درس ۹

PS>

try / catch

script.ps1

```
try {
  Get-Item x -EA Stop
} catch {
  "Error: $_"
}
```

همه را یاد بگیرید @MyWebUniversity-Dari

یک کورس رایگان از MyWebUniversity

اسکریپت‌نویسی PowerShell

درس ۹

مدیریت خطا

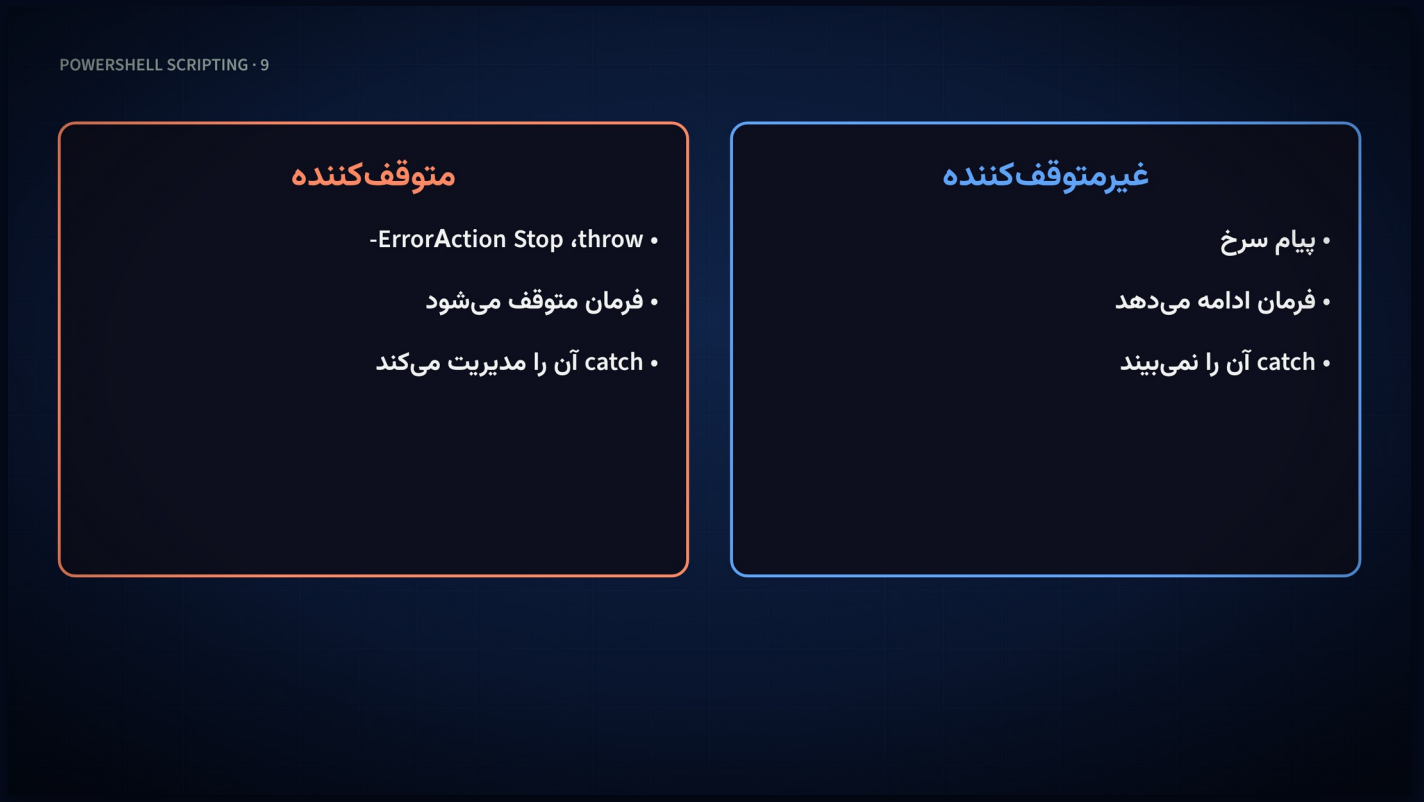
خطاهای متوقف‌کننده و غیرمتوقف‌کننده، try/catch/finally، ErrorAction- و $Error



دو نوع خطا

خطاهای غیرمتوقف‌کننده یک پیام سرخ چاپ می‌کنند اما فرمان ادامه می‌دهد، مثل Get-ChildItem روی ...

خطاهای متوقف‌کننده فرمان را متوقف می‌کنند. try و catch تنها خطاهای متوقف‌کننده را می‌بینند، ...



01

finally و try، catch

کود خطرناک را در try بگذارید. -ErrorAction Stop را اضافه کنید تا خطاهای عادی متوقف‌کننده ...

داخل catch، $_ خطا را نگه می‌دارد. finally همیشه اجرا می‌شود، پس اتصال‌ها و فایل‌ها را آنجا ...



02

-ErrorAction و ترجیح‌ها

-ErrorAction یک فرمان را کنترول می‌کند:
Stop، Continue، SilentlyContinue یا
... Ignore.

بسیاری از حرفه‌ای‌ها
$ErrorActionPreference = 'Stop' را در بالا
می‌گذارند، مثل set -e در Bash.

مقدار	اثر
Stop	متوقف‌کننده (قابل گرفتن)
Continue	خطا را نشان بده، ادامه بده (پیش‌فرض)
SilentlyContinue	خطا را پنهان کن، ادامه بده
Ignore	پنهان و ثبت نکن

03

throw و خطاهای خودتان

throw یک خطای متوقف‌کننده با پیام شما می‌سازد. وقتی ورودی نادرست است یا یک منبع ضروری موجود ...

Write-Warning و Write-Error نرم‌تر هستند: یک مشکل را گزارش می‌دهند اما به صدا کننده اجازه ...

throw.ps1

```powershell
function Get-Config {
    param([string]$Path)
    if (-not (Test-Path $Path)) { throw "Config not found: $Path" }
    Get-Content $Path -Raw | ConvertFrom-Json
}
try { Get-Config .\missing.json } catch { "Error: $_" }
```

04

$Error و لاگ کردن

متغیر خودکار $Error فهرستی از خطاهای اخیر را، جدیدترین اول، نگه می‌دارد. $Error[0] آخرین آن ...

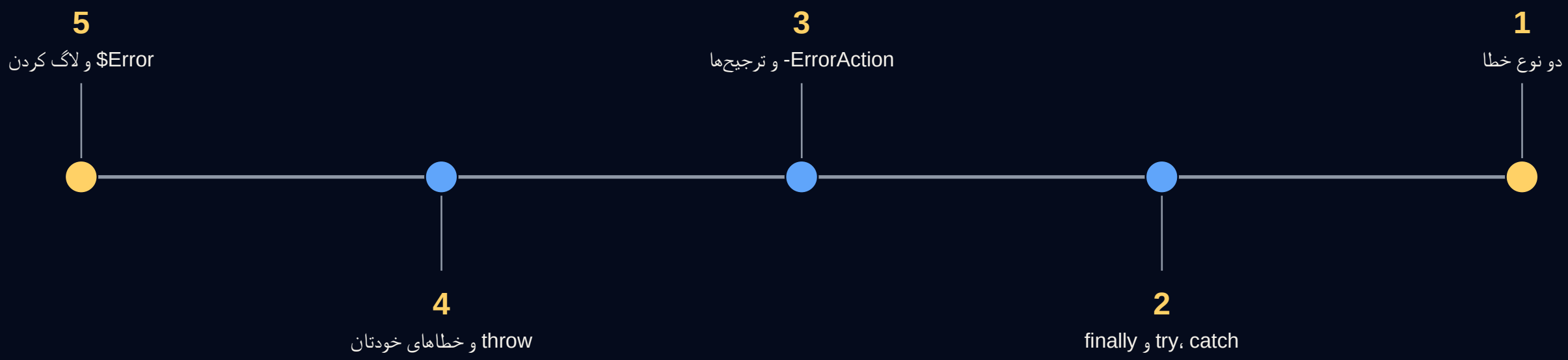
catch را با یک تابع log ترکیب کنید، و Start-Transcript را برای ثبت تمام session، تا ببینید ...

log.ps1

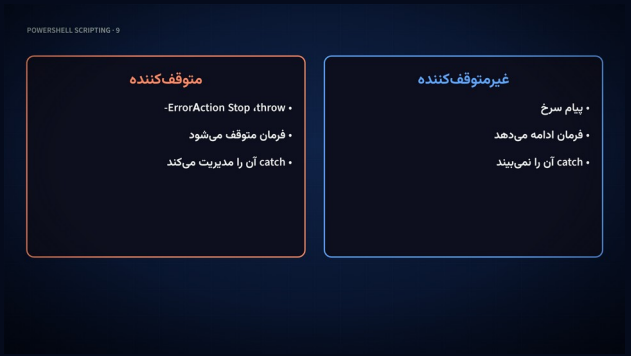
```
Start-Transcript -Path .\run.log -Append
function Write-Log($msg) { "$(Get-Date -Format s) $msg" | Add-Content .\app.log }
try { Restart-Service 'Spooler' -ErrorAction Stop; Write-Log 'restarted' }
catch { Write-Log "FAILED: $($_.Exception.Message)" }
$Error[0].InvocationInfo.ScriptLineNumber
Stop-Transcript
```

05

نقشهٔ درس


1
دو نوع خطا
2
try، catch و finally
3
ErrorAction- و ترجیح‌ها
4
throw و خطاهای خودتان
5
$Error و لاگ کردن

منابع و مطالعهٔ بیشتر





تشکر از تماشای شما

MyWebUniversity.com

آموزش رایگان تکنالوژی از سال ۲۰۰۰ — انگلیسی و دری

@MyWebUniversity-Dari · @MyWebUniversity