درس ۴ از ۱۲

Windows 10 & 11

Desktop · Workstation · Laptop

خط فرمان: CMD، PowerShell، Windows Terminal و NTFS

دوره کامل · Windows 10 22H2 · Windows 11 26H2



در درس چهارم چه یاد می‌گیرید

- shell ها: PowerShell 7، Windows PowerShell 5.1، CMD و WSL
- Windows Terminal: تب‌ها، profile ها، جلسات admin و sudo
- CMD در برابر PowerShell در برابر لینوکس: همان کارها کنار هم
- طرز فکر PowerShell: Verb-Noun، object ها و pipeline
- مسیرها، درایوها و سیستم فایل NTFS
- کار با فایل‌ها در CMD و PowerShell، از جمله robocopy
- اولین batch file و اسکریپت PowerShell شما

shell ها در ویندوز

Shell	برنامه	نکات
Command Prompt	cmd.exe	کلاسیک؛ batch file ها؛ هنوز همه جا
Windows PowerShell 5.1	powershell.exe	داخلی؛ در 5.1 متوقف شده
PowerShell 7.6 (LTS)	pwsh.exe	مدرن، متن‌باز، روی .NET 10؛ ویندوز، macOS، لینوکس
WSL (bash)	wsl.exe	یک shell و توزیع واقعی لینوکس (درس ۶)
Windows Terminal	wt.exe	پنجره‌ای که میزبان همه است؛ پیش‌فرض در ویندوز ۱۱

نصب PowerShell 7 و بررسی نسخه‌ها

```
PS C:\> winget install --id Microsoft.PowerShell --source winget
PS C:\> pwsh                                    # start PowerShell 7
PS C:\> $PSVersionTable.PSVersion
Major  Minor  Patch
7      6      4

C:\> ver                                        # CMD: Windows version
Microsoft Windows [Version 10.0.26300.xxxx]
```

شماره نسخه‌ها با آپدیت‌ها تغییر می‌کند — نسخه شما ممکن است جدیدتر باشد.

یک کار، سه shell

Linux bash	PowerShell	CMD	کار
ls -l	Get-ChildItem (ls, dir)	dir	لست فایل‌ها
cd	Set-Location (cd)	cd	تغییر فولدر
cp / mv	Copy-Item / Move-Item	copy / move	کاپی / انتقال
rm, rmdir	Remove-Item (rm)	del, rmdir	حذف
cat	Get-Content (cat)	type	نمایش فایل
grep	Select-String	findstr	جستجوی متن
clear	Clear-Host (cls)	cls	پاک کردن صفحه

طرز فکر PowerShell

- فرمان‌ها cmdlet های Verb-Noun هستند: New-Item، Stop-Process، Get-Service
- خروجی object هایی با property است، نه فقط متن
- pipeline یعنی | که object ها را به فرمان بعدی می‌دهد
- Get-Command فرمان‌ها را پیدا می‌کند؛ Get-Help آن‌ها را توضیح می‌دهد (-Examples)
- Get-Member property ها و method های هر object را نشان می‌دهد
- WhatIf- یک تغییر را پیش‌نمایش می‌کند؛ Confirm- اول سوال می‌کند

pipeline در عمل

```
PS C:\> Get-Process | Sort-Object CPU -Descending | Select-Object -First 5 Name, CPU, Id

PS C:\> Get-Service | Where-Object Status -eq 'Running' | Measure-Object

PS C:\> Get-ChildItem C:\Windows -File | Measure-Object Length -Sum

PS C:\> Get-Command *service*              # find commands
PS C:\> Get-Help Get-Service -Examples     # learn by example
PS C:\> Get-Service WinRM | Get-Member     # what is inside the object?
```

مسیرها، درایوها و NTFS

چه چیزی بدانید	موضوع
C: ویندوز است؛ D:، E: برای دیسک‌های دیگر، USB، DVD	حرف درایو
با بک‌اسلش؛ مسیر شبکه از نوع UNC	مسیرها
حساس نیست، اما حروف را همان طور نگه می‌دارد	حروف بزرگ و کوچک
اجازه‌ها (ACL)، journaling، فشرده‌سازی، hard link، EFS	ویژگی‌های NTFS
ReFS (ورک‌استیشن، سرور)، FAT32 و exFAT (فلش USB)	سیستم فایل‌های دیگر
نام‌های رزرو (CON، PRN، NUL)؛ محدودیت ۲۶۰ حرفی مسیر به طور پیش‌فرض	دام‌ها

فایل‌ها در CMD (از جمله robocopy)

```
C:\> mkdir C:\Lab\Reports & cd /d C:\Lab
C:\Lab> echo Hello Windows > notes.txt
C:\Lab> copy notes.txt Reports\           & dir /a Reports
C:\Lab> type notes.txt                     & attrib +h notes.txt
C:\Lab> tree /f C:\Lab

C:\> robocopy C:\Lab D:\Backup\Lab /E /R:1 /W:1 /LOG:C:\robocopy.log
#  /E copy subfolders   /MIR mirror (DELETES extra files at the destination!)
```

فایل‌ها در PowerShell

```
PS C:\> New-Item -ItemType Directory C:\Lab\Scripts
PS C:\> Set-Content C:\Lab\todo.txt 'Learn PowerShell'
PS C:\> Add-Content C:\Lab\todo.txt 'Practice every day'
PS C:\> Get-Content C:\Lab\todo.txt
PS C:\> Copy-Item C:\Lab C:\Lab2 -Recurse
PS C:\> Remove-Item C:\Lab2 -Recurse -WhatIf      # preview first!
PS C:\> Test-Path C:\Lab\todo.txt
PS C:\> Get-Content C:\Windows\Logs\CBS\CBS.log -Tail 20 -Wait   # follow a log
```

اولین اسکریپت‌های شما

```
:: hello.bat  — a batch file
@echo off
echo Hello %USERNAME%, today is %DATE%

# hello.ps1  — a PowerShell script
param([string]$Name = $env:USERNAME)
"Hello $Name, you have $((Get-Process).Count) processes running."

PS C:\Lab> Get-ExecutionPolicy                    # RemoteSigned on Windows 11
PS C:\Lab> .\hello.ps1 -Name Student              # dot-backslash runs it
```

Windows Terminal مانند یک متخصص

- Profile ها: PowerShell 7، Windows PowerShell، CMD، WSL، Azure Cloud Shell
- Settings > Startup: profile پیش‌فرض را انتخاب کنید (PowerShell 7)
- Ctrl + Shift + T تب جدید؛ Alt + Shift + D تقسیم پنجره
- کلیک راست روی یک profile > Run as administrator
- sudo برای ویندوز: Settings > System > Advanced > Enable sudo
- اسکریپت profile شما: notepad $PROFILE (alias ها، module، prompt ها)

خلاصه و کارخانگی

- CMD برای کارهای کلاسیک؛ PowerShell 7 برای مدیریت مدرن
- PowerShell = Verb-Noun + object؛ pipeline + ها؛ از Get-Help و WhatIf- استفاده کنید
- NTFS، مسیرها و UNC را بشناسید؛ robocopy برای کاپی مطمئن
- اسکریپت‌ها: .bat برای CMD، .ps1 برای PowerShell (execution policy)
- کارخانگی: PowerShell 7 را نصب کنید و پیش‌فرض Terminal بسازید
- کارخانگی: hello.ps1 را بنویسید و ۵ پروسه پرمصرف CPU را لست کنید

تشکر از شما! 🤲

ویدیوها و PDF های رایگان: www.MyWebUniversity.com

سلایدهای این درس: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity-Dari · @MyWebUniversity

وحیدالله لطفی – مؤسس و رئیس IT مای وب یونیورسیتی، و استاد علوم کامپیوتر

یاد بگیرید و به دیگران انتقال دهید