HyperActive Software Contact Us

Home What's New Who We Are What We Do Solutions Resources

**We make software for humans.** Custom Mac, Windows, iOS and Android solutions in HyperCard, MetaCard, and RunRev LiveCode

## Resources...

**Table of Contents**

(Partial listing)

- Introduction
- General Guidelines
- Pre-Import
- Post-Import
- The Title Card
- The About Card
- The Records Cards
- The Tips Card
- The Timer Cards
- About LiveCode Menus
- Adding a Menu Bar
- Editing the Stack Script
- Color Guidelines
- Optimizing - Timed Events
- Optimizing - About Box Substack
- Optimizing - Index Palette
- Other LiveCode Features

# Converting a HyperCard stack to LiveCode

## Moving existing objects down

**The first thing we need to do is increase the card height** to make room for the new menu group. Choose "Stack Properties" from the Objects menu.

There are width and height boxes in the palette that control the size of the card. LiveCode's default menu height is 22 pixels. Change the height dimension to 364, which is the current card height plus 22 for the menu group. The stack will resize as soon as you tab out of the field.

Now click the "Stack" tab and uncheck the "Resizable" box, which removes the size box from the bottom right of the stack window. Since we are not planning to write scripts that reposition card objects, we don't want the user to be able to resize the stack. If they could, all they would see outside the original card area would be white background.

Close the Stack Properties palette.

**Next we need to move all the existing card and background objects downward** by 22 pixels to make room for the menu group. This is best accomplished by a script.

The script of card 3 is empty, so we will use that space to write a temporary script that we will run only once. Type this into the script of the first card of the Records background:

```
on pushDown
  repeat with x = 1 to the number of bgs
    set the top of bg x to the top of bg x + 22
  end repeat
  repeat with x = 1 to the number of cds
    go cd x
    repeat with n = 1 to the number of cd parts
      set the top of cd part n to the top of cd part n + 22
    end repeat
  end repeat
end pushDown
```

For HyperCard compatibility, LiveCode accepts the use of the word `parts`, though the actual term is really `controls`. You can use either in the above script.

Compile and close the script editor and run `pushDown` from the message box. Every object in the stack will be repositioned, and a clear area at the top is now available for the menu group. Once the repositioning is done, you can delete the `pushDown` handler and save the stack to disk. You can use this script to help with any HyperCard stack conversion, not just this tutorial.

### Creating a menu bar and menu items

The easiest way to create a menu group in LiveCode is to use the Menu Builder tool. This utility creates all the menu buttons, sets the necessary properties of each so that it acts like a menu, and places it on the current card. Choose the Menu Builder from the Tools menu.

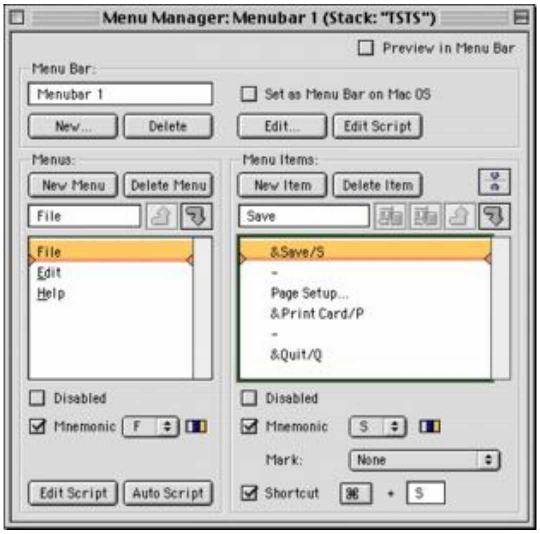
Click the New button to create a new, default menu bar. LiveCode defaults to a File, Edit, and Help menu. These are pretty much the minimums for any application on any platform, and we will use them. We want to change their contents though to suit our stack.

Click on the File listing in the left-hand field. The right-hand field displays the contents of that menu. So far, only Quit is listed.

We want to be able to save and print our stack, so we will add these lines:

```
&Save/S
-
Page Setup...
&Print Card/P
-
```

To add the first of these items, click the "New Item" button above the Menu Items field and enter "Save". The item will be inserted into the list wherever the insertion arrow is aligned. Click on the "Mnemonic" checkbox and choose "S", which will underline that character in the menu on a Windows machine. Click the Shortcut checkbox and enter "S", which will show that character as the Command-key shortcut in the menu.



Add the other four lines the same way (the second and fourth lines are just hyphens, which create divider lines.) If necessary, hilite each item in turn and use the up arrow button to move the item to the correct position in the menu items list. Position these four lines in the above order at the top of the menu item contents.

Click on the Edit menu listing in the Menu field to see its contents. Cut, copy, paste, and clear are already there, but let's add "undo". Use the "New Item" button to add these two lines to the Edit menu items field:

```
&Undo/Z
-
```

The second line is again just a hyphen. Use the up or down arrows as required to position these two items at the top of the Edit menu contents.

The Help menu needs no changes, so we don't need to do anything there.

We also want to add an additional menu which will hold the dynamic Index. Click the "New Menu" button above to the menu list and enter:

```
Index
```

This menu has no default items, so leave the menu items field empty. Set its mnemonic to "I". Use the arrow buttons to move the Index menu into position so that it is between the Edit and the Help menus.

**Remember:** to function correctly, the Help button must always be last in the list.

We want to use the automatic skeleton script that LiveCode can place into each menu for us. Click on each of the four Menu listings in turn and click the "Auto-script" button to insert a default handler.

Finally, check the "Set as menubar on Mac OS" checkbox. This tells LiveCode which group to insert into the menu bar when the stack is running on a Macintosh. This setting has no affect when running on other operating systems, since the menu group is always visible in the stack window on those systems. Close the Menu Manger.

Finally, select the "menubar 1" group in the Application Browser, click the Properties button, and select the Group tab in the palette. Check the `Background Behavior` checkbox. Setting `the backgroundBehavior` of a group to true causes that group to be automatically duplicated on every new card, in the same way that HyperCard handles backgrounds. Since we want a menu group on each card, this is the desired behavior. LiveCode sets this property to true automatically for all backgrounds that are imported from a HyperCard stack, but any new groups you create in LiveCode must be set manually if you want that behavior.

**Note about menu buttons:** The menu items of a menu button are simply a return-delimited list stored in the button contents. After the menu group is initially built, it is easy to edit the menus by changing the contents of each menu button. You do this just as you would in HyperCard. You can view the contents of each button by double-clicking the button with the edit tool and choosing the last tab in the properties window. The contents can be edited via script just as in HyperCard. You can also edit an existing menu group by simply opening it again in the Menu Builder.

### Putting a menu on each card

A custom menu will only appear in the menu bar if the current card contains a menu group, which means we need to place this group on every card. (The exception is if a `defaultMenuBar` has been assigned to the stack.) When you are building stacks from scratch in LiveCode, you can set LiveCode to create new cards with the same groups that were on the card you started from, just like HyperCard does. But in a HyperCard conversion where the cards have already been created without menus, we need to add the menu group to each imported card.

LiveCode offers a way to to do this. If it is not already open, choose the Application Browser from the Tools menu. Double-click on each card in the left-hand list to go to it, and use the "Place" menu item in the Objects menu to place "Menubar 1" on each card. Save the stack to disk.

**To see how the menus will look later:** You can see how the menu bar will look on a Mac by opening the menu group in the Menu Builder and checking the "Preview in menu bar" checkbox. The stack will collapse itself to hide the menu group. If you click on the stack to bring it frontmost, its menus will be visible in the Mac's menu bar. None of the menu items will work yet because they have no scripts. Un-check the "Preview in menu bar" checkbox to bring the menu group back into view so we can work on it some more.

### Writing the menu scripts

To edit the menu scripts, Option-Cmd-click each button to open it in the script editor, or you can access their scripts from the Application Browser. Edit each of the following button scripts.

**File Menu:**LiveCode provides a default `switch` script structure for menu buttons. We only need to fill in the details. (A `switch` control structure is like a streamlined `if-else` control structure, but more efficient.) Edit the File button script to look like this (changes are in red):

```
on menuPick pWhich
  switch pWhich
  case "Save"
    save this stack
    break
  case "Page Setup"
    -- shows Page Setup on Mac, choose printer on Windows:
    answer printer
    break
  case "Print Card"
    if the short name of this bg = "records"
    then print this card from "0,26" to "512,368"
    else beep
    break
  case "Quit"
    quit
    break
  end switch
end menuPick
```

The menuPick message is sent to a menu button whenever a menu item is chosen, along with a parameter which is the name of the menu item. The switch statement determines the resulting behavior.

**Edit Menu:** LiveCode provides simple commands for common editing functions. Change the script of the Edit menu to look like this:

```
on menuPick pWhich
  switch pWhich
  case "Undo"
    undo
    break
  case "Cut"
    cut
    break
  case "Copy"
    copy
    break
  case "Paste"
    paste
    break
  case "Clear"
    put empty into the selection
  end switch
end menuPick
```

Note that it is not necessary to specify the type of content that is to be cut, copied, or pasted as it is in HyperCard; i.e., paste picture and copy text are not menu commands in LiveCode. The copy, cut, and paste commands function universally, regardless of the content being manipulated.

**Index Menu:** The index menu contents are built dynamically by a handler in the stack script, which we will edit later. (And even further down the line, in the Optimizing for LiveCode section, we will be deleting this menu entirely and replacing it with a palette instead.) For now, we will just script the behavior that should occur when a user chooses one of the menu items. Each menu item will eventually be the name of a product that has been entered into the "Product Name" field on a Records card.

Change the Index menu button script to this:

```
on menuPick pWhich
  go cd lineOffset(pWhich,me) of bg "records"
end menuPick
```

Note the lineOffset function, which HyperCard doesn't provide. LineOffset() takes two parameters: the string to locate, and the container to locate it in. It returns the line number in which the string is found. In this case, we are getting the line number of a product name which is within a list stored in the button contents. LiveCode offers a similar itemOffset function.

**Help Menu:** The Help menu will take us to either the Help card or the About card, depending on the user's menu choice. Change the script to this:

```
on menuPick pWhich
  switch pWhich
  case "Help..."
    go cd "Help"
    break
  case "About..."
    go cd "About Card"
    break
  end switch
end menuPick
```

That takes care of the menus for now. You can turn on "Preview in menu bar" in the Menu Builder, bring the stack to the front, and test them (or you can just test the menus in place.) We will add more to the menus when we edit the stack script, which is next.

Up to top | Converting Stacks to LiveCode - TOC | Next Page

Home What's New Who We Are What We Do Solutions Resources Site Map Contact Us