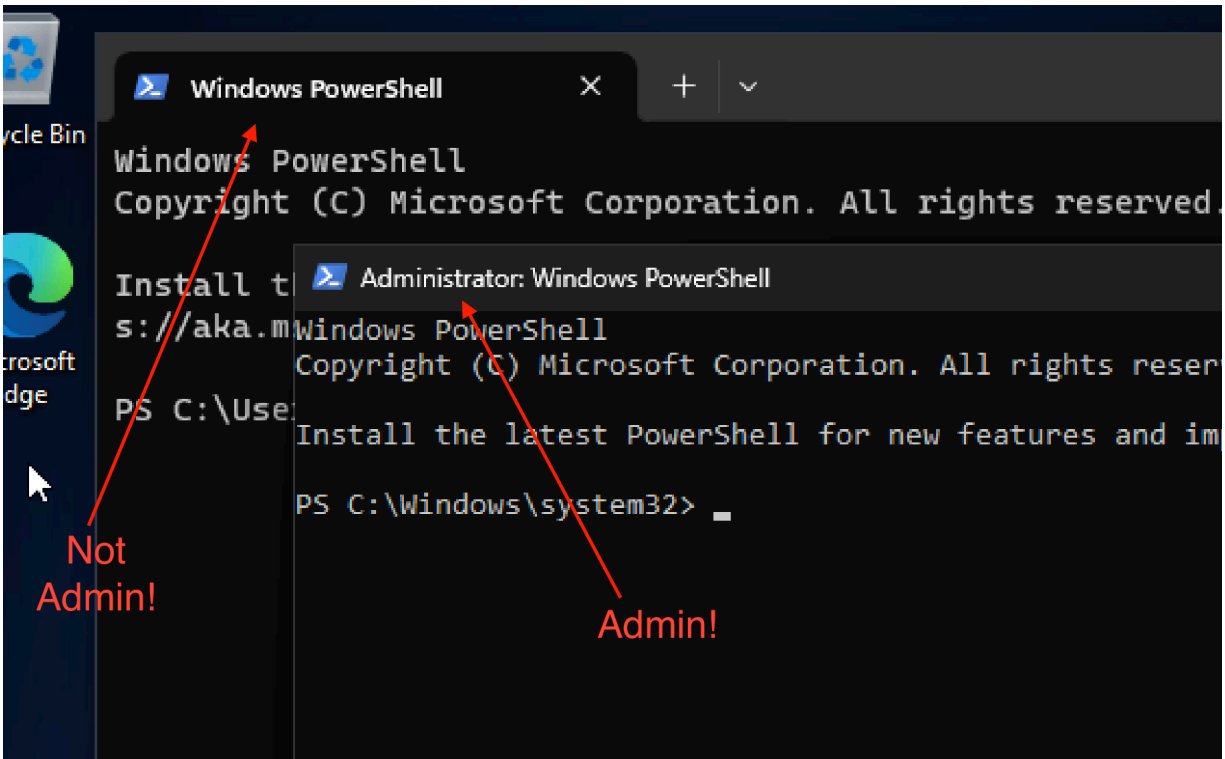
System



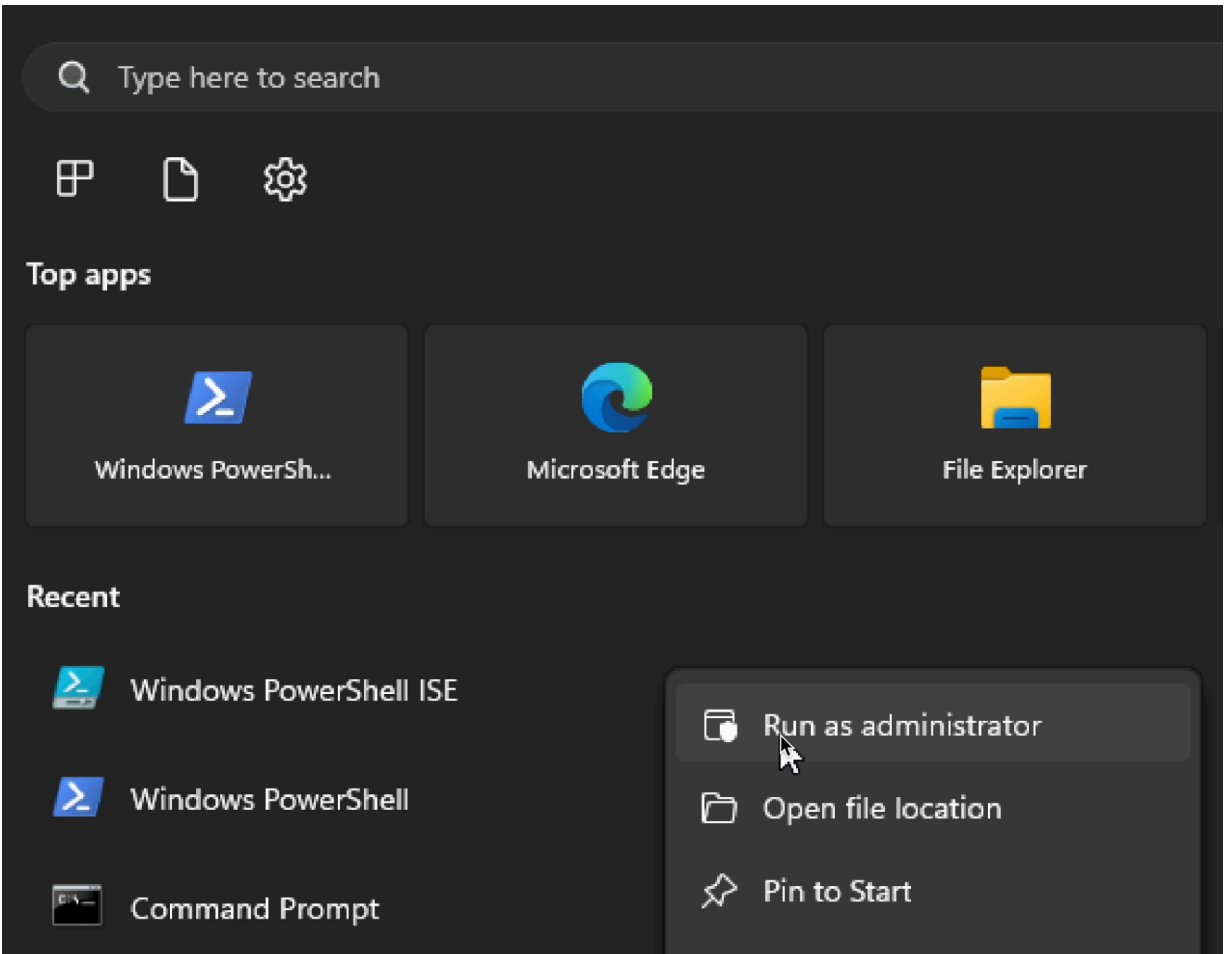
This GUI allows you to browse through cmdlets.

Note*

Look at the screen shot below. Note that there are 2 instances of PowerShell open. One is open as an admin and the other is not.



It is important to pay attention to this. It is best practice to only run any CLI as an admin only when needed. To run as as admin simply right click on the icon for PowerShell and click “Run as Administrator”.



PowerShell works with standard Windows commands

Use what you know already

Performs commands like:

Cd - change directory

Ls - list directory

Dir - directory listing

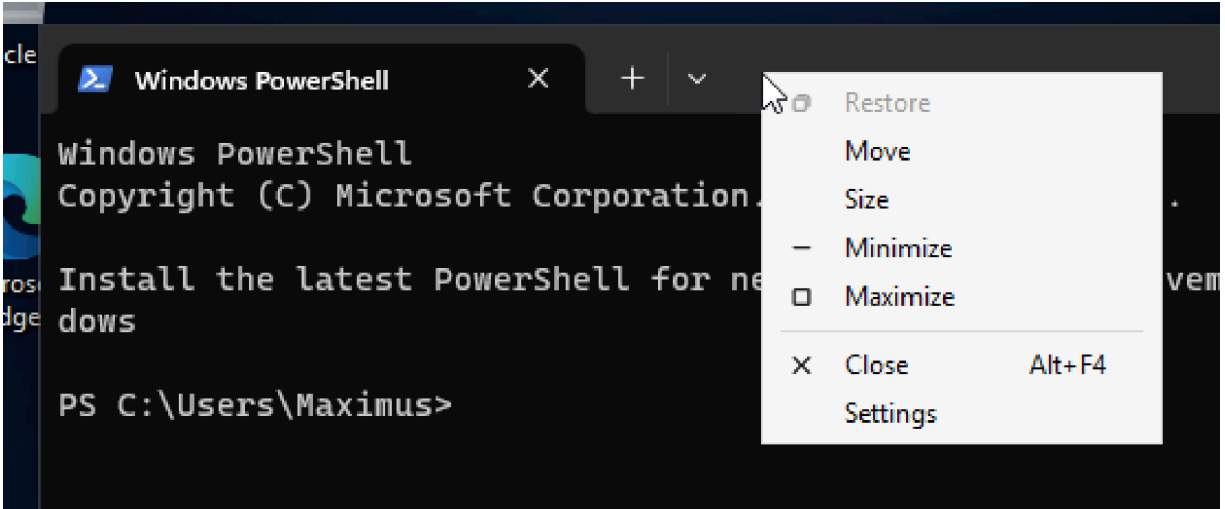
Works with Windows applications

Works with built-in commands called cmdlets

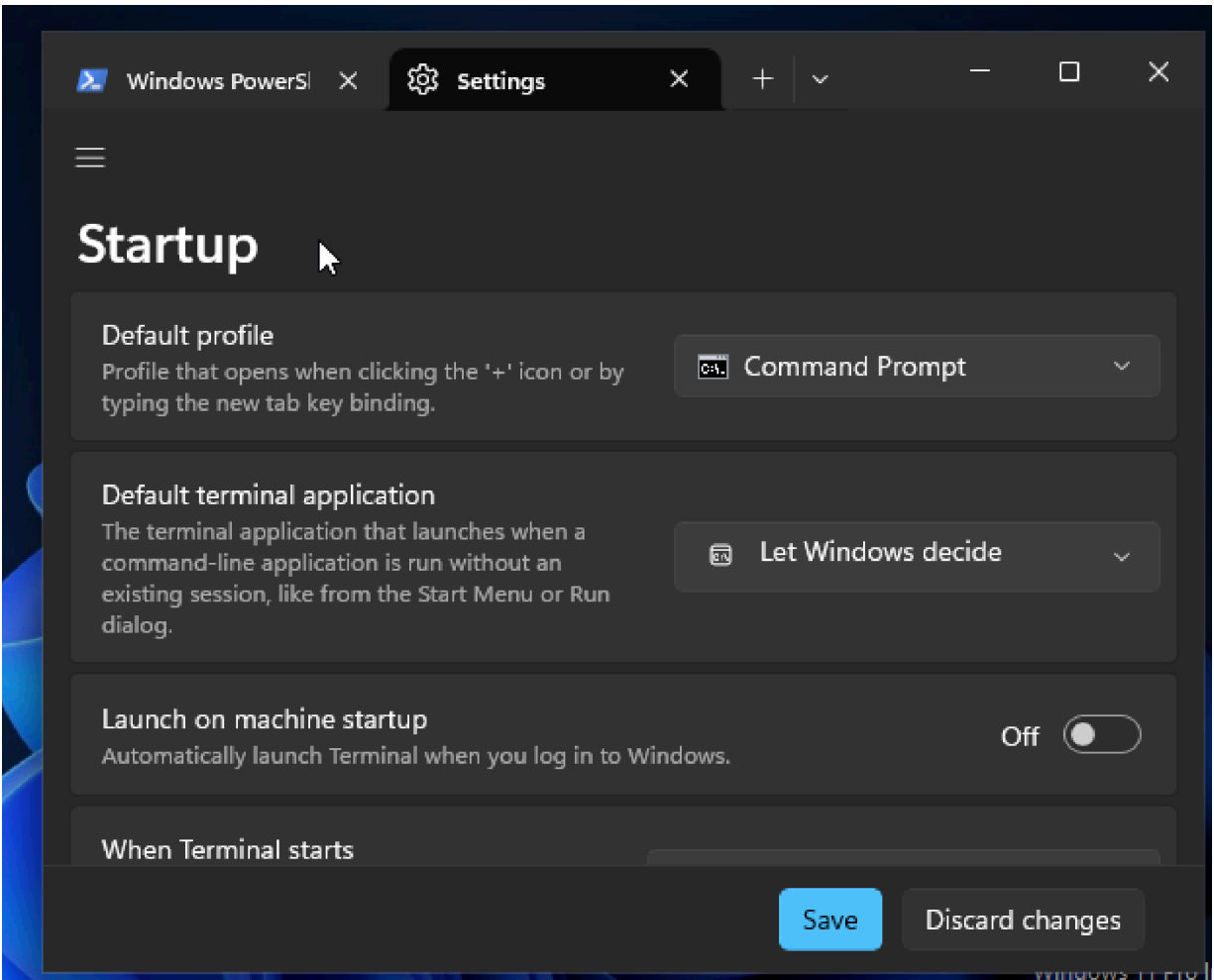
Can be used to navigate around file systems

Customize PowerShell

You can customize the look and overall environment of PowerShell (Or CMD). To do so right click the top bar and select settings.



This will bring up the settings panel/tab



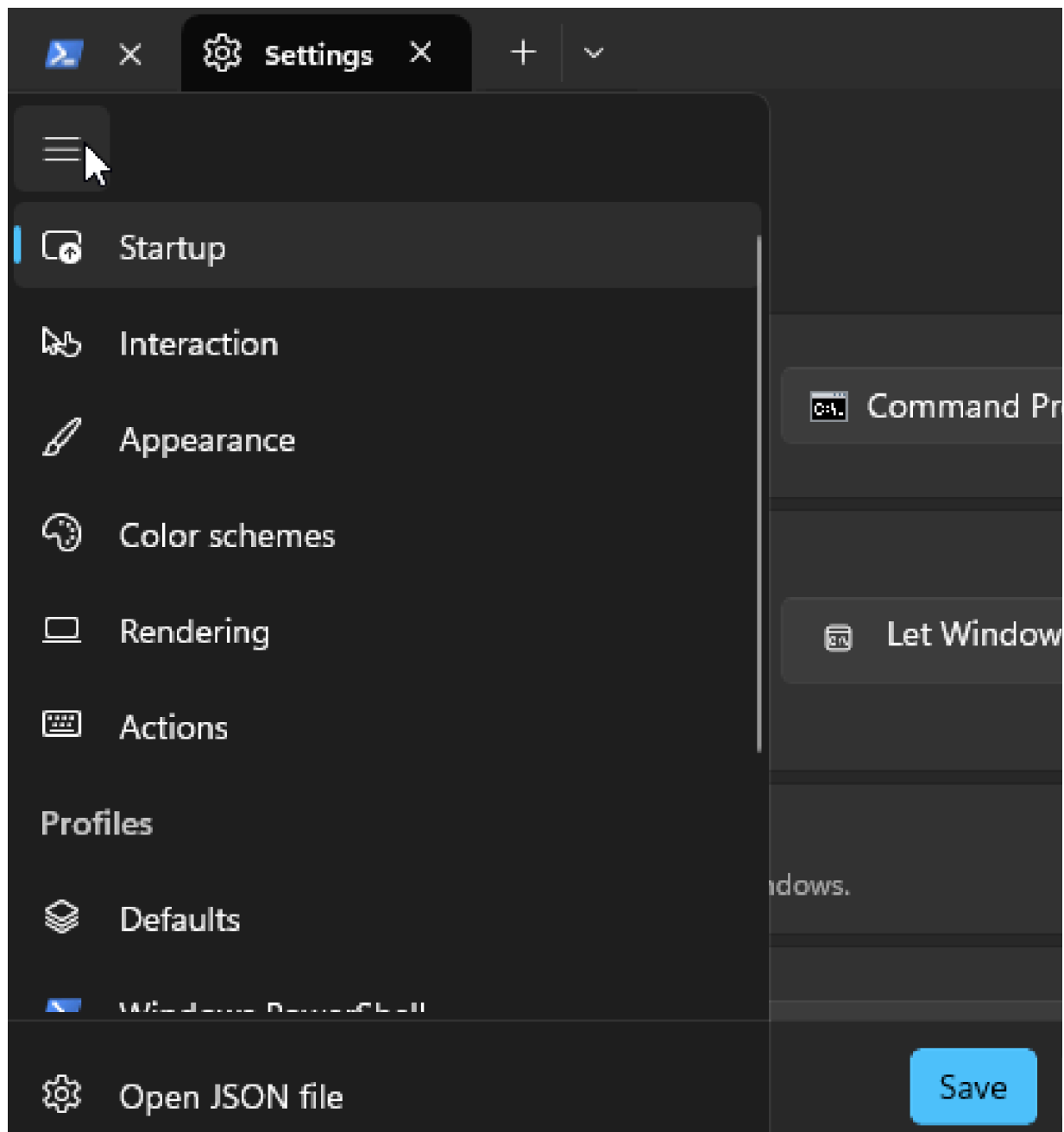
Here we can edit settings on Various things.

- The Console can be customized

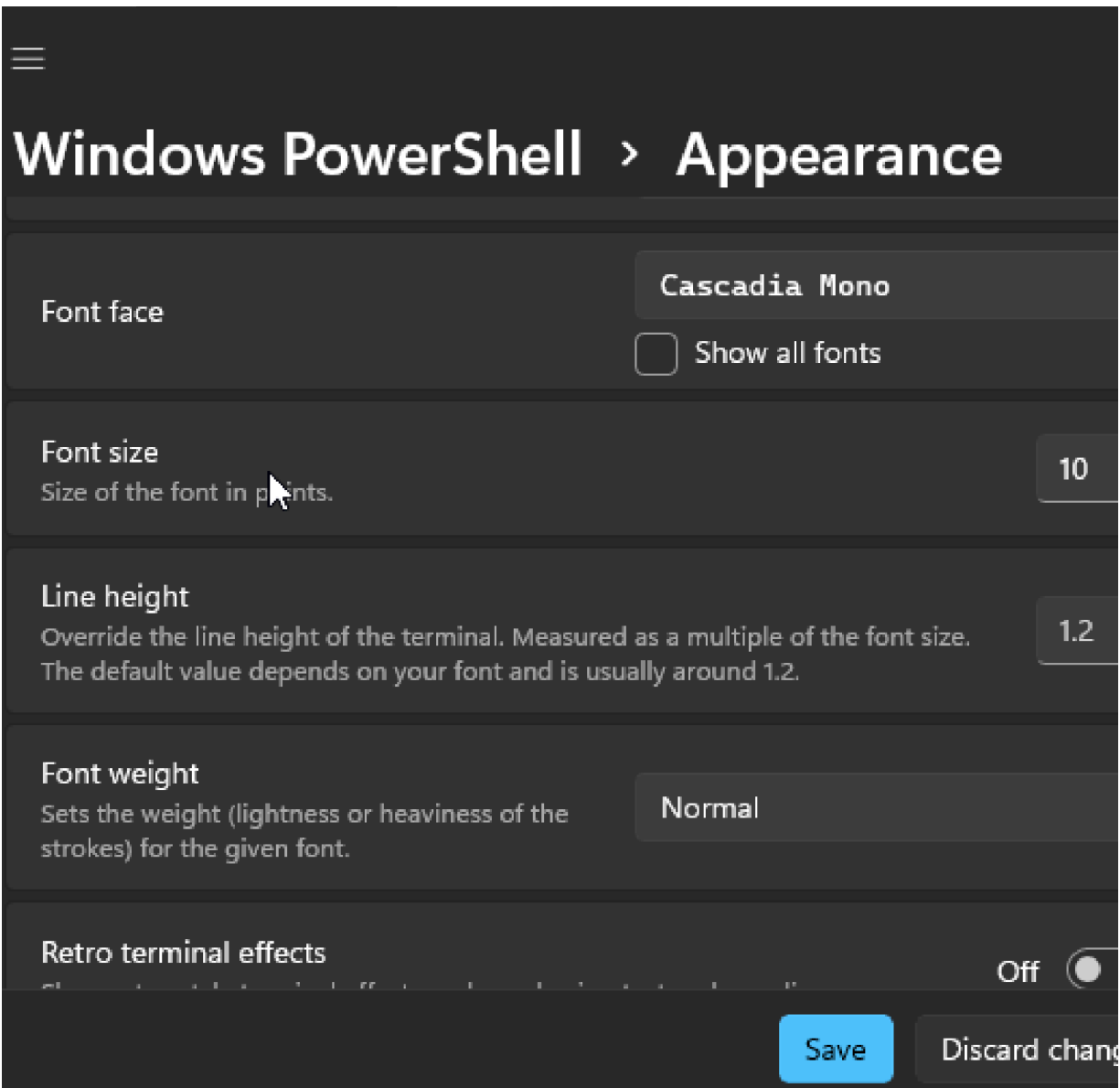
- Selecting the icon in the left corner of the console opens a dialog box

- Can be customized using PowerShell commands

*Note that this will be different on older versions.



In this version of CMD we can edit general settings here. However, we can also specifically go to profiles and click “Windows PowerShell” and edit settings specific to PowerShell.



In here we can edit:

- The Cursor
- Buffer
- Font Face
- Font Size
- Font Weight

- Screen Colors
- Font Colors
- Screen opacity

And more.

Customize From the CLI

```
-----  
-----
```

You can customize PowerShell directly from the CLI.

Using the `$Host` variable

Using the UI internal value: `$Host.UI`

Gets you to the RawUI class: `$Host.UI.RawUI`

```
PS C:\Users\Maximus> $host.ui.RawUI  
  
ForegroundColor      : Gray  
BackgroundColor      : Black  
CursorPosition       : 0,28  
WindowPosition      : 0,0  
CursorSize           : 25  
BufferSize           : 69,29  
WindowSize           : 69,29  
MaxWindowSize        : 69,29  
MaxPhysicalWindowSize : 784,32  
KeyAvailable         : False  
WindowTitle          : Command Prompt - powershell.exe
```

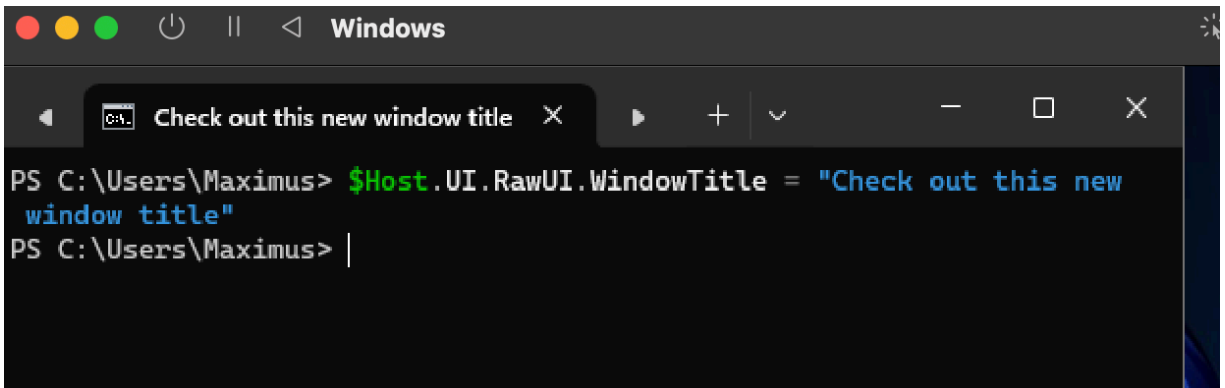
From this, you can see the interactions that are available

ForegroundColor
BackgroundColor
CursorPosition
WindowPosition
CursorSize
BufferSize
WindowSize
MaxWindowSize
MaxPhysicalWindowSize
KeyAvailable
WindowTitle

Using the \$Host

Change the window title:

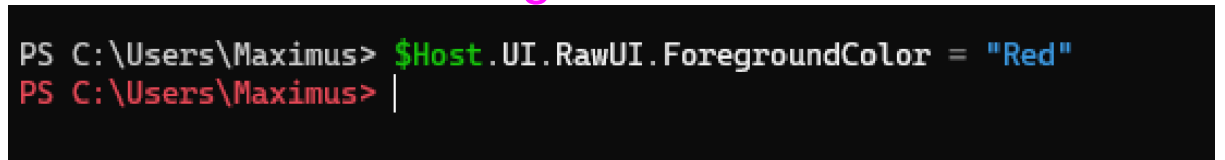
```
$Host.UI.RawUI.WindowTitle = "New Window Title Here"
```



A screenshot of a Windows PowerShell terminal window. The window title bar reads "Windows". The terminal content shows the command `$Host.UI.RawUI.WindowTitle = "Check out this new window title"` being executed. The prompt is `PS C:\Users\Maximus>`. The terminal text is displayed in a light blue color on a dark background.

Change foreground color:

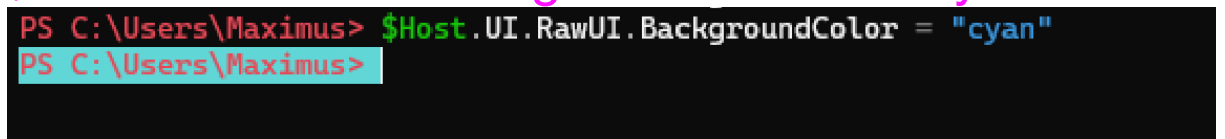
```
$Host.UI.RawUI.ForegroundColor = "Red"
```



A screenshot of a PowerShell terminal window. The terminal content shows the command `$Host.UI.RawUI.ForegroundColor = "Red"` being executed. The prompt is `PS C:\Users\Maximus>`. The terminal text is displayed in a red color on a dark background.

Change background color:

```
$Host.UI.RawUI.BackgroundColor = "Cyan"
```



A screenshot of a PowerShell terminal window. The terminal content shows the command `$Host.UI.RawUI.BackgroundColor = "cyan"` being executed. The prompt is `PS C:\Users\Maximus>`. The terminal text is displayed in a cyan color on a dark background.