

CHT Applications

[HOME](#) [CLASSES](#) [TPLS](#) [APPS](#)

TEMPLATES, CLASSES, UTILITIES, APPS, PROJECTS

CHT SNAP-IN & BATCH-BOT UTILITIES (BUILD: 24C.02.02) SEPTEMBER 17, 2020

CATEGORY: BATCH BOT

*** HNDENCRYPT.APP ***

NAME: HNDENCRYPT.APP
TITLE: BATCH-BOT ENCRYPTION [V10.20.09]
DATE: 9/17/2020
CATEGORY: BATCH BOT
CLASSES: [List App Classes](#)
TEMPLATES: [List App Templates](#)

DESCRIPTION:

This "CHT Batch Bot", HNDENCRYPT.APP, application runs from the DOS prompt or can be called from another application using Clarion RUN, or CHT HNDSnapCall Class, or from a batch file or windows .VBS script. It accepts several command-line parameters that describe the input file and and output path.

HNDENCRYPT.EXE source code, because it is in Clarion .APP format, is included as a standard part of the CHT toolkit, provided both in .APP format as well as ready-to-run, in .EXE format.

HNDENCRYPT.EXE command-line parameters are as follows:

```
** HELP **
--help [OPT]
    Pops up a splash help screen.
--silent [OPT]
    Suppresses log and help file popups.

** ENCRYPT/DECRYPT **
--input [REQ]
    Path and name of input file.
    If the path is omitted, the .EXE path is assumed.
--output [REQ]
    Output path only - local to this exe if omitted.
--mode [OPT]
    Encode or decode - defaults to encode if omitted.
--key [OPT]
    Encrypting password - read from registry if omitted.
```

The output file name is formatted from the input file with a .hzy extension. For example: test.pdf becomes testpdf.hzy.

The encrypted file is both compressed and encrypted by the operation of this CHT Batch Bot application. This operation is performed in a single pass whether encoding or decoding.

*** HNDFTPNET.APP ***

NAME: HNDFTPNET.APP
TITLE: BATCH-BOT FTP TOOL [V10.20.09]
DATE: 9/17/2020

CATEGORY: BATCH BOT

CLASSES: [List App Classes](#)

TEMPLATES: [List App Templates](#)

DESCRIPTION:

This "CHT Batch Bot", HNDFTPNET.EXE is a command-line FTP tool which can send, fetch and erase files to/from any standard FTP site. The application can be called from one of your Clarion applications, a Windows batch file or a Windows .VBS script to provide reliable single and multi file upload/download/services.

HNDFTPNET.EXE source code, because it is in Clarion .APP format, is included as a standard part of the CHT toolkit, provided both in .APP format as well as ready-to-run, in .EXE format.

HNDFTPNET.EXE is dependent on a CHT DLL called HNDFTPNET.DLL being present and visible on the system path, for its FTP capabilities.

The application communicates to this DLL via COM, requiring the HNDFTPNET.DLL to be registered on the host system. This DLL will auto-register on the host system when first called and found not to be registered.

CONFIGURING:

--config0
[CFG] Set up FTP configuration information #0

--config1
[CFG] Set up FTP configuration information #1

--config2
[CFG] Set up FTP configuration information #2

--set0
[CFG] Select and use configuration #0

--set1
[CFG] Select and use configuration #1

--set2
[CFG] Select and use configuration #2

COMMANDS:

--send
[OPT] Provides file name of the local file to be uploaded.
This file need not exist if used with the --mask tag to indicate that a pre-zip is performed using a mask.

--remote
[REQ] Names the remote directory with which to interact.
This is a required parameter which is formed as follows:
/ftprootdir/subdir1/subdir2/

--local
[OPT] Names a local directory when the --mask tag is used to describe one or more files for pre-zipping.
Defaults to the application directory if omitted.

--pwd
[OPT] Password used when pre-zipping with --mask.

--mask
[OPT] DOS-style file mask (eg: *.app) describes files to be zipped. Requires CHTSNAPZIP.EXE to be available.
--send parameter must be .zip file name

--fetch
[OPT] File name of a remote file to be downloaded.
Use instead of --send or --erase when downloading.

--erase
[OPT] Provides file to be erased from a remote directory.
Use instead of --fetch and --send when deleting.

--list
[OPT] Used by itself, this provides a file listing of the remote directory stipulated after the tag.
E.G. --list /web/subdir/

LOGGING:

All successful transactions or errors are logged into "hndftpnet.log" located in the same directory as the hndftpnet.exe.

HELP:
--help

[OPT] Used by itself, this pops up a splash help screen
Creates hndftpnethelp.txt which contains the contents of
this splash screen.

*** HNDMAILCDO.APP ***

NAME: HNDMAILCDO.APP
TITLE: BATCH-BOT CDO MAIL [V10.20.09]
DATE: 9/17/2020
CATEGORY: BATCH BOT
CLASSES: [List App Classes](#)
TEMPLATES: [List App Templates](#)

DESCRIPTION:

This "CHT Batch Bot", HNDMAILCDO.EXE, uses CDO mail which is a form of SMTP mail built into windows applications for use by scripting languages. Before you use it, the application must be configured with valid email parameters the same as any other email client requires configuration. This application is stand-alone and requires no extra DLLs.

The email template used to build this application is EMBED CDO MAIL FUNCTIONS. The CHT CDO Class is called HNDMSCDO located in files HNDMSCDO.INC and HNDMSCDO.CLW.

HNDMAILCDO.APP source code, because it is in Clarion .APP format, is included as a standard part of the CHT toolkit, provided both in .APP format as well as ready-to-run, in .EXE format.

HNDMAILCDO.EXE runs from the DOS prompt or can be called from another application using Clarion RUN or CHT's HNDSNAPCALL class. It accepts several command line parameters that describe the email to be sent. These parameters are as follows:

SENDING:

--to [REQ] mymail@address.com;mail@address.com;
--cc [OPT] mymail@address.com;mail@address.com;
--subj [REQ] This is the subject line.
--file [OPT] emailbody.txt/.html (contains body text)
--attach [OPT] somezip.zip (Attachment requires full path)

CONFIGURING:

--config0 [CFG] Set up mail configuration information #0
--config1 [CFG] Set up mail configuration information #1
--config2 [CFG] Set up mail configuration information #2
--set0 [CFG] Select and use configuration #0
--set1 [CFG] Select and use configuration #1
--set2 [CFG] Select and use configuration #2

LOGGING:

All transactions such as config changes, emails or errors are logged into "hndmailcdo.log" located in the same directory as the hndmailcdo.exe.

HELP:

--help Pops up a splash help screen and help file hndmailcdohelp.txt
--silent [OPT] Suppresses popup logs and help during mult-recipient sends

*** HNDMAILMAPI.APP ***

NAME: HNDMAILMAPI.APP
TITLE: BATCH-BOT MAPI MAIL [V10.20.09]
DATE: 9/17/2020
CATEGORY: BATCH BOT
CLASSES: [List App Classes](#)

TEMPLATES: [List App Templates](#)

DESCRIPTION:

This "CHT Batch Bot", HNDMAILMAPI.EXE, provides MAPI send capability using CHT's Clarion class-based MAPI functionality.

The CHT template illustrated in this application is called SOURCE_EMBED_MAPI_FUNCTIONS. The class underlying this template is called HNDIMAIL located in files HNDIMAIL.INC and HNDIMAIL.CLW.

HNDMAILMAPI.APP source code, because it is in Clarion .APP format, is included as a standard part of the CHT toolkit, provided both in .APP format as well as ready-to-run, in .EXE format.

When you have a properly configured MAPI email client like Outlook or Windows Live Mail configured, this application is able to connect with it to send email. This application is stand-alone and requires no other CHT DLLs.

HNDMAILMAPI.EXE runs from the DOS prompt or can be called from another application using Clarion RUN, or CHT's HNDSnapCall Class, from a batch file or even from a .VBS script. It is capable of sending standard email of the type(s) configured in your default email client.

It accepts several command line parameters that describe the email to be sent. These parameters are as follows:

SENDING:

--to [REQ] mail@address.com (or multiples + semi-colons)
--cc [OPT] mail@address.com (or multiples + semi-colons)
--subj [REQ] This is the subject line.
--file [REQ] emailbody.txt (Contains body text)
--attach [OPT] somezip.zip (Name of attachment requires full path)

LOGGING:

All transactions such as config changes, emails or errors are logged into "hndmailmapi.log" located in the same directory as the hndmailmapi.exe.

HELP:

--help Pops up a help text file hndmailmapihelp.txt
--silent Suppresses popup-help during multiple sends

*** HNDMAILNET.APP ***

NAME: HNDMAILNET.APP
TITLE: BATCH-BOT NET MAIL [V10.20.09]
DATE: 9/17/2020
CATEGORY: BATCH BOT
CLASSES: [List App Classes](#)
TEMPLATES: [List App Templates](#)

DESCRIPTION:

This "CHT Batch Bot", HNDMAILNET.EXE, uses SMTP mail provided by a CHT DLL called HNDNETSMTP.DLL written in C#.

Before you use it, the application must be configured with valid email parameters the same as any other smtp email client requires configuration. This application is stand-alone but requires the DLL mentioned above be located on the target machine and registered via REGASM.EXE.

HNDMAILNET.APP source code, because it is in Clarion .APP format, is included as a standard part of the CHT toolkit, provided both in .APP format as well as ready-to-run, in .EXE format.

The CHT email template used in this application is EMBEDNETSMTPCLASSES. The CHT class underlying this template is HNDNETSMTPCLASS located in files HNDNETSMTP.INC and HNDNETSMTP.CLW.

HNDMAILNET.EXE runs from the DOS prompt or can be called from another application using Clarion RUN, or CHT's HNDSNAPCALL class, or run from a batch file or even from a .VBS script. It is capable of sending standard SMTP email as well as SSL SMTP email as found with GMAIL, HOTMAIL and other similar services.

It accepts several command line parameters that describe the email to be sent. These parameters are as

follows:

**** SENDING:**

--to [REQ] mail@address.com
--cc [OPT] mail@address.com
--bcc [OPT] mail@address.com
--subj [REQ] This is the subject line.
--file [REQ] smtpbody.txt (Contains email body)
--attach [OPT] somezip.zip (Requires full path to file)

CONFIGURING:

--config0 [CFG] Set up mail configuration information #0
--config1 [CFG] Set up mail configuration information #1
--config2 [CFG] Set up mail configuration information #2
--set0 [CFG] Select and use configuration #0
--set1 [CFG] Select and use configuration #1
--set2 [CFG] Select and use configuration #2

LOGGING:

All transactions such as config changes, emails or errors are logged into "hndmailnet.log" located in the same directory as the hndmailnet.exe.

HELP:

--help [OPT] Pops up a help document
--silent [OPT] Suppress pop up log and help file during multi-recipient sends.

*** HNDMAILSMTP.APP ***

NAME: HNDMAILSMTP.APP
TITLE: BATCH-BOT SMTP MAIL [V10.20.09]
DATE: 9/17/2020
CATEGORY: BATCH BOT
CLASSES: [List App Classes](#)
TEMPLATES: [List App Templates](#)

DESCRIPTION:

This "CHT Batch Bot", HNDMAILSMTP.EXE uses CHT Native SMTP mail. It is not dependent on the host hardware having a MAPI-compliant email client. This application requires Clarion's CLARUNEXT.DLL but is otherwise stand-alone.

HNDMAILSMTP.EXE requires correct configuration of a valid SMTP email setup with UserName, Password, Email Server and Email Return Address. Our template and SMTP classes provide a configuration function with user-controllable registry location settings.

HNDMAILSMTP.APP source code, because it is in Clarion .APP format, is included as a standard part of the CHT toolkit, provided both in .APP format as well as ready-to-run, in .EXE format.

The email template illustrated is EMBEDSMTPFUNCTIONS. The CHT class underlying this template is HNDSMTP located in files HNDSMTP.INC and HNDSMTP.CLW.

HNDMAILSMTP.EXE runs from the DOS prompt or can be called from another application using Clarion RUN. It accepts several command line parameters that describe the email to be sent. These parameters are as follows:

SENDING:

--to [REQ] mail@address.com;mail@address2.com;
--cc [OPT] mailcc@address.com;mailcc@address2.com;
--bcc [OPT] mailbcc@address.com;mailbcc@address2.com;
--subj

[REQ] This is the subject line.
--file [REQ] emailbody.txt (Contains email body)
--attach [OPT] somezip.zip (Name of attachment requires full path)

CONFIGURING:

--config0 [CFG] Set up mail configuration information #0
--config1 [CFG] Set up mail configuration information #1
--config2 [CFG] Set up mail configuration information #2
--set0 [CFG] Select and use configuration #0
--set1 [CFG] Select and use configuration #1
--set2 [CFG] Select and use configuration #2

LOGGING:

All transactions such as config changes, emails or errors are logged into "hndmailsmtp.log" located in the same directory as the hndmailsmtp.exe.

HELP:

--help [OPT] Pops up a help splash screen
--silent [OPT] Suppresses log and help file popups

*** HNDMAKEHZO.APP ***

NAME: HNDMAKEHZO.APP
TITLE: BATCH-BOT HZO MAKER [V10.20.09]
DATE: 9/17/2020
CATEGORY: BATCH BOT
CLASSES: [List App Classes](#)
TEMPLATES: [List App Templates](#)

DESCRIPTION:

This "CHT Batch Bot", HNDMAKEHZO.EXE runs from the DOS prompt or can be called from another application using Clarion RUN, or from a batch file or windows .VBS script. It accepts several command line parameters that describe the input file and and output path.

HNDMAKEHZO.EXE builds compressed/encrypted file containers based on Clarion .TPS files. These file containers (.HZO extension) may be used like zip files, except they are relational, making them easy to browse, add to, or extract from in a non-sequential access fashion.

HNDMAKEHZO.EXE source code, because it is in Clarion .APP format, is included as a standard part of the CHT toolkit, provided both in .APP format as well as ready-to-run, in .EXE format.

HNDMAKEHZO.EXE command-line parameters are as follows:

--help [ALONE]
Pops up this splash help screen
--silent [OPT]
Supresses display of hndmakehzoHELP.txt file

** Build A Container **

--input [REQ]
Path of input directory
--output [REQ]
Path and name of output .HZO file

** Extract a Container **

--extract [REQ]
Name of container to extract
--target [REQ]
Name of target extraction directory

** Used wth Build or Extract **

--pass [OPT]
Encrypting/decrypting flag (true if present, false if missing).
--query [OPT]

CHT Query to describe files for input or output

*** HNDSEARCH.APP ***

NAME: HNDSEARCH.APP
TITLE: BATCH-BOT SEARCH TEXT [V10.20.09]
DATE: 9/17/2020
CATEGORY: BATCH BOT
CLASSES: [List App Classes](#)
TEMPLATES: [List App Templates](#)

DESCRIPTION:

This "CHT Batch Bot", HNDSEARCH.EXE, is a command-line utility that can be called from another application using, for example, Clarion RUN or one of CHT's many ShellRun() functions. It can also be leveraged from a DOS batch file, or a windows .VBS script or directly from the DOS prompt.

HNDSEARCH.EXE source code, because it is in Clarion .APP format, is included as a standard part of the CHT toolkit, provided both in .APP format as well as ready-to-run, in .EXE format.

The application accepts several command line parameters that describe search values, replace values and more. These parameters are as follows:

--help
 [OPT] Pops up application splash screen.
 When used, all other parameters are ignored.

--mask
 [REQ] Required parameter eg: c:\temp*.app
 Describes single or multiple files to be searched.
 Path required, unless file(s) in same directory as this exe.

--search
 [REQ] Required search text parameter.
 This describes what you're searching for.

--replace
 [OPT] Optional parameter if only a search
 This replaces the value under --search

--silent
 [OPT] Do not display log file after processing.

*** HNDSLFCMD.APP ***

NAME: HNDSLFCMD.APP
TITLE: BATCH-BOT HTTP TOOL [V10.20.09]
DATE: 9/17/2020
CATEGORY: BATCH BOT
CLASSES: [List App Classes](#)
TEMPLATES: [List App Templates](#)

DESCRIPTION:

This "CHT Batch Bot", HNDSLFCMD.EXE is a command-line HTTP tool which can send to, fetch and erase files from any standard CHT HTTP site managed by a Type 4 CHT Server such as HNDSLFSV.APP. A password and username are required for access to the server. Communications in both directions are encrypted.

The "CHT Batch Bot" application can be called from one of your Clarion applications, from a Windows batch file or from a Windows .VBS script to provide reliable file upload, download and delete services.

HNDSLFCMD.EXE source code, because it is in Clarion .APP format, is included as a standard part of the CHT toolkit, provided both in .APP format as well as ready-to-run, in .EXE format.

CONFIGURATION COMMANDS:

--config0 to --config9
 [CFG] Set up HTTP configuration information #0 to #9

SET COMMANDS:

"Set" commands may be combined with --send, --fetch or --erase.

--set0 to --set9
 [CFG] Select and use configuration #0 to #9
--set

[CFG] Manually inserted url and port, ignore config settings.
E.G. --set http://www.yoursite.org:88/

--userpass

[CFG] Manually inserted username and password
E.G. --userpass username:userpass

ACTION COMMANDS

--send

[REQ] Provides file name of the local file to be uploaded.
Use with --local to name the local pickup directory.
Use with --remote to name the remote target directory.
Multiple files may be listed, separated by commas.

--fetch

[OPT] File name of a remote file to be downloaded.
Use instead of --send or --erase for downloading files.
Use with --local to name the local target directory.
Use with --remote to name the remote pickup directory.
Multiple files may be listed, separated by commas.

--erase

[OPT] Provides file to be erased from a remote directory.
Use instead of --fetch and --send when deleting.
Use with --remote to name the remote directory.
Multiple files may be listed, separated by commas.

SOURCE/TARGET COMMANDS:

--remote

[OPT] Remote directory with which to interact.
Remote directory below server /root/ directory.
Example usage: subdir1/subdir2/
Defaults to the server /root/ directory if omitted.

--local

[OPT] Names a local directory to indicate --send pickup directory or to indicate --fetch target directory.
Defaults to the application directory if omitted.

HELP COMMANDS:

--help

[OPT] Pops up a splash help screen.
Creates hndslfcmdhelp.txt which contains the contents of this splash screen.

LOGGING:

All successful transactions and/or errors are logged into "hndslfcmd.log" located in the same directory as the hndslfcmd.exe.

*** HNDZIPNET.APP ***

NAME: HNDZIPNET.APP
TITLE: BATCH-BOT ZIP FILES [V10.20.09]
DATE: 9/17/2020
CATEGORY: BATCH BOT
CLASSES: [List App Classes](#)
TEMPLATES: [List App Templates](#)

DESCRIPTION:

This "CHT Batch Bot", HNDZIPNET.EXE is a command-line zip tool which can zip files alone or in bulk using a DOS-style file mask (eg: *.app) to describe the files to be zipped. The application can be called from one of your Clarion applications, a Windows batch file or a Windows .VBS script to provide reliable single and multi file zipping services.

HNDZIPNET.EXE source code, because it is in Clarion .APP format, is included as a standard part of the CHT toolkit, provided both in .APP format as well as ready-to-run, in .EXE format.

HNDZIPNET.EXE is dependent on one CHT DLL being present and visible on the system path, namely: HNDNETZIPPER.DLL.

The application communicates to this DLL via COM requiring the HNDNETZIPPER.DLL to be registered on the host system. The DLL will auto-register on the host system when first called and found not to be registered. Auto-registration can only be performed from an "admin" account.

** ZIPPING/UNZIPPING/LISTING ***

--zip
Provides file name of the target ZIP file to be created.
Note that the ESC key cancels ZIP operations in all cases.
Use --path to name the source files location.

--unzip
File name of the source ZIP file to be processed.
Use --path to name the UNZIP location.

--list [ALONE]
Lists the entire contents of the ZIP to "hndzipnet.log"
and displays it in CHT Snap Edit.
All other tags are ignored when --list is invoked.

--path [OPT]
Names the root where --ZIP picks up files for processing.
Names the root where --unzip places its files.
The Start-In path is used when this tag is omitted.

--subs [OPT]
Flags "true" that root path subdirectories be processed too.
Flags "true" that the ZIP must incorporate paths.

--win [OPT]
Flags "true" to indicate show a progress window.

--mask [OPT]
Standard dos-style file mask to describe files to be
zipped,, for example *.app

--qry [OPT]
CHT-Style file query to describe files to be zipped.
For example: NAME ENDSWITH APP OR NAME * .DCT

--silent [OPT]
Flags do not pop up a results log on completion of the operation.

HELP:
--help [ALONE]
Pops up this splash help screen.
Creates a help file called "hndzipnethelp.txt"
Displays "hndzipnethelp.txt" in CHT Snap Editor window.

LOGGING:
All successful operations or errors are logged into "hndzipnet.log" located in the same directory as the
hndzipnet.exe.

CATEGORY: SNAP-IN C#

*** CHTSNAP2PDF.EXE ***

NAME: CHTSNAP2PDF.EXE
TITLE: CHT Snap Edit CMD Utility -- Version: 23.19.09.01
DATE: 9/17/2020
CATEGORY: SNAP-IN C#

DESCRIPTION:

CHTSNAP2PDF.EXE (AKA: CHT Snap To PDF), is a command-line translation tool that is able to translate well-formed HTML 5 documents into exact-copy PDF documents that include full use of correctly formatted internal and external links.

CHTSNAP2PDF.EXE requires no pre-configuration and has no DLL dependencies in this version or later. It can be run from the DOS prompt, from a standard Windows short cut or from inside a Clarion application. A number of CHT demo and utility applications, for example *HNDDOCUMENTBUILDER* make extensive use of this document translation application.

CHTSNAP2PDF.EXE can be incorporated into your application from the CHT Snap-In interface where you can find a "CHTSnap2PDF" option. Our template sets an IDE directive to copy the CHTSNAP2PDF.EXE into your application's compile directory. The template also enables a CHT Class called *HNDSNAPCALL* which contains functions for calling the editor from your application.

For DEMO application which incorporate CHTSNAP2PDF.EXE as described above, see HNDDOCUMENTBUILDER.APP, HNDPREVIEWER.APP and HNDTPXHTNEXT.APP. These applications are described here: [HNDDOCUMENTBUILDER.APP](#).

Contact CHT at: support@cwhandy.ca for price and availability of C# source code information.

Gus M. Creces
The Clarion Handy Tools Page
<http://www.cwhandy.ca>
<http://news.cwhandy.ca>

*** CHTSNAPDISK.EXE ***

NAME: CHTSNAPDISK.EXE
TITLE: CHT Snap Disk CMD Utility -- Version: 23.19.08.02
DATE: 9/17/2020
CATEGORY: SNAP-IN C#

DESCRIPTION:

CHTSNAPDISK.EXE (AKA: CHT Snap Disk) is a command-line disk information tool included with the CHT toolkit. It may be used directly from the DOS prompt or it may be incorporated into your application via CHT templates.

CHT Snap Disk does not require any kind of pre-configuration involving a configuration dialog or maintaining registry entries

These DOS command-line inputs are required to use CHT Snap Disk.

(©) The Clarion Handy Tools Page
CHT Snap Disk -- Version: 23.19.08.03

These separate DOS command-line inputs are required to obtain disk or processor information.

To display CHTSNAPDISK.EXE instructions to console:
chtnapdisk --help, -help, or help

To display full disk information to console:
chtnapdisk --info, -info, or info

To generate full disk information to file: chtnapdisk.txt.
chtnapdisk --list, -list or list

To generate full disk information to file: chtnapdisk.txt,
and open with chtnapedit.exe:
chtnapdisk --listedit, -listedit or listedit

To display processor unique ID to console:
chtnapdisk --processor, -processor or processor

To change the LABEL of the designated drive number,
Drive 0 is assumed if [OPT] "DrvNUM" parameter is omitted:
chtnapdisk --label, -label or label "Your label inside quotes" [DrvNum]

To display free space information of all drives to console:
chtnapdisk --free, -free or free

CHT Snap Disk -- Version: 23.19.08.03

Contact CHT at: support@cwhandy.ca for price and availability of C# source code information.

Gus M. Creces
The Clarion Handy Tools Page
<http://www.cwhandy.ca>
<http://news.cwhandy.ca>

*** CHTSNAPEDIT.EXE ***

NAME: CHTSNAPEDIT.EXE
TITLE: CHT Snap Edit CMD Utility -- Version: 23.19.08.01
DATE: 9/17/2020

CATEGORY: SNAP-IN C#

DESCRIPTION:

CHT Snap Edit requires no pre-configuration and has no DLL dependencies in this version or later. It can be run from the DOS prompt, from a standard Windows short cut or from inside a Clarion application. A number of CHT demo and utility applications, for example *HNDDOCUMENTBUILDER* make extensive use of this editor application.

This editor is branched from the same C# source code from which the Clarion IDE code editor was originally built. In fact, if you view Clarion source .CLW files from this editor, editing will feel familiar. The hot keys provided, however, are optimized for editing raw HTML and XML files.

Hot keys are provided for most of standard mark-up operations such as **BOLD**, *ITALIC* Paragraph, Image Links, Web Anchors and numerous others.

CHTSNAPEDIT.EXE can be incorporated into your application from the CHT Snap-In interface where you can find a "CHT Snap Edit" option. Our template sets an IDE directive to copy the CHTSNAPEDIT.EXE into your application's compile directory. The template also enables a CHT Class called HNDSNAPCALL which contains functions for calling the editor from your application.

Two procedure-local templates called SNAPEDITSTARTBUTTON, RUNCHTSNAPEDIT provide an easy way to incorporate various text, HTML or XML editing tasks into a Clarion Window procedure.

For DEMO application which incorporate CHTSNAPEDIT.EXE as described above, see HNDDOCUMENTBUILDER.APP, HNDBULKMAILBATCHER.APP and HNDBULKNETMAILPROMO.APP. These applications are described here: [HNDDOCUMENTBUILDER.APP](#).

Contact CHT at: support@cwandy.ca for price and availability of C# source code information.

Gus M. Creces
The Clarion Handy Tools Page
<http://www.cwandy.ca>
<http://news.cwandy.ca>

*** CHTSNAPGET.EXE ***

NAME: CHTSNAPGET.EXE
TITLE: CHT Snap Get CMD Utility -- Version: 23.19.08.01
DATE: 9/17/2020
CATEGORY: SNAP-IN C#

DESCRIPTION:

CHTSNAPGET.EXE is added as a component to your Clarion application, from the CHT's global "Application Snap-Ins" template. A procedure-local template called SNAPGETSTARTBUTTON provides an easy way to configure the application two different ways from a Clarion window procedure and to trigger download of a file named in a procedure entry field.

CHTSNAPGET.EXE can also be used directly from the DOS command-line or in a DOS BATCH script.

For a DEMO application, see HNDTESTCHTSNAPGET.APP. That application is described here: [DOCS FOR HNDTESTCHTSNAPGET.APP](#).

These DOS command-line inputs are required to fetch a remote file:

** CHTSNAPGET.EXE ** (©) The Clarion Handy Tools Page

These DOS command-line inputs are required to fetch a remote file:
chtsnapget --fetch FILENAME

chtsnapget --fetchuz FILENAME (to unzip on receipt of .zip file)

FILENAME must be the raw name, no path, e.g. "myfile.pdf".
Remote subdirectories are configured in the setup dialog.
Wild card file names are not permitted. URL processing is stopped.

This application must be configured before use.
Configure manually via the command line, or by programmatically completing registry values with Clarion HNDSnapCall.CHTSnapGetConfig() class function (HNDSNAPCALL.INC/.CLW).
To configure from the DOS command line, type: "chtsnapget --config" and complete the provided dialog values.

** CHTSNAPGET.EXE ** Version: 23.19.08.01 CHT-XSA **

Contact CHT at: support@cwandy.ca for price and availability.

Gus M. Creces
The Clarion Handy Tools Page
<http://www.cwandy.ca>
<http://news.cwandy.ca>

*** CHTSNAPMAPI.EXE ***

NAME: CHTSNAPMAPI.EXE
TITLE: CHT Snap MAPI CMD Utility -- Version: 23.19.08.01
DATE: 9/17/2020
CATEGORY: SNAP-IN C#

DESCRIPTION:

CHTSNAPMAPI.EXE (AKA: CHT Snap MAPI) is able to send emails using the system's default MAPI client if such a client is already properly configured. Emails sent this way may be found, after sending, in the "Sent" mail repository of the default client as if they were sent manually from the client interface.

CHT Snap Mapi does not require any kind of pre-configuration involving a configuration dialog or maintaining registry entries. All information required for any individual email-send task can be passed on the application command line, since the installed email client is already pre-configured.

CHTSNAPMAPI.EXE is added as a component to your Clarion application, from the CHT's global "Application Snap-Ins" template. A procedure-local template called SNAPMAPISTARTBUTTON provides an easy way send MAPI email from a Clarion window procedure.

For a DEMO application, see HNDTESTCHTSNAPTOOLS.APP. That application is described here: [DOCS FOR HNDTESTCHTSNAPTOOLS.APP](#).

CHTSNAPMAPI.EXE accepts DOS command-line inputs in order to send text or HTML emails.

Three or four command-line inputs are required:

```
** CHTSNAPMAPI.EXE ** (© The Clarion Handy Tools Page - www.cwandy.ca)
-- Three or four command-line inputs are required:
-- CHTSNAPMAPI "TO" "SUBJECT" "MSG.TXT OR MSG.HTML" ["ATTACH.FILE"]
-- Inputs with spaces or extended characters must be surrounded by double quotes.
-- Otherwise, inputs may be provided without surrounding double quotes.
-- ATTACH.FILE is optional. This option represents optional attachment file(s).
-- Mail results are written to standard output, namely the console.
-- MSG.TXT, MSG.HTML and ATTACH.FILE may include full paths to different folder locations.
-- Multiple TO addresses may be separated by either semi-colons or commas.
-- Multiple attachments may be separated by either semi-colons or commas.
** CHTSNAPMAPI.EXE ** Version: [23.19.08.01] **
```

Contact CHT at: support@cwandy.ca for price and availability of C# source code information.

Gus M. Creces
The Clarion Handy Tools Page
<http://www.cwandy.ca>
<http://news.cwandy.ca>

*** CHTSNAPSEND.EXE ***

NAME: CHTSNAPSEND.EXE
TITLE: CHT Snap SEND CMD Utility -- Version: 22.18.01.00
DATE: 9/17/2020
CATEGORY: SNAP-IN C#

DESCRIPTION:

CHTSNAPSEND.EXE source code, because it is not in Clarion app format, is available from CHT via special request in the form of a C# wizard. Once imported as a project into Visual Studio, the developer can customize the application to his own specifications should that ever be necessary.

CHTSNAPSEND.EXE (AKA: CHT Snap Send), is an error-reporting tool which integrates, with your Clarion application via CHT templates. It is able to provide screen-snaps of error-causing procedure windows as well as lower-level error codes and error messages issued by the application when an error is encountered.

CHTSNAPSEND greatly increases the accuracy of error reporting from applications as it eliminates most of the error-reporting steps that end-users are likely to have difficulties with.

CHT Snap Send email settings can be configured manually using its internal setup dialog, or by leveraging a Clarion application to write the necessary registry entries from our Snap-Ins template interface.

This application does not take command-line inputs. It is leveraged by CHT's Snap-In template interface provided globally on the CHTSNAPGET tab and procedure-locally on a procedure tab that appears when the global CHTSNAPGET tab is enabled.

The "CHT Snapsend" template element, when enabled, writes global SnapSend configuration entries to the Windows Registry in order to customize some of the actions and messages that CHT SnapSend performs and displays, while taking application screen snaps. Further, this template causes the auto-population of multiple instances of a procedure-level template (one per window procedure) called "ProcedureSnapins(HndTools)". ProcedureSnapins(HndTools) instances are de-populated when ApplicationSnapins(HndTools) is removed. See the docs for ProcedureSnapIns to discover the procedure-specific configurations that are possible with that auto-populated template.

EmbedSnapCall: This template may be used to attach to your application, at the procedure level, a CHT Code module containing a single class called HNDSnapCall. This extension creates a single, local instance of the HNDSnapCall Class which is used to call a variety of "CHT SNAP-IN" executable tools in a manner that resembles standard DLL calls, using parameter passing on the command line.

ProcedureSnapIns: On a procedure-by-procedure basis, this template allows the developer to optionally expose the state of procedure-local and global variables while the procedure is running. These variables are added into a text file that is sent via email along with the procedure-window snapshot taken by CHT SnapSend. This template is also the parent of a button template called "HandyWindowsStopStartSnapSendButton(HndTools)". That means the button template does not appear in the IDE's control templates list until ApplicationSnapins' "Enable?" switch is checked.

Contact CHT at: support@cwhandy.ca for price and availability information.

Gus M. Creces
The Clarion Handy Tools Page
<http://www.cwhandy.ca>
<http://news.cwhandy.ca>

*** CHTSNAPSMTP.EXE ***

NAME: CHTSNAPSMTP.EXE
TITLE: CHT Snap SMTP CMD Utility -- Version: 23.19.08.04
DATE: 9/17/2020
CATEGORY: SNAP-IN C#

DESCRIPTION:

CHTSNAPSMTP.EXE (AKA: CHT Snap SMTP), is able to send email via SMTP protocol from any Windows system with an internet connection and a viable SMTP server. Even web-based email providers such as HOTMAIL, GMAIL and YAHOO may be leveraged by CHTSNAPSMTP since they provide also for unique SMTP setups. CHTSNAPSMTP can also send secure, encrypted email assuming that the SMTP server being accessed provides for it.

CHT Snap Get can be configured manually from the command line, or by leveraging a Clarion application to write the necessary registry entries. Once configured, to a specific SMTP server and account emails can be sent from the DOS command line.

CHTSNAPSMTP.EXE is added as a component to your Clarion application, from the CHT's global "Application Snap-Ins" template. A procedure-local template called SNAPSMTPSTARTBUTTON provides an easy way send SMTP email from a Clarion window procedure.

For a DEMO application, see HNDTESTCHTSNAPTOOLS.APP. That application is described here: [DOCS FOR HNDTESTCHTSNAPTOOLS.APP](#).

These DOS command-line inputs are required to send text or HTML emails:

Three or four DOS command-line inputs are required in order to send text or html mail with optional attachments:

```
** CHTSNAPSMTP.EXE ** (©) 2019 The Clarion Handy Tools Page
-- Three or four DOS command-line inputs are required
-- in order to send text or html mail with optional attachments:
-- CHTSNAPSMTP "TO" "SUBJECT" "MSG.TXT" OR "MSG.HTML" ["ATTACH.FILE"]
-- Inputs with spaces or extended characters use double quotes.
-- Otherwise, inputs may be provided without surrounding double quotes.
```

-- Individual inputs must be separated by one or more spaces.
-- ATTACH.FILE is optional. It represents optional attachment file(s).
-- CHT Snap SMTP must be configured before mail can be sent.
-- Type "CHTSNAPSMTP --config" at the DOS prompt to configure.
-- Mail results are written to the console and to chtsnapsmtp.log.
-- MSG.TXT or MSG.HTML and ATTACH.FILE [OPT] may include full paths to folders.
-- Multiple TO addresses may be separated with either semi-colons or commas.
-- Multiple attachments may be separated with either semi-colons or commas.
-- Issue "--rcpt" alone on command line prior to sending for read-receipt.
-- Issue "--norcpt" prior to sending to cancel read-receipt setting.

** CHTSNAPSMTP.EXE ** Version: [23.19.08.04] **

Contact CHT at: support@cwandy.ca for price and availability of C# source code information.

Gus M. Creces
The Clarion Handy Tools Page
<http://www.cwandy.ca>
<http://news.cwandy.ca>

*** CHTSNAPZIP.EXE ***

NAME: CHTSNAPZIP.EXE
TITLE: CHT Snap Zip CMD Utility -- Version: 23.19.08.03
DATE: 9/17/2020
CATEGORY: SNAP-IN C#

DESCRIPTION:

CHTSNAPZIP.EXE (AKA: CHT Snap Zip) is a command-line zip/unzip tool included with the CHT toolkit. It may be used directly from the DOS prompt or it may be incorporated into your application via CHT templates.

CHT Snap Zip does not require any kind of pre-configuration involving a configuration dialog or maintaining registry entries. All information required for any individual "zipping" task can be passed on the application command line.

CHTSNAPZIP.EXE is added as a component to your Clarion application, from the CHT's global "Application Snap-Ins" template. Three procedure-local templates called SNAPZIPSTARTBUTTON, SNAPUNZIPSTARTBUTTON and SNAPZIPDIRSTARTBUTTON, provides an easy way to incorporate various zipping tasks into a Clarion Window procedure.

For a DEMO application, see HNDTESTCHTSNAPZIP.APP. That application is described here: [DOCS FOR HNDTESTCHTSNAPZIP.APP](#).

These DOS command-line inputs are required to zip files or directories.

** CHTSNAPZIP.EXE - A CHT Snap Utility ** Version: [23.19.08.05]
** (©) 2019 The Clarion Handy Tools Page - www.cwandy.ca

-- The ZR parameter recurses subdirectories and accepts 3 inputs.
-- It produces a zip with folder names included inside the zip.
-- The second input may include a target path for placement of the output zip file.
-- When no zip target path is given, the zip file is created in the same directory
-- as the CHTSNAPZIP.EXE executable.
-- The "rootdir" parameter used without a mask must end with a backslash.
-- [mask] is optional after the backslash [e.g. \hnd*.tps] to filter files.

chtsnapzip <[\path\]zipfilename>

-- The ZZ parameter skips subdirectories and accepts 3 inputs.
-- It produces a flat zip without the folder names in the zip.
-- The second input may include a target path for placement of the output zip file.
-- When no zip target path is given, the zip file is created in the same directory
-- as the CHTSNAPZIP.EXE executable.
-- The "sourcedir" parameter used without a mask must end with a backslash.
-- [mask] is optional after the backslash [e.g. \hnd*.tps] to filter files.

chtsnapzip <[\path\]zipfilename>

-- The ZL parameter lists zip contents and accepts 2 inputs.
It produces a comma-delimited list of zip files for pickup by calling apps.

chtsnapzip <[\path\]zipfilename>

-- The ZF parameter lists zip contents and accepts 2 inputs.
It produces a fully formatted list intended for easy reading from the DOS console.

chtnapzip <[\path\]zipfilename>

-- The UZ parameter unzips the zip file contents and accepts 3 inputs.
-- The second input may include a full path to the zip file.
-- When no zip path is given, the zip file is assumed to be located
-- in the same directory as the CHTSNAPZIP.EXE executable.
-- The "targetdir" parameter used without a mask must end with a backslash.
-- [mask] is optional after the backslash [e.g. \hnd*.tps] to filter files.

chtnapzip <[\path\]zipfilename>

** CHTSNAPZIP.EXE - A CHT Snap Utility ** Version: [23.19.08.05]

Contact CHT at: support@cwandy.ca for price and availability of C# source code information.

Gus M. Creces
The Clarion Handy Tools Page
<http://www.cwandy.ca>
<http://news.cwandy.ca>

*** HNDCLEAN.EXE ***

NAME: HNDCLEAN.EXE
TITLE: CHT Clarion Gen-Code Cleaner -- Version: 22.18.01.00
DATE: 9/17/2020
CATEGORY: SNAP-IN C#

DESCRIPTION:

HNDCLEAN.EXE source code, because it is not in Clarion app format, is available from CHT via special request in the form of a C# wizard. Once imported as a project into Visual Studio, the developer can customize the application to his own specifications should that ever be necessary.

HNDCLEAN.EXE (AKA: CHT Clarion Gen-Time Code Cleaner), is a replacement for the Clarion IDE's inadequately functioning "Clean Solution" operation. It erases APPNAME*.CLW and APPNAME*.OBJ files, and more, when configured to match the IDE's .RED file [RELEASE] or [DEBUG] configurations.

HNDCLEAN.EXE can be installed to the IDE's "External Tools" interface under the name "CHT Clarion Gen-Code Cleaner". The "Arguments" field should be completed as explained in our document located at <http://www.cwandy.ca/howtousehndclean.htm>.

If you've made modifications to your application that requires the application to fully regenerate, with this tool installed, you can erase the .CLW and .OBJ files associated with the application currently loaded. Then by using the IDE's "Build Generate and Make" menu, the entire application is regenerated from scratch, and compiled with all hand-embedded changes and template interface changes being correctly applied.

Contact CHT at: support@cwandy.ca for price and availability of C# source code information.

Gus M. Creces
The Clarion Handy Tools Page
<http://www.cwandy.ca>
<http://news.cwandy.ca>

CATEGORY: TOOLKIT UTILITY

*** HNDBASE64CVT.APP ***

NAME: HNDBASE64CVT.APP
TITLE: Base64 File Encrypt Utility [V10.20.09]
DATE: 9/17/2020
CATEGORY: TOOLKIT UTILITY
CLASSES: [List App Classes](#)
TEMPLATES: [List App Templates](#)

DESCRIPTION:

This "CHT Batch Bot", HNDBASE64CVT.EXE is a command-line utility that can be called from another

application using, for example, Clarion RUN or one of CHT's many ShellRun() functions. It can also be leveraged from a DOS batch file, or a windows .VBS script.

HNDBASE64CVT.EXE source code, because it is in Clarion .APP format, is included as a standard part of the CHT toolkit, provided both in .APP format as well as ready-to-run, in .EXE format.

The application accepts several command line parameters that describe the input file and output path. These parameters are as follows:

```
--help
    [OPT] Displays a help screen. Writes hndbase64cvthelp.txt.
--input
    [REQ] Path and name of input file
--output
    [OPT] Output path only - local to this exe if omitted.
--mode
    [OPT] Encode or decode - defaults to encode if omitted.
```

The output file name is formatted from the input file with a .h64 extension. For example: test.pdf becomes testpdf.h64

CATEGORY: XML2HTML

*** HNDXML2HTMLGEN.APP ***

NAME: HNDXML2HTMLGEN.APP
TITLE: XML2HTML Command Utility [V10.20.09]
DATE: 9/17/2020
CATEGORY: XML2HTML
CLASSES: [List App Classes](#)
TEMPLATES: [List App Templates](#)

DESCRIPTION:

This "CHT Batch Bot", HNDXML2HTMLGEN.EXE, runs from various other CHT Batch-Bots, Snap-Ins and Utility Applications in order to generate user-embedded XML template files into stand-alone HTML pages and forms.

It may also be run directly from a DOS batch file or Clarion application implementing XML to HTML processing.

CHT Snap-Edit now calls this HTML generator exclusively to transform XML template files into HTML before previewing.

This generator implementation can access and call remote procedures established in a CHT client data server to request data packages which are then merged into the HTML document. Several example XML templates have been provided to illustrate insertion of remote data from a CHT server by UCR (user custom request): 000CHTRECEIPT.XML for example.

A document is in preparation which will explain in detail the structuring of various interchangeable XML template designs with which this generator-application will operate to create HTML. For example, mail-merge letters, static web pages, interactive web forms, tile menus and more.

** GENERATING **

--input [REQ]
Name of the XML input template file.

--output [OPT]
Name of the HTML output template file.

--prew [OPT]
Preview HTML after generate.

--noprew [OPT]
No preview HTML after generate.

** CONFIGURE DATA SERVER **

--config [OPT]
Configure remote data server for secure access.

** HELP **
--help [CFG]

Pops up a splash help screen.

©1996-2020
XML2HTML™
Clarion Handy Tools