TagList

COLLABORATORS			
	TITLE : TagList		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		February 8, 2025	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

TagList

1.1 TagList V1.00

```
Pure Basic TagList library V1.00

  Taglists are widely used in the Pure Basic library and by the
  way, in the AmigaOS. It's simply a way to supply a list of
  paramaters to a functions. You can have any parameters you want
  in any order so, it's very flexible. These functions simplify to
  the max the use of taglists, so I hope you willn't have any
  problems with them.

Commands summary:

  AddTag
  ChangeTag
  FreeTagList
  InitTagList
  ResetTagList
  TagListID
```

1.2 addtag

```
   SYNTAX
AddTag(#TAG_ITEM, #TAG_DATA)

   STATEMENT
Add the given tags to the taglist.
```

1.3 changetag

```
   SYNTAX
ChangeTag(Position, #TAG_ITEM, #TAG_DATA)

   STATEMENT
Change the tag value at the given position in the current TagList
```

1.4   freetaglist

```
   SYNTAX
FreeTagList
```

```
   STATEMENT
```
Free the memory allocated by NInitTagList(). After this, you CAN'T
use the TagList functions any more, you must recall InitTagList()
if you need to build a taglist.

1.5   inittaglist

```
   SYNTAX
TagList.l = InitTagList()
```

```
   COMMAND
```
Init all the TagList environments for later use. You   must   put   this
function at the top of your source code if you want to use the TagList
commands. This command allocates a memory zone of (#NumMaxTags+2)*8 bytes.
It returns the memory start address or 0 if it can't allocate the memory
(should never happen). You can Free this memory when you don't need to
use the TagList any more with the NFreeTagList command.

1.6   resettaglist

```
   SYNTAX
ResetTagList(#TAG_ITEM, #TAG_DATA)
```

```
   STATEMENT
```
Rebuild a new taglist from scratch and put the given tag inside.

1.7   taglistid

```
   SYNTAX
TagList = TagListID()
```

```
   FUNCTION
```
Returns the TagList pointer for passing to any command requiring the
taglist (Window(), Screen(), ASL()...)