Windows PowerShell Get-Help on Cmdlet 'Set-WinUserLanguageList'

PS:\>Get-HELP Set-WinUserLanguageList -Full

NAME

   Set-WinUserLanguageList

SYNOPSIS

   Sets the language list and associated properties for the current user account.

SYNTAX

                                                         Set-WinUserLanguageList   [-LanguageList]
<System.Collections.Generic.List`1[Microsoft.InternationalSettings.Commands.WinUserLanguage]>   [-Confirm]   [-Force]
[-WhatIf]

   [<CommonParameters>]

DESCRIPTION

   The Set-WinUserLanguageList cmdlet sets the current user language settings. These settings include input method,
spelling setting, text prediction setting, and

   handwriting input mode. For more information, see CultureInfo Class (https://go.microsoft.com/fwlink/?LinkID=242306)
and [Configurable Language and Cultural

   Settings](https://go.microsoft.com/fwlink/?LinkID=242307).

PARAMETERS

   -Confirm [<SwitchParameter>]

      Prompts you for confirmation before running the cmdlet.

      Required?   false

      Position?   named

      Default value   False

      Accept pipeline input?   False

      Accept wildcard characters? false

   -Force [<SwitchParameter>]

      Required?   false

      Position?   named

      Default value   False

      Accept pipeline input?   False

      Accept wildcard characters? false

   -LanguageList <System.Collections.Generic.List`1[Microsoft.InternationalSettings.Commands.WinUserLanguage]>

      Specifies a language list.

      Required?   true

      Position?   0

      Default value   None

      Accept pipeline input?   True (ByValue)

      Accept wildcard characters? false

   -WhatIf [<SwitchParameter>]

      Shows what would happen if the cmdlet runs. The cmdlet is not run.

Required?   false

Position?   named

Default value   False

Accept pipeline input?   False

Accept wildcard characters? false

  <CommonParameters>

    This cmdlet supports the common parameters: Verbose, Debug,

    ErrorAction, ErrorVariable, WarningAction, WarningVariable,

    OutBuffer, PipelineVariable, and OutVariable. For more information, see

    about_CommonParameters (https:/go.microsoft.com/fwlink/?LinkID=113216).

INPUTS

  System.Collections.Generic.List<Microsoft.InternationalSettings.Commands.WinUserLanguage>

      You can pipe a list of WinUserLanguage objects that contain one or more languages and associated properties from the current user account's language list. The

    language object contains the following properties:

      - BCP-47 (READ). A standard language tag that is used to identify languages. For more information, see the Internet Engineering Task Force (IETF) BCP 47 RFC

      (https://go.microsoft.com/fwlink/?LinkID=242207). - Autonym (LP database) (READ). The name of the language in the language itself. - English name (LP database)

      (READ). The name of the language in English. - Localized name (LP database) (READ). The name of the language in the current Windows display language. - Script

      (LP database) (READ). The writing system of the language. - Input methods (READ/WRITE). A list of input method Tablet Input Panel (TIP) strings that are enabled

      for this language. The enabled input methods are listed in the format `Language ID:Keyboard layout ID`. See Default input profiles (input locales) in Windows

              (/windows-hardware/manufacture/desktop/default-input-locales-for-windows-language-packs). - Handwriting recognition input mode (READ/WRITE). This value is either

      0 (freehand) or 1 (write each character separately).

OUTPUTS

NOTES

---- Example 1: Add a language to an existing language list ----

PS C:\> $OldList = Get-WinUserLanguageList

PS C:\> $OldList.Add("fr-FR")

PS C:\> Set-WinUserLanguageList -LanguageList $OldList

This example adds the language French (France) to the language list of the user.

The first command gets the user language list by using the Get-WinUserLanguageList cmdlet. The command stores the result in the $OldList variable.

The second command adds a language to the object in $OldList.

The final command sets the language list of the current user to the revised value of $OldList.

----------- Example 2: Change a value in a language -----------

PS C:\> $UserLanguageList = New-WinUserLanguageList -Language en-US

PS C:\> $UserLanguageList[0].Handwriting = $True

PS C:\> Set-WinUserLanguageList -LanguageList $UserLanguageList

This example changes the handwriting setting for a language in the user language list.

The first command creates a language list for US English that has default values, and then stores it in the $UserLanguageList variable.

The second command modifies the Handwriting property of the first member of the language list in $UserLanguageList.

The final command sets the language list of the current user to the revised value of $UserLanguageList.

RELATED LINKS

Get-WinUserLanguageList

New-WinUserLanguageList