Deleting a file or folder (or other type of object)

The Remove-Item cmdlet does exactly what the name implies: it enables you to get rid of things once and for all. Tired of the file C:\Scripts\Test.txt? Then delete it:
[crayon-6a9d6f7f2541f583779154/]

You can also use wildcard characters to remove multiple items. For example, this command removes all the files in C:\Scripts:
[crayon-6a9d6f7f25425996421500/]

There is one catch, however. Suppose C:\Scripts contains subfolders. In that case, you'll be prompted as to whether or not you really do want to delete everything in the Scripts folder:
[crayon-6a9d6f7f25427060974491/]

Is there a way to bypass this prompt? Yep; just tack the -recurse parameter onto the end of your command:
[crayon-6a9d6f7f25429538187983/]

Here's an interesting variation. Suppose the Scripts folder contains a bunch of files and you'd like to delete them all. Wait: maybe not all, maybe you'd like to leave any .wav files. No problem; just use the -exclude parameter and indicate the file types that should be excluded from the deletion:
[crayon-6a9d6f7f2542b830424808/]

What's that? Now you'd like to remove just .wav and .mp3 files, leaving all other file types alone? All you had to do was ask (and use the -include parameter):

[crayon-6a9d6f7f2542c271904740/]

As you've probably figured out, if you use the -include parameter the cmdlet will act only on those items specified as part of that parameter. (And, yes, you can specify multiple items; just separate items using commas.) By contrast, if you use the -exclude parameter those items will be exempt from the cmdlet actions.
And yes, if you want to get really fancy you can use both -include and -exclude in the same command. What do you suppose will happen when we run this command?
[crayon-6a9d6f7f2542e770430174/]

You got it: all the .txt files (as indicated by the -include parameter) in the C:\Scripts folder will be deleted, except for any files that have the string value test anywhere in the file name (as indicated by the -exclude parameter). Try some different variations, and it will soon make perfect sense.
Incidentally, the Remove-Item cmdlet has a -whatif parameter that doesn't actually remove anything but simply tells you what would happen if you did call Remove-Item. What do you mean that doesn't make any sense? Here, take a look at this command:
[crayon-6a9d6f7f25430572306293/]

If we run this command, none of the .vbs files in the folder C:\Scripts will be removed; however, we will get back information like this, which lets us know which files would be removed if we called Remove-Item without the -whatif parameter:
[crayon-6a9d6f7f25431926740497/]

In addition, you can remove things other than files and folders. For example, this command gets rid of an alias named

show:
[crayon-6a9d6f7f25433729744954/]

Make a note of how you specify the location: alias:\. That’s standard notation for all Windows PowerShell drives: the drive letter followed by a colon followed by a \. For example, to switch to the Windows PowerShell drive for environment variables use this command:
[crayon-6a9d6f7f25434794724256/]