**Part 1**

1.

@echo off

\\ This command is used in batch scripts to prevent each command line in the script from being echoed or displayed as it is executed. It makes the output cleaner and more readable.

------------------------------------------------------------------------------------------------------------------------------------------

2.

SETLOCAL ENABLEEXTENSIONS ENABLEDELAYEDEXPANSION

\\ This command starts the localization of environment changes in a batch file, meaning any changes to environment variables will remain local to the batch file and not affect the global environment. ENABLEEXTENSIONS enables the use of certain extended features in batch files. ENABLEDELAYEDEXPANSION allows for the use of dynamic variable names.

------------------------------------------------------------------------------------------------------------------------------------------

3.

SET REG\_PATH1=HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall

\\ This line sets the variable REG\_PATH1 to the registry path where uninstall information for software is generally stored in the Windows Registry under the HKLM (HKEY\_LOCAL\_MACHINE) hive.

------------------------------------------------------------------------------------------------------------------------------------------

4.

SET REG\_PATH2=HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall

\\ This line sets the variable REG\_PATH2 to the registry path for uninstall information specifically in the Wow6432Node. This is used on 64-bit systems to store information about 32-bit applications.

------------------------------------------------------------------------------------------------------------------------------------------

5.

SET NETSUPPORT\_VERSION=

\\ This line initializes the variable NETSUPPORT\_VERSION to an empty string. This variable is intended to store the version number of NetSupport software if found in the registry.

------------------------------------------------------------------------------------------------------------------------------------------

6.

SET UNINSTALL\_STRING=

\\ This line initializes the variable UNINSTALL\_STRING to an empty string. This variable will be used to store the uninstallation command for NetSupport software if found in the registry.

------------------------------------------------------------------------------------------------------------------------------------------

7.

FOR /F "tokens=\*" %%A IN ('REG QUERY %REG\_PATH1% /s /f "NetSupport" 2^>nul') DO (

\\ This line starts a FOR loop that executes a command to query the Windows Registry. It searches for "NetSupport" in the path stored in REG\_PATH1. The 'tokens=\*' part ensures that the entire line of output is captured. '2^>nul' redirects any error output to null, effectively silencing error messages.

------------------------------------------------------------------------------------------------------------------------------------------

8.

FOR /F "tokens=2,\*" %%B IN ('REG QUERY "%%A" /v "DisplayVersion" 2^>nul') DO SET NETSUPPORT\_VERSION=%%C

\\ Inside the first FOR loop, this line is another FOR loop that queries the registry key found in the previous loop for a value named "DisplayVersion". If found, it sets NETSUPPORT\_VERSION to the value of "DisplayVersion".

------------------------------------------------------------------------------------------------------------------------------------------

9.

FOR /F "tokens=2,\*" %%B IN ('REG QUERY "%%A" /v "UninstallString" 2^>nul') DO SET UNINSTALL\_STRING=%%C

\\ Similar to the previous line, this FOR loop queries the registry key for a value named "UninstallString". If found, it sets UNINSTALL\_STRING to the uninstall command for NetSupport software.

------------------------------------------------------------------------------------------------------------------------------------------

10.

IF NOT DEFINED NETSUPPORT\_VERSION (

\\ This line checks if the NETSUPPORT\_VERSION variable is still undefined, which would indicate that the NetSupport software was not found in the first registry path (REG\_PATH1).

------------------------------------------------------------------------------------------------------------------------------------------

11.

FOR /F "tokens=\*" %%A IN ('REG QUERY %REG\_PATH2% /s /f "NetSupport" 2^>nul') DO (

\\ Similar to line 7, this FOR loop searches for "NetSupport" in the second registry path (REG\_PATH2). This is executed if the software wasn't found in the first path.

------------------------------------------------------------------------------------------------------------------------------------------

12.

FOR /F "tokens=2,\*" %%B IN ('REG QUERY "%%A" /v "DisplayVersion" 2^>nul') DO SET NETSUPPORT\_VERSION=%%C

\\ Inside the IF condition, this line is similar to line 8, but it operates on the results from the second registry path.

------------------------------------------------------------------------------------------------------------------------------------------

13.

FOR /F "tokens=2,\*" %%B IN ('REG QUERY "%%A" /v "UninstallString" 2^>nul') DO SET UNINSTALL\_STRING=%%C

\\ Similar to line 9, this line operates on the results from the second registry path and sets UNINSTALL\_STRING if an UninstallString is found.

**Part 2**

1.

IF DEFINED NETSUPPORT\_VERSION (

\\ This line checks if the NETSUPPORT\_VERSION variable is defined (i.e., if the NetSupport version was found in the registry). If it is defined, the script executes the commands within the parentheses.

------------------------------------------------------------------------------------------------------------------------------------------

2.

ECHO NetSupport version found: %NETSUPPORT\_VERSION%

\\ This line outputs a message to the console displaying the found NetSupport version.

------------------------------------------------------------------------------------------------------------------------------------------

3.

SET /P REMOVE="Would you like to completely remove NetSupport from your system? (Y/N)? "

\\ This line prompts the user with a question and sets the user's input to the variable REMOVE.

------------------------------------------------------------------------------------------------------------------------------------------

4.

IF /I "%REMOVE%"=="Y" OR /I "%REMOVE%"=="YES" (

\\ This line checks if the user's input (stored in REMOVE) is either "Y" or "YES" (case-insensitive), indicating they want to proceed with the uninstallation.

------------------------------------------------------------------------------------------------------------------------------------------

5.

ECHO Checking for running processes that need to be terminated...

\\ Outputs a message indicating that the script is checking for processes that need to be terminated.

------------------------------------------------------------------------------------------------------------------------------------------

6.

CALL T:\TechServices-FP3\Techs\Jared\Scripts\_J\NetSuppRemove\killtasks.bat

\\ This line calls an external batch file named `killtasks.bat` located at a specific path. This file is presumably responsible for terminating related processes.

------------------------------------------------------------------------------------------------------------------------------------------

7.

IF ERRORLEVEL 1 (

\\ Checks if the last command (`CALL`) returned an error level greater than or equal to 1, indicating an error occurred.

------------------------------------------------------------------------------------------------------------------------------------------

8.

ECHO Error occurred in killtasks.bat.

\\ If there was an error, this line outputs a message indicating an error in `killtasks.bat`.

------------------------------------------------------------------------------------------------------------------------------------------

9.

powershell -NoProfile -ExecutionPolicy Bypass -File "T:\TechServices-FP3\Techs\Jared\Scripts\_J\RegBangChecker\NetSuppRemoveRegCheck.ps1"

\\ Executes a PowerShell script (`NetSuppRemoveRegCheck.ps1`) with certain parameters to bypass the execution policy and not load any profiles.

------------------------------------------------------------------------------------------------------------------------------------------

10.

IF %ERRORLEVEL% NEQ 0 (

\\ Checks if the PowerShell script execution resulted in an error (error level not equal to 0).

------------------------------------------------------------------------------------------------------------------------------------------

11.

ECHO Error occurred in regCheck.ps1

\\ Outputs an error message if there was an error in executing the PowerShell script.

------------------------------------------------------------------------------------------------------------------------------------------

12.

ECHO Attempting to remove NetSupport installation folder...

\\ Outputs a message indicating the script is attempting to remove the NetSupport installation folder.

------------------------------------------------------------------------------------------------------------------------------------------

13.

RMDIR /S /Q "C:\Program Files (x86)\NetSupport"

\\ This command removes the NetSupport installation directory and all of its contents quietly without prompting for confirmation.

------------------------------------------------------------------------------------------------------------------------------------------

14.

ECHO Folder removed or not found.

\\ Outputs a message indicating that the folder has been removed or it was not found.

------------------------------------------------------------------------------------------------------------------------------------------

15.

ECHO Proceeding with NetSupport uninstallation...

\\ Outputs a message indicating that the script is proceeding with the uninstallation of NetSupport.

------------------------------------------------------------------------------------------------------------------------------------------

16.

START "" %UNINSTALL\_STRING%

\\ This line starts the uninstallation process for NetSupport using the uninstallation command string found in the registry.

------------------------------------------------------------------------------------------------------------------------------------------

17.

ECHO Uninstallation and cleanup complete.

\\ Outputs a message indicating that the uninstallation and cleanup process is complete.

------------------------------------------------------------------------------------------------------------------------------------------

18.

) ELSE (

\\ This marks the beginning of the 'else' block, which is executed if the user's input was not "Y" or "YES".

------------------------------------------------------------------------------------------------------------------------------------------

19.

ECHO Uninstallation aborted.

\\ Outputs a message indicating that the uninstallation has been aborted.

------------------------------------------------------------------------------------------------------------------------------------------

20.

)

\\ Marks the end of the 'if' block.

------------------------------------------------------------------------------------------------------------------------------------------

21.

) ELSE (

\\ Marks the beginning of an 'else' block which is executed if the NETSUPPORT\_VERSION variable is not defined.

------------------------------------------------------------------------------------------------------------------------------------------

22.

ECHO Unable to determine the installed version of NetSupport.

\\ Outputs a message indicating that the script was unable to determine the installed version of NetSupport.

------------------------------------------------------------------------------------------------------------------------------------------

killtasks.bat

1.

@echo off

\\ This command turns off the display of commands in the batch script, making the output cleaner.

-------------------------------------------------------------------------------------------------------------------------------

2.

SETLOCAL ENABLEEXTENSIONS ENABLEDELAYEDEXPANSION

\\ This line enables local environment settings within the batch file. ENABLEEXTENSIONS enables the use of extended features and ENABLEDELAYEDEXPANSION allows for the delayed expansion of variables.

-------------------------------------------------------------------------------------------------------------------------------

3.

TASKLIST | FINDSTR /I "client32.exe" >nul 2>&1

\\ This line checks if the process "client32.exe" is running. TASKLIST lists all running tasks, piped into FINDSTR which searches for "client32.exe" case-insensitively (/I). Output is redirected to null (>nul) to hide it, and standard error (2) is redirected to standard output (&1).

-------------------------------------------------------------------------------------------------------------------------------

4.

IF NOT ERRORLEVEL 1 (

\\ This line checks if the previous command (finding "client32.exe") was successful (ERRORLEVEL not 1). If successful, it means "client32.exe" is running.

-------------------------------------------------------------------------------------------------------------------------------

5.

TASKKILL /F /IM "client32.exe" >nul 2>&1

\\ Inside the IF block, this command forcefully (/F) terminates the "client32.exe" process (/IM specifies the image name of the process). Output and errors are again redirected to null.

-------------------------------------------------------------------------------------------------------------------------------

6.

IF NOT ERRORLEVEL 1 (

\\ This line checks if TASKKILL was successful in terminating "client32.exe" (ERRORLEVEL not 1).

-------------------------------------------------------------------------------------------------------------------------------

7.

ECHO client32.exe process terminated.

\\ If TASKKILL was successful, this command outputs a confirmation message.

-------------------------------------------------------------------------------------------------------------------------------

8.

) ELSE (

\\ The ELSE clause handles the case where TASKKILL failed to terminate "client32.exe".

-------------------------------------------------------------------------------------------------------------------------------

9.

ECHO Failed to terminate client32.exe.

\\ This line outputs a message indicating failure to terminate "client32.exe".

-------------------------------------------------------------------------------------------------------------------------------

10.

) ELSE (

\\ This ELSE corresponds to the outer IF and handles the case where "client32.exe" was not found running.

-------------------------------------------------------------------------------------------------------------------------------

11.

ECHO client32.exe is not running.

\\ Outputs a message indicating "client32.exe" is not currently running.

-------------------------------------------------------------------------------------------------------------------------------

12.

TASKLIST | FINDSTR /I "RevitAccelerator.exe" >nul 2>&1

\\ This line is similar to line 3, but it checks for the process "RevitAccelerator.exe".

-------------------------------------------------------------------------------------------------------------------------------

13.

IF NOT ERRORLEVEL 1 (

\\ Checks if the previous command (finding "RevitAccelerator.exe") was successful.

-------------------------------------------------------------------------------------------------------------------------------

14.

TASKKILL /F /IM "RevitAccelerator.exe"

\\ Forcefully terminates "RevitAccelerator.exe" if it was found running.

-------------------------------------------------------------------------------------------------------------------------------

15.

ECHO RevitAccelerator.exe process terminated.

\\ Outputs a confirmation message indicating the termination of "RevitAccelerator.exe".

-------------------------------------------------------------------------------------------------------------------------------

16.

) ELSE (

\\ Handles the case where "RevitAccelerator.exe" was not found running.

-------------------------------------------------------------------------------------------------------------------------------

17.

ECHO RevitAccelerator.exe is not running.

\\ Outputs a message indicating "RevitAccelerator.exe" is not currently running.

-------------------------------------------------------------------------------------------------------------------------------

It then goes on to do the same but for other tasks: Runplugin64.exe, runplugin.exe, node.exe, msedge.exe, smartscreen.exe, and CollaborationKeysController.exe