

File Renaming Script

Sharjeel Faiq

June 7, 2023

1 Introduction

This document presents a PowerShell script that renames files in a given directory based on a specific pattern or criteria.

2 PowerShell Script

```
1 $directoryPath = "C:\Path\To\Directory"
2 $pattern = "pattern" # Specify the pattern you want to match
3
4 # Get the list of files in the directory
5 $files = Get-ChildItem -Path $directoryPath
6
7 # Loop through each file and rename based on the pattern
8 foreach ($file in $files) {
9     # Check if the file name matches the pattern
10    if ($file.Name -match $pattern) {
11        # Generate the new file name
12        $newFileName = $file.Name -replace $pattern, "replacement"
13
14        # Construct the new file path
15        $newFilePath = Join-Path -Path $directoryPath -ChildPath $newFileName
16
17        # Rename the file
18        Rename-Item -Path $file.FullName -NewName $newFileName -Force
19    }
20 }
```

3 Explanation

The provided PowerShell script performs the following steps:

1. Assigns the target directory path and the pattern to match to variables.

2. Retrieves the list of files in the specified directory using the **Get-ChildItem** cmdlet.
3. Iterates through each file in a loop.
4. Checks if the file name matches the specified pattern using the **-match** operator.
5. Generates a new file name by replacing the matched pattern with the desired replacement value using the **-replace** operator.
6. Constructs the new file path using the **Join-Path** cmdlet, combining the target directory path with the new file name.
7. Renames the file using the **Rename-Item** cmdlet with the new file name and the **-Force** parameter to enforce the renaming operation.

The script allows you to automate the renaming of files in a directory based on a specified pattern, providing flexibility and efficiency.