http://www.autohotkey.com/forum/viewtopic.php?t=4888

Usage:

To create a structure, first use the StructCreate function. This will create the structure itself and create AutoHotkey variables that can be used to send/retrieve values from the structure. The variables are named by using the name of the structure, then a ? character, then the name of the variable.

For example, if StructCreate was used to create a RECT structure with the variables: Left, Top, Right and Bottom

Code (Copy):

```
StructCreate("RECT", "Int", "Left", "Int", "Top", "Int", "Right", "Int",
"Bottom")
;
; or using the alternate syntax (either is valid with version 2.0)
;
StructCreate("RECT", "Left As Int", "Top As Int", "Right As Int", "Bottom As
Int")
```

The resulting variables would then be:
RECT - The RECT structure created
RECT?Left
RECT?Top
RECT?Right
RECT?Bottom

To change the values in the RECT structure, use the Struct@ function. For example, to change the Left value to 100

Code (Copy):

```
Struct@("RECT?Left", "100")
```

To save making multiple calls if many changes need to be made, change the values of the variables first then use the Struct@ function but specify the name of the structure instead of the variable to change:

Code (Copy):

```
RECT?Left = 100
RECT?Top = 50
RECT?Right = 200
RECT?Bottom = 250
; This will change all values in the structure to the values currently assigned to
the variables
Struct@("RECT")
```

If a structure has been used in DllCall and the contents of the structure has been changed, use the Struct? function to retrieve the new values. Similar to using the Struct@ function, if you specify the name of the structure only then the variables are updated with the new values from the structure or you can specify a specific variable to only overwrite the value of that variable with the value from the structure. For example:

Code (Copy):

```
StructCreate("POINT1", "x as long", "y as long")
DllCall("GetCursorPos", "Str", POINT1)
; get the new values from the structure
Struct?("POINT1")
MsgBox, Mouse X position: %POINT1?x%
MsgBox, Mouse Y position: %POINT1?y%
```

That's about it  . To help in troubleshooting potential problems when working with structures I also put together an optional function called struct_enum. This function retrieves the current names and values of each part of the structure. For example, to view the values from the example above, you could use:

Code (Copy):

```
; Display values in POINT1 structure
MsgBox, % struct_enum("POINT1", "``=")
; Display values in variables for POINT1 structure
MsgBox, % struct_enum("POINT1", "``=", "V")
```