

Linked

COLLABORATORS

	<i>TITLE :</i> Linked	
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>
WRITTEN BY		February 8, 2025
<i>SIGNATURE</i>		

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Linked	1
1.1	Linked List V1.00	1
1.2	addelement	1
1.3	countlist	2
1.4	firstelement	2
1.5	killelement	2
1.6	lastelement	2
1.7	listbase	2
1.8	listindex	3
1.9	nextelement	3

Chapter 1

Linked

1.1 Linked List V1.00

Pure Basic - Linked List library V1.00

The Linked Lists are object which are dynamically allocated depending of your need. This is a list of elements and each elements is fully independant of the other. You can add any elements you want, inserting elements at th position you need deleting some other and more... These kind of datamanagement is very used in the AmigaOS as it's the best way to handle data when you don't know how many there are.

Commands summary in alphabetical order:

```
AddElement
CountList
FirstElement
KillElement
LastElement
ListBase
ListIndex
NextElement
```

Example:

```
Linked List demo
```

1.2 addelement

SYNTAX

```
AddElement(linkedlist())
```

COMMAND

Add a new empty element after the actual position. This new element become the current element of the list.

1.3 countlist

SYNTAX

```
CountList(linkedlist())
```

COMMAND

Count how many elements there is in the linked list. It doesn't change the actual current element.

1.4 firstelement

SYNTAX

```
FirstElement(linkedlist())
```

FUNCTION

Change the current list element to the first list element.

1.5 killelement

SYNTAX

```
KillElement(linkedlist())
```

FUNCTION

Remove the current element from the list. After this call, the new current element is the element which follow or null if you've kill the last element.

1.6 lastelement

SYNTAX

```
LastElement(linkedlist())
```

FUNCTION

Change the current list element to the last list element.

1.7 listbase

SYNTAX

```
*ListBase = ListBase(linkedlist())
```

FUNCTION

It returns the address of the list base structure ('List' on the AmigaOS)

1.8 listindex

SYNTAX

```
Index = ListIndex(linkedlist())
```

STATEMENT

Return the current list element position, considering that the first element is at the position 1.

1.9 nextelement

SYNTAX

```
Result = NextElement(linkedlist())
```

STATEMENT

Change the current list element with the next element and return its address or return NULL if there is no more elements.
