

Cómo depurar scripts en ISE de Windows PowerShell

<https://docs.microsoft.com/es-es/powershell/scripting/windows-powershell/ise/how-to-debug-scripts-in-windows-powershell-ise?view=powershell-7.1&viewFallbackFrom=powershell-6>

Como administrar los puntos de interrupción

Un punto de interrupción es una zona designada en un script que desee que la operación entre en pausa para poder examinar el estado actual de las variables y el entorno en que se ejecuta el script. Cuando un punto de interrupción pausa un script, puede ejecutar comandos en el panel de consola para examinar el estado del script. Puede generar variables o ejecutar otros comandos. Incluso puede modificar el valor de las variables que son visibles para el contexto del script que se está ejecutando. Después de examinar lo que desea ver, puede reanudar la operación del script.

Puede establecer tres tipos de puntos de interrupción en el entorno de depuración de Windows PowerShell:

1. **Punto de interrupción de línea.** El script se pausa cuando se alcanza la línea designada durante la operación del script.
2. **Variable breakpoint.** El script se pausa cuando cambia el valor de la variable designada.
3. **Command breakpoint.** El script se pausa cada vez que el comando designado está a punto de ejecutarse durante la operación del script. Puede incluir parámetros para filtrar aún más el punto de interrupción solo para operación deseada. El comando también puede ser una función que ha creado.

De estos, en el entorno de depuración Windows PowerShell ISE, solo se pueden establecer puntos de interrupción de línea usando el menú o los métodos abreviados de teclado. Los otros dos tipos de puntos de interrupción se pueden establecer, pero debe hacerse desde el panel de consola mediante el cmdlet [Set-PSBreakpoint](#). En esta sección se describe cómo puede realizar la depuración de tareas en Windows PowerShell ISE mediante los menús, cuando están disponibles, y ejecutar una serie más amplia de comandos desde el panel de consola usando scripts.

Para establecer un punto de interrupción

Solo se puede establecer un punto de interrupción en un script después de guardarlo. Haga clic con el botón derecho en la línea donde desee establecer un punto de interrupción y después haga clic en **Alternar punto de interrupción**. O bien, haga clic en la línea donde desee establecer un punto de interrupción y presione F9 o, en el menú **Depurar**, haga clic en **Alternar punto de interrupción**.

El script siguiente es un ejemplo de cómo establecer un punto de interrupción de variable desde el panel de consola mediante el cmdlet [Set-PSBreakpoint](#).PowerShellCopiar

```
# This command sets a breakpoint on the Server variable in the
Sample.ps1 script.
Set-PSBreakpoint -Script sample.ps1 -Variable Server
```

Enumerar todos los puntos de interrupción

Muestra todos los puntos de interrupción de la sesión actual de Windows PowerShell.

En el menú **Depurar**, haga clic en **Mostrar puntos de interrupción**. El script siguiente es un ejemplo de cómo enumerar todos los puntos de interrupción desde el panel de consola mediante el cmdlet [Get-PSBreakpoint](#).PowerShellCopiar

```
# This command lists all breakpoints in the current session.
```

Get-PSBreakpoint

Quitar un punto de interrupción

Al quitar un punto de interrupción, este se elimina.

Si piensa que podría querer usarlo más adelante, considere la posibilidad de [deshabilitar un punto de interrupción](#) en su lugar. Haga clic con el botón derecho en la línea donde desee quitar un punto de interrupción y después haga clic en **ToggleBreakpoint**. O bien, haga clic en la línea donde desee quitar un punto de interrupción y, en el menú **Depurar**, haga clic en **Alternar punto de interrupción**. El script siguiente es un ejemplo de cómo quitar un punto de interrupción con un identificador especificado en el panel de consola mediante el cmdlet [Remove-PSBreakpoint](#).PowerShellCopiar

```
# This command deletes the breakpoint with breakpoint ID 2.  
Remove-PSBreakpoint -Id 2
```

Quitar todos los puntos de interrupción

Para quitar todos los puntos de interrupción definidos en la sesión actual, en el menú **Depurar**, haga clic en **Quitar todos los puntos de interrupción**.

El script siguiente es un ejemplo de cómo quitar todos los puntos de interrupción del panel de consola mediante el cmdlet [Remove-PSBreakpoint](#).PowerShellCopiar

```
# This command deletes all of the breakpoints in the current session.  
Get-PSBreakpoint | Remove-PSBreakpoint
```

Deshabilitar un punto de interrupción

Al deshabilitar un punto de interrupción, este no se quita; permanece desactivado hasta que se vuelve a habilitar. Para deshabilitar un punto de interrupción de línea específico, haga clic con el botón derecho en la línea donde desee

deshabilitar un punto de interrupción y luego haga clic en **Deshabilitar punto de interrupción**. O bien, haga clic en la línea donde desee deshabilitar un punto de interrupción y presione F9 o, en el menú **Depurar** , haga clic en **Deshabilitar punto de interrupción**. El script siguiente es un ejemplo de cómo quitar un punto de interrupción con un identificador especificado del panel de consola mediante el cmdlet [Disable-PSBreakpoint](#).PowerShellCopiar

```
# This command disables the breakpoint with breakpoint ID 0.  
Disable-PSBreakpoint -Id 0
```

Deshabilitar todos los puntos de interrupción

Al deshabilitar un punto de interrupción, este no se quita; permanece desactivado hasta que se vuelve a habilitar. Para deshabilitar todos los puntos de interrupción de la sesión actual, en el menú **Depurar** , haga clic en **Deshabilitar todos los puntos de interrupción**. El script siguiente es un ejemplo de cómo deshabilitar todos los puntos de interrupción desde el panel de consola mediante el cmdlet [Disable-PSBreakpoint](#).PowerShellCopiar

```
# This command disables all breakpoints in the current session.  
# You can abbreviate this command as: "gbp | dbp".  
Get-PSBreakpoint | Disable-PSBreakpoint
```

Habilitar un punto de interrupción

Para habilitar un punto de interrupción específico, haga clic con el botón derecho en la línea donde desee habilitar un punto de interrupción y después haga clic en **Habilitar punto de interrupción**. O bien, haga clic en la línea donde desee habilitar un punto de interrupción y presione F9 o, en el menú **Depurar** , haga clic en **Habilitar punto de interrupción**. El script siguiente es un ejemplo de cómo

habilitar puntos de interrupción desde el panel de consola mediante el cmdlet [Enable-PSBreakpoint](#).PowerShellCopiar

```
# This command enables breakpoints with breakpoint IDs 0, 1, and 5.
```

```
Enable-PSBreakpoint -Id 0, 1, 5
```

Habilitar todos los puntos de interrupción

Para habilitar todos los puntos de interrupción definidos en la sesión actual, en el menú **Depurar** , haga clic en **Habilitar todos los puntos de interrupción**. El script siguiente es un ejemplo de cómo habilitar todos los puntos de interrupción desde el panel de consola mediante el cmdlet [Enable-PSBreakpoint](#).PowerShellCopiar

```
# This command enables all breakpoints in the current session.  
# You can abbreviate the command by using their aliases: "gbp | ebp".
```

```
Get-PSBreakpoint | Enable-PSBreakpoint
```

Cómo administrar una sesión de depuración

Antes de iniciar la depuración, debe establecer uno o varios puntos de interrupción. No se puede establecer un punto de interrupción si no se guarda el script que desea depurar. Para obtener instrucciones acerca de cómo establecer un punto de interrupción, vea [Cómo administrar los puntos de interrupción](#) o [Set-PSBreakpoint](#). Después de iniciar la depuración, no se puede editar un script hasta que la depuración se detenga. Un script con uno o más puntos de interrupción establecidos se guarda automáticamente antes de ejecutarse.

Para iniciar la depuración

Presione F5 o haga clic en el icono **Ejecutar script** en la barra de herramientas, o bien, en el menú **Depurar** , haga clic en **Ejecutar o continuar**. El script se ejecuta hasta que encuentra el primer punto de interrupción. Detiene la operación en este punto y resalta la línea en la que se produce la pausa.

Para continuar con la depuración

Presione F5 o haga clic en el icono **Ejecutar Script** en la barra de herramientas, o bien, en el menú **Depurar** , haga clic en **Ejecutar o continuar**. También puede escribir C en el panel de consola y presionar ENTRAR. Esto hace que el script se siga ejecutando hasta el punto de interrupción siguiente o hasta el final si no se encuentran más puntos de interrupción.

Para ver la pila de llamadas

La pila de llamadas muestra la ubicación de ejecución actual en el script. Si el script se ejecuta en una función que llamó una función diferente, se representa mediante filas adicionales en la salida. La última fila muestra el script original y la línea en la que se llamó a una función. La siguiente línea muestra esa función y la línea en la que se podría haber llamado a otra función. La primera fila muestra el contexto actual de la línea actual en la que se estableció el punto de interrupción.

Mientras está en pausa, para ver la pila de llamadas actual, presione CTRL+MAYÚS+D o, en el menú **Depurar** , haga clic en **Mostrar pila de llamadas**. También puede escribir K en el panel de consola y presionar ENTRAR.

Para detener la depuración

Presione MAYÚS+F5 o, en el menú **Depurar** , haga clic en **Detener**

el depurador. También puede escribir Q en el panel de consola y presionar ENTRAR.

Cómo depurar paso a paso por procedimientos, por instrucciones y para salir durante la depuración

La ejecución paso a paso es el proceso de ejecutar una instrucción cada vez. Puede detenerse en una línea de código y examinar los valores de las variables y el estado del sistema. En la tabla siguiente se describen las tareas de depuración comunes, como la depuración paso a paso por procedimientos, por instrucciones y para salir.

Tarea de depuración	Descripción	Cómo llevarla a cabo en PowerShell ISE
Depurar paso a paso por instrucciones	Ejecuta la instrucción actual y, luego, se detiene en la instrucción siguiente. Si la instrucción actual es una llamada de función o script, el depurador ejecuta la depuración paso a paso por instrucciones en la función o el script. De lo contrario, se detiene en la siguiente instrucción.	Presione F11 o, en el menú Depurar , haga clic en Paso a paso por instrucciones . También puede escribir S en el panel de consola y presionar ENTRAR.

Tarea de depuración	Descripción	Cómo llevarla a cabo en PowerShell ISE
Depurar paso a paso por procedimientos	Ejecuta la instrucción actual y, luego, se detiene en la instrucción siguiente. Si la instrucción actual es una llamada de función o script, el depurador ejecuta la función o el script completo y se detiene en la siguiente instrucción después de la llamada de función.	Presione F10 o, en el menú Depurar , haga clic en Paso a paso por procedimientos . También puede escribir V en el panel de consola y presionar ENTRAR.
Depurar paso a paso para salir	Sale de la función actual y sube un nivel si la función está anidada. Si se encuentra en el cuerpo principal, el script se ejecuta hasta el final o hasta el punto de interrupción siguiente. Las instrucciones omitidas se ejecutan, pero no se depuran paso a paso.	Presione MAYÚS+F11 o, en el menú Depurar , haga clic en Depurar paso a paso para salir . También puede escribir O en el panel de consola y presionar ENTRAR.

Tarea de depuración	Descripción	Cómo llevarla a cabo en PowerShell ISE
Continuar	Continúa la ejecución hasta el final o hasta el punto de interrupción siguiente. Las funciones omitidas y las invocaciones se ejecutan, pero no se ejecutan paso a paso.	Presione F5 o, en el menú Depurar , haga clic en Ejecutar o continuar . También puede escribir C en el panel de consola y presionar ENTRAR.

Cómo mostrar los valores de variables durante la depuración

Puede mostrar los valores actuales de las variables en el script mientras realiza la depuración paso a paso del código.

Para mostrar los valores de las variables estándar

Utilice uno de los métodos siguientes:

- En el panel de scripts, mantenga el puntero sobre la variable para mostrar su valor como una información sobre herramientas.
- En el panel de consola, escriba el nombre de la variable y presione ENTRAR.

Todos los paneles de ISE están siempre en el mismo ámbito. Por lo tanto, mientras está depurando un script, los comandos que se escriben en el panel de consola se ejecutan en el ámbito del script. Esto le permite usar el panel de consola para

buscar los valores de variables y llamar a funciones que solo se han definido en el script.

Para mostrar los valores de las variables automáticas

Puede usar el método anterior para mostrar el valor de casi todas las variables al depurar un script. Sin embargo, estos métodos no funcionan con las siguientes variables automáticas.

- `$_`
- `$Input`
- `$MyInvocation`
- `$PSBoundParameters`
- `$Args`

Si intenta mostrar el valor de cualquiera de estas variables, obtendrá el valor de esa variable en una canalización interna usada por el depurador, no el valor de la variable en el script. Puede solucionar esto para algunas variables (`$_`, `$Input`, `$MyInvocation`, `$PSBoundParameters` y `$Args`) con el método siguiente:

1. En el script, asigne el valor de la variable automática a una nueva variable.
2. Para mostrar el valor de la nueva variable, mantenga el puntero sobre la nueva variable en el panel de scripts o escriba la nueva variable en el panel de consola.

Por ejemplo, para mostrar el valor de la variable `$MyInvocation`, en el script, asigne el valor a una nueva variable, como `$scriptName` y, a continuación, mantenga el puntero encima o escriba la variable `$scriptName` para mostrar su valor.

```
# In C:\ps-test\MyScript.ps1
```

```
$scriptName = $MyInvocation.MyCommand.Path
```

```
PowerShellCopiar
```

```
# In the Console Pane:
```

```
.\MyScript.ps1
```

```
$scriptName
```