Execut-It ver. 2.0

Script & Task Processor *for Windows95®*

HELP FILE CONTENTS

Welcome to Execut-It help. It is a little but at times very useful application which can save you a lot of hassle when it comes to performing simple yet not that easily done tasks. Select a help area from topics listed below:

Installation

Running with command-Line parameters

Processing a script

Registering default Execut-It script extension

Note from the authors

Execut-It ver. 2.0

Script & Task Processor ***for Windows95®***

Running With Command-Line Parameters

One of the modes in which Execut-It can operate is a *command line parameter mode.* Unlike in previous version these parameters are not case sensitive, remember however that any errors will cause Execut-It to completely ignore the command or it may result in a system error. Execut-It will not notify you of processing errors at any time.

Below you will find a list of command line parameters together with required additional parameters where necessary.

Syntax:

```
executit.exe [COMMAND] fParam sParam
```

There might be cases wher *fParam* and/or *sParam* are optional depending on the *Command*. Below is a list of available commands. To find out more about their meaning and performed action, click on a name:

- **About** or **?**
- **Animate**
- **LogOff**
- **Play**
- **Power**
- **ReBoot**
- **Shutdown**
- **SSav**
- **Suspend**
- **WallPaper**

- **AutoConnect**

- **Script**

Note that results are computer dependent. Not all machines will be able to use all Execut-It command line parameter actions. All long file names used as additional parameter must be enclosed in double-quotes

About

Usage

executit.exe ABOUT

Action

Displays Execut-It about screen

Animate

Usage

executit.exe ANIMATE fParam

fParam	Could either be YES to turn window animation on or none to turn it off

Action

Turn on/off Window minimize/maximize animation. This action is not performed immediately

LogOff

Usage

executit.exe LOGOFF

Action

Close all running applications and log back on for a new/different user

Play

Usage

executit.exe PLAY fParam

fParam Filename of a Wave file. Using a complete path may be necessary. If using long filenames with spaces between them, using double quotes is necessary.

Action

Play a wave file

Power

Usage

executit.exe POWER fParam sParam

fParam	Timeout value in minutes for Low Power mode. 0 turns this mode off.
sParam	Timeout value in minutes for Power Off mode. Must be 0 or greater or equal to fParam

Action

Set monitor power saving modes. This action does not go into effect immediately.

Reboot

Usage

executit.exe REBOOT fParam

fParam Timeout value in seconds for user intervention. If no fParam specified action is performed immediately. If fParam [1-90] warning screen appears prior to actual reboot for fParam seconds, giving user a chance to abort.

Action

Reboot system

ShutDown

Usage

executit.exe SHUTDOWN fParam

fParam Timeout value in seconds for user intervention. If no fParam specified action is performed immediately. If fParam [1-90] warning screen appears prior to actual shutdown for fParam seconds, giving user a chance to abort.

Action

Shut the system down for complete power-off

SSav

Usage

executit.exe SSAV fParam

fParam Timeout value in minutes for Screen Saver activation.
If no fParam specified Screen Saved is deactivated.

Action

Shut the system down for complete power-off

Suspend

Usage

executit.exe SUSPEND

Action

Enter Suspend mode. You computer must support APM (Advanced Power Management)

WallPaper

Usage

```
executit.exe WALLPAPER fParam
```

fParam Filename of a BMP file. Using a complete path may be necessary. If using long filenames with spaces between them, using double quotes is necessary.

Action

Set new WallPaper for current desktop.

AutoConnect

Usage

executit.exe AUTOCONNECT fParam

fParam Can either be REBOOTABLE or none.

Action

AutoConnect will monitor your desktop for Dial-Up Networking connection dialogs and should any of them appear. Fo more information click here.

Running A Script

Usage

executit.exe ScriptFilename

Using a complete path may be necessary.
If using long filenames with spaces between them, using double quotes is necessary.

Action

Process Execut-It script specified by *ScriptFilename*. For more information click here

RUNNING A COMMAND SCRIPT

One of the two modes Execut-It can operate in is a *Script-Process mode*. In this mode you will be able to process a list of operations with one execution. This feature might be extremely useful when performing routine tasks consisting of several subcommands and sub-tasks. As we all know, under Windows95 old batch file system does not perform well at all, plus it would not give you the simplicity and flexibility of Execut-It script processing.

Number of features and their advancement are limited so that we could keep quite simple syntax. Execut-It is by no means a compiler-like product and having a very simple structure it is also very easy to use.

There are only a couple of restrictions when writing your Execut-It script but we have really tried to make it as fool-proof as possible while not sacrificing the size of this program too much. Unlike in previous version, Execut-It is less sensitive to selective command syntax. What it means that if a command, mode, option etc is optional you may simply skip that line specifying nothing. Except in [RUN] set <program name> and [MSG] <Title> which in both cases must be used as first entry of the set, order of rest of the parameters and directives is not fixed. Try to avoid empty lines except between end of one set and beginning of another, this however is again not regarded as an error and will be processed.

Wherever you see the {o} mark, it means that a parameter, mode or directive is optional and may or may not have to be used. Others are required.

If you do not wish to remember all this syntax you can use Execut-It script editor (*eiedit.exe*), which features automated set creation for quick script writing.

Below is a list of all the commands available for you to use in your scripts. At the end of this page we have also enclosed a sample script to give you an idea how to write your own.

Script sets may be one or more of the following:

[RUN]
<program name (200 chars long max)> (full path if necessary)
<parameters (200 chars long max)> {o}
WAIT {o}
<mode> {o}
Cmd:<command list> {o}

> Execute program specified by <program name> or program associated with this type of file. Calling it using full path may be necessary. You can also start <program name> with optional parameters such as document name or options. Note that some programs may require /option notation. If no parameters are to be used simply skip this command. You can also open application in specific <mode> which can either be NORMAL, MAXIMIZED or MINIMIZED. Long filenames must be enclosed in double-quotes(). It will also execute a shortcut (link). <command list> may contain a list of keys you wish to simulate pushing upon opening of an application. To view Cmd syntax click here. If you include WAIT directive, script process will be paused till launched application terminates. Note however that WAIT will not work if you launching files other than executables themselves. In other words you cannot use MyFile.txt as program name as you normally could. You must specify program name (in this case it may be Notepad.exe) and use file to open as a parameter (MyFile.txt).
>
> (*) Note that you can use Execut-It itself in [RUN] set to call itself with different parameters or scripts. This will not terminate or disturb one it is called from

[DIRECTORY]

<full path (200 chars long max)>

<full path> will be used for all [RUN] and [DELETE] sets to follow until end or next [DIRECTORY] set is specified. This way you can execute/delete programs by calling only their filenames providing that they are in the same directory specified by <full path>; Note that specifying full path filename in [RUN] or [DELETE] will override this setting in that instance. Long filenames must be enclosed in double-quotes()

[DELETE]

NORECYCLE {o}
<file to delete>
<file to delete> {o}
<file ...> {o}

Deletes file(s) specified by <file to delete>. If NORECYCLE command is present, deleted files will not be placed in the Recycle Bin. Multiple files can be declared using any combination of name.ext. Wildcards (*) are also allowed. You can either explicitly specify full path\filename or use [DIRECTORY] to set path and then only refer to file by its name. Long filenames must be enclosed in double-quotes()

(*) If you trying to delete files to which access is denied while NORECYCLE directive is not used, you will get a System Error Message. You should use NORECYCLE in such places where this error might occur - such as your TEMP directory.

[MSG]

<Title (25 char max)>
<ResponseStyle> {o}
<Type> {o}
<Message text (100 chars per line max)>
<Message text> {o}
CR {o}
<Message text> {o}

Display a message on screen. <Title> is message`s title. <Message text> can be in multiple lines. This way you can sort of design its appearance. If you put CR between lines it will place a space between corresponding lines. Multiple CR's are allowed. Simplified <ResponseStyle> is available *YES_NO*, *OK_CANCEL* or *OK*. <Type> may be: *EXCLAMATION*, *CRITICAL*, *INFORMATION*, *QUESTION* or *SYSTEM* (which may be useful at times because it stays on top of all windows until OK pressed). If you use this set in the middle of script, it will be halted till you close the message (press button). Unless OK <ResponseStyle> is used, Execut-It will try to jump to the first <ResponseDirective> which must be one or more of the following:

[BN OK] - if OK button pressed in OK_CANCEL style;
[BN CANCEL] if CANCEL button is pressed in OK_CANCEL style;
[BN YES] if YES button is pressed in YES_NO style;
[BN NO] if NO button is pressed in YES_NO style;

[WAIT]

<Seconds>

Script execution will be halted for specified amount of seconds. It may be important if you want to ensure proper execution sequence or whatever else you might think of.

[PLAY]

<mode> {o}
<mode> {o}

<Sound>

Play either WAV file specified by <Sound> (using full path may be necessary) or a system sound. For example you could specify <Sound> as

```
c:\windows\sounds\somesong.wav
or
SystemExit
```

In the second case Execut-It will play the sound assigned as you Exit Windows sound. The two <mode> directives may be specified as follows:

NOWAIT - to tell Execut-It to play the sound and proceed while playing
FORCE - to force other sound playing at that time to stop.

You may specify either, both or none of those modes. Note however that if *FORCE* mode is not specified sound may not be played if another sound is being played by another application/system. If *NOWAIT* mode is not specified Execut-It will wait with processing the script until the sound is finished. Long filenames must be enclosed in double-quotes()

[JUMP]
<label>

[JUMP] set will cause Execut-It to move pointer to label skipping instructions and sets calling label is simply a string. Label itself however must be the same string surrounded by square brackets like [LABEL]
For example:

```
[JUMP]
ComeHere

this will be skipped

[ComeHere]
[NEXT EXECUTED SET]
```

[LABEL]

Jump point to which to send a script processing. Word label can be replaced by any. Must be enclosed in square brackets and with no spaces (use underscore or dash for separation). [LABEL] can be put anywhere in the script even before [JUMP] specifying that label which may be useful for repeated tasks based on [MSG] responses.

[END]

Terminates the script execution. This command is required at the end of the script. We noted a couple of cases when users forgot to put the [END] command and Execut-It ran indefinitely. To prevent this we have limited script to 400 lines/operations maximum which should be more than enough for an actual script and will cause Execut-It to terminate itself whether you use [END] or not. Note, that [END] can be placed anywhere else in the script which may be useful in case of jumps or using message responses

Comments

Can be placed anywhere by preceding them with a pound (#) character

```
# This is my comment
```

Sample Script

```
#
# My first Execut-It script file.
# Sample script by Webster Image Group
#

[RUN]
        # Run myapp.exe minimized
        # but wait with script execution
        # until it terminates
myapp.exe
WAIT
MINIMIZED

[RUN]
        # Run someapp.exe maximized and then tell it simulate
        # pressing keys: O;D;D and then SHIFT-F2
someapp.exe
Maximized
Cmd: ODD+{F2}

[WAIT]
        # Wait now for 5 seconds
5

[PLAY]
        # Play sound mysound.wav in directory c:\sounds
        # If anything else is playing stop it and give priority
        # Wait with script processing until sound is finished
FORCE
c:\sounds\mysound.wav

[MSG]
        # Ask the user something and act upon it
Information
YES_NO
QUESTION
Hello Joe.
CR
Did you like the sound I was playing?

[BN NO]
        # User answered NO; End this script
        # Skip to place marked by [QUICK-END]
        # Note that if jump is to be made to end the script you can
        # simply put [END] here instead of [JUMP]
[JUMP]
QUICK-END

[BN YES]
        # User answered YES; Proceede from here

[PLAY]
        # Play other.wav in d:\extra; Process rest of script without
```

```
            # waiting for sound to end, however if the device is busy,
            #sound will not be heard
NOWAIT
d:\extra\other.wav

[DIRECTORY]
            # Change current script directory to c:\temp
c:\temp

[DELETE]
            # Delete all files (from c:\temp).
            # [DIRECTORY] should always be specified with relative
            # file specification so you dont remove files from
            # a wrong directory
*.*

[DELETE]
            # Remove some more files but this time do not place
            # them in the Recycle Bin.
            # Remove all BAK files from c:\windows
            # and all TMP files from c:\temp as this is my current directory
NORECYCLE
c:\windows\*.bak
*.tmp

[RUN]
            # Run Execut-It itself with a command line parameter SSAV
10
            # so that it sets Screen Saver timeout to 10 minutes
executit.exe
SSAV 10

[QUICK-END]
            # Remember? The JUMP will be here.
[END]
            # Finally Im done
```

Registering Default Execut-It ScriptExtension

If you want to create a default extension for Execut-It script files (**.eis*) you can do so by running its script editor installed in application`s directory with a /reg parameter

```
eiedit.exe /reg
```

What will it do:

- Add *.EIS* extension to the assigned extensions list
- Associate this extension with Execut-It application. By double clicking or calling the script alone it will launch Execut-It to actually run it.
- When clicking in Explorer with right mouse button on a script file, one of the options available will be *Edit Script...* which will open Execut-It script editor with its contents.

Command List Syntax

Command list may contain a series of keys(buttons) to be passed to an application. Convention used is that of Visual Basic, C++ and other common programming language procedures so if you familiar with that syntax you will have no problem using this list.

Key	Code
BACKSPACE	{BS}
BREAK	{BREAK}
CAPS LOCK	{CAPSLOCK}
DEL	{DELETE} or {DEL}
DOWN ARROW	{DOWN}
END	{END}
ENTER	{ENTER}
ESC	{ESC}
HELP	{HELP}
HOME	{HOME}
INS	{INSERT}
LEFT ARROW	{LEFT}
NUM LOCK	{NUMLOCK}
PAGE DOWN	{PGDN}
PAGE UP	{PGUP}
PRINT SCREEN	{PRTSC}
RIGHT ARROW	{RIGHT}
SCROLL LOCK	{SCROLLLOCK}
TAB	{TAB}
UP ARROW	{UP}
F1	{F1}
F2	{F2}
F3	{F3}
F4	{F4}
F5	{F5}
F6	{F6}
F7	{F7}
F8	{F8}
F9	{F9}
F10	{F10}
F11	{F11}
F12	{F12}
F13	{F13}
F14	{F14}
F15	{F15}
F16	{F16}

To specify keys combined with any combination of the SHIFT, CTRL, and ALT keys, precede the regular key code with one or more of the code keys in following table:

Key	Code
SHIFT	+
CTRL (CONTROL)	^

ALT	%

If any combination of SHIFT, CTRL, and ALT should be held down while several other keys are pressed, enclose the code for those keys in parentheses.

For example:

use **+(AB)**	to specify to hold down SHIFT while A and B are pressed
use **+AB**	to specify to hold down SHIFT while A is pressed and B without SHIFT

Command list Examples:

Cmd: %FOmyfile.txt{ENTER}
will send Alt-F;O;myfile.txt;ENTER as a result of which you would open file myfile.txt

Cmd: +(PC)K
will send SHIFT-P; SHIFT-C;K

Cmd: ^{F10}%{ENTER}^CK
will send CTRL-F10;ALT-ENTER;CTRL-C;K

Execut-It ver. 2.0

Script & Task Processor ***for Windows95®***

NOTE FROM THE AUTHORS

We hope you find some good use for this application. When the first version of Execut-It has been written by one of our programmers for his immediate needs it turned out that much more people can put program like this into their own computing routine. So we decided to better it up a bit.

Version 2.0 of Execut-It is what we would like to call an *Appreciationware*. It means that if you like what it is doing for you we would like you to make some financial contribution to its current and future development. How much? That is the best part - you decide. Just to not to leave you guessing anywhere between $1 and $5 is absolutely what we find reasonable. Of course if you appreciate it for much more we will not refuse. If you cannot afford a donation would you at least please let us know if you happen to happily use it. We promise we will never bother you for any payments it will however, give us some reason to work on it later. Execut-It is not some *crippleware,* it will not nag or bother you with any screens. It works silently in the background for as long as necessary and terminates. Main objective was less user intervention so it never reports any errors(*) and if your parameters or scripts are incorrect, they are in most cases simply ignored.

If you decide to make you donation, please send Check or M.O. to

Webster Image Group
69-28 62nd Drive
Middle Village, NY 11379-1102

and we thank you in advance for doing so.

For any other information, additional files upgrades and updates visit us on the web at:
http://www.webster-image.com/clients/ei

(*) System errors should they occur are possible. We have tried to eliminate any such possibility however keep in mind that Execut-It is a quite simple script processor not a sophisticated compiler. Error that is definitely shown is one coming from the [DELETE] set in case files are to be moved to the Recycle Bin and yet access is being forbidden. See more in Script Processing section

Execut-It ver. 2.0

Script & Task Processor ***for Windows95®***

EXECUT-IT 2.0 INSTALLATION

Run Execut-It setup program (*setup.exe*) to install its components. Main file (*executit.exe*) will be installed into your Windows directory.

Execut-It can work only on computers running Microsoft® Windows95®.

Execut-It ver. 2.0

Script & Task Processor ***for Windows95®***

DUN AutoConnect

Even though AutoConnect is one of Execut-It`s features it has properties of its own and deserves separate help page to explain how it works. Since the build 2.0.0220 AutoConnect will do following thigs:

- Initialize called DUN connection;
- Initialize connection upon disconnect;
- Attempt to connect inspite of reported DUN errors;
- Close connection call if modem already in use;
- Run designated script upon successful connection;
- Reboot and connect again if severe DUN error;

Initializing DUN connection:

When a DUN dialog comes up, AutoConnect will initialize this connection. If process is cancelled during execution AutoConnect will attempt to connect again. 5sec timeout is set so that you can actually cancel the connection if you choose to.

Initializing connection upon disconect:

If you get disconnected AutoConnect will reinitialize this connection automnatically

DUN Errors:

There are two most common errors which may come up during initialization:
Unable to negotiate protocol and *Modem already in use*. If Modem is already in use AutoConnect will close this connection so that you dont get constant errors. Connection obviously cannot be made. If DUN is unable to negotiate a protocol, connection will be reset and tried again (up to 15 times). *However*, this particular error may be a result of one of three things:

- Problems on the ISP end
- Incorrectly setup DUN connection
- Windows95 goes nuts (very likely)

If you are sure that your connections are setup right and your ISP is unlikely to experience problems you can run AutoConnect in *ReBootable* mode.

ReBootable Mode:

If the error discussed above is a result of a glitch in Windows95, you can bet your life you will not be able to establish a connection since, unless you reboot your machine. If you run AutoConnect in this mode, it will do that for you and then initialize this connection again. Note that two attempt will be made. If after second attempt this error comes up, it is only then when reboot will take place.

It may not always produce expected results however, this is one more way you can attempt automate the process. This would be particularly useful if you often leave your machine unattented to perform Internet tasks and nothing messes things up more than some bizzare message which nobody reads anyway. Prior to reboot you will be warned with a message box giving you 3 seconds to abort so you can still have some saying in this if you want to. Otherwise, after 3 seconds System will be rebooted. Be careful since all your unsaved work will be lost in this case.

Running a script upon connect:

Starting with the build 2.0.0220 you can also run a script when connection is finalized. To do so, you must create a file called AutoConn.acn and place it in your WINDOWS directory. To run a script this file must contain valid entries as in example below:

i.e. You have following DUN connections created

My SLIP Connection
My PPP Connection
Some other connection

To be able to run scripts for them AutoConn.acn needs to look somewhat like this:

```
[AC Scripts]
My SLIP Connection=C:\MyDir\SLIP_script.eis
MY PPP Connection=D:\OtherDir\SomeScript.eis
Some other connection=C:\MyDir\DefaultConn.eis
```

Got the idea? It is not necessary that you have an entry for every connection, just those you wish to run the script for. The first part (before = sign) must be an *exact* connection name.

Note that script will not be ran after successful *reconnection*.