

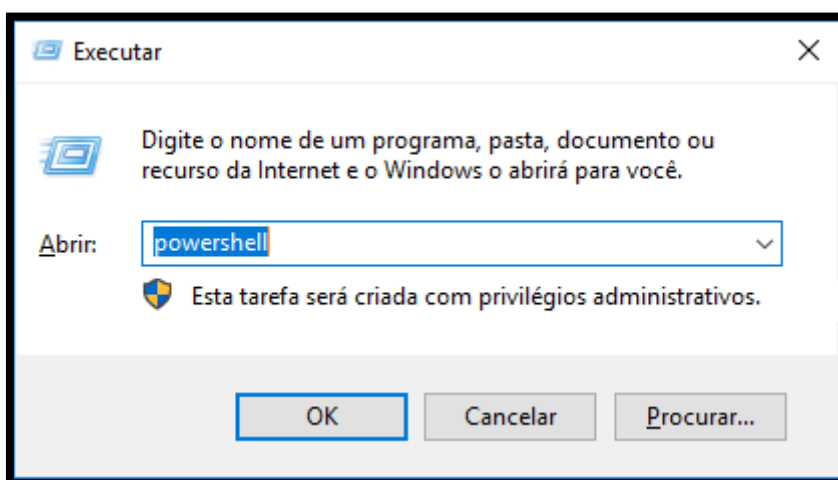
2º Laboratório - PowerShell

Veremos agora como criar uma lista de todos os programas que estão instalados em sua máquina, de forma bem fácil e prática usando comandos no PowerShell.

Start

Vamos a alguns comandos do WPS usando o Power Shell

1. Pressione “Win+R” para abrir o Executar, digite **powershell** e pressione “Enter” ou clique em “OK”.



2. Copie o comando abaixo e cole na janela do PowerShell aberta;



```
Get-ItemProperty HKLM:\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall\* | Select-Object DisplayName, DisplayVersion | Sort-Object -Property DisplayName -Unique | Format-Table -AutoSize
```

Então esse comando mostra todos os programas instalado na máquina local.

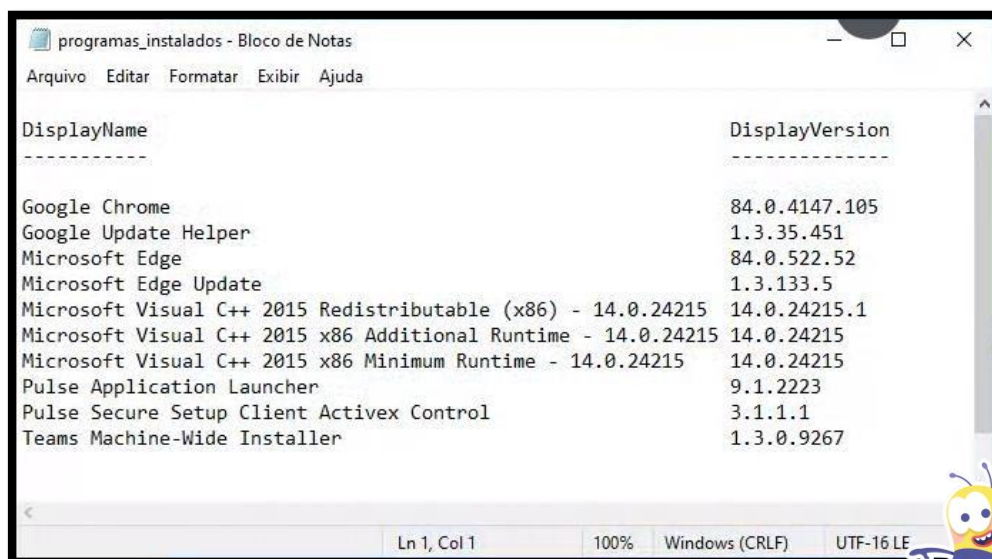
DisplayName	DisplayVersion
-----	-----
Google Chrome	84.0.4147.105
Google Update Helper	1.3.35.451
Microsoft Edge	84.0.522.52
Microsoft Edge Update	1.3.133.5
Microsoft Visual C++ 2015 Redistributable (x86) - 14.0.24215	14.0.24215.1
Microsoft Visual C++ 2015 x86 Additional Runtime - 14.0.24215	14.0.24215
Microsoft Visual C++ 2015 x86 Minimum Runtime - 14.0.24215	14.0.24215
Pulse Application Launcher	9.1.2223
Pulse Secure Setup Client Activex Control	3.1.1.1
Teams Machine-Wide Installer	1.3.0.9267

3. Agora vamos exportar essas informações.

Para isso, copie e cole o comando abaixo no PowerShell para criar o arquivo "programas_instalados.txt" na área de trabalho.

```
Get-ItemProperty HKLM:\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall\* | Select-Object DisplayName, DisplayVersion | Sort-Object -Property DisplayName -Unique | Format-Table -AutoSize > "C:\Users\Aluno\Desktop\programas_instalados.txt"
```

Então agora temos um arquivo que pode servir como relatório.



DisplayName	DisplayVersion
-----	-----
Google Chrome	84.0.4147.105
Google Update Helper	1.3.35.451
Microsoft Edge	84.0.522.52
Microsoft Edge Update	1.3.133.5
Microsoft Visual C++ 2015 Redistributable (x86) - 14.0.24215	14.0.24215.1
Microsoft Visual C++ 2015 x86 Additional Runtime - 14.0.24215	14.0.24215
Microsoft Visual C++ 2015 x86 Minimum Runtime - 14.0.24215	14.0.24215
Pulse Application Launcher	9.1.2223
Pulse Secure Setup Client Activex Control	3.1.1.1
Teams Machine-Wide Installer	1.3.0.9267

End

Agora vamos fazer de outra formar...

Temos a possibilidade de criar um script semi-automatizado.

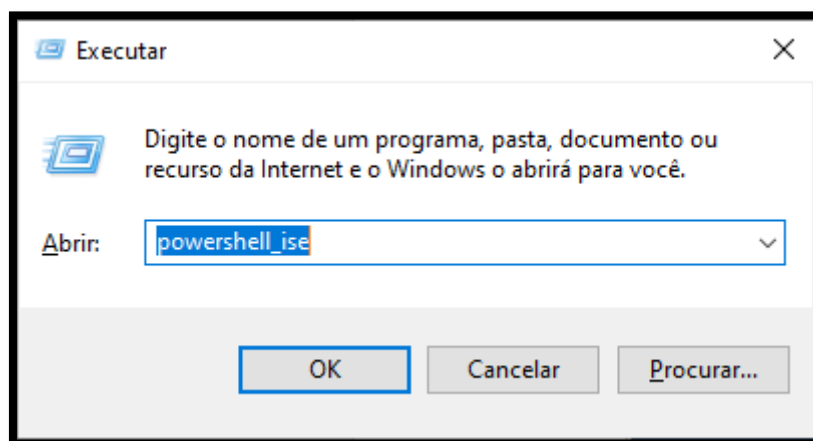
Para isso, vamos utilizar o Power Shell ISE que nos oferece mais recursos.

Start

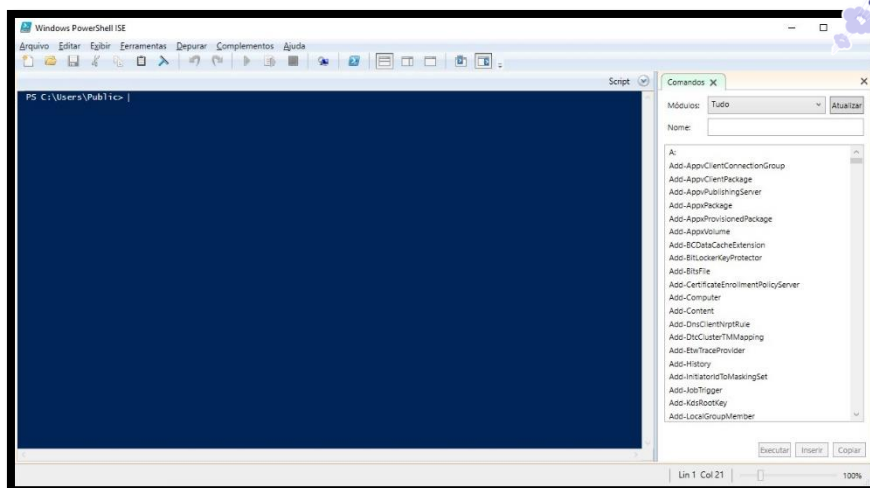
Vamos a alguns comandos do WPS usando o Power Shell ISE

1.

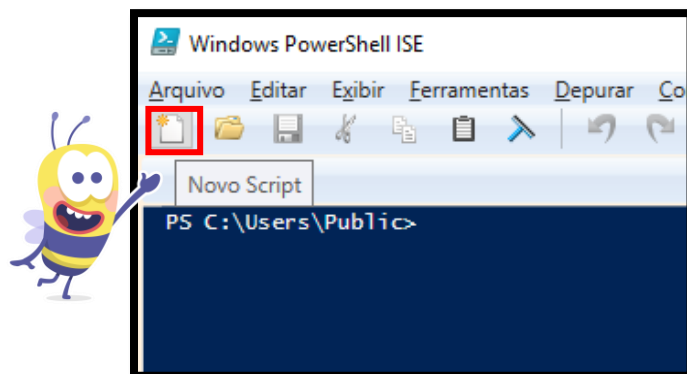
Pressione “Win+R” para abrir o Executar, digite **powershell_ise** e pressione “Enter” ou clique em “OK”.



Logo o Windows PowerShell ISE começa a inicializar.



2. Selecione a opção “Novo Script”.

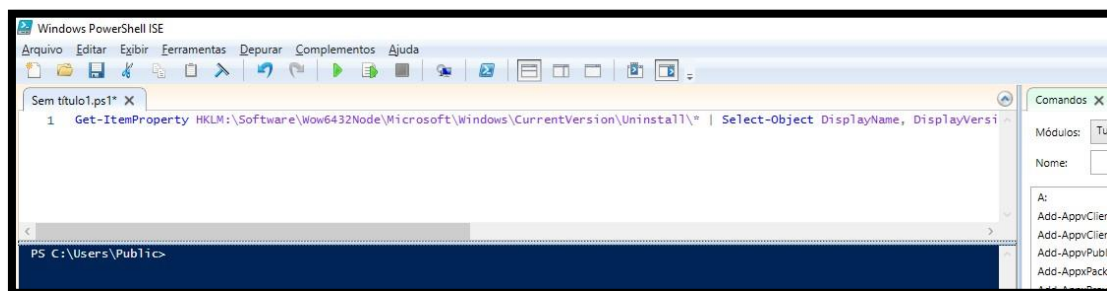


Agora você tem um arquivo do tipo **.ps1**, nele você pode adicionar comandos.

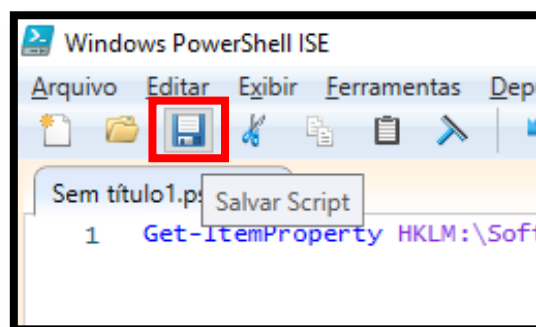
3. Agora vamos criar um script executável semi-automatizado.
Para isso, copie e cole o comando abaixo no PowerShell ISE.

```
Get-ItemProperty HKLM:\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall\* | Select-Object DisplayName,
DisplayVersion | Sort-Object -Property DisplayName -Unique | Format-Table -AutoSize >
"C:\Users\Aluno\Desktop\programas_instalados.txt"
```

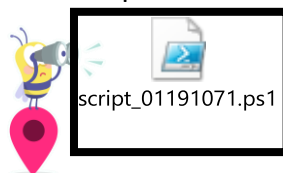
Deve ficar como a imagem abaixo:



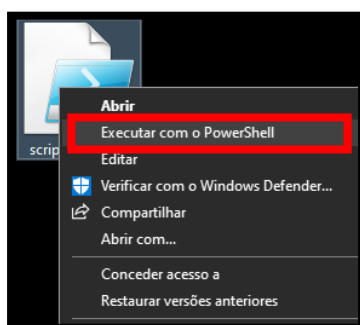
4. Salve esse script na área de trabalho com o nome “**script_seuRA**”.



Você deve encontrar um arquivo como esse na área de trabalho:



5. Clique com o botão direito do mouse sobre esse arquivo.ps1, e selecione **“Executar com o PowerShell”**.



Logo em seguida, **um novo arquivo.txt será criado**, contendo uma lista com o nome e versão dos programas instalados em sua máquina.



End

Descrição

Explicação dos comandos utilizados:

Get-ItemProperty

Obtém as propriedades de um item especificado.

O **Get-ItemProperty** cmdlet obtém as propriedades dos itens especificados. Por exemplo, você pode usar este cmdlet para obter o valor da propriedade LastAccessTime de um objeto de arquivo. Você também pode usar este cmdlet para exibir as entradas do registro e seus valores.

Select-Object

Seleciona objetos ou propriedades do objeto.

O **Select-Object** cmdlet seleciona propriedades especificadas de um objeto ou conjunto de objetos. Também pode selecionar objetos exclusivos, um número especificado de objetos ou objetos em uma posição especificada em uma matriz.

DisplayName, captura o nome do objeto.

DisplayVersion, captura a versão do objeto.

Sort-Object

Classifica objetos por valores de propriedade.

O **Sort-Object** cmdlet classifica objetos em ordem crescente ou decrescente com base nos valores de propriedade do objeto.

ATENÇÃO ERROS

COMANDO

```
Get-ItemProperty HKLM:\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall\* | Select-Object DisplayName,
DisplayVersion | Sort-Object -Property DisplayName -Unique | Format-Table -AutoSize >
"C:\Users\Aluno\Desktop\programas_instalados.txt"
```

ERRO

```
PS C:\Users\Guilherme\Desktop> C:\Users\Guilherme\Desktop\script2.ps1
```

File C:\Users\Guilherme\Desktop\script2.ps1 cannot be loaded. The file
C:\Users\Guilherme\Desktop\script2.ps1 is not

digitally signed. You cannot run this script on the current system. For more information
about running scripts and

setting execution policy, see about_Execution_Policies at
<https://go.microsoft.com/fwlink/?LinkID=135170>.

```
+ CategoryInfo          : SecurityError: (:) [], ParentContainsErrorRecordException  
+ FullyQualifiedErrorId : UnauthorizedAccess
```

RESOLUÇÃO

1. Abrir PowerShell ISE como administrador
2. No prompt de comando digite "cd .." e aperte Enter.
3. No prompt de comando digitar o código "set-executionpolicy remotesigned" e
apertar Enter.
4. Depois executar, vai aparecer uma tela e você deve clicar em "Sim para todos" ou
"Yes to all".
5. Adicionar o comando e apertar Enter:

```
Get-ItemProperty
```

```
HKLM:\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall\* |  
Select-Object DisplayName, DisplayVersion | Sort-Object -Property DisplayName-  
Unique | Format-Table -AutoSize >
```

```
"$env:userprofile\desktop\programas_instalados.txt"
```

6. Ir no local onde salvou o arquivo. Na hora de abrir o arquivo clique com o botão
direito do mouse e selecione "executar com PowerShell" ou "Run with PowerShell".
7. Pronto. O arquivo foi criado na Área de Trabalho ou Desktop.

Referências:

<https://docs.microsoft.com/pt-br/powershell/scripting/windows-powershell/starting-windows-powershell?view=powershell-7> – acessado em 09/08/2020.

<https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.utility/get-date?view=powershell-7> – acessado em 09/08/2020.

https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_execution_policies?view=powershell-7