To add content to an existing file using PowerShell, you can use the `Add-Content` cmdlet or the `Out-File` cmdlet with the `-Append` parameter. Here are examples of both methods:

1. Using `Add-Content`:

```powershell

Add-Content -Path "C:\Path\To\Your\File.txt" -Value "This is the new content you want to add" -Encoding UTF8

```

Replace `"C:\Path\To\Your\File.txt"` with the path to your existing file and `"This is the new content you want to add"` with the content you want to append to the file.

2. Using `Out-File` with the `-Append` parameter:

```powershell

"New content to append" | Out-File -Append -FilePath "C:\Path\To\Your\File.txt" -Encoding UTF8

```

Again, replace `"C:\Path\To\Your\File.txt"` with the path to your existing file and `"New content to append"` with the content you want to add.

Both of these methods will append the specified content to the end of the existing file. Make sure to specify the correct file path and encoding if needed, as shown in the examples.

In PowerShell, there are several cmdlets and methods for sending output to the console, including `Write-Host` and `Write-Output`. However, they serve different purposes and have different behaviors.

1. `Write-Host`:

- `Write-Host` is primarily used for displaying information directly to the console or host. It sends the specified text or data to the host (console) and doesn't return any objects.

- The output generated by `Write-Host` is not typically captured by other cmdlets or processes because it's sent directly to the host and not included in the pipeline.

- It's often used when you want to display messages, progress information, or user prompts in a script or interact with users via the console.

Example of `Write-Host`:

```powershell

Write-Host "This is a message displayed directly to the console."

```

2. `Write-Output`:

- `Write-Output` is used to send objects down the PowerShell pipeline. It's a more versatile cmdlet as it allows you to pass data from one part of a script to another, making it available for further processing.

- The output generated by `Write-Output` can be captured, filtered, and manipulated by other cmdlets and scripts.

- It's commonly used to output data that needs to be processed further or sent to other cmdlets for additional actions.

Example of `Write-Output`:

```powershell

$myVariable = "This is some data."

Write-Output $myVariable

```

In summary, the main difference between `Write-Host` and `Write-Output` is their purpose and behavior:

- Use `Write-Host` to display information directly to the console for user interaction or displaying messages.

- Use `Write-Output` to send objects down the PowerShell pipeline for further processing by other cmdlets or scripts.