MODULE 19: SAVING & LOADING

On completion of this module you will be able to save and load the data as entered by the user and that is created by your program.

MODULE 19.1: SAVING & LOADING

Subject Outcome 1: Introduction

Subject Outcome 2: Saving

Subject Outcome 3: Loading

19.1 SAVING & LOADING

All programs, even games, must be able to save and load information from a file. You may either save/load directly using the element (SAVETOFILE / LOADFROMFILE) or indirectly using a QSTRINGLIST to store information that may be saved/loaded to/from a file.

19.1.1 SAVING

19.1.1.1 Direct Saving

The following elements are able to save directly to a file:

- QLISTBOX
- QRICHEDIT (settings format is stored with file)
- QSTRINGGRID (will be addressed with database module).

SNO	CODING	EXPLANATION

1	$TYPECHECK ON
	$INCLUDE <RapidQ2.inc>
	declare sub savelist
	declare sub saveedit
	CREATE Form AS QFORM
	  Caption = "Form":Width = 640:Height = 480:Center
	  create lister as qlistbox
	    left=10:top=10
	    additems "item 1","item 2","item 3"
	  end create
	  create edit as qrichedit
	    left=10:top=150:width=400:height=150
	    addstrings "Hi, how are you doing?"
	  end create
	  create menu as qmainmenu
	    create menua as qmenuitem
	      caption="Save List":onclick=savelist
	    end create
	    create menub as qmenuitem
	      caption="Save Edit":onclick=saveedit
	    end create
	  end create
	END CREATE
	SetWindowLong(Form.Handle, -8, 0)
	SetWindowLong(Application.Handle, -8, Form.Handle)
	Form.ShowModal
	sub savelist
	lister.savetofile "c:\ms3\mybis\listme.txt"
	showmessage "saved"
	end sub
	sub saveedit
	edit.savetofile "c:\ms3\mybis\edit.txt"
	showmessage "saved"
	end sub	

19.1.1.2 In-Direct Saving

The element we will use to store information is the QSTRINGLIST. It is like a QLISTBOX with items – however it is much more stable and meant for storing information fast and safely. You may have multiple string lists within your program.

SNO	CODING	EXPLANATION
1	$TYPECHECK ON	
	$INCLUDE <RapidQ2.inc>	
	dim blank as qstringlist	Establish element.
	declare sub recordinfo	Sub program to record file.
	CREATE Form AS QFORM	
	Caption = "Form":Width = 640:Height = 480:Center	
	create entry1 as qedit	
	left=10:top=10:onkeyup=recordinfo	As user types, it will save the file.
	end create	
	create entry2 as qedit	
	left=10:top=40:onkeyup=recordinfo	As user types, it will save the file.
	end create	
	END CREATE	
	SetWindowLong(Form.Handle, -8, 0)	
	SetWindowLong(Application.Handle, -8, Form.Handle)	
	Form.ShowModal	
	sub recordinfo	
	blank.clear	Clear the **stringlist**
	blank.additems entry1.text	Add entry 1
	blank.additems entry2.text	Add entry 2
	blank.savetofile "c:\ms3\mybis\stringlist.txt"	Save the file as **STRINGLIST.TXT**
	end sub	
2	**CLEAR** - (clears the string list).	**Additional commands related to QSTRINGLIST regarding the item entries.**
	ADDITEMS "…","…" - (add items).	
	SAVETOFILE "path$" - (saves the file).	
	ITEMCOUNT - (this holds the value of how many items are stored within the stringlist).	
	DELITEMS(index%) - (this will delete the indicated item entry).	**Blank.delitems(2)**
	EXCHANGE(index1%,index2) - (swap index ? and index ??).	**Blank.exchange(3,5)**
	INSERTITEM(index%,ENTRY$) - (Add an entry (string) at index number).	**Blank.insertitem(3,"Test")**
	SORT - This will sort the entries (numerical & alphabetically)	**Blank.sort**

19.1.2 LOADING

19.1.2.1 Direct Loading

The following elements are able to load directly from a file:

- QLISTBOX
- QRICHEDIT (settings format is stored with file)
- QSTRINGGRID (will be addressed with database module).

SNO	CODING	EXPLANATION
1	$TYPECHECK ON	
	$INCLUDE <RapidQ2.inc>	
	declare sub loadlist	
	declare sub loadedit	
	CREATE Form AS QFORM	
	Caption = "Form":Width = 640:Height = 480:Center	
	create lister as qlistbox	
	left=10:top=10	
	end create	
	create edit as qrichedit	
	left=10:top=150:width=400:height=150	
	end create	
	create menu as qmainmenu	
	create menua as qmenuitem	
	caption="Load List":onclick=loadlist	
	end create	
	create menub as qmenuitem	
	caption="Load Edit":onclick=loadedit	
	end create	
	end create	
	END CREATE	
	SetWindowLong(Form.Handle, -8, 0)	
	SetWindowLong(Application.Handle, -8, Form.Handle)	
	Form.ShowModal	
	sub loadlist	
	lister.loadfromfile "c:\ms3\mybis\listme.txt"	
	end sub	Load file into ListBox.
	sub loadedit	
	edit.loadfromfile "c:\ms3\mybis\edit.txt"	
	end sub	Load file into RichEditor.

19.1.2.2 In-Direct Loading

The element we will use to store information is the QSTRINGLIST. It is like a QLISTBOX with items – however it is much more stable and meant for storing information fast and safely. You may have multiple string lists within your program.

SNO	CODING	EXPLANATION
1	$TYPECHECK ON	
	$INCLUDE <RapidQ2.inc>	
	dim blank as qstringlist	
	declare sub recordinfo	
	declare sub loadinfo	The loading of the file into the
	CREATE Form AS QFORM	EDIT boxes.
	Caption = "Form":Width = 640:Height = 480:Center	
	create entry1 as qedit	
	left=10:top=10:onkeyup=recordinfo	
	end create	
	create entry2 as qedit	
	left=10:top=40:onkeyup=recordinfo	
	end create	
	END CREATE	
	SetWindowLong(Form.Handle, -8, 0)	
	SetWindowLong(Application.Handle, -8, Form.Handle)	
	loadinfo	Load before window appears.
	Form.ShowModal	

SNO	CODING	EXPLANATION

	sub loadinfo
	if fileexists("c:\ms3\mybis\stringlist.txt")=1 then
	    blank.clear
	    blank.loadfromfile "c:\ms3\mybis\stringlist.txt"
	    entry1.text=blank.item(0):entry2.text=blank.item(1)
	end if
	end sub
	
	sub recordinfo
	blank.clear
	blank.additems entry1.text:blank.additems entry2.text
	blank.savetofile "c:\ms3\mybis\stringlist.txt"
	end sub
		Clear the register
		Load from the file
		Assign entry 0 (first entry is zero) to ENTRY1 Edit box, next entry (1) to ENTRY2 edit box.