CHT Applications

HOME CLASSES TPLS APPS

TEMPLATES, CLASSES, UTILITIES, APPS, PROJECTS

CHT APPLICATIONS
FULL SET
(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
    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 mult-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 setttings.

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.
            Multipe 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 oprerations or errors are logged into "hndzipnet.log" located in the same directory as the
hndzipnet.exe.
```

CATEGORY:
BROWSES EXPB

***** HND2.APP *****

NAME:	HND2.APP
TITLE:	BROWSES EXPB (ISAM) [V10.20.09]
DATE:	9/17/2020
CATEGORY:	BROWSES EXPB
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This application demonstrates primarily, the use of a Clarion Handy Tools Template called ExplorerBrowse (EXPB); hence the name of the application.

EXPB is an extension template that extends a standard ABC browse with some 25+ features including header-click browse column sorting, built-in locators, greenbar support, and much more.

EXPB itself can also be extended by adding a locator template called ExplorerBrowseLocatorControl and a query control called QueryParsingFilterControl (QPFC). Examples of both are provided. Ancillary templates called HandyQueryProcessLimiter and HandyQueryReportLimiter allow you to pass queries directly from the browse to a report or process via a window button template called HandyQueryProcessButton. See the procedure called Demobrowse4 for examples of this powerful feature.

The major CHT classes involved are: HNDBrowse, HNDBrwFilter and HNDParse. Subscribing developers should take some time to study the header files for these three classes to become familiar with available properties and methods in order to gain the maximum amount of utility from them.

*** HNDISAMPOPFAVEXPB.APP ***

NAME: HNDISAMPOPFAVEXPB.APP
TITLE: BROWSES EXPB (ISAM) [V10.20.09]
DATE: 9/17/2020
CATEGORY: BROWSES EXPB
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application demonstrates primarily, the use of a Clarion Handy Tools Template called ExplorerBrowse. This is an extension template that extends a standard ABC browse with some 25+ features including header-click browse column sorting, built-in locators, greenbar support, and much more.

ExplorerBrowse itself can also be extended by adding a variety of extension templates. This application adds PopFavoriteQueries_EXPB to provide a way to inject prebuilt queries into the browse to limit the records being displayed.

PopFavoriteQueries_EXPB provides a query control, a clear button, a favorites query selection control, a query builder button, a query save button and a query delete button.

The application also illustrates a procedure built with a CHT Jumpstart that allows you to manage and modify or insert pre-built queries.

*** HNDSQL1.APP ***

NAME: HNDSQL1.APP
TITLE: BROWSES EXPB (SQL) [V10.20.09]
DATE: 9/17/2020
CATEGORY: BROWSES EXPB
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

* BRWSQL1_ExplorerBrowse * - CHT Query Builder in return-SQL mode populated with Explorerbrowse.

* BRWSQL2_ExplorerBrowse * - ExplorerBrowse Locator ControlEx in drill-down mode.

* Hiding/unhiding columns programmatically.
* Hidden columns are hidden also in CHT Query Builder.
* SQL Tracing set to ON in both browses, trace with HNDSQLTRACE.EXE.

This demonstration application requires the sample data base provided with MICROSOFT ACCESS, called Northwind. The application will find the data base and register it with ODBC if not already registered.

*** HNDSQL3.APP ***

NAME: HNDSQL3.APP
TITLE: BROWSES EXPB (SQL) [V10.20.09.0379]
DATE: 9/17/2020
CATEGORY: BROWSES EXPB
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

* A CHT ExplorerBrowse Template.

* A CHT ExplorerBrowseQueryBuilder Template.

* A CHT PopFavoriteQueries_EXPB Template.

* A CHT ExplorerBrowseLocatorControlEx Template.

* A CHT ExplorerBrowseOneColumnQueryControl Template.

* A CHT Explorerbrowse range-limiting query example.

This demonstration application requires the sample data base provided with MICROSOFT ACCESS, called Northwind. The application will find the data base and register it with ODBC if not already registered.

*** HNDSQLPOPFAVEXPB.APP ***

NAME: HNDSQLPOPFAVEXPB.APP
TITLE: BROWSES EXPB (SQL) [V10.20.09]
DATE: 9/17/2020
CATEGORY: BROWSES EXPB
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application demonstrates CHT Explorerbrowse used in an SQL setting in conjunction with query template PopFavoriteQueries_EXPB.

It illustrates these concepts:
*** A Explorerbrowse implementation running on SQL Data Tables.
*** CHT Query template PopFavoriteQueries_EXPB.

This demonstration application requires the sample data base provided with MICROSOFT ACCESS, called Northwind. The application will find the data base and register it with ODBC if not already registered.

CATEGORY:
BROWSES HMB

*** HND3.APP ***

NAME: HND3.APP
TITLE: BROWSES HMB (ISAM) [V10.20.09]
DATE: 9/17/2020
CATEGORY: BROWSES HMB
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

HMB extends a normal ABC browse template configured with very minimal settings (see procedure: BRW1_PreparingForMarkerTemplates).

Varous procedures in this application illustrate different implementations of HMB designed to achieve significantly improved ABC browse performance.

* BRW2_HandyMarkerBrowseBasicDemo * HMB with auto-filter preset queries illustating Clarion filter syntax and CHT query language syntax

* BRW3_HandyMarkerBrowseLocator * HMB with 3 column-specifc query controls for NAME, ADDRESS and CITY. Queries are additive until cleared.

* BRW4_HandyMarkerBrowseLocator * HMB with Handy Marker Browse Locator Control Template

* BRW5_HandyMarkerBrowseDemo * HMB with Query Parsing Filter Control (QPFC) template as well as 5 different export examples using Handy Query Process Limiter template.

* BRW6_HandyMarkerBrowseJoins * 2 HMB Browses on separate tabs but joined as parent and child.

* BRW7_HandyMarkerBrowseWithSetsControl * HMB with Marker Sets Control template and QPFC.

Other marking-related templates ancillary to HMB allow you pass marked records to processes for export to HTML, TEXT, FILE, EXCEL and print reports limited to only the records marked in your browse.

These templates are:
HandyMarkerBrowesProcessLimiter, HandyMarkerBrowseReportLimiter, HandyMarkerBrowseProcessButton,

HandyMarkerBrowseReportButton.

An HMB ancillary template called HandyMarkerBrowseSetsControl allows you to store and recall sets of marked records by name (see procedure BRW7_*).

*** HNDSCHOOL.APP ***

NAME: HNDSCHOOL.APP
TITLE: BROWSES HMB (SQL) [V10.20.09]
DATE: 9/17/2020
CATEGORY: BROWSES HMB
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This is an CHT demo actively re-built with CHT templates on ABC demo application SCHOOL.APP; hence the name HNDSCHOOL.APP.

Its purpose is to study the use of the SQLite driver with CHT browses, and query templates set to UseSQL=True. We chose to "CHT-ify" this application specifically because it uses several SQLite data tables located in a repository file called SCHOOL.SQLITE.

A second purpose is, of course, to provide some visual and functional contast between a raw, wizard-produced, really UGLY and semi-functional, ABC application and ABC applications extended, enhanced and functionalized by a variety of CHT templates.

To analyse the allocation of templates in this application, open the application in the Clarion IDE and navigate to Global Extensions, select AACHTControlPanel and click on the menu item "Templates By Proc". This sequence of actions triggers an HTML document to display which enumerates and names all templates used in the application by procedure.

We chose to apply, as always, our "minimalist" screen design which avoids unnecessary use of tab controls, and shuns MDI child windows. Control buttons are for the most part kept to the toolbar and when placed on a window are always above the edit fields or list box.

This application applies, for the most part, CHT's browse extension template HandyMarkerBrowse (HMB); hence comes the name of the application. However, one procedure utilizes ListBoxBrowseExtender (LBX) which illustrates that an underlying ABC browse is no longer necessary to build fantastic Clarion browses. LBX browses are the bright future of data browsing with Clarion, ABC browses are the clunky past.

*** HNDSQL2.APP ***

NAME: HNDSQL2.APP
TITLE: BROWSES HMB (SQL) [V10.20.09]
DATE: 9/17/2020
CATEGORY: BROWSES HMB
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

* SQLFilter_HandyMarkerBrowse * - Illustrates a hand-built SQL query control using a standard entry control coded to post an SQL query to the data table.

* Hiding/unhiding columns programmatically.
* SQL Query injection from a button click event
* SQL Query tracing on and off from a button click event

This demonstration application requires the sample data base provided with MICROSOFT ACCESS, called Northwind. The application will find the data base and register it with ODBC if not already registered.

*** HNDSQL2_DUTCH.APP ***

NAME: HNDSQL2_DUTCH.APP
TITLE: BLADEREN DOOR HMB (SQL) [V10.20.09]
DATE: 9/17/2020

CATEGORY:	BROWSES HMB
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

* SQLFilter_HandyMarkerBrowse * - Illustreert een met de hand gebouwde SQL-querybesturingselement met behulp van een standaardvermeldingsbeheer dat is gecodeerd om een SQL-query in de gegevenstabel te plaatsen.

* Verbergen / verbergen kolommen programmatisch.
* SQL Query-injectie van een knop klik gebeurtenis
* SQL Query tracing in en uit van een knop klik gebeurtenis

Deze demonstratietoepassing vereist de voorbeeldgegevensbank die is voorzien van MICROSOFT ACCESS, genaamd Northwind. De applicatie vindt de databank en registreert deze bij ODBC als deze nog niet is geregistreerd.

*** HNDSQL2_FRENCH.APP ***

NAME:	HNDSQL2_FRENCH.APP
TITLE:	NAVIGATEURS HMB (SQL) [V10.20.09]
DATE:	9/17/2020
CATEGORY:	BROWSES HMB
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

* SQLFilter_HandyMarkerBrowse *
- Illustre un contr�le de requ�te SQL construit � la main � l�aide d�un contr�le d�entr�e standard cod� pour afficher une requ�te SQL dans la table de donn�es.

* Masquer/masquer les colonnes par programme.
* Injection SQL � partir d�un clic de bouton
* SQL tra�ant sur et hors tension � partir d�un �v�nement de clic de bouton.

Cette application de d�monstration n�cessite l�exemple de base de donn�es fournie avec MICROSOFT ACCESS, appel�e Northwind. L�application trouvera la base de donn�es et l�enregistrera aupr�s d�ODBC si elle n�est pas d�j� enregistr�e.

CATEGORY:
BROWSES LBX

*** HNDLBXDM.APP ***

NAME:	HNDLBXDM.APP
TITLE:	BROWSES LBX (ISAM) [V10.18.08]
DATE:	9/17/2020
CATEGORY:	BROWSES LBX
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This application demonstrates primarily a record-marking template, that assists in marking records using ordinary Clarion list boxes.

That template is called ListBoxBrowseExtender or "LBX" for short.

Some Features Illustrated In This Application:
* ListBoxBrowseExtender Has Green Bar.
* LBX has multiple optional Query Control templates (_LBX)
* Optionally use ORDER BY In Query Control For Mult-Column Sort.
* ListBoxBrowseExtender Honours Retain-Record.
* ListBoxBrowseExtender Generates Data Queue

* ListBoxBrowseExtender connects update forms
* ListBoxBrowseExtender has queue auto-fill on startup
* LBX has optional pre-filter to limit auto-fill on startup
* LBX has selectable fill-queue methods for local or remote data

Four example procedures illustate some powerful LBX features;

* PROCEDURE: ListBoxBrowseExtenderInQueueQuery()
This procedure illustrates a query template called InQueueQueryControl_LBX. This control filters the records already in the list box queue, into a secondary queue called "the shadow queue" and displays the result in the same browse. Clearing this filter resets to the starting or "Main" queue and displays the original record set. All of this except the original fill-queue happens without ANY further file access being needed.

* PROCEDURE: ListBoxBrowseExtenderSortOrderFieldFilter()
This procedure illustrates a template called SortOrderFieldFilter_LBX. This control requires the user to enter only a string of characters to match something contained in the sort-order field. As above, matching data are inserted into the shadow queue and are displayed in the same browse. As above, also, clearing this filter resets to and displays the "Main" queue record set.

* PROCEDURE: ListBoxBrowseExtenderUniversalQueryControl()
This procedure illustrates a template called UniversalQueryControl_LBX. This control when empty fills the queue with all table/view records unless a pre-filter is added to the template config. Click the "Queue Data" button to switch to the "Shadow Queue" and enter a query. These records are filtered from the Main queue into the Shadow queue. Click back on the "Queue Data" button and the browse switches back to the Main queue with all original records intact.

* PROCEDURE: ListBoxBrowseExtenderPopFavoriteQueries()
This procedure populates a template called PopFavoriteQueries_LBX. This control opens a table called PreQueries and scans it for queries designated for the view name defined on the template. Queries may be optionally added, or deleted at run time and can be selected from a popup menu. The PreQueries table is defined in HNDO.DCT from which it can be imported into your own dictionary. Prequeries may be ISAM or SQL as required by your application.

*** HNDLBXSQL.APP ***

NAME: HNDLBXSQL.APP
TITLE: BROWSES LBX (SQL) [V10.20.09]
DATE: 9/17/2020
CATEGORY: BROWSES LBX
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

It illustrates these concepts:
*** Filling the browse queue directly from a single file
*** Filling the browse queue from multiple files using a view
*** Filling the browse queue manually by embedding
*** Filling the browse queue using class method SQLSelect
*** Filling the browse queue from a remote CHT Server Type 2
*** Filling the browse queue from a remote CHT Server Type 3
*** Incorporating an Update Form with UpdateButtons_LBX.
*** A hand-code created/managed query control.

This demonstration application requires the sample data base provided with MICROSOFT ACCESS, called Northwind. The application will find the data base and register it with ODBC if not already registered.

*** HNDLBXSQLEX.APP ***

NAME: HNDLBXSQLEX.APP
TITLE: BROWSES LBX (SQL) [V10.20.09]
DATE: 9/17/2020
CATEGORY: BROWSES LBX
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

It illustrates these concepts:

*** An LBX browse with UniversalQueryControl_LBX
*** An LBX browse with InQueueQueryControl_LBX
*** Connecting an update form with UpdateButtons_LBX

*** PROCEDURE: LBXWithUniversalQueryControl()
The browse queue is loaded without pre-filter initially from the data table. The query control provides for entry of CHT queries which will re-fill the "Main" queue with a new subset of data directly from the data table while "Main Queue Data" is displayed. Click over to the "Shadow Queue Data" and CHT queries in the query control will display subsets of the "Main" data queue without data table access being excercised. Clearing the Main Queue Data query control reloads the Main queue from the data table.

*** PROCEDURE: LBXWithInQueueQueryTemplate()
The browse queue is loaded without pre-filter initially from the data table. The query control provides for entry of a CHT query string. Records matching the query string are inserted into the "Shadow" queue which is then displayed. "Main" queue records remain undisturbed. Clearing the query empties the shadow queue and re-displays the Main queue. No file access is required after the initial data load from the data tables. A reset from file button, reloads the main queue from the data tables.

This demonstration application requires the sample data base provided with MICROSOFT ACCESS, called Northwind. The application will find the data base and register it with ODBC if not already registered.

*** HNDPEOPLE_LBX.APP ***

NAME:	HNDPEOPLE_LBX.APP
TITLE:	BROWSES LBX (ISAM) [V10.20.09]
DATE:	9/17/2020
CATEGORY:	BROWSES LBX
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

LBX "FillFrom" tabs used:
FillFromFile() -- Local Data Tables
FillFromView() -- Local Data Tables
FillFromCHTServer() -- Remote Data Tables

Upcoming versions of this application will add additional browses based on LBX "FillFrom" tabs:
FillFromSQLSelect -- Local Data Tables
FillByEmbedding -- Local or Remote Data Tables
FillFromUCRServer -- Remote Data Tables

Hot Key Alerts:
** MouseLeft - Mark a single record, unmarks any others.
** ShiftMouseLeft - Mark range between selected and click.
** CtrlShiftMouseLeft - Mark from selected to click position.
** CtrlMouseLeft - Toggle record marked/unmarked.
** CtrlMouseLeft - On header, pops up column hide menu.
** CtrlSpace - Same effect as CtrlMouseLeft
** ShiftUp - Mark next record up from selected.
** ShiftDown - Mark next record down from selected.
** UpKey - Switch to single record mode and move up.
** DownKey - Switch to single record mode and move down.
** CtrlUp - Move cursor up with no marking or unmarking.
** CtrlDown - Move cursor down with no marking.
** ShiftPgDn - Mark all from selected to top of current page.
** ShiftPgUp - Mark all from selected to end of current page.
** ShiftHome - Mark all from selected to first record.
** ShiftEnd - Mark all from selected to last record.
** HomeKey - Unmark all and select first record.
** CtrlHome - Unmark all and select first record.
** EndKey - Unmark all and select last record.
** CtrlEnd - Unmark all and select first record.
** PgUpKey - Unmark all and select last record on page.
** PgDnKey - Unmark all and select first record on page.
** CtrlA - Select all.

CATEGORY:
BROWSES MIXED

*** HNDACCES.APP ***

NAME: HNDACCES.APP
TITLE: BROWSES MIXED (SQL) [V10.20.09]
DATE: 9/17/2020
CATEGORY: BROWSES MIXED
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

The following listing names and provides a summary of the illustative procedures provided with this application:

** ABCBRW_ABCBrowseOnly ** - Procedure illustrates ABC browse only with no CHT browse extensions, PLUS Query Parsing Filter Control (QPFC).

** EXPB1_ExplorerBrowse ** - Procedure uses ExplorerBrowse with Explorer Browse Locator ControlEx template.

** EXPB2_ExplorerBrowse ** - Procedure uses ExplorerBrowse with Explorer Browse Locator Control template.

** EXPB3_ExplorerBrowse ** - Procedure uses Explorerbrowse with built-in filtering locator enabled.

** EXPB4_ExplorerBrowse ** - Procedure uses Explorerbrowse with serveral ExplorerBrowse One Column Query Control templates populated on the window.

** EXPB5_ExplorerBrowse ** - Procedure uses Explorerbrowse with Query Parsing Filter Control template.

** HMB1_HandyMarkerBrowse ** - Procedure uses HandyMarkerBrowse with built in filtering locator enabled.

** HMB2_HandyMarkerBrowse ** - Procedure uses HandyMarkerBrowse with serveral Handy Marker Browse One Column Query Control templates populated on the window.

All procedures in this application are SQL-Trace enabled, allowing you, with HNDSQLTRACE.EXE to trace all back-end SQL calls from the application to the data tables.

This demonstration application requires the sample data base provided with MICROSOFT ACCESS, called Northwind. The application will find the data base and register it with ODBC if not already registered.

CATEGORY:
CDROM BURNING

*** HNDCDROMEXAMPLE.APP ***

NAME: HNDCDROMEXAMPLE.APP
TITLE: Handy Backup To CDROM [V10.20.09.0455]
DATE: 9/17/2020
CATEGORY: CDROM Burning
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application uses CHT template EmbedHNDCDROMClass and multiple copies of HandyRegionButton.

With this application it is possible to select (or drag and drop) one or more directories to be backed up to CD ROM. Note that sub-directories of selected directories are automatically backed up too.

Target CDROM disks may be CD-R, CD-RW, DVD-R, DVD-RW and other, less commonly used variations of these.

CATEGORY:
CHT BROWSES REMOTE

*** HNDPPLLBX_REMOTE.APP ***

NAME: HNDPPLLBX_REMOTE.APP
TITLE: PEOPLE_LBX CLNT SAFE [V10.20.09]
DATE: 9/17/2020
CATEGORY: CHT Browses Remote
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

The base browse template is ListBoxBrowseExtender (LBX) configured as: "FillFromCHTServer() -- Remote Data Tables".

The remote server is CHT demo application called HNDPPLLBX_SERVER.APP which you can configure and run on your own hardware.

For easy testing however, we're running an instance of this same server application on our hardware, located in Toronto, Canada.

Instructions that will allow you to connect to our server application are included with the installation instructions in CHTTODO.CLW dated (August 11, 2020).

*** HNDPPLLBX_SERVER.APP ***

NAME: HNDPPLLBX_SERVER.APP
TITLE: PEOPLE_LBX SVR SAFE [V10.20.09]
DATE: 9/17/2020
CATEGORY: CHT Browses Remote
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

You can also compile, configure and run this serverapplication locally on your test computer or on your network, and also connect to it with HNDPPLLBX_REMOTE.APP.

The necessary components for this server and the matching client mentioned above are listed in your CHT installation information file CHTTODO.CLW. See the entry dated (August 11, 2020). All required components are shipped with your CHT toolkit installation

CATEGORY:
CHT CONTROL TEMPLATES EXAMPLE

*** HNDCONTROLTPLS.APP ***

NAME: HNDCONTROLTPLS.APP
TITLE: CHT Controls Test Bench [V10.20.09]
DATE: 9/17/2020
CATEGORY: CHT Control Templates Example
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

CHT CONTROL Templates illustrated so far in HNDCONTROLTPLS.APP, in no particular order of importance:

** AppBuildNumberStamp,
** AppCopyRightStamp
** AppBuildDateTimeStamp
** HandyAddressControls
** HandyCOMProgressBarControl
** HandyCOMSliderControl
** HandyThreadLimiter
** HandyToolBarGradient
** HandyWindowGradient

** VisitHandyControl
** HandyHelpFileFetch
** HandyMailMeControl_EWF
** HandyCOMMAPIControl
** HandyRulerControl
** HandySliderControl
** HandyTellMeControl
** VisitMeControl
** HandyHyperLink_EWF
** HandyWindowDismissButton
** HandyWindowsNotePadButton_EWF
** HandyCOMDigitalClockControl
** HandyHelpButtonControl
** HandyRegionButton
** HandyCOMAnimationControl
** HandyCOMDatePickerControl
** HandyDriveSelectorControl
** HandyFileSelectorControl
** HandyGradientEllipse
** HandyGradientRoundBox
** HandyHelpFileTopicButton
** HandyPopupCalendarButton
** Still many more to be added next update

CATEGORY:
CHT FIRST START EXAMPLE

*** HNDSTART.APP ***

NAME: HNDSTART.APP
TITLE: Start New Application [V10.20.09]
DATE: 9/17/2020
CATEGORY: CHT First Start Example
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application demonstrates primarily, the use of several templates that help you enhance the visual appeal of your application.

ApplicationImagesEx keeps track of the images used in your application and controls consistency of your visual style in a format that can be saved for reuse and more.

EmbedWindowFunctions provides, advanced icon menus, application-in-tray, window boxes and more.

Procedures in this application are available as JUMP START procedures via the CHTJumpStartProceduresUtility Template under the topic Main And Splash.

CATEGORY:
CHT JUMPSTARTS TEST BENCH

*** HNDJUMPBENCH00.APP ***

NAME: HNDJUMPBENCH00.APP
TITLE: CHT Jumpstarts Bench (00) [V10.20.09]
DATE: 9/17/2020
CATEGORY: CHT Jumpstarts Test Bench
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application is one of several applications which we use internally to build and maintain CHT Jumpstarts.

Hence, the procedures in this application are available as JUMP START procedures via one of CHT's three Utility Templates named:
CHTJumpStartProceduresUtility1
CHTJumpStartProceduresUtility2
CHTJumpStartProceduresUtility3

You are able to identify the name of the Jump Start Procedure being illustrated and tested here from the MAIN MENU of the application.

We're making this application available to you as an alternative to using CHT JUMP START templates. Use our AAExportProcedure Utility Template to export a procedure from this application as a .TXA. Then, use our AAImportProcedure Utility Template to import the resulting .TXA to your own application.

*** HNDJUMPBENCH01.APP ***

NAME:	HNDJUMPBENCH01.APP
TITLE:	CHT Jumpstarts Bench (01) [V10.20.09]
DATE:	9/17/2020
CATEGORY:	CHT Jumpstarts Test Bench
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This application is one of several applications which we use internally to build and maintain CHT Jumpstarts.

Hence, the procedures in this application are available as JUMP START procedures via one of CHT's three Utility Templates named:
CHTJumpStartProceduresUtility1
CHTJumpStartProceduresUtility2
CHTJumpStartProceduresUtility3

You are able to identify the name of the Jump Start Procedure being illustrated and tested here from the MAIN MENU of the application.

We're making this application available to you as an alternative to using CHT JUMP START templates. Use our AAExportProcedure Utility Template to export a procedure from this application as a .TXA. Then, use our AAImportProcedure Utility Template to import the resulting .TXA to your own application.

CATEGORY: CALCULATORS

*** HNDCALCULATOR.APP ***

NAME:	HNDCALCULATOR.APP
TITLE:	CHT Calculator Demo [V10.18.01.0287]
DATE:	9/17/2020
CATEGORY:	Calculators
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

Extension template EmbedCalculatorFunctions attaches a CHT class called HNDCalcControl to your application procedure, making its functions available to the procedure.

Control template HandyCalcEntryControl_ECF drops an entry control and a button on your procedure window. These controls work in tandem. A numerical value in the entry control is passed to the calculator - accessed via the button control - and the calculator passes a calculated result back into the control. A thread option on the tempate provides for passing data back to the entry control without closing the calculator, which can be left open on top of the application window.

Until the the close of 4th quarter 2008, this template and the HNDCalcControl class should be considered beta and feature incomplete. If you have feature suggestions, for the class or the templates, please do feel free to make them on the CHT forum using the "Suggestions" setting.

Any questions, feel free to ask them on the CHT forum.
Gus M. Creces
The Clarion Handy Tools Page

CATEGORY: DATES AND CALENDARS

*** HNDCAL.APP ***

NAME: HNDCAL.APP
TITLE: HNDDates Class Demonstration [V10.18.07.0317]
DATE: 9/17/2020
CATEGORY: Dates And Calendars
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

As usual, subscribers with a SOURCE code subscription receive the code module as well as the header. DLL version subscribers receive only the header module and the class code is in one of the CHT DLLs.

The application illustrates the syntax used to call many of the functions available in the HNDDates class. There are also two separate implementations of the calendar showing how you can modify the look of the CHT calendar by calling methods like OBJ.SetCalendarIcon() and embedding new calendar image properties in that method.

*** HNDCMCAL.APP ***

NAME: HNDCMCAL.APP
TITLE: COM Calendar Demo [V10.18.08.0209]
DATE: 9/17/2020
CATEGORY: Dates And Calendars
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

A CHT control template called HandyCOMCalendarControl helps you drop this control onto any application window and instantiate a Clarion-style interface to the OCX ICalendar control.

Any questions, feel free to ask: news.cwhandy.com

CATEGORY: EMAIL EXAMPLE

*** HNDBULKSMTPMAIL.APP ***

NAME: HNDBULKSMTPMAIL.APP
TITLE: Bulk SMTP Email [V10.19.01.0118]
DATE: 9/17/2020
CATEGORY: Email Example
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

The application also includes a set of separate attachment encryption/decryption procedures. While attachment encryption is invoked on the fly as you send mail, the decryption procedure can be used to unencrypt attachments received from this application.

The Clarion Handy Tools offer a very broad spectrum of email support in the form of templates, OOP classes, examples and Jump Start procedures.

Any questions, feel free to ask: news.cwhandy.com

*** HNDSOURCEMAPIDEMO.APP ***

NAME: HNDSOURCEMAPIDEMO.APP
TITLE: MAPI / ZIP / ABC Source Procedure [V10.18.12.0322]
DATE: 9/17/2020
CATEGORY: Email Example
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

Source_EmbedMAPIFunctions template
Source_EmbedZipFunctions template

The SMTP template provides an embeddable CHT class called HNDSmtp located in HNDIMAIL.INC and HNDIMAIL.CLW.

The ZIP template provides an embeddable CHT Class called HNDComZipClass located in HNDZIPPR.INC and HNDZIPPR.CLW.

If you want to learn to hand code with these two classes, this application is a perfect learning platform. The procedures provided may also be imported directly into your apps to provide pre-built functionality.

***** HNDSOURCESMTPDEMO.APP *****

NAME: HNDSOURCESMTPDEMO.APP
TITLE: SMTP / ZIP / ABC Source [V10.18.0011]
DATE: 9/17/2020
CATEGORY: Email Example
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

Source_EmbedSMTPFunctions template
Source_EmbedZipFunctions template

The SMTP template provides an embeddable CHT class called HNDSmtp located in HNDSMTP.INC and HNDSMTP.CLW.

The ZIP template provides an embeddable CHT Class called HNDComZipClass located in HNDZIPPR.INC and HNDZIPPR.CLW.

If you want to learn to hand code with these two classes, this application is a perfect learning platform. The procedures provided may also be imported directly into your apps to provide pre-built functionality.

CATEGORY:
FILE AND MEMORY COMPRESSION

***** HNDZDEMO.APP *****

NAME: HNDZDEMO.APP
TITLE: BulkShrinkFiles/BulkGrowFiles [V10.20.09]
DATE: 9/17/2020
CATEGORY: File And Memory Compression
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application uses the generic template EmbedDiskFunctions to attach a class called HNDDISK to your application procedures. The application illustrates the use of two HNDDISK functions:
OBJ.BulkShrinkFiles() -> OBJ.ShrinkFile()
OBJ.BulkGrowFiles() -> OBJ.GrowFile()

The HNDDISK Class provides several different Compression and Decompression methods including memory and file compression and decompression as well as ZIP compression and decompression.

The SHRINK and GROW functions provide CHT proprietary file compression and decompression capabilities that are pure Clarion code using internal Clarion functions created for that purpose.

During SHRINK, the target file is read into a memory buffer, the memory buffer is compressed in-memory (by Clarion's COMPRESS function) and then written to disk as binary bytes.

During GROW, the target file is read into a memory buffer, the memory buffer is decompressed in-memory (by Clarions DECOMPRESS function) and then written to disk as binary bytes.

The SHRINK version of a file is given the extension .HNZ, and the original file name and extension ae retained inside a file header embedded in the .HNZ file. Hence, the GrowFile() function is able to re-establish a fully expanded version of the original file including the file extension.

SHRINK and GROW are also optionally encryption capable. When encryption is enabled, the compressed file is given the extension .HZY.

CHT proprietary compression and encryption provided by these functions are extremely useful for secure file transport via email, FTP and HTTP since the decompression phase can only be performed by another CHT application that is made aware of the encryption password or by a CHT Clarion procedure built into your CHT email application, CHT FTP application or CHT Web Server.

See also HNDCPYDM.APP for more examples including file encryption.

CATEGORY:
FILE BLOB FUNCTIONS

*** HNDFBLOB.APP ***

NAME: HNDFBLOB.APP
TITLE: File To Blob Example [V10.18.08.0243]
DATE: 9/17/2020
CATEGORY: File BLOB Functions
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

Please note, that the code in the EntryPointMain() procedure has very little to do with file to blob or blob to file. This code looks up a file name to be sent to a blob and manages the data base that stores the blob record. Your blob data base may look nothing like the file ZStore that we are using here.

The salient point is that when calling the functions JumpStartFileToBlob() and its inverse, JumpStartBlobToFile(), you are responsible for creating, fetching and saving the data base records in the blob-containing data table. The functions provided, will insert the files to, and extract files independently from, your blob record. Data base I/O is up to you. However the I/O required is standard ABC stuff which you should already know how to do.

We have demonstrated here a couple of HNDDisk class functions to help us with our popup menu, when someone selects a file to be restored or deleted, for example. These have no direct relationship with the two blob functions being demonstrated.

Any questions, feel free to ask: news.cwhandy.ca

CATEGORY:
FILE MOVING, COPYING

*** HNDJUMPSTARTSBULK.APP ***

NAME: HNDJUMPSTARTSBULK.APP
TITLE: Jumpstarts #3 Bulk-Tab [V10.20.09]
DATE: 9/17/2020
CATEGORY: File Moving, Copying
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application demonstrates the following CHT Jumpstarts available from our Utility Template #3 interface, on the "Bulk" tab.

** JumpStartBulkFileCompressExpandWithProgress()
-Bulk file compression and expansion with encryption and decryption
** JumpStartBulkFileContainerizeWithProgress()
-Bulk file compression to a container (.HZO) file with encryption.
** JumpStartBulkFileDeContainerizeWithProgress()
-Bulk file decrompression from container (.HZO) file
** JumpStartBulkFileCopyWithProgress ()
-Bulk file copying to a target directory
** JumpStartBulkFileEncryptDecryptWithProgress()
-Bulk file encryption/decryption with mpression, decompression
** JumpStartBulkFileEraseWithProgress()
-Bulk file removal
** JumpStartBulkFileMoveWithProgress()
-Bulk file moving
** JumpStartBulkFileRenameWithProgress()
-Bulk file renaming

** A query allows you to describe the files to be acted on. Or files to be acted on may be dropped from MS File Explorer.

CATEGORY: FILE SEARCH AND REPLACE

*** HNDSCHDM.APP ***

NAME: HNDSCHDM.APP
TITLE: CHT Search And Replace Demo [V10.18.08.0230]
DATE: 9/17/2020
CATEGORY: File Search And Replace
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

- OBJ.BulkSearchReplaceText()
- OBJ.BulkFileRename()

BulkSearchReplaceText() -
Searches through text and program code files of all types in the target directory to replace words, commands, phrases.

BulkFileRename() -
Adds a prefix and removes spaces and extended characters from filenames in the target directory.

CATEGORY: GENPACK

*** HNDLBXGENPACK.APP ***

NAME: HNDLBXGENPACK.APP
TITLE: BROWSES LBX GENPAK [V10.20.09]
DATE: 9/17/2020
CATEGORY: GENPACK
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This app has only a splash screen for self-documentation purposes and serves as a repository with which to group and multi-compile a set of related CHT apps, without using CHT Project Builder.

It contains the following 4 CHT LBX-related applications:
hndlbxdm.app
hndpeople_lbx.app
hndlbxsql.app
hndlbxsqlex.app

CATEGORY:
HTTP CLIENT TYPE 1

*** HNDBRWCL.APP ***

NAME: HNDBRWCL.APP
TITLE: HTTP CLIENT (FORUM) [V10.20.09]
DATE: 9/17/2020
CATEGORY: HTTP CLIENT TYPE 1
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDBRWCL.EXE is designed to provide subscriber support services, by way of our web forum. This application implements an instance of the IE browser inside the application window and reads your CHT forum parameters from the windows registry to automatically log you in.

The URL being accessed is https://news.cwhandy.ca/.

The above URL may also be accessed directly in your favorite browser. We suggest, for convenience and speed, you use this application since it can log you into our forum automatically, without you needing to know your subscriber credentials.

The user interface looks and interacts exactly as it would were you to use IE directly but without some of the navigational clutter that normally appears on the IE window.

This client application serves to illustrate how to log in and then make a variety of data requests and data updates from a Clarion client application using the IE browser class called HNDInternetExplorer and a template called HandyComInternetExplorer.

This application now includes procedure(s) using the template CHTBestPracticeEmbedClass as an example for developers.

CATEGORY:
HTTP CLIENT TYPE 3

*** HNDCLIENTCL.APP ***

NAME: HNDCLIENTCL.APP
TITLE: Data Client TYPE 3 (Forum) [V10.20.09]
DATE: 9/17/2020
CATEGORY: HTTP CLIENT TYPE 3
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This client application serves to illustrate how to log in and then make a variety of data requests and data updates from a Clarion client application (using the HNDCLNT class) to a CHT Server configured for Client Server Mode.

NOTE: This client uses a two-piece login (Serial number, Last Name) and does not require the email address. The server (HNDCLIENTCL.APP) omits the Email Address Key on the server login configuration dialog to make this possible.

This applicaiton uses templates HandyWebClientBrowseProcedure and HandyWebClientUpdateFormProcedure to build the web browse and web update form, respectively.

*** HNDFILCL.APP ***

NAME: HNDFILCL.APP
TITLE: CHTTP File Client (T3) [V10.20.09]
DATE: 9/17/2020
CATEGORY: HTTP CLIENT TYPE 3
CLASSES: List App Classes

TEMPLATES: List App Templates

DESCRIPTION:

This application is designed to act as a starting point for all CHT "Secure File Clients" intended to access basic login-enforced file upload and download services from a CHT TYPE 3 file server application like HNDFILSV.APP.

This "Basic" version has had all non-essential componentry removed and focusses entirely on CHT templates and classes required to build a fast, efficient CHT secure file client. It uses a 3-piece login.

This client application has a matching demonstration server called HNDFILSV.APP which is configured to provide secure upload and download services.

This client is able to provide both compressed and encrypted file movement services without requiring either pre-compression or pre-encryption of the upload or download files.

This application is entirely template-built and contains almost no hand-embedded code.

CATEGORY: HTTP CLIENT TYPE 4

*** HNDSLFCL.APP ***

NAME: HNDSLFCL.APP
TITLE: HTTP CLIENT - TYPE 4 (FILES) [V10.20.09]
DATE: 9/17/2020
CATEGORY: HTTP CLIENT TYPE 4
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application is a file client to be used with HNDSLFSV.APP to test some basic principles surrounding secure, file exchanges with a CHT Type 4 file server.

The application illustrates how to make secure file upload and download requests to and from a CHT server root directory or any directory below server root.

A log-in dialog has been added, under an unlock image button on the tool bar. This produces an XSA, 2-part authentication dialog for server-level authentication prior to transfer access being allowed. The USER NAME and PASS CODE #, entered must match the same two values established on the server.

HNDSLFCL.APP illustrates two secure file transfer functions: HTTPGetFileSecure() and HTTPPutFileSecure(). The functions provide a progress window with a cancel button. This application requires no .DCT (Clarion Dictionary).

The functions are hosted in CHT Class HNDCLIENT and are made available in the application by a template called EmbedHTTPClientFunctions.

See also C# client CHTSNAPGET.EXE for command-line and batch-script downloading from HNDSLFSV.APP.

CATEGORY: HTTP SERVER TYPE 2

*** HNDTYPE4SERVERPEOPLE.APP ***

NAME: HNDTYPE4SERVERPEOPLE.APP
TITLE: TYPE 4 UCR SERVER - PEOPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: HTTP SERVER TYPE 2
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

In this particular example, application functionality has been limited to acting as a dynamic data, page server and providing several example UCR$ (User Custom Request) procedure calls that may be called from any HTML web browser, desktop, tablet or phone.

This server was branched from HNDSLFSV.APP to demonstrate building dynamic-data web pages based on CHT XML2HTMLGEN techniques. Take note that it has had it's default base page changed from INDEX.HTML to PEOPLEBROWSE.UCR. This is still a TYPE 4 server, however, because these UCR$ requests do not require prior authentication against a server password or against a user file. Page transactions with this server do not require prior login, and do not operate on a SESSION identifier.

This server utilizes a function called AutoGenHTMLFromXML_VER02 exported from HNDXML2HTMLGEN.APP to convert standard CHT XML template files into HTML. This is the same functionality that triggers an HTML file to be generated from CHTSNAPEDIT.EXE when the "Preview" menu key is clicked.

Calling any one of this server's specialized UCR$ functions, triggers the return of a dynamic data page based on one of CHT's .XML page templates.

Data rows inserted into a temporary copy of "000peoplebrowse.xml" file before processing to an HTML file, are read from a TPS file called "HNDO.TPS" which contains, a table called "PEOPLE".

*** HNDTYPE4UCRSERVERTEMPLATES.APP ***

NAME:	HNDTYPE4UCRSERVERTEMPLATES.APP
TITLE:	HTTP TYPE 4 UCR SERVER [V10.20.09]
DATE:	9/17/2020
CATEGORY:	HTTP SERVER TYPE 2
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

In this particular example, application functionality has been limited to acting as a dynamic data, page server and providing several example UCR$ (User Custom Request) procedure calls that may be called from any HTML web browser, desktop, tablet or phone.

This server was branched from HNDSLFSV.APP to demonstrate building dynamic-data web pages based on CHT XML2HTMLGEN techniques. Take note that it has had it's default base page changed from INDEX.HTML to TEMPLATESBROWSE.UCR. This is still a TYPE 4 server, however, because these UCR$ requests do not require prior authentication against a server password or against a user file, page transactions with this server do not require prior login, and do not operate on a SESSION Identifier.

This server utilizes a function called AutoGenHTMLFromXML_VER02 exported from HNDXML2HTMLGEN.APP to convert standard CHT XML template files into HTML. This is the same functionality that triggers an HTML file to be generated from CHTSNAPEDIT.EXE when the "Preview" menu key is clicked.

Calling any one of this server's specialized UCR$ functions, for example, "XML2HTMLTemplatesBrowse" triggers the return of a dynamic data page based on one of CHT's .XML page templates.

Data rows inserted into a temporary copy of "000templatesbrowse.xml" file before processing to an HTML file, are read from a TPS table called "TPLHEADER".

CATEGORY:
HTTP SERVER TYPE 3

*** HNDCLIENTSV.APP ***

NAME:	HNDCLIENTSV.APP
TITLE:	CHTTP CLIENT SERVER (T3) [V10.20.09]
DATE:	9/17/2020
CATEGORY:	HTTP SERVER TYPE 3
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This server application requires a matching demonstration client called HNDCLIENTCL.APP and a dictionary called HNDCLIENTSV.DCT.

The server provides both compressed and encrypted data streams consisting of browse data, form data, report and process data based on interactive queries placed from the client side targeted at pre-determined data views (single or multi-table) provided in the server.

NOTE: This server is configured for a two-piece login. It omits the Email Address Key on the BrowserServerInitialization controls template. In a two-piece login only the Login ID and Last Name (in this case) are used. This is a client server for reading and posting remote messages -- from which to both learn and with which to build forward. The back end views portions of the server are already installed and working.

This server is entirely template-built and contains almost no hand-embedded code. It may be used alone, via a standard HTTP communications link, or via an encrypted HTTPS tunnel communications link, when combined with a STUNNEL Proxy configured with an SSL certificate.

*** HNDFILSV.APP ***

NAME:	HNDFILSV.APP
TITLE:	CHTTP FILE SERVER (T3) [V10.20.09]
DATE:	9/17/2020
CATEGORY:	HTTP SERVER TYPE 3
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This application is intended as a starting point for developers to build file transfer servers of various sorts that act like FTP servers but utilize the HTTP protocol including encryption and compression. This server (TYPE 3) is a level above CHTTP page server, HNDSLFSV.APP (TYPE 4), in that it incorporates a server members file with unique member access credentials.

Several demo client applications are provided to CHT subscribers: HNDFLCLN.APP, HNDLIVUP.APP, HNDFILCL.APP and HNDINSTALLMYFILES.APP. Another client application called HNDHTGT.APP illustrates making on-the-fly file requests from this server using the HNDHTTP and HNDHTTPClient classes.

The server maintains complete control of which files and directories are visible to the client. File transfers are optionally compressed and/or encrypted at the request of the client application.

This is an "Authenticating" server. The client logs in by providing "LastName", "LoginID" and "EmailAddress". The server, unless reconfigured uses a member file called HNDFILSVMEM.TPS located in the \HNDAPPS\HNDFILSV\ directory. New members may be added as required to customize the users list.

This application is entirely template-built and contains almost no hand-embedded code.

NOTE: This is a starting-point server from which to build forward. The "file transfer" portion of the server is already installed and working.

CATEGORY: HTTP SERVER TYPE 4

*** HNDSLFSV.APP ***

NAME:	HNDSLFSV.APP
TITLE:	HTTP SERVER TYPE 4 [V10.19.07]
DATE:	9/17/2020
CATEGORY:	HTTP SERVER TYPE 4
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

HNDSLFSV.APP is a TYPE 4 CHT server, which means it serves files. It is our default web page server. Web pages and images, are after all, files. Normal HTTP page views from a browser are public and do not require password enabled access privileges. File transactions into and out of server-visible directories are security-controlled via a 2-part server login.

HNDSLFSV.AP can securely transact file transfers with several CHT file client applications. All secure clients require this server to set up a single, 2-part server-level login (username + password), to control file upload/download permissions.

HNDSLFCMD.APP is a CHT "Batch-Bot" upload/download application, enabling transfers into and out of the server root and any server subdirectory below the server root. "Batch-Bot" style applications may be used from the DOS command line, from DOS batch scripts, and, naturally, inside Clarion applications where they are known as "CHT Snap-Ins".

HNDSLFCL.APP is a simple, example client single-file uploader, downloader illustrating four different client-side upload/download methods.

CHTSNAPGET.EXE is a pre-compiled C# utility application included in the CHT Toolkit. It may be used to securely download files from this server when a server username:password combination is configured.

This HNDSLFSV.APP is entirely template-built and contains no significant hand-embedded code.

*** HNDSLFSVFX.APP ***

NAME: HNDSLFSVFX.APP
TITLE: CHTTP FILE SERVER (T4) [V10.20.09]
DATE: 9/17/2020
CATEGORY: HTTP SERVER TYPE 4
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

HNDSLFSVFX.APP is a TYPE 4 CHT server, which means it serves files. It is our default web page server. Web pages and images, are after all, files. Normal HTTP page views from a browser are public and do not require password enabled access privileges. File transactions into and out of server-visible directories are security-controlled via a 2-part server login.

HNDSLFSVFX.AP can securely transact file transfers with several CHT file client applications. All secure clients require this server to set up a single, 2-part server-level login (username + password), to control file upload/download permissions.

HNDSLFCMD.APP is a CHT "Batch-Bot" upload/download application, enabling transfers into and out of the server root and any server subdirectory below the server root. "Batch-Bot" style applications may be used from the DOS command line, from DOS batch scripts, and, naturally, inside Clarion applications where they are known as "CHT Snap-Ins".

HNDSLFCL.APP is a simple, example client single-file uploader, downloader illustrating four different client-side upload/download methods.

This server is entirely template-built and contains almost no hand-embedded code. It may be used alone, via a standard HTTP communications link, or via an encrypted HTTPS tunnel communications link, when combined with a STUNNEL Proxy, configured with an SSL Certificate.

*** HNDTEAMDOCSERVER.APP ***

NAME: HNDTEAMDOCSERVER.APP
TITLE: CHTTP DOC SERVER (T4) [V10.20.09]
DATE: 9/17/2020
CATEGORY: HTTP SERVER TYPE 4
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDTEAMDOCSERVER.EXE, acts as an HTML page server intended to display application information generated from your applications via CHT Clarion Project Builder"

For developers working in a team, especially teams with remote members, it's useful for members to know which versions of the applications for which they are responsible have been centrally compiled for QA. With each re-build, using CHT Compile Manager, a set of HTML pages is generated into this server's "run" directory, detailing the applications with a variety of information from version number, to procedure docs, to templates in use.

Note that the base page required by this server is docindex.html. CHT compile manager generates a matching index page with this name.

This server is entirely template-built and contains almost no hand-embedded code. It may be used alone, via a standard HTTP communications link, or via an encrypted HTTPS tunnel communications link, when combined with a STUNNEL Proxy, configured with an SSL Certificate.

CATEGORY:
INTERNET - HTTP SERVERS

*** HNDSTUBFILESERVER.APP ***

NAME: HNDSTUBFILESERVER.APP
TITLE: CHT Static Page Web Server [V9.14.05.0110]
DATE: 9/17/2020
CATEGORY: Internet - HTTP Servers
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

These extension/control templates provide TCP/IP connectivity with a browser or with a Clarion client application using HNDHttp or HNDClient classes. In this particular example, application functionality has been limited to acting as a static page browser server and providing two example UCR$ (User Custom Request) procedure calls that may be called from client applications HNDHTGT.APP or HNDSLFCL.APP.

The BrowserServerInitializationControls template provides a mode switch that lets your application act as follows:

* A Static Page Web Server (Clarion Server To Browser)
* A Client Data Server (Clarion App Both Ends)
* A Browser Data Web Server (Clarion Server To Browser)

This application is entirely template-built and contains almost no hand-embedded code.

NOTE: This is a starting-point server from which to build forward. It proceeds to the point of delivering static HTML/JS/CSS pages. Many "extra" operations and features are ready to be grafted in by the developer using a CHT training kit called "CHT Static Page Server Construction Kit" available from our "Purchase" page.

The kit includes several hours of explanatory video, a manual, this server, and an expanded server complete with source code and sample static pages.

Ask about this and other "CHT Server Construction Kits" available or under development at CHT headquarters.

CATEGORY:
MS OFFICE EXCEL

*** HNDTESTEXCEL.APP ***

NAME: HNDTESTEXCEL.APP
TITLE: CHT Excel Class Example [V10.19.02.0300]
DATE: 9/17/2020
CATEGORY: MS Office Excel
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

The HNDEXCELCLASS class is introduced to the procedure via a template called EmbedHndExcelClassFunctions.

Various controls are provided on the window that call into class functions to illustrate HNDEXCELCLASS features.

Any questions, feel free to ask.

CATEGORY:
MS OFFICE OUTLOOK

*** HNDOUTLOOK.APP ***

NAME: HNDOUTLOOK.APP
TITLE: MS Outlook View Control [V10.19.11.0158]
DATE: 9/17/2020
CATEGORY: MS Office Outlook
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

A CHT control template called HandyCOMOutLookControl helps you drop this control onto any application window and instantiate a Clarion-style interface to the COM IViewCtl control.

With the HndComOutlookControl template and it's underlying CHT OOP classes you can have a fully interactive MS Outlook application running in your Clarion Window, including such Outlook features as Inbox, OutBox, Sent Items, Calendar, Contacts, Journal, Notes, Tasks and Drafts.

Any questions, feel free to ask: news.cwhandy.com

CATEGORY:
SMS MESSAGES

*** HNDSMSMESSAGE.APP ***

NAME: HNDSMSMESSAGE.APP
TITLE: CHT SMS Messaging Utility [V10.20.09]
DATE: 9/17/2020
CATEGORY: SMS Messages
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application" illustrates both CHT Native SMS messaging as well as SMS messages via SMTP. This application requires Clarion's CLARUNEXT.DLL but is otherwise stand-alone.

Further it is dependent on correct configuration of a valid SMTP server or SMS gateway setup with UserName, Password, Email/SMS Server and Email/SMS Return Address. Our template and SMS classes provide a configuration function with user-controllable registry location setttings.

The SMS template illustrated is EMBEDSMSFUNCTIONS. The CHT class underlying this template is HNDSMS located in files HNDSMS.INC and HNDSMS.CLW.

Please read and understand the proper formation of SMS addresses before experimenting, using this URL: (https://en.wikipedia.org/wiki/SMS_gateway).

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 incorpopated 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:
chtsnapdisk --help, -help, or help

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

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

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

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

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

To display free space information of all drives to console:
chtsnapdisk --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@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

*** 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@cwhandy.ca for price and availability.

Gus M. Creces
The Clarion Handy Tools Page
http://www.cwhandy.ca
http://news.cwhandy.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 configurtion 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.cwhandy.ca)
-- Three or four command-line inputs are required:
-- CHTSNAPMAPI "TO" "SUBJECT" "MESG.TXT OR MESG.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.
-- MESG.TXT, MESG.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@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

*** 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 configued 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 configued 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@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

*** 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 incorpoprated into your application via CHT templates.

CHT Snap Zip does not require any kind of pre-configuration involving a configurtion 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.cwhandy.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 paramter list zip contents and accepts 2 inputs.
It produces a fully formatted list intended for easy reading from the DOS console.

chtsnapzip <[\path\]zipfilename>

-- The UZ parameter unzips the zip file contants 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.

chtsnapzip <[\path\]zipfilename>

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

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

*** 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.cwhandy.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@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

CATEGORY:
SNAP-IN EXAMPLE

*** HNDTESTCHTBOTCLACDO.APP ***

NAME: HNDTESTCHTBOTCLACDO.APP

TITLE: HNDMAILCDO EXAMPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: SNAP-IN EXAMPLE
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application demonstrates the CHT global template called ApplicationSnapIns.

This global template provides an interface to your application whereby several "Snap-In" or "Batch-Bot" components may be incorporated.

This application illustrates CHTSNAPEDIT.EXE and HNDMAILCDO.EXE.

CHT "Batch-Bots" like HNDMAILCDO.EXE provide various functionalies, that can be used standalone from the DOS prompt , from a DOS BATCH file, or programmatically from a Clarion application. The above "Snap-Ins/Batch-Bots" provide text editing and SMTP-style email, respectively.

Procedure-level button control templates are provided for each of the "Snap-In" executables in order to pass information to them and launch them from your application, seamlessly, just like dll-based procedures. Button templates are: SnapEditStartButton and BotCDOStartButton.

External binary dependencies: CHTSNAPEDIT.EXE, HNDMAILCDO.EXE and HNDCAPCON.DLL

*** HNDTESTCHTBOTCLAMAPI.APP ***

NAME: HNDTESTCHTBOTCLAMAPI.APP
TITLE: HNDMAILMAPI EXAMPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: SNAP-IN EXAMPLE
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application demonstrates the CHT global template called ApplicationSnapIns.

This global template provides an interface to your application whereby several "Snap-In" or "Batch-Bot" components may be incorporated.

This application illustrates CHTSNAPEDIT.EXE and HNDMAILMAPI.EXE.

CHT "Batch-Bots" like HNDMAILMAPI.EXE provide various functionalies, that can be used standalone from the DOS prompt , from a DOS BATCH file, or programmatically from a Clarion application. The above "Snap-Ins/Batch-Bots" provide text editing and MAPI email, respectively.

Procedure-level button control templates are provided for each of the "Snap-In" executables in order to pass information to them and launch them from your application, seamlessly, just like dll-based procedures. Button templates are: SnapEditStartButton and BotMAPIStartButton.

External binary dependencies: CHTSNAPEDIT.EXE, HNDMAILMAPI.EXE and HNDCAPCON.DLL

*** HNDTESTCHTBOTCLASMTP.APP ***

NAME: HNDTESTCHTBOTCLASMTP.APP
TITLE: HNDMAILSMTP EXAMPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: SNAP-IN EXAMPLE
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application demonstrates the CHT global template called ApplicationSnapIns. This global template provides an interface to your application whereby several "Snap-In" components may be incorporated.

This application illustrates CHTSNAPEDIT.EXE and HNDMAILSMTP.EXE.

CHT "Batch-Bots" like HNDMAILSMTP.EXE provide various functionalies, that can be used standalone from the DOS prompt , from a DOS BATCH file, or programmatically from a Clarion application. The above

"Snap-Ins" provide text editing and SMTP email, respectively.

Procedure-level button control templates are provided for each of the "Snap-In" executables in order to pass information to them and launch them from your application, seamlessly, just like dll-based procedures. Button templates are: SnapEditStartButton and BotSMTPStartButton.

External binary dependencies: CHTSNAPEDIT.EXE, HNDMAILSMTP.EXE and HNDCAPCON.DLL

*** HNDTESTCHTBOTHNDENCRYPT.APP ***

NAME:	HNDTESTCHTBOTHNDENCRYPT.APP
TITLE:	CHTSNAPCRYPT EXAMPLE [V10.20.09]
DATE:	9/17/2020
CATEGORY:	SNAP-IN EXAMPLE
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This application illustrates how to incrorporate HNDENCRYPT.EXE into a Clarion application via the global SNAP-IN interface template "CHT ApplicationSnapIns".

HNDENCRYPT.EXE runs from the DOS prompt or it can be called from another application using Clarion RUN, or using CHT's HNDSNAPCALL Class, or from a Windows batch file or a windows .VBS script. It accepts several command line parameters that describe the input file and and output path.

HNDENCRYPT.EXE, is configured with a --KEY established saved and configured in the registry which is used in the encrypt operation. Files encrypted and decrypted on the same machine are processed easily.

Files passed to other machines must first configure the correct encryption key before they can decrypt files passed to them.

External binary dependencies: HNDENCRYPT.EXE and HNDCAPCON.DLL

*** HNDTESTCHTBOTHNDNETZIP.APP ***

NAME:	HNDTESTCHTBOTHNDNETZIP.APP
TITLE:	HNDZIPNET EXAMPLE [V10.20.09]
DATE:	9/17/2020
CATEGORY:	SNAP-IN EXAMPLE
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This application illustrates how to incrorporate HNDZIPNET.EXE into a Clarion application via the global SNAP-IN interface template "CHT ApplicationSnapIns".

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

This "test" application presents a BotZipStartButton template a target zip field, a source directory name, zip mask field, zip query field and an unzip target directory field. Complete the necessary fields and click the --zip button or the --unzip button.

HNDZIPNET.EXE error messages, warnings and help information from HNDZIPNET.EXE appear in a window text control to tell you the outcome of pushing the zip button.

External binary dependencies: HNDZIPNET.EXE, HNDCAPCON.DLL AND HNDNETZIPPER.DLL

*** HNDTESTCHTBOTHNDSEARCH.APP ***

NAME:	HNDTESTCHTBOTHNDSEARCH.APP
TITLE:	HNDSEARCH EXAMPLE [V10.20.09]
DATE:	9/17/2020
CATEGORY:	SNAP-IN EXAMPLE
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This application demonstrates the CHT global template called ApplicationSnapIns.

This global template provides an interface to your application whereby several "Snap-In" and "Batch-Bot" components may be incorporated.

This application illustrates incorporating HNDSEARCH.EXE into a Clarion application using CHT's HNDSNAPCALL class.

CHT "Batch-Bots" like HNDSEARCH.EXE provide various functionalies, that can be used standalone from the DOS prompt , from a DOS BATCH file, or programmatically from a Clarion application. The above "Snap-In" provides text file search and replace capability.

Procedure-level button control templates are provided for each of the "Snap-In" executables in order to pass information to them and launch them from your application, seamlessly, just like dll-based procedures. Button templates are: BotSEARCHStartButton.

External binary dependencies: HNDSEARCH.EXE, CHTSNAPEDIT.EXE and HNDCAPCON.DLL

*** HNDTESTCHTBOTMAKEHZO.APP ***

NAME: HNDTESTCHTBOTMAKEHZO.APP
TITLE: HNDMAKEHZO EXAMPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: SNAP-IN EXAMPLE
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application illustrates how to incrorporate HNDMAKEHZO.EXE into a Clarion application via the global SNAP-IN interface template "CHT ApplicationSnapIns".

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, runs from the DOS prompt or can be called from another application such as this one using Clarion RUN, or CHT's HNDSNAPCALL class, or from a Windows BATCH file or windows .VBS script.

External binary dependencies: HNDMAKEHZO.EXE and HNDCAPCON.DLL

*** HNDTESTCHTBOTNETSMTP.APP ***

NAME: HNDTESTCHTBOTNETSMTP.APP
TITLE: HNDMAILNET EXAMPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: SNAP-IN EXAMPLE
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application demonstrates the CHT global template called ApplicationSnapIns.

This global template provides an interface to your application whereby several "Snap-In" and "Batch-Bot" components may be incorporated.

This application illustrates CHTSNAPEDIT.EXE and HNDMAILNET.EXE.

CHT "Batch-Bots" like HNDMAILNET.EXE provide various functionalies, that can be used standalone from the DOS prompt , from a DOS BATCH file, or programmatically from a Clarion application. The above "Snap-Ins/Batch-Bots" provide text editing and SMTP email, respectively.

Procedure-level button control templates are provided for each of the "Snap-In" executables in order to pass information to them and launch them from your application, seamlessly, just like dll-based procedures. Button templates are: SnapEditStartButton and BotNetSMTPStartButton.

External binary dependencies: CHTSNAPEDIT.EXE, HNDMAILNET.EXE, HNDNETSMTP.DLL and HNDCAPCON.DLL

*** HNDTESTCHTBOTSLFCMD.APP ***

NAME: HNDTESTCHTBOTSLFCMD.APP
TITLE: HNDSLFCMD EXAMPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: SNAP-IN EXAMPLE
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application illustrates how to incrorporate HNDSLFCMD.EXE into a Clarion application via the global SNAP-IN interface template "CHT ApplicationSnapIns".

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. A username and password are required for server access to be given. Communications with the server are encrypted. The application can be called from one of your Clarion applications, a Windows batch file or a Windows .VBS script to provide reliable file upload/download/delete services.

HNDSLFCMD.EXE, is configured by using parameters, --config0 to --config9 established saved and configured in the registry

External binary dependencies: HNDSLFCMD.EXE and HNDCAPCON.DLL

*** HNDTESTCHTSNAPCRYPT.APP ***

NAME: HNDTESTCHTSNAPCRYPT.APP
TITLE: CHTSNAPCRYPT EXAMPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: SNAP-IN EXAMPLE
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application illustrates how to incrorporate CHTSNAPCRYPT.EXE into a Clarion application via the global SNAP-IN interface template "CHT ApplicationSnapIns".

CHTSNAPCRYPT.EXE is a command-line utility with which to encrypt files (to .HYN extension) and optionally wrap the encrpted file into a .ZIP.

CHTSNAPCRYPT.EXE is configured with a --KEY established saved and configured in the registry which is used in the encrypt operation. Files encrypted and decrypted on the same machine are processed easily.

Files passed to other machines must first configure the correct encryption key before they can decrypt files passed to them.

External binary dependencies: CHTSNAPCRYPT.EXE and HNDCAPCON.DLL

*** HNDTESTCHTSNAPDISK.APP ***

NAME: HNDTESTCHTSNAPDISK.APP
TITLE: CHTSNAPDISK EXAMPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: SNAP-IN EXAMPLE
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application illustrates how to incrorporate CHTSNAPDISK.EXE into a Clarion application via the global SNAP-IN interface template "CHT ApplicationSnapIns".

CHTSNAPDISK.EXE is a command-line disk-drive information utility which provides: Drive Number, Drive Letter, Drive Type, Volume Label, File System, Space Available to Current User, Total Space Available, Total Size of Drive, and Disk Hardware Serial with the "--info" command.

The "--list" command provides a file called "chtsnapdisk.txt" containing the above information.

The "--listedit" provides a file called "chtsnapdisk.txt" containing the above information and on completion, opens the file with CHTSNAPEDIT.EXE.

The "--processor" command also identifies the current computer's processor ID.

The "--label" command provides for changing the disk-drive label.

External binary dependencies: CHTSNAPDISK.EXE, CHTSNAPEDIT.EXE and HNDCAPCON.DLL

*** HNDTESTCHTSNAPGET.APP ***

NAME: HNDTESTCHTSNAPGET.APP
TITLE: CHTSNAPGET EXAMPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: SNAP-IN EXAMPLE
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application illustrates how to incrorporate CHTSNAPGET.EXE into a Clarion application via the global SNAP-IN interface template "CHT ApplicationSnapIns".

CHTSNAPGET.EXE is a command-line utility to download files from any CHT file server. It uses either the HTTP or HTTPS protocol.

CHTSNAPGET.EXE also has secure-file access capabilities. This requires user knowledge of the user-name/password combination protecting the file or files being downloaded. This secure mode can optionally use HTTPS protocol or HTTP protocol depending on server configuration.

File and HTTP header data, regardless of protocol, is encrtyped on request.

This application can be tested with CHT demonstration servers HNDSLFSV.APP or HNDFILSV.APP available in your \HNDAPPS\ directory.

External binary dependencies: CHTSNAPGET.EXE and HNDCAPCON.DLL

*** HNDTESTCHTSNAPMAPI.APP ***

NAME: HNDTESTCHTSNAPMAPI.APP
TITLE: CHTSNAPMAPI EXAMPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: SNAP-IN EXAMPLE
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application demonstrates the CHT global template called ApplicationSnapIns.

This global template provides an interface to your application whereby several "Snap-In" components may be incorporated.

This application illustrates CHTSNAPEDIT.EXE and CHTSNAPMAPI.EXE.

CHT "Snap-Ins" are C# executables (EXE's) which provide various functionalies, that can be used standalone from the DOS prompt , from a DOS BATCH file, or programmatically from a Clarion application. The above "Snap-Ins" provide text editing and SMTP email, respectively.

Procedure-level button control templates are provided for each of the "Snap-In" executables in order to pass information to them and launch them from your application, seamlessly, just like dll-based procedures. Button templates are: SnapEditStartButton and SnapSMTPStartButton

External binary dependencies: CHTSNAPMAPI.EXE, CHTSNAPEDIT.EXE and HNDCAPCON.DLL

*** HNDTESTCHTSNAPSMTP.APP ***

NAME: HNDTESTCHTSNAPSMTP.APP
TITLE: CHTSNAPSMTP EXAMPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: SNAP-IN EXAMPLE

CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application demonstrates the CHT global template called ApplicationSnapIns.

This global template provides an interface to your application whereby several "Snap-In" components may be incorporated.

This application illustrates CHTSNAPEDIT.EXE and CHTSNAPSMTP.EXE.

CHT "Snap-Ins" are C# executables (EXE's) which provide various functionalies, that can be used standalone from the DOS prompt , from a DOS BATCH file, or programmatically from a Clarion application. The above "Snap-Ins" provide text editing and SMTP email, respectively.

Procedure-level button control templates are provided for each of the "Snap-In" executables in order to pass information to them and launch them from your application, seamlessly, just like dll-based procedures. Button templates are: SnapEditStartButton and SnapSMTPStartButton

External binary dependencies: CHTSNAPSMTP.EXE, CHTSNAPEDIT.EXE and HNDCAPCON.DLL

*** HNDTESTCHTSNAPZIP.APP ***

NAME: HNDTESTCHTSNAPZIP.APP
TITLE: CHTSNAPZIP EXAMPLE [V10.20.09]
DATE: 9/17/2020
CATEGORY: SNAP-IN EXAMPLE
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This application illustrates how to incrorporate CHTSNAPZIP.EXE into a Clarion application via the global SNAP-IN interface template "CHT ApplicationSnapIns".

CHTSNAPZIP.EXE is a command-line utility that auto-zips a target directory while OPTIONALLY including any sub-directories of the source directory. CHTSNAPZIP.EXE also unzips to a target directory. This application illustrates how to do all of those things from inside a Clarion application using CHT Templates to write the code.

This application presents a SnapZipStartButton template and two input fields, a target zip file name field and a source directory name field. Complete these two fields and click the zip start button.

Other related templates used are SnapZipDirStartButton and SnapUnZipStartButton.

CHTSNAPZIP.EXE warns if the target zip file exists and if the source directory doesn't exist. Messages from CHTSNAPZIP.EXE appear in a window text control to tell you the outcome of pushing the zip button.

External binary dependencies: CHTSNAPZIP.EXE and HNDCAPCON.DLL

CATEGORY:
TOOLKIT UTILITY

*** HNDBACKUPCONFIG.APP ***

NAME: HNDBACKUPCONFIG.APP
TITLE: Auto-Backup Config [V10.20.09]
DATE: 9/17/2020
CATEGORY: TOOLKIT UTILITY
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDBACKUPCONFIG.EXE is intended to configure the CHT toolkit to back up your Clarion applications any time your app file is changed and generated.

Settings include, the ability to:
1) establish the auto-backup timer,

2) modify the target backup directory,
3) modify the target zip directory
4) modify the target restore directory
5) maintain, delete and zip backup files
6) locate and restore backup files from zip

From this application you are able to select a range of backups and compress them into a single zip file for archiving to a configured location using HNDNETZIPPER.DLL

You are also able to search and examine zip archives and restore backed up applications and dictionaries to a configurable location.

This application now includes procedure(s) using the template CHTBestPracticeEmbedClass as an example for developers.

*** HNDBACKUPCONFIGSNAP.APP ***

NAME: HNDBACKUPCONFIGSNAP.APP
TITLE: Auto-Backup Config (SNAP) [V10.20.09]
DATE: 9/17/2020
CATEGORY: TOOLKIT UTILITY
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDBACKUPCONFIGSNAP.EXE, is intended to configure the CHT toolkit to back up your Clarion applications any time your app file is changed and generated.

Settings include, the ability to:
1) establish the auto-backup timer,
2) modify the target backup directory,
3) modify the target zip directory
4) modify the target restore directory
5) maintain, delete and zip backup files
6) locate and restore backup files from zip

From this application you are able to select a range of backups and compress them into a single zip file for archiving to a configured location.

You are also able to search and examine zip archives and restore backed up applications and dictionaries to a configurable location.

This specially named version (i.e. SNAP in the app name) is built on CHTSNAPZIP.EXE incorporated as a CHT "Snap-In" on the global template "ApplicationSnapins".

*** 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

*** HNDBATCHRUNNER.APP ***

NAME: HNDBATCHRUNNER.APP
TITLE: Batch File Run Utility [V10.20.09]
DATE: 9/17/2020
CATEGORY: TOOLKIT UTILITY
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDBATCHRUNNER.EXE, is an automation tool designed to let you pull together various command line tools and "Batch Bots" to perform tasks using batch file scripts you create that are executed from the application.

Rather than writing a program to perform a repetetive task such as zipping and uploading to an FTP site or sending an email announcement, HNDBATCHRUNNER.EXE lets you quickly write a script, save it to disk and execute the script when needed.

"Batch Bots" include a number of command-line utilities that at time of writing are as follows:

HNDBASE64CVT.EXE (Base 64 encoder/decoder)
HNDENCRYPT.EXE (File encrypter/decrypter with compression)
HNDMAKEHZO.EXE (CHT Container building utility)
HNDFTPNET.EXE (.NET-based FTP utility)
HNDMAILCDO.EXE (Windows CDO-based email send)
HNDMAILNET.EXE (.NET based SMTP email send)
HNDMAILSMTP.EXE (SMTP-based email send)
HNDMAILMAPI.EXE (MAPI-based email send)
HNDZIPNET.EXE (.NET-based file zip utility)
HNDSEARCH.EXE (Text file search/replace utility)
HNDSLFCMD.EXE (HTTP upload utility)
CHTSNAPGET.EXE (HTTP download utility)
CHTSNAPZIP.EXE (Zip, Unzip utility)
CHTSNAPSMTP.EXE (SMTP-based email send)
CHTSNAPMAPI.EXE (SMTP-based email send)
CHTSNAPEDIT.EXE (Clarion code and HTML editor)
CHTSNAPSEND.EXE (Error reporting utility)

All of the above (starting with HND) are provided to CHT developers pre-compiled as well as in source .APP format so they can be expanded, emulated or customized by CHT developers.

Many command-line utilities of this sort are available that perform a variety of useful tasks. For example:
CLARIONCL.EXE (Clarion generate/compile utility)
SIGNCODE.EXE (Windows code-signing utility)
REGASM.EXE (.NET assembly registration utility)

HNDBATCHRUNNER.EXE supports writing, storing, searching and executing of batch file scripts that support your day-to-day use of command-line utiliites of the sort described above.

*** HNDBULKSIGNER.APP ***

NAME: HNDBULKSIGNER.APP
TITLE: Bulk Code Signing Utility [V10.20.09]
DATE: 9/17/2020
CATEGORY: TOOLKIT UTILITY
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDBULKSIGNER.EXE, is a batch-oriented BULK CODE SIGNING TOOL designed to provide easy and fast, post-compile bulk code signing of your .EXE, .DLL and *.OCX binaries before final delivery.

Select a target directory containing several to-be-signed .EXE, .DLL or .OCX files; click select all, followed by the bulk sign button, and watch as in a few seconds the entire batch is signed with your designated code signing certificate.

Log files (*_SIGN.LOG) are created to document each executable as it is signed. If any single executable fails signing, an err file (*_SIGN.ERR) is created indicating what went wrong.

*** HNDCMP.APP ***

NAME:	HNDCMP.APP
TITLE:	Clarion Project Builder [V10.20.09]
DATE:	9/17/2020
CATEGORY:	TOOLKIT UTILITY
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDCMP.EXE comes to you courtesy of The Clarion Handy Tools Page. It interacts with CLARIONCL.EXE (Clarion EE Edition Only) to help you automate your development projects.

With CHT Clarion Project Builder you can perform the following operations directly without having to open your applications to change their internal configuration settings:

1) Switch between EXE and DLL compilation of an app
2) Assign a DLL memory load address for your DLL (Optional)
3) Change link mode from LIB (library) to DLL
4) Switch app compile between Debug and Release mode
5) Switch app compile between expiration and non-expiration mode
6) Switch from HNDFBACK.TRN (English) to HNDCUSTOM.TRN (User)
7) Adjust processor affinity of the EXE for multi processor machines
8) Assign code signing information into the executable
9) Export the app to TXA
10) Compress the executable [Optional]
11) Copy the executable to a configured target directory
12) Stop compilation on error or continue to next
13) Generate app discovery HTML into a configured target directory
14) View a compile log for each application giving full details of what happened during compile, code-signing, affinity assigning and compression.

CHT Clarion Project Builder requires that the HNDTOOLS.TPL (template) be present in your Clarion template directory and a CHT template called HNDCMPControlPanel has been applied to your applications. This template and file are supplied only to full-fledged CHT subscribers.

*** HNDFBACKBUILDER.APP ***

NAME:	HNDFBACKBUILDER.APP
TITLE:	Translation File Builder [V10.20.09]
DATE:	9/17/2020
CATEGORY:	TOOLKIT UTILITY
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDFBACKBUILDER.EXE, is a translation-file tool that every CHT developer who builds non-English applications will want to begin using right away.

In the past, when CHT shipped a HNDFBACK.TRN (CHT Classes Translation File) that contained some new variables, it was the responsibility of the developer to copy these new variables to their personal HNDCUSTOM.TRN to avoid missing-variable errors during compilation.

This application automates that once manual process.

By opening this application and importing their HNDCUSTOM.TRN under the appropriate language menu, developers are able to generate a revised version of their own HNDCUSTOM.TRN which merges their personal HNDCUSTOM.TRN translations with the latest version of HNDFBACK.TRN.

Already-translated variables replace the English language versions and new variables are present and marked as "NEW" to make them easy to find for subsequent translation from the interface provided in this application.

After merging, and generating a new HNDCUSTOM.TRN based on a combination of their already-translated variables and new ones from HNDFBACK.TRN, developers are able to continue using their own HNDCUSTOM.TRN without having to manually synch new variables to avoid compilation errors.

*** HNDFTPWZ.APP ***

NAME: HNDFTPWZ.APP
TITLE: FTP Wizard (CLARION) [V10.20.09]
DATE: 9/17/2020
CATEGORY: TOOLKIT UTILITY
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDFTPWZ.EXE, comes to you courtesy of The Clarion Handy Tools Page. It demonstrates a combination of six FTP Templates.

* EmbedInternetFTP (parent template)
* HandyFTPInitializationControls (gather logon and user information)
* HandyFTPLocalFilesControls (gather upload file information)
* HandyFTPLocalFilesULControls (upload files control panel)
* HandyFTPRemoteFilesControls (gather download file information)
* HandyFTPRemoteFilesDLControls (download files control panel)

Note that the application can be easily rebuilt for developer customization with the CHTJumpStartProceduresUtility template. Navigate to the FTP Functions tab and select item four, "FTP Wizard".

Or the app may be reproduced and modified by making a copy of HNDFTPWZ.APP and introducing your own modifications.

This source code applicaiton is a fully functional FTP wizard with file upload and download capabiltity, proxy support, directory listing and creation, file deletion and moving. While this application shares the features of many commercial FTP applications, it alone uses the CHT query control which allows you to filter your upload and download file lists and create recallable queries that really boost your productivity.

*** HNDSETUP.APP ***

NAME: HNDSETUP.APP
TITLE: Installation Container Utility [V10.20.09]
DATE: 9/17/2020
CATEGORY: TOOLKIT UTILITY
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDSETUP.EXE, compresses all kinds of files into a structured container (.HZO) built around a Clarion .TPS file. Optional encryption is available.

Our CHT Toolkit installer/updater, for your Clarion version also assigns this application as the default Windows handler for .HZO files so that you can click on any .HZO file to open it and even unpack it to a target directory if the optional password is known.

A query allows you to describe the files to be containerized. Or files to be containerized may be dropped from MS Windows Explorer.

This application can be used to build standard .HZO installation and update files for transport and decontainerization by applications like HNDSETUPMKR.APP to provide continuous, on-line updating of your software and data products.

Our utility application called HNDSETUPMKR.APP (intended to be cloned and customized) uses a new install-maker template to generate a custom web-installer using the .HZO containers created by this HNDSETUP.APP application.

This applicaton now incorporates three CHT-XSA security settings and a "Post File To Server" dialog, in order to make it quick and easy to upload your .HZO container to a waiting CHT installation server enabled with a CHT-XSA security dialog.

*** HNDSQLTRACE.APP ***

NAME: HNDSQLTRACE.APP
TITLE: SQL Trace Utility [V10.20.09]
DATE: 9/17/2020
CATEGORY: TOOLKIT UTILITY
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDSQLTRACE.EXE, is designed to help you with your SQL application development projects by assisting with easy SQL tracing.

Start tracing by configuring the specific driver(s) used by your SQL application followed by a click on the "Minimize" button . Once this application is started, then start your SQL application.

Navigate in your SQL application to the specific area to be tested and bring HNDSQLTRACE.EXE back up using the windows task bar icon menu provided.

Manual tracing:
When HNDSQLTRACE.EXE re-appears all SQL statements executed to that point are displayed in reverse chronological order, with newest statements at the top. From this point forward, to trace specific actions, click the "Clear" button and interact briefly with your SQL application. Click the "Read Next" button to view the most recent SQL activity raised by your application. Repeat as necessary.

Automatic tracing:
Check the "Auto Trace" switch. A window timer will periodically refresh to the latest SQL select or group of selects issued by your application.

File Trace or Direct Trace:
These modes are mutually exclusive.

File trace is performed by communicating with SQL trace information issued by Clarion's file drivers into a text file. In "File Trace" mode a developer can trace any and all SQL select outputs from various browses in an application as long as HNDSQLTRACE.EXE is started and configured before starting the application.

"Direct Trace" mode is performed by low-level communication, usually from a specific procedure under scrutiny. This mode requires one of CHT's browse templates to be configured to issue trace information during interaction with the user.

This application includes the template CHTBestPracticeEmbedClass as an example for developers to study.

CATEGORY:
VIDEO PLAYER

*** HNDVIDEOEDUCATION.APP ***

NAME: HNDVIDEOEDUCATION.APP
TITLE: VIDEO PLAYER (EDU) [V10.20.09]
DATE: 9/17/2020
CATEGORY: VIDEO PLAYER
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDVIDEOEDUCATION.EXE, is designed to be CHT's official media player. It streams .MP4 versions of CHT Educational Videos from our CHT Video Server (also a Clarion application).

At the same time HNDVIDEOEDUCATION.EXE also serves as a model for developers wishing to build their own streaming video players.

The player is built on CHT Template HandyComMediaControl which uses the Windows Media Player COM object based in WMP.DLL.

The player is capable of playing off-line any videos which have been previously viewed to completion. On-line viewing of uncached videos begins quickly, once approximately 5 percent of the video has downloaded, and already-cached videos begin immediately since they are not downloaded again.

If you plan to adapt this application to act as a video player for your own videos, a useful, accompanying application called HNDBUILDVIDEOLIST.APP can build video index files on your CHT-based video server, of the type utilized by this application and by the HNDWindowsMediaPlayer class (HNDWMP.INC, HNDWMP,CLW, HNDWMP.TLB).

CATEGORY:
WINDOWS API

*** HND5.APP ***

NAME: HND5.APP
TITLE: CHT HNDDisk Classes Demo [V10.19.02.0517]
DATE: 9/17/2020
CATEGORY: Windows API
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

Some examples of the things illustrated:
Creating nested directories
Copying and moving files
Finding the directory of a file using only its file name
Determining drive size and space remaining, the number of drives and drive types
Determining the unique ID of a system's hard disk
Launching Windows File Explorer from inside your app
For file copying, moving, erasing, renaming in bulk see HNDCPYDM.APP.
For file compression and decompression in bulk see HNDZDEMO.APP.
For file containerization and decontainerization in bulk see HNDSETUP.APP.

CATEGORY:
XML2HTML

*** HNDBULKMAILBATCHER.APP ***

NAME: HNDBULKMAILBATCHER.APP
TITLE: XML2HTML Docs Mail [V20.09.15]
DATE: 9/17/2020
CATEGORY: XML2HTML
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDBULKMAILBATCHER.EXE, or CHT Bulk Mail Batcher, is a batch-oriented approach to email sending via SMTP from Clarion applications or from DOS batch files using one or the other of two CHT Command Line utility applications called HNDMAILSMTP.EXE and CHTSNAPSMTP.EXE.

Generate HTML Mail With Internal Style Sheets
It is important for HTML email to use internal style sheets so that the email, on arriving in the recipient's inbox, does not need to request external style sheets from the web. It is therefore deemed "safe" by email clients and it displays immediately, like text mail, without the user needing to click any kind of "allow" switch.

Style Sheet Generating Template
This application illustrates several "_mail_" XML2HTML example templates that allow the developer to change every important aspect of numerous useful HTML style sheets so that different "looks" can easily be incorporated into different HTML-generating-procedures. To create HTML outputs with new and different looks, simply copy the existing example procedure and add your own CSS styles (or modify ours) by adjusting template switches and by removing or adding to the various "sections" incorporated into our examples.

Select Single Or Multipe Email Addresses
Select one or more email addresses from the email browse(s) using standard MS Windows record marking and an .HTML file written ahead of time using CHTSNAPEDIT.EXE or generated by one of this application's "AutoGen" procedures.

A batch file is generated, which in a loop, successively calls HNDMAILSMTP.EXE or CHTSNAPSMTP.EXE (depending which one you choose to use) to send the HTML email body to the addresses selected above. The batch file may be run either from the DOS prompt or from a simple desktop link or from inside your own Clarion applications, or using CHT's HNDBATCHRUNNER.EXE. utility.

An example of this run-from-Clarion concept, is provided in the "Run (Send Mail With) Last Generated Batch Script" menu of this application.

CHT Bulk Mail Batcher is able to import email addresses exported from other applications such as Windows Live Mail.

Preconfigure your SMTP email-send account beforehand and select that account for sending before generating your batch file.

*** HNDBULKNETMAILPROMO.APP ***

NAME:	HNDBULKNETMAILPROMO.APP
TITLE:	XML2HTML Docs Mail [V10.20.09]
DATE:	9/17/2020
CATEGORY:	XML2HTML
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDBULKNETMAILPROMO.APP, or CHT Bulk Net Mail Promo, is a HNDNETSMTP.DLL based approach to HTML-email-sending via SMTP from Clarion applications. It shares the same XML-based mail "outline" files used by HNDBULKMAILSMTP.APP.

This application processes the HTML-email directly via the above-mentioned DLL, rather than by starting an executable batch utility. Email creation and generation features, and XML and HTML source files are idential to those provided in HNDBULKMAILBATCHER.APP for cross-compatibility.

Generate HTML Mail With Internal Style Sheets
It is important for HTML email to use internal style sheets so that the email, on arriving in the recipient's inbox, does not need to request external style sheets from the web. It is therefore deemed "safe" by email clients. It displays immediately, like text mail without the user needing to click any kind of "allow" switch.

Style Sheet Generating Template
This application illustrates a CHT template that allows the developer to change every important aspect of numerous useful HTML style sheets so that different "looks" can easily be incorporated into different HTML-generating-procedures. To create HTML outputs with new and different looks, simply copy the existing example procedure and add your own style and flare by adjusting template switches and by removing or adding to the various "sections" incorporated into our example procedures.

Select Single Or Multipe Email Addresses
Select one or more email addresses from the email browse(s) using standard MS Windows record marking and an .HTML file written ahead of time using CHTSNAPEDIT.EXE or generated by one of this application's HTML-generating procedures.

When "Send" is invoked, on a group of selected records, the HNDNETSMTP.DLL sends the HTML email body to the addresses selected. Each email is processed and a success or failure outcome string is written into the email file. Images inserted into the email body are processed and included as always.

CHT Bulk Net Mail Promo is able to import email addresses exported from other applications such as Windows Live Mail. And up to 9 different email address source files can be maintained in this application. These files are shared with CHT Bulk Mail Batcher.

Preconfigure your SMTP email-send account beforehand and select that account for sending before attempting to send any mail.

*** HNDDOCUMENTBUILDER.APP ***

NAME:	HNDDOCUMENTBUILDER.APP
TITLE:	XML2HTML Docs Builder [V10.20.09]
DATE:	9/17/2020
CATEGORY:	XML2HTML
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDDOCUMENTBUILDER.EXE, CHT HTML Document Builder, is an excellent tool for creating HTML documents, for example, book chapters, manuals, and standard, static web pages. It creates portable documents that do not need to access the web in order to pick up CSS classes to handle formatting. These are generated by this application and travel inside the document.

To create a PDF of any HTML document generated by CHT HTML Document Builder, just print it to disk with Windows native PDF printer driver.

Style Sheet Generating Template
This application illustrates a new template that allows the developer to change every important aspect of numerous useful HTML style sheets so that different "looks" can easily be incorporated into different HTML-generating-procedures. To create HTML outputs with new and different looks, simply copy the existing example procedure and add your own style and flare by adjusting template switches and by removing or adding to the various "sections" incorporated into our examples.

*** HNDPREVIEWER.APP ***

NAME:	HNDPREVIEWER.APP
TITLE:	XML2HTML Previewer [V10.20.09]
DATE:	9/17/2020
CATEGORY:	XML2HTML
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDPREVIEWER.EXE, is designed to act as a specialized previewer for use with our CHT Document Builder suite of applications, such as "CHT Document Builder", "CHT Snap-Edit" and "CHT Bulk Net Mail Promo" to assist with HTML interface development.

Menus are provided to instantly set the previewer's window size to one of a variety of viewport sizes for common device sizes from small phones like the Apple iPhone to larger phones, tablets and, of course, Windows desktop.

Interacting with "CHT Document Builder" applications, like CHTSNAPEDIT.EXE and HNDDOCUMENTBUILDER.EXE, this application picks up an HTML directory location or URL from the command line passed to it from the calling application, and displays the HTML content just generated.

This provides instant feedback relative to the the document changes just made.

*** HNDPRINTHTML.APP ***

NAME:	HNDPRINTHTML.APP
TITLE:	XML2HTML Preview-Print [V10.20.09]
DATE:	9/17/2020
CATEGORY:	XML2HTML
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDPRINTHTML.EXE, can display .HTM and .HTML files and typical web content such as produced by the CHT Forum Web Application whether located locally or at a remote URL (website).

Buttons are supplied:
1) to select and load another local HTML file
2) to print the currently loaded file to the print device (PRN, PDF) selected by the user from a printer select dialog.

A URL entry control is supplied with which a remote web-site HTML file may be loaded, displayed and printed.

This app picks up an HTML file name (with path) from the command line if you wish to utilize the app as a command-line utility using Clarion RUN() or from a batch file.

If you wish to import the display/print procedure PreviewPrintHTMLMain() into your application from our app, that's easy enough to do as well.

*** HNDTILEAPPLAUNCHER.APP ***

NAME:	HNDTILEAPPLAUNCHER.APP
TITLE:	XML2HTML EXAMPLES [V10.20.09]
DATE:	9/17/2020
CATEGORY:	XML2HTML
CLASSES:	List App Classes
TEMPLATES:	List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDTILEAPPLAUNCHER.EXE, illustrates an HTML-based MAIN procedure from which other applications, and/or application procedures can be started.

The browser control on the MAIN window, launches "hndtileapplaunchermenus.html" automatically for purposes of demonstration but changing the HTML file being displayed by this application is trivial. This file is assumed to be in your application directory and if missing it is regenerated by HNDXML2HTMLGEN.EXE from an XML template we've provided called "000hndtileapplauncher.xml".

The finished HTML file is copied by the XML template to the name hndtileapplaunchermenus.html for the purposes of this example. Finished HTML names of generated entities is totally under developer control.

This application, illustrates six other .XML templates by opening them into CHTSNAPEDIT.EXE and suggesting you preview the XML file as HTML by pressing the F5 key in the editor after first clicking anywhere in the editor work surface.

Dependency files required for this application are listed in the "Version Notes To Myself" on the AACHTControlPanel information tab.

*** 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.

--prevw [OPT]
Preview HTML after generate.

--noprevw [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.

CATEGORY:
ZIP-N-POST

*** HNDZIPNEMAILBBOTMAPI.APP ***

NAME: HNDZIPNEMAILBBOTMAPI.APP
TITLE: ZIP-N-POST (BOT MAPI) [V10.20.09]
DATE: 9/17/2020
CATEGORY: ZIP-N-POST
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application" combines CHT's zip file creation capabilities using HNDNETZIPPER.DLL via CHT Classes HNDNetZipperClass with CHT's Batch-Bot SMTP executable HNDMAILMAPI.EXE launched via CHT's Global and Local templates GlobalEmbedHNDSnapCallFunctions, ProcedureSnapIns and SnapHNDMAILMAPISnapButton, respectively.

The application can be pre-configured using up to 20 potential configurations to back up a particular directory using a CHT Query that describes the files to be backed up, resulting in a standard zip file that can include source subdirectories right inside the zip.

This application's configuration button reminds only that MAPI email leverages the default email client to send mail, so it is only necessary that a working email client (e.g. Windows Live, Outlook) be present in order for email sending to operate correctly.

Once configured, your application can build a zip with one button click and send it to one or more recipients with a second click. Up to twenty separate, unique configurations are retained for instant re-use.

*** HNDZIPNEMAILCDO.APP ***

NAME: HNDZIPNEMAILCDO.APP
TITLE: ZIP-N-EMAIL (CDO) [V10.20.09]
DATE: 9/17/2020
CATEGORY: ZIP-N-POST
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDZIPNEMAILCDO.EXE<< combines CHT's zip file creation capabilities using HNDNETZIPPER.DLL via CHT Classes HNDNetZipperClass with CHT's native Clarion-code CDO classes located in HNDMSCDO.INC/HNDMSCDO.CLW.

The application can be pre-configured using up to 20 potential configurations to back up a particular directory using a CHT Query that describes the files to be backed up, resulting in a standard zip file that can include source subdirectories right inside the zip.

An SMTP/CDO configuration dialog lets you preconfigure up to three target SMTP servers to handle the sending of your email and zip attachment.

Your application can build a zip with one button click and send it to one or more recipients with a second click. Up to twenty separate, unique configurations are retained for instant re-use.

*** HNDZIPNEMAILMAPI.APP ***

NAME: HNDZIPNEMAILMAPI.APP
TITLE: ZIP-N-EMAIL (MAPI) [V10.20.09]
DATE: 9/17/2020
CATEGORY: ZIP-N-POST
CLASSES: List App Classes

TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application",HNDZIPNEMAILMAPI.EXE, combines CHT's zip file creation capabilities using HNDNETZIPPER.DLL via CHT Classes HNDNetZipperClass with CHT's native Clarion-code MAPI classes located in HNDMAPI.INC/HNDMAPI.CLW.

The application can be pre-configured using up to 20 potential configurations to back up a particular directory using a CHT Query that describes the files to be backed up, resulting in a standard zip file that can include source subdirectories right inside the zip.

Email configuration is carried out on the system's default MAPI client software. Success with this MAPI client is dependent on a MAPI-compliant email application being installed and configured on the hosting computer.

This application's email configuration button reminds only that MAPI email leverages the default email client to send mail, so it is only necessary that a working email client (e.g. Windows Live, Outlook) be present in order for email sending to operate correctly.

Your application can build a zip with one button click and send it to one or more recipients with a second click. Up to twenty separate, unique configurations are retained for instant re-use.

*** HNDZIPNEMAILNET.APP ***

NAME: HNDZIPNEMAILNET.APP
TITLE: ZIP-N-EMAIL (NET) [V10.20.09]
DATE: 9/17/2020
CATEGORY: ZIP-N-POST
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDZIPNEMAILNET.EXE, combines CHT's zip file creation capabilities using HNDNETZIPPER.DLL via CHT Classes HNDNetZipperClass with CHT's HNDNETSMTP.DLL via HNDNetSMTPClass to provide a "Handy" utility that assists with sending zips via .NET based SMTP email.

The application can be pre-configured using up to 20 potential configurations to back up a particular directory using a CHT Query that describes the files to be backed up, resulting in a standard zip file that can include source subdirectories right inside the zip.

An SMTP configuration dialog lets you preconfigure up to three target SMTP servers to handle the sending of your email and zip attachment.

Once configured, your application can build a zip with one button click and send it to one or more recipients with a second click. Up to twenty separate, unique configurations are retained for instant re-use.

*** HNDZIPNEMAILSMTP.APP ***

NAME: HNDZIPNEMAILSMTP.APP
TITLE: ZIP-N-EMAIL (SMTP) [V10.20.09]
DATE: 9/17/2020
CATEGORY: ZIP-N-POST
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDZIPNEMAILSMTP.EXE, combines CHT's zip file creation capabilities using HNDNETZIPPER.DLL via CHT Classes HNDNetZipperClass with CHT's native Clarion SMTP Classes HNDSmtp to provide a "Handy" utility that assists with sending zips via pure Clarion-code SMTP email.

The application can be pre-configured using up to 20 potential configurations to back up a particular directory using a CHT Query that describes the files to be backed up, resulting in a standard zip file that can include source subdirectories right inside the zip.

An SMTP configuration dialog lets you preconfigure up to three target SMTP servers to handle the sending of your email and zip attachment.

Once configured, your application can build a zip with one button click and send it to one or more recipients with a second click. Up to twenty separate, unique configurations are retained for instant re-use.

*** HNDZIPNEMAILSNAP.APP ***

NAME: HNDZIPNEMAILSNAP.APP
TITLE: ZIP-N-POST (SNAP MAIL) [V10.20.09]
DATE: 9/17/2020
CATEGORY: ZIP-N-POST
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application" combines CHT's zip file creation capabilities using HNDNETZIPPER.DLL via CHT Classes HNDNetZipperClass with CHT's Snap-In SMTP executable CHTSNAPSMTP.EXE launched via CHT's Global and Local templates GlobalEmbedHNDSnapCallFunctions and ProcedureSnapIns respectively.

The application can be pre-configured using up to 20 potential configurations to back up a particular directory using a CHT Query that describes the files to be backed up, resulting in a standard zip file that can include source subdirectories right inside the zip.

An SMTP configuration dialog lets you preconfigure up to three SMTP servers to handle the sending of your email and zip attachment.

Once configured, your application can build a zip with one button click and send it to one or more recipients with a second click. Up to twenty separate, unique configurations are retained for instant re-use.

*** HNDZIPNEMAILSNAPMAPI.APP ***

NAME: HNDZIPNEMAILSNAPMAPI.APP
TITLE: ZIP-N-POST (MAPI MAIL) [V10.20.09]
DATE: 9/17/2020
CATEGORY: ZIP-N-POST
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application" combines CHT's zip file creation capabilities using HNDNETZIPPER.DLL via CHT Classes HNDNetZipperClass with CHT's Snap-In SMTP executable HNDMAILMAPI.EXE launched via CHT's Global and Local templates GlobalEmbedHNDSnapCallFunctions, ProcedureSnapIns and SnapMAPIStartButton respectively.

The application can be pre-configured using up to 20 potential configurations to back up a particular directory using a CHT Query that describes the files to be backed up, resulting in a standard zip file that can include source subdirectories right inside the zip.

This application's email configuration button reminds only that MAPI email leverages the default email client to send mail, so it is only necessary that a working email client (e.g. Windows Live, Outlook) be present in order for email sending to operate correctly.

Once configured, your application can build a zip with one button click and send it to one or more recipients with a second click. Up to twenty separate, unique configurations are retained for instant re-use.

*** HNDZIPNPOSTTOCHTSERVER.APP ***

NAME: HNDZIPNPOSTTOCHTSERVER.APP
TITLE: ZIP-N-POST (HTTP CLIENT) [V10.20.09]
DATE: 9/17/2020
CATEGORY: ZIP-N-POST
CLASSES: List App Classes
TEMPLATES: List App Templates

DESCRIPTION:

This "CHT Utility Application", HNDZIPNPOSTTOCHTSERVER.EXE, combines CHT's zip file creation capabilities using HNDNETZIPPER.DLL and HNDNETZIPPER source classes, along with CHT's wide-ranging HTTP powers to provide a "Handy" utility that assists with backups to any 2-Part CHT XSA File Server like HNDSLFSVFX.APP and others.

The application can be pre-configured to 20 different settings to back up a variety of directories using a CHT File Query that describes the files to be backed up. Each setting results in a standard zip file that can optionally include source subdirectories right inside the zip. Moving from one configuration to the next requires only one button click.

The HTTP configuration dialog also lets you preconfigure the target CHT file server designated to accept your zip file(s).

This application utilizes CHT-XSA secure, encrypted authentication, so that you can authenticate against any CHT Server file server that has been provided with a CHT-XSA configuration dialog.

Once configured, your application can build a backup with one button click and PUT it to a CHT-Style secure XSA web server with a second click.