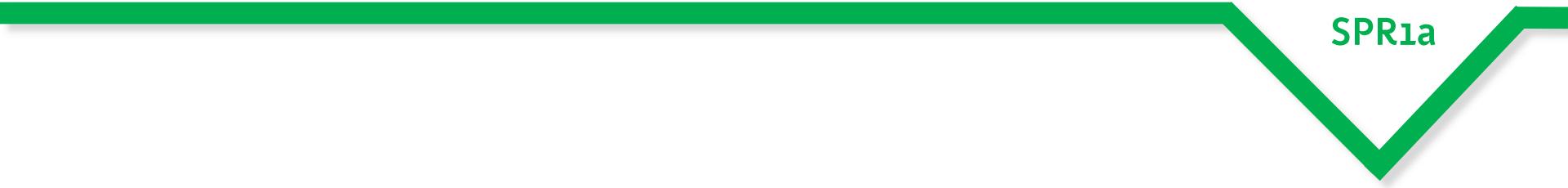


Systémové programovanie /

ÚINF/SPR1a



SPR1a

Róbert Novotný
robert.novotny@upjs.sk

5. 10. 2010

Skriptovacie finty

SPR1a

- PowerShell umožňuje mnohoraké finty uľahčujúce skriptovanie
- hlavne:
 - aliasy
 - skracovanie názvov parametrov

Názvy cmdletov

- štandardný názov cmdletu:

*sloveso-vec
prísudok-predmet*

- Get-ChildItem
- Set-Date
- Copy-Item

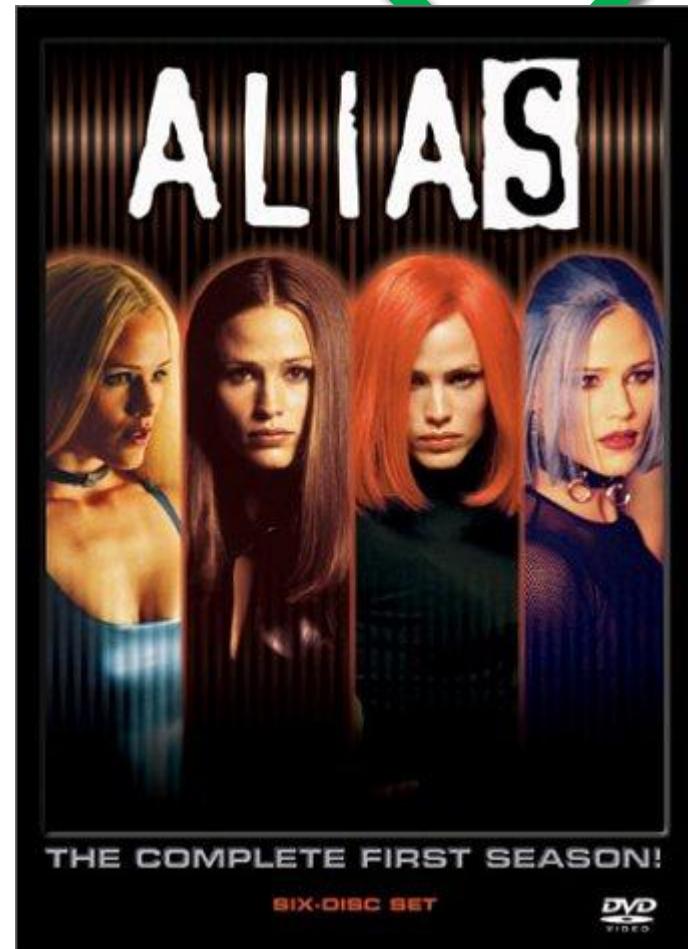
Get-Command



Aliases

SPR1a

- cmdlet možno volať pod rozličnými názvami
- aliasy šetria písanie
- uľahčujú kultúrny šok pri prechode z iných systémov
- zneprehľadňujú kód



Aliasy

SPR1a

Definícia	Alias
Get-ChildItem	ls, dir, gci
Get-Command	gcm
Copy-Item	cp, copy, cpi
Where-Object	where, ?
ForEach-Object	foreach, %
Get-Command	gcm
Get-Alias	alias, gal

Skracovanie názvov parametrov

SPR1a

- názvy parametrov možno skracovať dovtedy, pokiaľ sú jednoznačné
- v niektorých prípadoch môžeme vyniechať meno pomenovaného parametra

```
Get-ChildItem -Path D:/MP3 -Filter o* -Recursive
```



```
dir d:/MP3 o* -r
```

Problém: zneprehľadňovanie

SPR1a

- Čo robí tento one-liner?

```
dir d:/mp3 | ? {$_._PSIsContainer}  
| select FullName,  
    { (dir $_.FullName -r  
        | measure -s length).Sum }
```

PowerShell a viacriadkové skripty

SPR1a

- PowerShell podporuje plnoprávne programovanie
- vlastná syntax + arzenál .NET frameworku
 - objekty
 - funkcie
 - ...
- dodáva sa PowerShell ISE (Integrated Script Environment)

Vypíšte veľkosti adresárov

SPR1a

- PowerShell podporuje plnoprávne programovanie
- vlastná syntax + arzenál .NET frameworku
 - objekty
 - funkcie
 - ...
- dodáva sa PowerShell ISE (Integrated Script Environment)

Riešenie na viacerých riadkoch

```
function Velkost($adresar) {  
    $meno = $adresar.FullName  
    $statistiky = Get-ChildItem $meno -Recurse  
        | Measure-Object -Property Length -Sum  
    $statistiky.sum  
}  
  
$adresare = Get-ChildItem  
    | Where {$_.PSIsContainer}  
  
foreach($adresar in $adresare) {  
    $velkost = Velkost($adresar)  
    Write-Host $adresar.FullName $velkost  
}
```

Poznámky k funkciám

SPR1a

- funkcie sa vedia tváriť ako cmdlety
 - pozor na syntax volania!

```
function Get-Sum($a, $b) {  
    return $a + $b  
}
```

```
Get-Sum(3, 5)
```

3
5

Get-Sum 3 5

Parametre funkcií

SPR1a

```
function Get-Sequence([int] $from, [int] $to,  
                      [int] $step = 1)  
{  
    for($i = $from; $i -le $to; $i = $i + $step) {  
        $i  
    }  
}
```

```
Get-Sequence 0 20 -step 2
```

Funkcie, ktoré spracovávajú pipeliny

SPR1a

- špeciálna premenná **\$input** s objektami, ktoré prídu z rúry

```
function Measure-Directory {  
    foreach($item in $input) {  
        $size = (dir $item.FullName -Recurse  
            | Measure-Object -Sum -Property Length).Sum  
  
        @{$item.FullName = $size}  
    }  
}
```

```
dir D:\MP3 | Measure-Directory
```

Príklad: očíslujte položky z rúry

SPR1a

```
function Enumerate-Items {  
    $itemIndex = 0;  
  
    foreach($item in $input) {  
        @{$itemIndex = $item}  
        $itemIndex++  
    }  
}  
  
dir | Enumerate-Items
```

Príklad: očíslujte položky z rúry II.

SPR1a

- alternatívna syntax funkcií nad rúrou
 - **begin**: vykoná sa na začiatku rúry
 - **process**: pre každú položku v rúre.
 - aktuálna položka je v premennej `$_`
 - **end**: na konci rúry

```
function Enumerate-Items {  
    begin {  
        $itemIndex = 0;  
    }  
  
    process {  
        @{$itemIndex = $_}  
        $itemIndex++  
    }  
}
```

Príklad: očíslujte položky z rúry III.

- alternatívna syntax funkcií nad rúrou

SPR1a

```
filter Get-CpuHeavyProcess {  
    if($_.PercentProcessorTime -ge 2) {  
        $obj = New-Object PSObject | Select Name, CPU  
        $obj.Name = $_.Name;  
        $obj.CPU = $_.PercentProcessorTime;  
  
        $obj  
    }  
}  
  
gwmi Win32_PerfFormattedData_PerfProc_Process  
| Get-CpuHeavyProcess
```

Príklad: očíslujte položky z rúry III.

SPR1a

- filter je sémanticky ekvivalentný funkcie s jediným blokom **process**

```
filter Get-FilteredObject  
{  
    ...  
}
```

```
function Get-FilteredObject  
{  
    process {  
    }  
}
```

Aktuálna položka: `$_.`

Aktuálna položka:
`$input.current`