

Building Advanced Regular Expression Commands with the Regex Object



Jeff Hicks

AUTHOR/TEACHER/SENSEI

@jeffhicks | <https://jdhitsolutions.com>





PowerShell uses
`System.Text.RegularExpressions`

Use the `[regex]` type accelerator

Use the same regular expression pattern

Used for more complex regular expression
operations



[Regex]

```
PS C:\> [regex]$rx = "^[a-zA-Z]+-\d{1,3}$"
```

```
PS C:\> $rx | Get-Member
```



[Regex]

TypeName: System.Text.RegularExpressions.Regex

Name	MemberType	Definition
----	-----	-----
Equals	Method	bool Equals(System.Object obj)
GetGroupNames	Method	string[] GetGroupNames()
GetGroupNumbers	Method	int[] GetGroupNumbers()
GetHashCode	Method	int GetHashCode()
GetObjectData	Method	void Serializable.GetObjectData(System.Runtime...
GetType	Method	type GetType()
GroupNameFromNumber	Method	string GroupNameFromNumber(int i)
GroupNumberFromName	Method	int GroupNumberFromName(string name)
IsMatch	Method	bool IsMatch(string input), bool IsMatch(str...
Match	Method	System.Text.RegularExpressions.Match Match(...
Matches	Method	System.Text.RegularExpressions.MatchCollecti...
Replace	Method	string Replace(string input, string replace...
Split	Method	string[] Split(string input), string[] Spl...
...		



[Regex]

```
PS C:\> [regex]$rx = "^[a-zA-Z]+-\d{1,3}$"  
PS C:\> $rx | Get-Member  
PS C:\> $rx.IsMatch("PHI-101")  
True
```



Demo



[Regex] In Action



Summary



Use when you need more than simple matching

Use for more complex splitting and replacing

Will be useful with even more complex regular expressions techniques in the next module

