



## CNS Barcode

---

CNSBC\_AddProfile

CNSBC\_Configure

CNSBC\_ConfigureProfile

CNSBC\_Generate

CNSBC\_GetLastProfileName

CNSBC\_GetProfileNames

CNSBC\_GetProfileOption

CNSBC\_Register

CNSBC\_RemoveProfile

CNSBC\_ScanImage

CNSBC\_ScanWindow

CNSBC\_SetProfileOption

CNSBC\_Version

CNSBC\_VersionAutoUpdate

# CNSBC\_AddProfile

## Description

---

This function will add a new scanning profile to the plug-in. Profiles tell the plug-in what to do with the data it scans from a barcode.

## Return Type

---

Text

## Format

---

CNSBC\_AddProfile ( **ProfileName** )

## Required Parameters

---

### **ProfileName**

This is the name for the new profile you are adding.

## Related Items

---

CNSBC\_ConfigureProfile, CNSBC\_GetProfileNames, CNSBC\_GetProfileOption,  
CNSBC\_RemoveProfile, CNSBC\_SetProfileOption

## Example

---

### **Code:**

```
CNSBC_AddProfile( "MyProfile" )
```

### **Result:**

This will add a new profile named "MyProfile".

# CNSBC\_Configure

## Description

---

Calling this function with no parameters will open the CNS Barcode Configuration Dialog. You can optionally open to a specific tab by specifying the name of the tab as the "Option" parameter. This function also allows you to get or set any preference found in the Configuration Dialog.

Valid PrefNames:

"AddHelp" - If True, Function specific Help will be added to Calculations when inserting the Plug-in's Functions.

## Return Type

---

Text

## Format

---

`CNSBC_Configure ( Option ; PrefName ; PrefValue )`

## Optional Parameters

---

### Option

Specify the name of a Tab in the Configuration Dialog to show it opened to that tab. Specify "Get" with the PrefName parameter to get a preference value. (If not found, and PrefValue is defined, PrefValue will be returned.) Specify "Set" with the PrefName and PrefValue parameters to set a preference value.

### PrefName

The Name of the Preference to Get or Set. (See the Function Description for a list of valid PrefNames.)

### PrefValue

The Value of the Preference to Set. (See the Function Description for some possible values.)

## Examples

---

### Example 1

#### Code:

```
CNSBC_Configure
```

#### Result:

Opens the CNS Barcode Configuration Dialog. (Because the "Option" parameter is not used, the dialog will open to the "Basics" tab.)

### Example 2

#### Code:

```
CNSBC_Configure( "Get" ; "AddHelp" )
```

#### Result:

Returns the value of the 'Add Help Comments to External Functions' setting from the Configuration Dialog.

## Example 3

### Code:

```
CNSBC_Configure( "Set" ; "AddHelp" ; "False" )
```

### Result:

Sets the 'Add Help Comments to External Functions' setting in the Configuration Dialog to false.

# CNSBC\_ConfigureProfile

## Description

---

This function will bring up a dialog allowing you to configure the settings for a profile. If you call this function without any parameters, you will be able to configure all the profiles currently defined as well as add and remove profiles. If you call this function with the name of an existing profile, you will be able to configure that profile by itself.

## Return Type

---

Text

## Format

---

CNSBC\_ConfigureProfile ( **ProfileName** )

## Optional Parameters

---

### ProfileName

The name of an existing profile to configure. Leave off this parameter to edit all profiles.

## Related Items

---

CNSBC\_AddProfile, CNSBC\_GetProfileNames, CNSBC\_GetProfileOption, CNSBC\_RemoveProfile, CNSBC\_SetProfileOption

## Examples

---

### Example 1

#### Code:

```
CNSBC_ConfigureProfile( "MyProfile" )
```

#### Result:

This allows you to edit the "MyProfile" profile by itself.

### Example 2

#### Code:

```
CNSBC_ConfigureProfile
```

#### Result:

This will bring up a dialog that allows you to edit, add, and/or remove profiles in the plug-in.

# CNSBC\_Generate

## Description

---

This function will take the data you provide it and generate a barcode image (in PNG format).

## Return Type

---

Container

## Format

---

`CNSBC_Generate ( Type ; Data ; Format )`

## Required Parameters

---

### Type

This specifies the type of barcode to generate. Valid values are "Code39", "Code39+", "Code93", "Code128", "Databar14", "Databar14Stacked", "EAN", "EAN14", "EAN128", "Interleaved2of5", "ISBN", "QRCode", "UPCA", or "UPCE".

### Data

This is the data to encode into the barcode.

## Optional Parameters

---

### Format

The output Format of the Barcode Image. Valid values are "PNG" (the default), "EPS", or "SVG".

## Examples

---

### Example 1

#### Code:

```
CNSBC_Generate( "QRCode" ; "http://www.cnsplug-ins.com/" )
```

#### Result:

This generates a QR Code with an embedded URL to the CNS Plug-ins website.

### Example 2

#### Code:

```
CNSBC_Generate( "UPCA" ; "03506152005" )
```

#### Result:

This generates a UPC A barcode.

### Example 3

## Code:

```
CNSBC_Generate( "ISBN" ; "9781595544346" ; "EPS" )
```

## Result:

This generates an ISBN barcode in the EPS format (a vector format that can be cleanly resized).

# CNSBC\_GetLastProfileName

## Description

---

This function will return the name of the profile used to process the last scanned barcode. This is useful when the profile is set to call a script after reading a barcode.

## Return Type

---

Text

## Format

---

CNSBC\_GetLastProfileName

## Related Items

---

CNSBC\_ScanImage, CNSBC\_ScanWindow

## Example

---

### Code:

```
CNSBC_GetLastProfileName
```

### Result:

This returns the name of the last profile used to process a scanned barcode.



# CNSBC\_GetProfileNames

## Description

---

Use this function to get a list of names of all the currently defined profiles.

## Return Type

---

Text

## Format

---

`CNSBC_GetProfileNames`

## Related Items

---

CNSBC\_AddProfile, CNSBC\_ConfigureProfile, CNSBC\_GetProfileOption, CNSBC\_RemoveProfile, CNSBC\_SetProfileOption

## Example

---

### Code:

```
CNSBC_GetProfileNames
```

### Result:

This returns a list of names of all currently defined profiles.

# CNSBC\_GetProfileOption

## Description

---

This function will return the setting for one of the options set in the given profile.

## Return Type

---

Text

## Format

---

CNSBC\_GetProfileOption ( **ProfileName** ; **OptionName** )

## Required Parameters

---

### **ProfileName**

This is the name of the profile to get the option from.

### **OptionName**

This is the name of the option to return. Valid values are "Action", "ScriptDB", "ScriptName", "FieldDB", "FieldTable", "FieldName", or "CameraName".

## Related Items

---

CNSBC\_AddProfile, CNSBC\_ConfigureProfile, CNSBC\_GetProfileNames, CNSBC\_RemoveProfile, CNSBC\_SetProfileOption

## Examples

---

### Example 1

#### Code:

```
CNSBC_GetProfileOption( "MyProfile" ; "Action" )
```

#### Result:

This returns the current action set on the "MyProfile" profile.

### Example 2

#### Code:

```
CNSBC_GetProfileOption( "MyProfile" ; "FieldName" )
```

#### Result:

This returns the name of the field to set for the "SetField" action in the "MyProfile" profile.

# CNSBC\_Register

## Description

---

You can use this function to Register the plug-in from a script instead of through the Configuration Dialog. This is useful when the plug-in is being distributed to many computers, allowing you to install and register the plug-in without having to physically visit each computer. This function also allows you to check if the plug-in is already registered or clear the current registration. The plug-in always requires you to accept the License Agreement to use the plug-in. This is usually done by presenting the License Agreement Dialog, but that can be suppressed by using the special option value "I Accept the License Agreement".

## Return Type

---

Text

## Format

---

CNSBC\_Register ( **FirstName** ; **LastName** ; **LicenseKey** ; **Option** )

## Required Parameters

---

### FirstName

The Registration First Name you specified when you ordered. (See your Receipt.)

### LastName

The Registration Last Name you specified when you ordered. (See your Receipt.)

### LicenseKey

The License Key from your Receipt.

## Optional Parameters

---

### Option

Specify "Dialog" to enter your Registration Information in a dialog. Specify "Check" to see if the plug-in is already Registered. Specify "Clear" to remove the Registration. Specify "I Accept the License Agreement" to automatically accept the License Agreement dialog without showing it to the end user. Notes: The "Dialog", "Check", and "Clear" options can also be specified as the first parameter. If you have a Developer License, do not use the "I Accept the License Agreement" value here. See your Developer Instructions file for more information.

## Examples

---

# Example 1

## Code:

```
CNSBC_Register( "My First Name" ; "My Last Name" ; "My License Key" )
```

## Result:

Registers the plug-in with the provided registration information (obviously the above is not valid registration information; please see your Receipt).

## Example 2

### Code:

```
CNSBC_Register( "Dialog" )
```

### Result:

Displays a dialog for you to enter your First Name, Last Name, and License Key as it appears on your Receipt.

## Example 3

### Code:

```
CNSBC_Register( "Check" )
```

### Result:

Returns "Not Registered." or "Registered to <name> for a <license>."

## Example 4

### Code:

```
CNSBC_Register( "My Company Name" ; "My Company Name" ; "My Site License Key" ; "I Accept
```

### Result:

Registers the plug-in and uses the "I Accept the License Agreement" option to keep the License Agreement dialog from appearing.

# CNSBC\_RemoveProfile

## Description

---

Use this function to remove an existing profile.

## Return Type

---

Text

## Format

---

CNSBC\_RemoveProfile ( **ProfileName** )

## Required Parameters

---

### **ProfileName**

This is the name of the profile you want to remove.

## Related Items

---

CNSBC\_AddProfile, CNSBC\_ConfigureProfile, CNSBC\_GetProfileNames, CNSBC\_GetProfileOption, CNSBC\_SetProfileOption

## Example

---

### **Code:**

```
CNSBC_RemoveProfile( "MyProfile" )
```

### **Result:**

This removes the profile named "MyProfile".

# CNSBC\_ScanImage

## Description

---

This function will scan an image for barcodes and process them according to the specified profile. The image can be a stored or by reference container, or it can be a path to an image file on the hard drive. If you specify a path to an image on the hard drive, it must be in the PNG or JPEG format. For other formats, insert the image into a container field first and specify the container. If the ProfileName parameter is not specified, the function will return the data from the barcode in the image, otherwise, it will use the action defined in the profile.

## Return Type

---

Text

## Format

---

CNSBC\_ScanImage ( **Image** ; **ProfileName** )

## Required Parameters

---

### Image

This is the image to scan. You can specify a stored or by reference container object, or you can specify the path to an image file.

## Optional Parameters

---

### ProfileName

If specified, the function will look to the profile to see what action to take after scanning the barcode. If this parameter is not specified, the function will return the data found in the barcode.

## Related Items

---

CNSBC\_GetLastProfileName, CNSBC\_ScanWindow

## Examples

---

### Example 1

#### Code:

```
CNSBC_ScanImage ( MyDB::MyImage )
```

#### Result:

This will scan an image in the MyImage field and if it finds a barcode in the image, it will return the data from that barcode.

### Example 2

#### Code:

```
CNSBC_ScanImage ( "/somefolder/someimage.jpg" )
```

## Result:

This will scan the image on the hard drive and if it finds a barcode in the image, it will return the data from that barcode.

## Example 3

### Code:

```
CNSBC_ScanImage ( MyDB::MyImage ; "MyProfile" )
```

## Result:

This will scan the image from the container field and if it finds a barcode, it will use the action defined in the MyProfile profile for processing the data from the barcode.

# CNSBC\_ScanWindow

## Description

---

This function is used to work with a scanning window that contains a live video stream from your webcam. The Action parameter determines what will be done as well as what other parameters are needed. The following Actions are supported:

"Show" - This will show the Scanning Window. With this action, the ProfileName parameter is also required and set to the name of the profile to use when a barcode is scanned. Use the Left and Top parameters to define where the window appears on screen. Use the WindowTitle parameter to specify the window title. You can use the "Show" action multiple times with different ProfileName and/or WindowTitle parameters to change those settings. If the ProfileName parameter specifies a profile with the "Return" action, then the scanning window will be presented modally with a "Cancel" button. Once a barcode is scanned (or if the "Cancel" button is pressed), the window will automatically hide.

"Hide" - This will hide the Scanning Window.

"State" - This will return the current state of the Scanning Window, whether it is visible or hidden.

"Move" - This will move the Scanning Window. Use the Left and Top parameters to define where to move the window on the screen. The other parameters are ignored.

## Return Type

---

Text

## Format

---

CNSBC\_ScanWindow ( **Action** ; **ProfileName** ; **Left** ; **Top** ; **WindowTitle** )

## Required Parameters

---

### Action

This parameter tells the plug-in what to do with the Scanning Window. Valid values are "Show", "Hide", "State", or "Move". To change the profile or the window title, you can call the function with the "Show" action again with different ProfileName and/or WindowTitle parameters.

## Optional Parameters

---

### ProfileName

When using the "Show" action, you must use this parameter to specify the name of the profile to use to process the barcodes that are scanned. This parameter is only valid with the "Show" action.

### Left

This is the left pixel coordinate of the left-top corner of the scanning window. If you specify 0 or "", the last known left pixel coordinate will be used. If you specify -1, the window will be centered horizontally on the screen. This parameter is only valid with the "Show" and "Move" actions.

### Top

This is the top pixel coordinate of the left-top corner of the scanning window. If you specify 0 or "", the last known top pixel coordinate will be used. If you specify -1, the window will be centered vertically on the screen. This parameter is only valid with the "Show" and "Move" actions.

### WindowTitle

Use this parameter to specify an alternate title for the scanning window. This parameter is only valid with the "Show" action.



## Related Items

---

CNSBC\_GetLastProfileName, CNSBC\_ScanImage

## Examples

---

### Example 1

#### Code:

```
CNSBC_ScanWindow( "Show" ; "MyProfile" )
```

#### Result:

This will show the scanning window and use the "MyProfile" profile to process any barcodes that are scanned. If the scanning window was already visible, this would switch it to using the "MyProfile" profile instead of whatever profile it may have been using before.

### Example 2

#### Code:

```
CNSBC_ScanWindow( "Hide" )
```

#### Result:

This will hide the scanning window.

### Example 3

#### Code:

```
CNSBC_ScanWindow( "Move" ; "" ; -1 ; -1 )
```

#### Result:

This will move the scanning window (if it is open) to the center of the screen.

### Example 4

#### Code:

```
CNSBC_ScanWindow( "Show" ; "MyProfile" ; 0 ; 0 ; "Hold a barcode up to the camera" )
```

#### Result:

This will open the scanning window, using the "MyProfile" profile and setting the window title to "Hold a barcode up to the camera".

# CNSBC\_SetProfileOption

## Description

---

Use this function to set a profile option in an existing profile.

## Return Type

---

Text

## Format

---

CNSBC\_SetProfileOption ( **ProfileName** ; **OptionName** ; **OptionValue** )

## Required Parameters

---

### ProfileName

This is the name of the profile to set the option in.

### OptionName

This is the name of the option to set. Valid values are "Action", "ScriptDB", "ScriptName", "FieldDB", "FieldTable", "FieldName", or "CameraName".

### OptionValue

This is the value for the option.

## Related Items

---

CNSBC\_AddProfile, CNSBC\_ConfigureProfile, CNSBC\_GetProfileNames, CNSBC\_GetProfileOption, CNSBC\_RemoveProfile

## Example

---

### Code:

```
CNSBC_SetProfileOption( "MyProfile" ; "Action" ; "SetField" )
```

### Result:

This sets the "Action" option of the "MyProfile" profile to "SetField".

# CNSBC\_Version

## Description

---

This function returns the current version of CNS Barcode. This function is useful for testing whether or not the plug-in is installed and enabled. If you call this function and a question mark ("?.") is returned, then the plug-in is either not installed or not enabled.

## Return Type

---

Text

## Format

---

CNSBC\_Version

## Example

---

### Code:

```
CNSBC_Version
```

### Result:

Returns the CNS Barcode version like "CNS Barcode v.1.0.0.3".

# CNSBC\_VersionAutoUpdate

## Description

---

This function returns an Auto Update friendly Version number of CNS Barcode. The format of this version number is always exactly 8 digits long. The first two digits represent the major version of the plug-in (zero-filled). The third and fourth digits represent the minor version of the plug-in (zero-filled). The fifth and sixth digits represent the update portion of the version (zero-filled). The final two digits represent a special build number or a beta version number and will usually be zeros.

As an example, for CNS Barcode 1.2.5.0, the major version is 1, the minor version is 2, the update number is 5, and there is no special build or beta version defined. So, the resulting Auto Update friendly version number would be 01020500.

## Return Type

---

Text

## Format

---

CNSBC\_VersionAutoUpdate

## Example

---

### Code:

```
CNSBC_VersionAutoUpdate
```

### Result:

Returns 01030000 for CNS Barcode version 1.3.0.