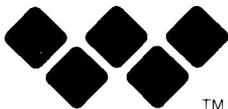
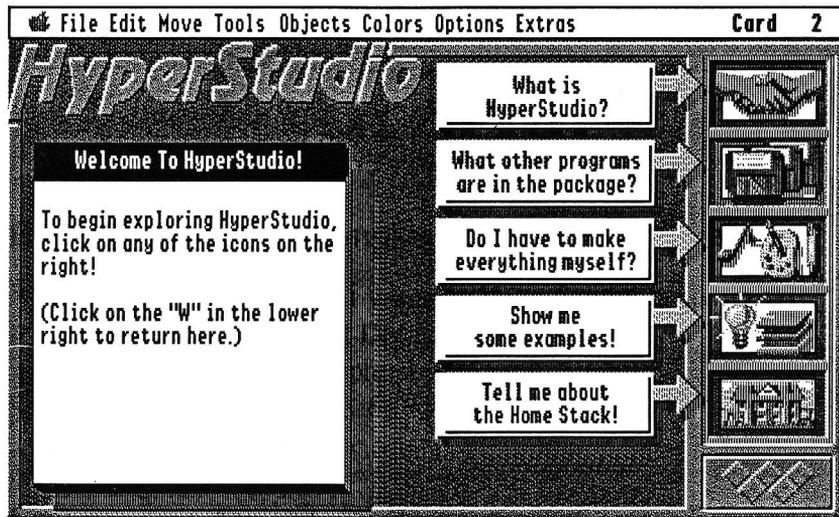


HyperStudio[®] tutorial

HyperStudio comes complete with...

- Award-winning HyperStudio Software
- Sound Digitizing Hardware
- Microphone
- Clip-Art and Clip-Sounds
- Sample Stacks



Roger Wagner[™]
PUBLISHING, INC.

HyperStudio[®]

Tutorial

the premier hypermedia/
multimedia system for
the Apple IIGS

by Michael O'Keefe, Dave Klimas, and Jeff Smith
copyright 1998-'92 Roger Wagner Publishing, Inc.
(619) 442-0522

Design and Interface: Roger Wagner and Michael O'Keefe

Programmers: Michael O'Keefe, Dave Klimas,
and Jeff Smith

Contributing Programmers: Roy Bannon, Garrick Toubassi, Ken
Kashmarek, Steve Allen, Mike Nuzzi, Terry
Romine, and Eric Mueller

Contributing Artists: Bob Cavey, Bo Monroe, Donald McIntosh,
and Greg Roach

Contributing Stack Designers: Bo Monroe, Steve Allen, Donald McIntosh,
Greg Roach, and Dean Esmay

HyperStudio Tutorial: Della Smith and Eric Mueller

HyperStudio Reference: Eric Mueller, Bert Klimas, and Roger
Wagner

Special Thanks: Garland Buckingham, Valerie Laird, Leslie
O'Brien, Sandy Goez, and Debbie Midgley

Dedicated To: Pam Wagner, Bert Klimas, Andrea
Marcucci, and Della Smith

Produced by:
Roger Wagner Publishing, Inc.
1050 Pioneer Way, Suite P
El Cajon, California 92020

© Copyright 1988-'92 by Roger Wagner Publishing, Inc. **HyperStudio**
is a registered trademark of Roger Wagner Publishing, Inc. All rights
reserved worldwide.

Customer Service and Technical Support
(619) 442-0522
9:30am - 5:30pm Pacific time
please have your HyperStudio serial number ready

Apple, HyperCard, Apple IIGS, and GS/OS are registered trademarks of
Apple Computer, Inc.

All other copyrights and trademarks are held by their respective owners.
We apologize for any omissions; they are not intentional.

ISBN 0-927796-40-6

revision 1.2 12/13/91; first printing, December 1991

10 9 8 7 6 5 4 3 2 1

Table of Contents

Introduction	1
Type Styles and Icons.....	1
Changing Disks	1
The Data Disk.....	1
Keyboard Equivalents.....	1
Moving from card to card.....	1
HyperStudio Tutorial Goals.....	3
HyperStudio Hands-On Walk Through.....	4
Starting the Program.....	4
Exploring the Paint Tools.....	4
Deleting Items on the Screen	5
CARD #1—The Title Screen.....	6
Adding the Border.....	7
Adding the Text	7
Adding a Button.....	9
Adding Clip Art.....	13
Adding the Text	14
Adding a Button.....	15
Adding Sound to a Button.....	16
CARD #3—USA Map.....	18
Adding an Invisible Button.....	19
CARD #4—The Help Card	25
Adding a Border.....	25
Adding a Text Item	26
Adding a Button.....	28
Digitizing Images with LightningScan	33
Setting the Scanner	33
Importing Scanned Images into HyperStudio	34
Digitizing Images with the Quickie Scanner.....	37
Importing Scanned Images into HyperStudio	38
Digitizing an Image from a Video Camera.....	41
Setting Up ComputerEyes.....	41
Using A ComputerEyes Image in HyperStudio.....	42
Adding a Text Item	43

Button Connections.....	44
Creating an Animation.....	47
The Text.....	47
The Animation—Creating the “Movie”.....	48
Adding a Button to Activate the Animation.....	50
Aligning the Graphic.....	52
Using SimpleScript.....	55
Your First Program.....	55
The First Line: Setting the Position.....	56
The Second Line: Setting the Font.....	59
The Third Line: Setting the Color.....	60
The Final Line: Drawing the Text.....	60
Running the Program.....	61
Looping and Variables.....	62
Recording the Sound.....	62
Writing the Program.....	63
How Does It Work?	63
Modifications.....	65
The Quiz Stack.....	66
Preparing the Stack.....	66
Creating the Button.....	69
How it Works.....	70
Suggested Changes.....	73
Where To Go From Here?	73
Conclusion.....	74

Introduction

Welcome to the HyperStudio tutorial! We think you're going to enjoy the upcoming activities. Here are some miscellaneous hints and tips on navigating within HyperStudio and this workbook.

Type Styles and Icons: All menu items that you must use are in *italics*, while all menus themselves are in **boldface** type. Disk names and filenames will be in this special type.

★ One-Drive System

This icon appears next to all steps required on a one-drive system. If you are running HyperStudio off of a hard drive or network, you may skip these steps.

⌘ Two-Drive System

This icon appears next to all steps required on a two-drive system. If you are running HyperStudio off of a hard drive or network, you may skip these steps.

⇒ Hard Drive or Network

This icon appears next to all steps required on a hard drive or network. If you are running HyperStudio off of a one-drive system, you may skip these steps.

At certain points in this tutorial, a particularly important procedure or concept will be discussed, and highlighted with a box like this. **Be sure to read these sections very carefully, and make sure you understand them completely, before going on.**

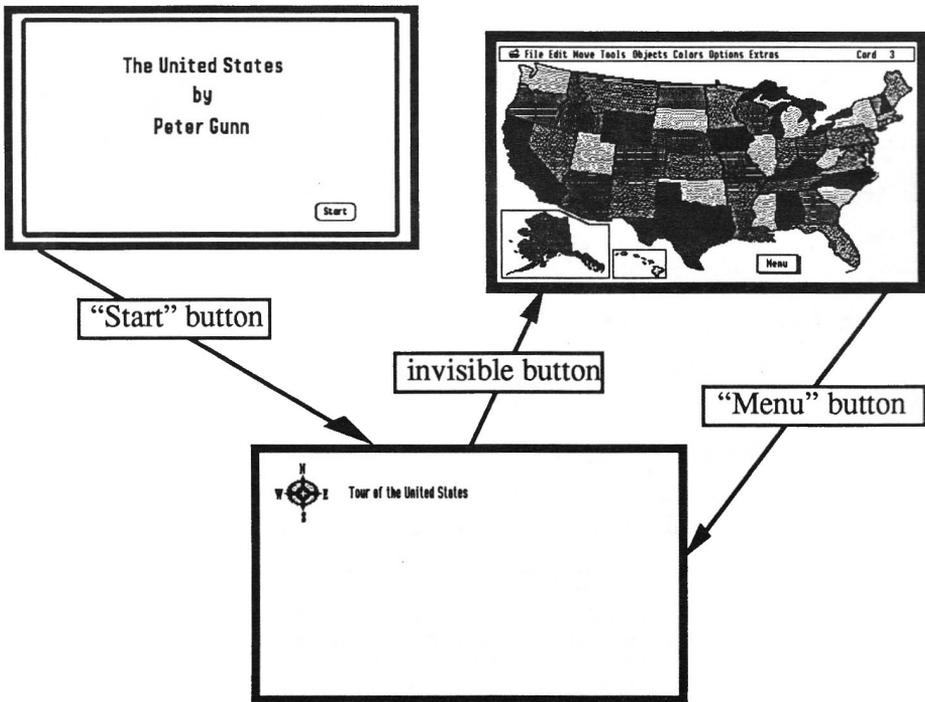
Changing Disks: Occasionally, while going through this tutorial, HyperStudio will ask for a specific disk. (This almost only happens the first time you perform an action, such as adding a button or text field.) Simply follow the directions on the screen; usually, they will involve inserting the specified disk and then clicking an "OK" button. Also note that on the HyperStudio disk labels, the name of the disk starts with a slash and is on the third line from the top.

The Data Disk: You will save the results of all of the workbook exercises on a 3.5" floppy disk, known as the *data disk*. You need to have this disk formatted before you begin working through this tutorial. (To format a disk, simply run the Finder and insert the disk in the drive. If the disk is already formatted, it will appear on the desktop; otherwise, the Finder will ask if you want to format it.)

Keyboard Equivalents: Many of the the HyperStudio commands can be duplicated with keyboard equivalents. In HyperStudio's pull-down menus, these are indicated next to the command with the Apple symbol (⌘). They are also referred to within the instructions as *Apple-*.

Moving from card to card: In order to move from card to card within the stack, use either the Move menu commands or the Apple-> and Apple-< keys. The > key moves to the next card in the stack, while the < key moves to the previous card.

Here's a flow chart of how your first stack will look and operate when it's complete:



HyperStudio Tutorial Goals

- HyperStudio Hands-On Walk Through—pages 5–32**

This section will teach you the eight basic HyperStudio skills. They are:

- ❖ **Making a card**
- ❖ **Loading a background**
- ❖ **Making a button**
- ❖ **Linking a button to a card or sound**
- ❖ **Importing clip art graphics**
- ❖ **Cutting and pasting objects**
- ❖ **Saving a stack**
- ❖ **Running a stack**

Other activities included in this tutorial are:

- Scanning with the Lightning Scan or the Quickie Hand Scanner—pages 33–40**
- Using the ComputerEyes Video Digitizer—pages 41–46**

Additionally, the following advanced subjects are covered:

- Adding an Animation—pages 47–54**
- Using SimpleScript—page 55–73**

HyperStudio Hands-On Walk Through

Note that this tutorial assumes that you have HyperStudio ready to launch. This means that you have the disks backed up, and, if you are using a hard disk or network, installed. Additionally, you need to be certain that your Control Panel is set correctly.

For more information, please see chapter 2 of the HyperStudio Reference Manual, “Setting Up.”

Starting the Program

If HyperStudio has already been launched, go directly to the “Exploring the Paint Tools” section, below.

⇒ Hard Drive or Network

Note to hard drive or network users: simply launch HyperStudio and skip directly to “Exploring the Paint Tools,” below.

- Insert the /HS.System disk into the disk drive.
- Hold down the Apple and Control keys, and then press and release the Reset key (while continuing to hold down the Apple and Control keys).
- Release the Apple and Control keys.

The disk drive will be accessed, and you will see the HyperStudio startup screens. Finally, you will be at the welcome stack.

- Click the “W” icon, in the lower left corner of the screen.
- You will be prompted to insert the /HyperStudio disk. Go ahead and insert the disk and click “OK.”

Now you’re in the HyperStudio program!

Exploring the Paint Tools

The purpose of this section is to give you an opportunity to explore all of the various HyperStudio paint tools. If you haven’t used a computer paint program before, you should spend at least five minutes trying the various tools. **Do make a point to try out each of the options described.** Many of the instructions in the remainder of this tutorial assume that you are comfortable using the paint tools.

If you are familiar with paint programs, you may wish to skip this section and go on to Card #1—The Title Screen.

- Choose *New Stack* from the **File** menu.
- Choose the *Round Rectangle* from the **Tools** menu.

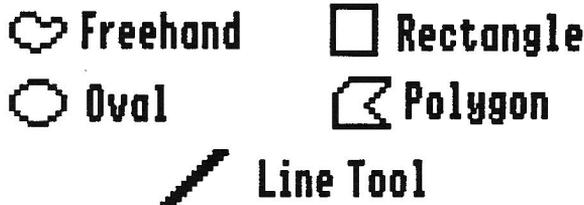
The Round Rectangle looks like this:



If you've never used this type of paint tool before, this is a good time to practice drawing on the screen. To do so, position the cursor, which appears as a cross-hair, wherever you'd like the upper-left corner of the rectangle to appear. As you press down on the mouse button, drag the mouse diagonally across the screen to the lower-right side of where you want the rectangle to end. Draw several rectangles on the screen.

- Choose different line widths by choosing *Line Size...* from the **Options** menu.
- Choose different colors from the **Colors** menu.

You may also wish to draw with some of the other paint tools that work in a similar manner. Try the following tools if you'd like:



Deleting Items on the Screen

When the screen is full, you have several choices for erasing what you've drawn. They are using the *Eraser Tool* or *Selector Tool* from the **Tools** menu or choosing *Erase Background* or *Undo* from the **Edit** menu. The next section details using each process.

- Choose *Undo* from the **Edit** menu. This will undo the last action you did. Use the line tool to draw three lines. Now, choose "Undo." Note that undo only reverses the very last action—in this case, removing the third (and most recent) line drawn. If at any point while creating things with HyperStudio, you realize you made a mistake in adding or deleting any portion of the screen, *immediately* choosing "Undo" will reverse your last action.
- Choose the *Eraser* from the **Tools** menu. It looks like this:



While pressing the mouse button, move the eraser over the area you wish to erase. Erase part of the screen. You can also erase items by selecting them with the

Selector Tool and then pressing the Delete key. Everything that is selected will be deleted.

- Choose the *Selector Tool* from the **Tools** menu. The *Selector Tool* looks like this:



This is one of the most often-used tools in HyperStudio. Make sure you understand how to use it!

To use the *Selector Tool*, position the mouse at the upper-left corner of some graphic image on your screen. While pressing on the mouse button, drag the mouse diagonally across the screen to just past the lower-right side of what you wish to select. A dotted, rectangular box will appear around the area you have selected. If you didn't get the area you wanted, simply click outside the box and try again.

To move the image, place the cursor over the inside of the selected area. Now, hold down the mouse button while you drag the mouse, and you'll notice that the selected area of the screen will follow the mouse around. When you are done moving, click the mouse outside of the selected rectangle to "drop" the graphic at that position.

From the **Edit** menu, choose *Undo*. If you haven't selected any new area before choosing *Undo*, the image will return to its original position.

Use the *Selector* tool to select a portion of the screen again. Suppose you wanted to erase the selected area? Press the Delete key while a section of the screen is selected, and it will be removed. Choose *Undo* to restore the selected area.

- Choose *Erase Background* from the **Edit** menu. This will erase everything on the screen.

CARD #1—The Title Screen

In this activity, you will create a small, three-card stack. You will use many of HyperStudio's basic features while creating this stack.

- From the **File** menu, choose *New Stack*.
- Look under the **Options** menu, and be sure that "Draw Filled," "Draw Multiple," and "Draw Centered" all do *not* have checkmarks next to them. If an option has a checkmark next to it, select that option to remove the checkmark.

In this stack, we will have a yellow background color on each card. The **normal**, default background color is white, so we need to change it.

- Set *Background Color...* from the **Options** menu to yellow, then click "Erase Now" to immediately erase the background. (If you were to click "OK" instead of

“Erase Now,” HyperStudio would simply change the background color that it knows internally without the card being erased and re-drawn: effectively, doing nothing on the screen until a new card is added to the stack.)

Adding the Border

- Choose the *Round Rectangle* from the **Tools** menu.
 - Choose a medium width *Line Size* from the **Options** menu.
 - Choose the *red color* from the **Colors** menu.
- Draw a border, starting at the upper-left corner of the screen. To do so, position the cursor, which appears as a cross-hair, on the upper-left side of the screen. As you press down on the mouse button, drag the mouse diagonally across the screen to the lower-right side of the screen. Release the mouse button. Make the border as big as the monitor screen, following the edges of the screen.

If you decide you want to draw the border again, choose *Undo* from the **Edit** menu, and try again. “Undo” only undoes the last step, so you may need to choose the *Eraser Tool* or *Erase Background* from the **Edit** menu for a completely fresh start.

Adding the Text

- Choose *Text Style* from the **Options** menu.
- Choose “Shaston”, which has already been highlighted as the default font.
- Click “16” as the font size.
- Click in the box next to “Bold” to choose it as the style.
- Click “OK” to exit the Text style dialog box.
- Choose the *Text* tool from the **Tools** menu.

It looks like this:

T Text

- Choose the *red color* from the **Colors** menu.
- Click the mouse in the left side of the screen about 1 inch from the top to set where you will start typing.
- Type **The United States** (Press Return.)
(Press the space bar until the cursor is centered under the first line.)
by (Press Return.)
(Press the space bar until the cursor is centered under this line.)
Your Name (Press Return.)
- If necessary, choose the *Selector Tool* from the **Tools** menu to move the text. If you need to erase the text, select it with the *Selector Tool* and press the Delete key.

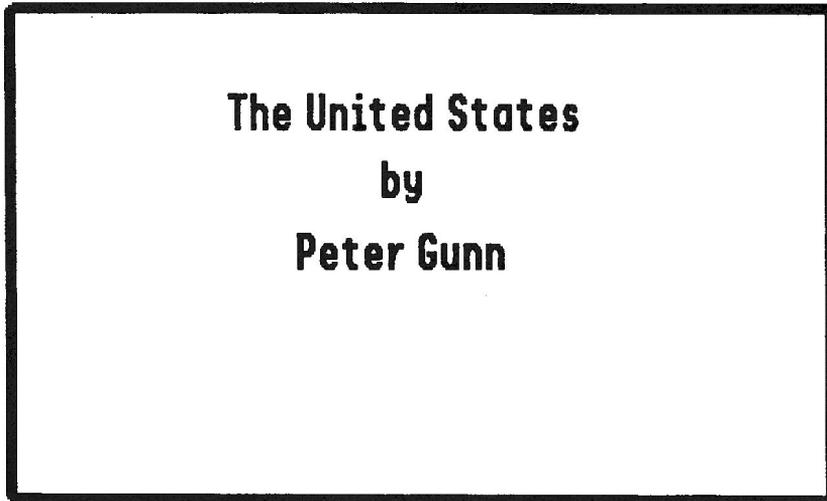
The *Selector Tool* looks like this:

 **Selector**

To use the *Selector Tool*, click the upper-left side of the area you wish to select (in this case, **just to the upper-left** of any of the text you just typed). You may select all of the text, or just one word, depending on what needs to be repositioned. While pressing on the mouse button, drag the mouse diagonally across the screen to just past the lower-right side of what you wish to select. Release the mouse button when the desired area is selected. A dotted, rectangular box will appear around the area you have selected. If you didn't get the area you wanted, simply click outside the box and try again. To move the selected area, place the cursor inside the box. While pressing on the mouse button, drag the box to the new location. Click outside the box to drop it in the new location.

If you need to start over, simply select the text and press the delete key. This will erase all of the selected text and give you a fresh screen to use.

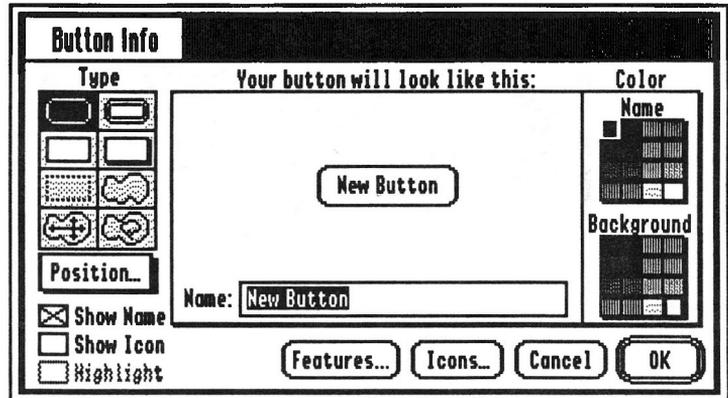
Your screen should look like this (this is an example of the entire screen):



Adding a Button

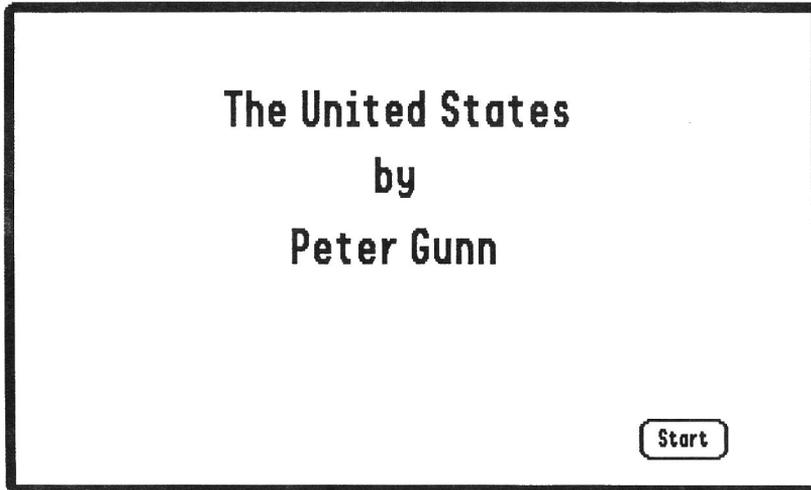
Now we will add a button to this page. It will connect to the next card.

- Choose *Add a button* from the *Objects* menu. The “*Button Appearance*” dialog box will appear. This dialog box is where you control how your button will appear—selecting different options will affect the button you are building, which is displayed in the middle of the dialog box.



- Press the “Clear” key. It is to the right of the Delete key, above the number seven on the key pad.
- Type *Start*
- Click “*Position...*” to give the button a position.
- If an instructional dialog box appears, read it, then click “*Continue*”. (This dialog box will only appear the first time a button is added.)
- Drag the button to the lower-right corner of the screen by placing the cursor inside the button, pressing down on the mouse button, and moving the mouse so that the button is placed on the lower-right corner of the screen.

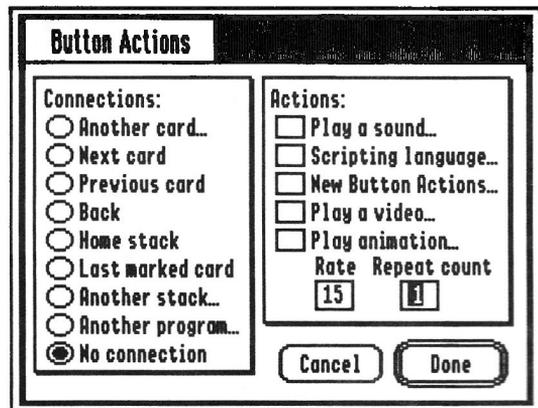
Your screen should look like this:



• Click outside the button to place it. You must always click outside the button to tell the computer that you've chosen a position for the button.

You will be returned to the Button Info screen. Since you're done defining the button's name, position, and style, you can continue, so click the "OK" button to proceed to the Button Actions screen.

The Button Actions screen has many options for what you wish the button to do. For now, we will connect it to the next card.



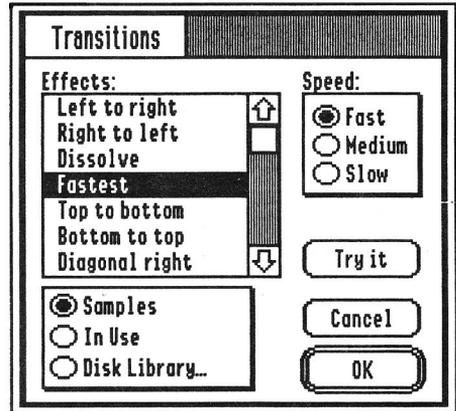
• In the upper-left corner of the Button Actions screen, under the words "Connections:", click the radio button by "Next card."

- The “Transitions” dialog box will then appear.
- Click next to the “Blocks” transition from the list on the left side of the dialog box.
- Click “OK”.

If you wanted this button to do more, you could choose more actions, but connecting to the next card is all we want this button to do.

- Click “Done” on the lower-right section of the Button Actions screen.

• Make sure that your data disk is not write-protected!



★ One-Drive System

Note to one-drive system users: remove the /HyperStudio disk and insert your data disk into the drive.

- Choose *Save Stack As...* from the **File** menu.

⌘ Two-Drive System

Note to two-drive system users: insert your data disk into the second drive, and be sure to select the location of your data disk by holding down the Apple key and pressing the TAB key until the name of your data disk (/DATA) appears at the top of the dialog box. **This is very important!** If you don't select your data disk as the destination, you will save onto the hard drive.

⇨ Hard Drive or Network

Note to hard drive or network users: be sure to select the location of your data disk by holding down the Apple key and pressing the TAB key until the name of your data disk (/DATA) appears at the top of the dialog box. **This is very important!** If you don't select your data disk as the destination, you will save onto the hard drive.

- Press the “Clear” key.
- Type **USA** in the entry box. You must type the file name exactly like this. GS/OS does not allow spaces in names; you may only enter letters, periods, or numbers.
- Click “Save.”

You have just created a title screen for your stack. In doing so, you have used paint tools to create a border and add text. You have also created your first button and saved your stack onto your data disk. Now we'll make the next card in your stack.

CARD #2—The Menu Card

Adding Clip Art

This menu card will have buttons that connect to the rest of this stack. We'll add one button for now, with a sound effect as a transition from this card to the next. Later, you can add more buttons to this page when you add to this stack. For now, we'll add a clip-art drawing that we'll use as part of a button.

- Choose *New Card* from the **Edit** menu. This creates the second card in your stack.
- Choose *Add Clip Art* from the **File** menu.
- You will see a dialog box asking you to select a file for the clip art. You want a picture from a different disk, though.

★ | One-Drive System

Note to one-drive system users: insert the /HS.Art disk into the disk drive. A dialog box will appear, informing you that there is no device on-line. Click "OK".

🎧 | Two-Drive System

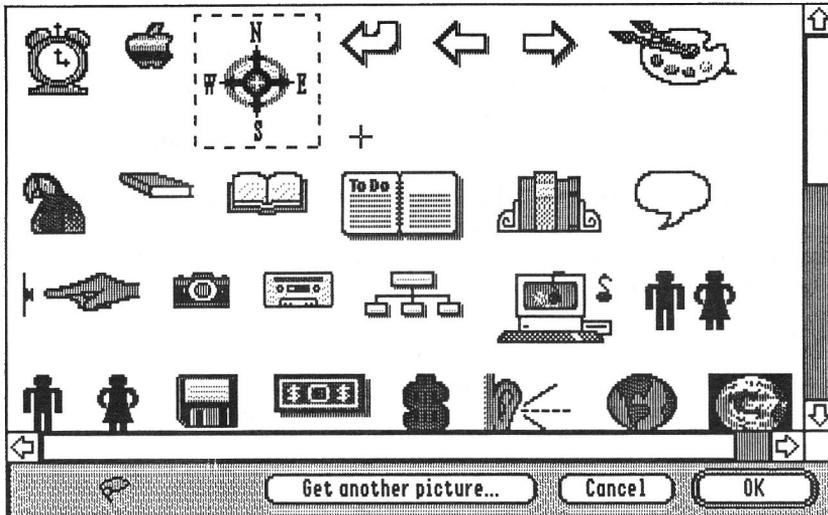
Note to two-drive system users: remove your data disk from the second drive and insert the /HS.Art disk into the disk drive.

⇒ | Hard Drive or Network

Note to hard drive or network users: locate and open the HS.Art folder on the hard drive. It should be in the HyperStudio folder.

- Scroll downward through the files by putting the mouse cursor on the "down" arrow, and pressing the mouse button. Click the `Icon.Library` file to select it. (There is a folder called "Icons" on the /HyperStudio disk. This is not the file you're looking for. Make sure you've located the `Icon.Library` file on the /HS.Art disk or folder.
- Click "Open".
- After a short delay, the clip art selection screen will appear. The upper portion of the screen will contain the Icons picture, and the lower part will have several options for working with the screen. The *Selector* tool will automatically be selected.
- *Select* the compass by using the *Selector* tool to draw a rectangle carefully around the compass, and nothing else on the screen. To draw the line, drag the cursor from one corner of the compass to the other.

The entire compass should be selected, and look like this:



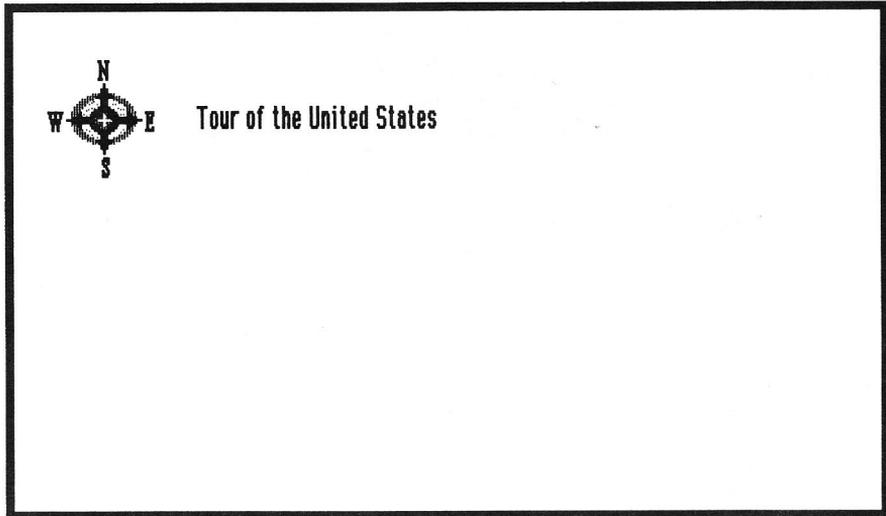
- Click “OK” at the bottom of the screen.
- Position the compass on the upper-left side of the screen by placing the cursor inside the compass, pressing the mouse button, and dragging the compass up to the corner.
- Click outside the compass to place it on the screen.
- If there are any stray marks on the screen, select them with the *Selector Tool* and press the Delete key to erase them, or use the *Eraser Tool*.
- Finally, as you can see, the compass is still surrounded by a white background, though the rest of the card is yellow. To fix this, you need to change the white around the compass to yellow. To do this, select the *Fill* tool from the Tools menu, and yellow from the Colors menu.
- With the *Fill* tool selected (which looks like a little paint bucket tipped over), click in the white area around the compass, and it will be “filled” with the yellow color.

Adding the Text

- Move the mouse back up to the menu bar. Make sure that you’ve moved all the way to the top of the screen and that your cursor has changed to a pointer.
- Choose the *Text* tool from the Tools menu.
- Choose *Text Style* from the Options menu.
- Click “Geneva” in the list of fonts.
- Click “12” to select the size.
- Verify that “Bold” is still marked and if it isn’t, click “Bold” to select it.
- Click the red square below the words “Text Color” to set the color to red.
- Click “OK” to exit the Text style dialog box.

- Click the mouse to the right of the compass icon to position the cursor. Type **Tour of the United States**

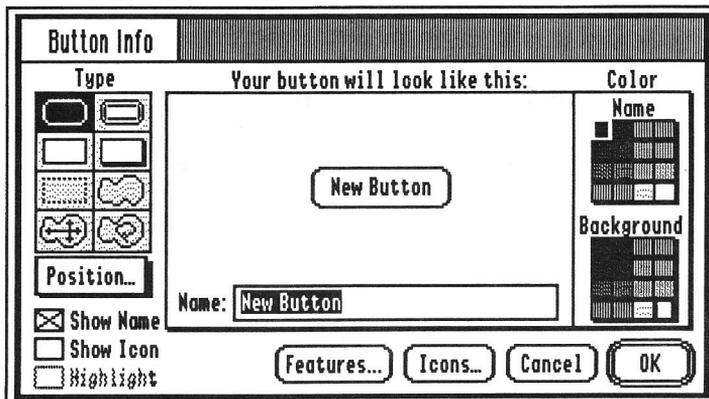
Your screen should look like this:



- Use the *Selector Tool* to re-position the words, if necessary. Use the same technique as you did on the Title Screen.

Adding a Button

- Choose *Add a button* from the **Objects** menu. The “Button Info” dialog will appear.



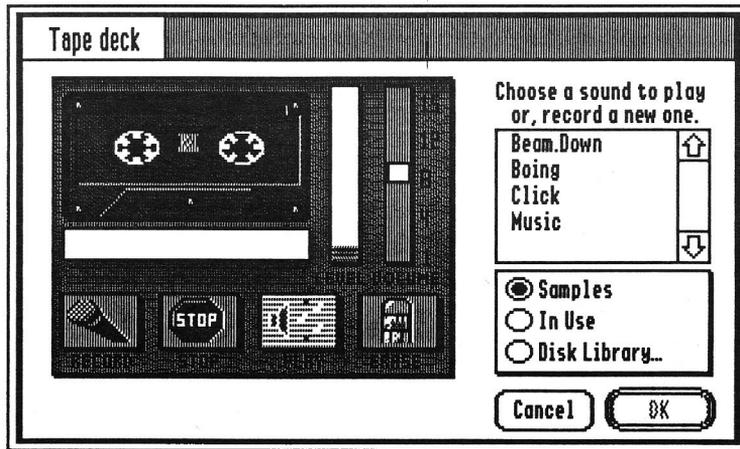
- Click the icon for the rectangular invisible button style. It's the third button type from the top on the left side of the dialog box, and looks like this: 
- Click "Position..." to accept your choice and set the position of the button.
- Drag the dotted box over the compass by placing the cursor inside of the box, pressing down on the mouse button, and dragging the mouse so that the box covers the compass.

The invisible button must now be sized to better fit the compass. To size the button, you must place the tip of the pointer cursor on one of the box corners. The tip is the only portion of the pointer cursor that is "hot". (In other words, the part that the computer recognizes when it is clicked.) If you click outside the box, HyperStudio will think that you are finished placing the box. If this happens, don't worry. Simply choose "Cancel" on the Button Info screen and start over by choosing *Add a button* from the **Objects** menu again.

- Place the tip of the cursor on one of the box corners.
- Press on the mouse button.
- Drag the mouse towards the compass icon (while pressing on the mouse button). Make the box fit the compass by moving the sides of the box in this manner.
- Click outside the dotted box when you have sized it correctly (to place the button on the screen).
- You will be returned to the Button Info dialog box. Click "OK" to proceed with the button creation process.
- Click next to "Next card" in the upper-left "Connect to:" box on the Button Actions screen. The "Transitions" screen will appear.
- Choose "Fade to black" as the transition on the "Transition" screen.
- Click "OK" to accept your visual effect.
- Don't click "Done" yet, since we want this button to also play a sound.

Adding Sound to a Button

- Click in the box next to "Play a sound" in the upper-right "Actions" box on the Button Actions screen.
- The HyperStudio tape deck screen will appear with a list of the sample (built-in) sounds on the right. We could use one of those sounds for our button, but instead, let's go to the disk and load one from `/HS.Sounds`.



the HyperStudio Tape deck

- Click the “Disk library...” button.
- Now we need to locate the Dramal file from the /HS.Sounds disk.

★ | One-Drive System

Note to one-drive system users: insert the /HS.Sounds disk into the drive. A dialog box will appear, informing you that there is no device on-line. Click “OK”.

🔗 | Two-Drive System

Note to two-drive system users: remove the /HS.Art disk and insert the /HS.Sounds disk into the second disk drive.

⇒ | Hard Drive or Network

Note to hard drive or network users: locate the /HS.Sounds folder. It will probably be in your HyperStudio folder.

- Click Dramal to select it, then click “Open”.

- Click “Play” to hear the sound you’ve just added. Adjust the volume, if necessary, with the slide volume control to the right of the level meter. When you are finished listening to the sound,

- Click “OK” to leave the Tape deck.

You will now see the Button Actions screen again. You could choose more actions, but this is all we want this button to do.

- Click “Done” on the lower-right portion of the Button Actions screen.

🔗 | Two-Drive System

Note to two-drive system users: remove the /HS.Sounds disk and insert your data disk into the second disk drive.

*** | One-Drive System**

Note to one-drive system users: insert your data disk into the disk drive.

- Press *Apple-S* or choose *Save Stack* from the **File** menu. (It isn't necessary to use *Save As...* any longer, because you've already given your stack a name. You can now use *Save* at any time—this is a quick way to save your work in progress.)

CARD #3—USA Map

In your last card for this stack, we will add a map of the United States. You will also add the name of a state with your own voice!

- Choose *New Card* from the **Edit** menu.
- Choose *Load Background* from the **File** menu.

*** | One-Drive System**

Note to one-drive system users: insert the /HS.Art disk. A dialog box will appear, informing you that there is no device on-line. Click "OK".

🔗 | Two-Drive System

Note to two-drive system users: remove your data disk and insert the /HS.Art disk.

⇒ | Hard Drive or Network

Note to hard drive or network users: locate and open the HS.Art folder.

- Scroll down through the files by clicking on the "down" arrow. Click the U.S.A. file to select it.
- Click "Open".

Now we are going to add an invisible button to the state of Arizona. This button will play a sound that you are going to record, using HyperStudio's sound digitizer card and microphone.

Before we do that, though, you should save your stack.

🔗 | Two-Drive System

Note to two-drive system users: remove the /HS.Art disk and insert your data disk into the second disk drive.

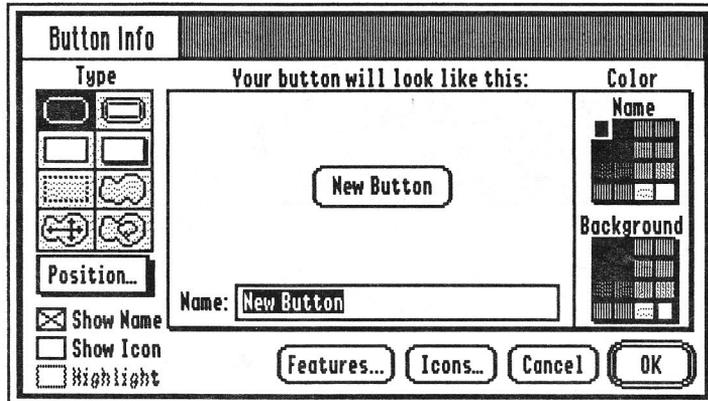
* One-Drive System

Note to one-drive system users: insert your data disk into the disk drive.

- Press *Apple-S* or choose *Save Stack* from the **File** menu.

Adding an Invisible Button

- Choose *Add a button* from the **Objects** menu. The “Button Info” dialog box will appear:



Because the button is invisible, we don't need to give it a name, so we'll go right to setting the type and position of the button.

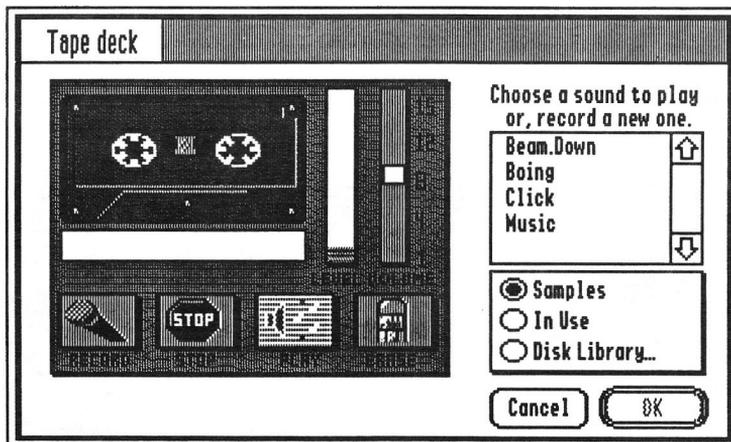
- You need to select what kind of button type you want from the options on the far right side of the dialog box. Click the expanding area button, which looks like a small four-headed arrow cursor inside of a circle. It's in the bottom row on the left side, directly above the letter “P” in the “Position...” button, and looks like this:



- Click “Position...” to set the position of the button.
- Move the cursor over the state of Arizona. Click inside of Arizona. (Arizona is immediately to the right of Southern California.) HyperStudio will make a button that expands to “fill” Arizona, then display a dialog box letting you know that the button shape has been defined.
- Now, click “OK” to accept the button and return to the Button Info screen, and then click “OK” a second time to continue to the Button Actions screen.
- Leave the “No connection” button checked on the lower-right corner of the Button Actions screen.

Adding Sound to a Button

- Click in the box next to “Play a sound” in the upper-right “Actions” section of the Button Actions screen. The HyperStudio Tape deck will appear:



the HyperStudio Tape deck

Click the “Record” button. You must hold the microphone very close to your mouth! Say firmly into the microphone, “This is Arizona.” As soon as you finish speaking, press any key to stop recording. It is important to press a key quickly to stop recording after you are finished speaking, since sound files take up large amounts of memory—about 5K per second—and you don’t want to waste memory by recording “silence” at the end of your phrase.

If you cannot easily hear your voice, adjust the volume slide control. If you still cannot hear your recording, click the “Record” button again and try holding the microphone closer to your mouth and speaking more firmly. You must project your voice—use a “radio announcer” voice. If there is extra “space” at the end of the recording, you should practice pressing a key more quickly after speaking so as not to waste memory in your stack.

- Now, you need to give your sound a name. Press the Clear key (to erase the suggested name of “Untitled”) and type “Arizona”. This name is used inside of HyperStudio to refer to the sound; it’s useful if you want to share the same sound with several buttons—you could make another button and tell HyperStudio that you wanted it to play the sound called “Arizona” that you recorded earlier.
- For now, however, you’re finished, so click “OK.”
- Click “Done” in the lower-right corner of the Button Actions screen.

Adding a Visible Button

- Choose *Add a button* from the **Objects** menu. The “Button Info” dialog box will appear.
- You need to select what kind of button type you want. Select the shadow button, the second one from the top on the right side of the button types.

It looks like this: 

- Next, the button needs a name. Press the “Clear” key and type **Menu**.
- Click “Position...” so that you can specify the button’s position.
- Move the button to the Gulf of Mexico (between the bottom of Texas and Florida) by placing the cursor inside the button, pressing the mouse button, and dragging the button.
- Click outside the button to place it, then click “OK” for the Button Actions dialog box.

Normally, you would choose “Previous card” as your destination. However, occasionally you will need to connect cards that are not next to each other. The following procedure is used to connect cards that are not immediately next to each other.

• Choose “Another card...” in the “Connect to:” section on the upper-left corner of the Button Actions screen. **Important: DO NOT click “OK” in the dialog box until you’ve read the following instructions!**

You will first see a dialog box instructing you to move to your destination. It is very important that you move to the card you want this button connected to before you click “OK.” Once you click “OK,” you will tell HyperStudio that this is the card you want your button connected to.

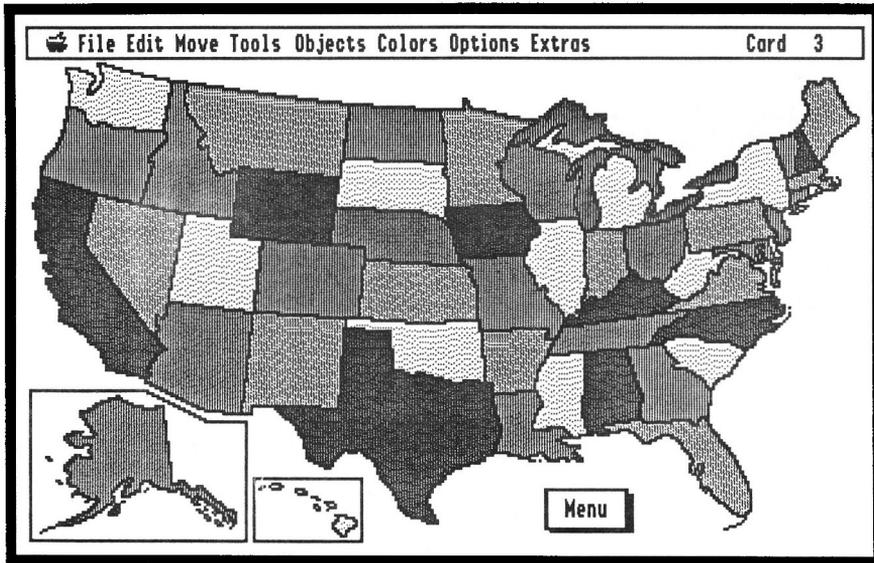
- Use the “Prev” button to move to the next card.

Continue to move, by clicking the “Prev” button, until you see the card that says “Tour of the United States” (with the compass icon). It will say “Card 2” in the upper-right corner.

• **When you reach the card that is your destination, click “OK” in the dialog box.**

- Choose “Fastest” as the visual effect on the “Select a visual effect” screen.
- Click “OK”.
- Click “Done” on the lower-right corner of the Button Actions screen.

Your card should look like this:



- Try out your button by clicking it. If it doesn't connect to the Menu card, you will need to edit the button (see below).

Editing or Deleting A Button

If you create a button that doesn't work the way you intended, you can change the way it works or simply delete it and make a new button.

IMPORTANT! If you do not want to change the button you just created, go on to "Saving Your Stack"!

- Choose the *Editing Tool* from the **Tools** menu. It is the pointer, and looks like an arrow.
- Click the button that says "Menu".

A dotted, moving box will enclose the button, showing that it is the active item.

At this point, to delete the button it's as simple as pressing the "Delete" key. The button will be removed from the card. To edit the button, double-click the button, and you'll be given the Button Info dialog box. At this point, you can change the button's appearance. If you want to change the button's actions, click the "Actions..." button.

Note that, frequently, it may be easier to simply delete a button and create the button again instead of editing it, particularly if the button is very simple and only performs one small action (such as moving to another card).

When you're done editing a button, choose the Browse tool (the hand) from the Tools menu, to return to the usual browse mode.

Saving Your Stack

★ | One-Drive System

Note to one-drive system users: insert your data disk into the disk drive.

⌘ | Two-Drive System

Note to two-drive system users: remove the /HS.Art disk and insert your data disk.

- Press *Apple-S* or choose *Save Stack* from the **File** menu.

That's all there is to it!

You have just made a three card stack that incorporates HyperStudio's eight basic skills. To review, they are:

- ❖ Making a card
- ❖ Loading a background
- ❖ Making a button
- ❖ Linking a button to a card or sound
- ❖ Importing clip art graphics
- ❖ Cutting and pasting objects
- ❖ Saving a stack
- ❖ Running a stack

With these eight basic skills, you can do almost anything in HyperStudio!

In the following pages, you can learn to use more of HyperStudio's features and also learn how to use other devices (such as a ComputerEyes video digitizer or Quickie hand scanner) with HyperStudio.

You don't need to go through the following chapters in order—for example, if you're interested in animation, you may wish to skip directly to that section.

Eventually, you can create a masterpiece that incorporates all of HyperStudio's major features!

CARD #4—The Help Card

In this activity, we will add a Help screen to the stack. It will be accessed by a button on the menu card. This Help screen will contain a border and a text item.

Adding a Border

- Choose *Background Color* from the **Options** menu. Set the color to white.
- Choose *New Card* from the **Edit** menu.
- Choose *Load Background* from the **File** menu.

★ | One-Drive System

Note to one-drive system users: insert the /HS.Art disk into the disk drive. Click “OK” in the dialog box that appears informing you that no volume is on-line.

☞ | Two-Drive System

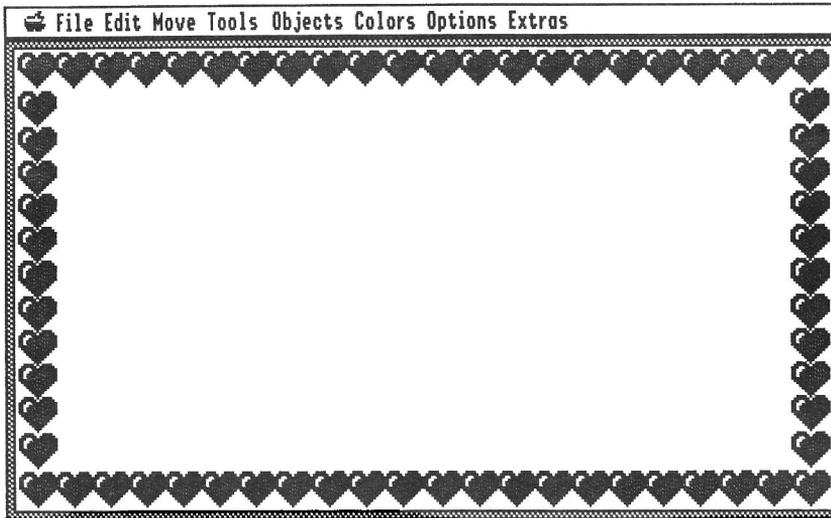
Note to two-drive system users: remove your data disk and insert the /HS.Art disk.

⇒ | Hard Drive or Network

Note to hard drive or network users: locate the /HS.Art disk.

- Click *Education.2* to highlight (and choose) the file. The *Education.2* file has the border we want to use; we’re going to erase the actual content of the screen (which is all education-related).
- Click “Open” to open the file.
- Choose the *Selector Tool* from the **Tools** menu.
- Enclose all of the clip-art within the heart border with the selector tool. To do so, click just inside the heart border. While pressing the mouse button, drag the mouse diagonally to just inside the lower-right corner of the border.

- Once you have successfully selected the clip-art, press the delete key. Your screen should look like this:



- If the interior of the border turns black, simply choose the *Fill Tool* and fill the area with white. (The area turned black because you happened to start the selection of the clip-art on a black pixel. If you select an area and then delete it, the selected area becomes whatever color on which your selector box begins.)
- If you needed to use the *Fill Tool*, move the mouse back up to the menu bar. Make sure that you've moved all the way to the top of the screen and that your cursor has changed to a pointer. Choose the hand from the *Tools* menu.

If you accidentally erase part of the heart border, press Apple-Z (to Undo your last action) and try it again.

If there are any stray marks on the screen that need to be erased, select and delete them, or use the *Eraser tool*.

That's all there is to it! You have just added a border to your screen. The next step will be to add some text to it.

Adding a Text Item

In this activity, we are going to add a text item to this help screen. Text items are very easy to use, and they function almost exactly like a word processor within HyperStudio. Since they can have a scroll bar, text items can also hold large amounts of information, without taking up more of the screen than you desire.

- Choose *Text style* from the **Options** menu.
 - Choose Shaston 16, Bold as your *text style*. If you don't have this font available, choose any font you'd like.
 - Click the black square below the words "Text Color" to set the text to black.
 - Click "OK" to accept these choices and exit the Text style dialog box.
 - Choose *Add a text item* from the **Objects** menu.
-
- An instructional dialog box may appear—if it does, read the dialog box and click "Continue". (This will only happen the first time you add a text item.)
 - Place the cursor inside the box. Do not click outside the box! If you do, HyperStudio will think that you are finished placing this text item.
 - While pressing the mouse button, drag the corner of the box to the upper-left corner of the screen within the heart border.
 - Place the tip of the arrow on the very edge of the lower-right corner of the box.
 - While pressing the mouse button, drag the corner of the box down and to the right so that it extends to the bottom of the screen (the top of the lower row of hearts) and about half-way across the screen. (Your box should take up about one-half of the screen.) If you accidentally click outside the box while doing this, simply click "Cancel" on the next screen and try again.
-
- Click outside the box when you have sized it correctly.

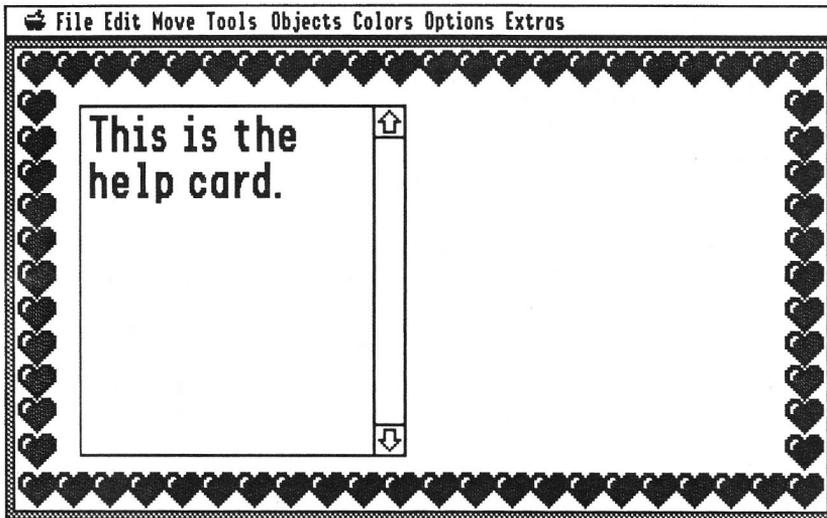
You will now see a dialog box that details information about your text item.

If you had some information already stored as a word processing file, you could click "Get file..." and import the text from a disk. You can directly import any AppleWorks Classic file or ASCII text file.

- Since you're going to enter the text now instead of getting it from a disk file, though, just click "OK".
- Click with the cursor inside the text box.
- Type **This is the help card**.

Now you may type in some information about how this stack works. You may also type what this stack is about. You may type as much information as you wish. Notice that as you reach the end of the box, HyperStudio automatically scrolls up another line. If you wish to see the beginning of your text, move the scroll bar on the right with your mouse. Intentionally make a typing error and continue typing. You may correct your error as easily as you can with a word processing program: simply move the cursor to where the error is and correct it! (Click the cursor after the mistake, press the Delete key as needed, and retype.)

Your screen should look like this:



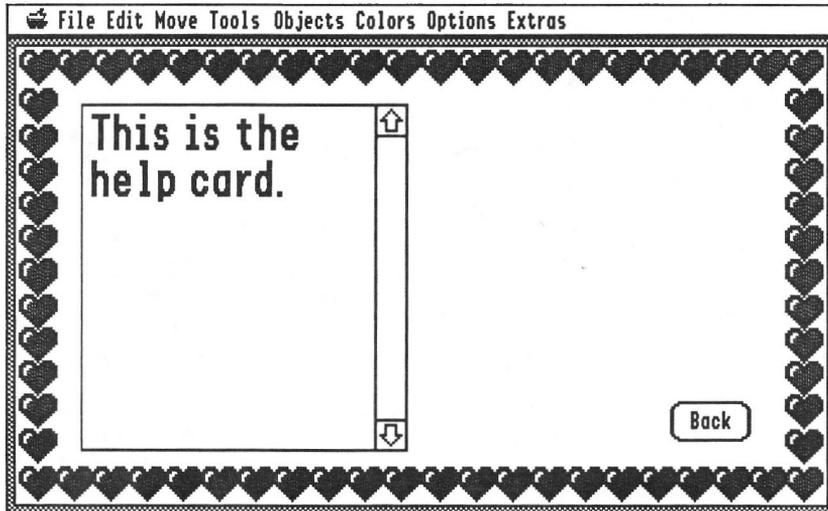
Adding a Button

- Choose *Add a button* from the **Objects** menu.
- Choose the rounded rectangle button style by clicking its icon.

It looks like this: 

- Press the Clear key and type **Back**.
- Click "Position..."
- Place the cursor inside the button, press the mouse button, and drag the button to the lower-right corner of the screen (instead of the heart border).
- Click outside the button to place it, then click "OK" to continue in the button creation process by leaving the proceed to the Button Actions screen.
- Choose *Connect to: another card* on the Button Actions screen.
- Use the **Move** menu to move to card #2, the menu card.
- Click "OK" when you see the menu card.
- Choose any visual effect you'd like from the "Select a visual effect" dialog box.
- Click "Done" on the lower-right portion of the Button Actions screen since we don't want this button to do anything else.

Your screen should look like this:



Two-Drive System

Note to two-drive system users: remove the /HS.Art disk and insert your data disk.



One-Drive System

Note to one-drive system users: insert your data disk into the disk drive.

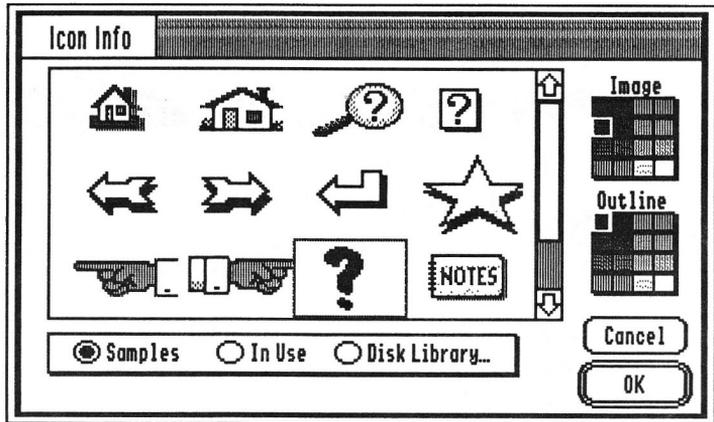
- Choose *Save stack* or press *Apple-S*.

Connecting to the Menu Card

Now we will connect this screen to the Menu Card. The button will be an invisible button with a “question mark” icon.

- Select *Previous card* (from the *Move* menu) twice to go to the Menu card (card #2).
- Choose *Add a button* from the *Objects* menu.
- Choose the rectangular invisible button style. It looks like this: 
- Now, to add an icon, click the button marked “Icons...”, in the bottom of the “Button Info” screen that you’re looking at.

- The “Icon Info” screen will appear, as shown to the right.
- Click the question mark icon, then click the red color below the word “Image.”
- Click “OK.”
- You will be returned to the “Button Actions” menu. Click “Position...” to accept your choice and specify a position for the button.



• As usual, you will be returned to your card to position your button. Note that, though the question mark icon isn't shown, the button's size is pre-set to fit the icon. You don't want to change the size of the button, you just want to position it on the screen.

• To position the button on the screen, drag inside the rectangle until the button is in the lower-right corner of the screen.

• Click outside of the button rectangle to place it. You will be returned to the “Button Actions” screen.

• Choose “Another card” (under the “Connections” heading) on the Button Actions screen.

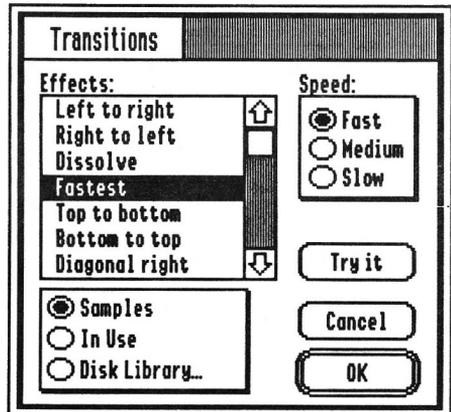
• Connect it to the Help card by clicking “Next” in the connection box until you see the Help card (the card with the text item on it).

• Click “OK” when you see the Help card. The “Transitions” dialog box, shown to the right, will appear.

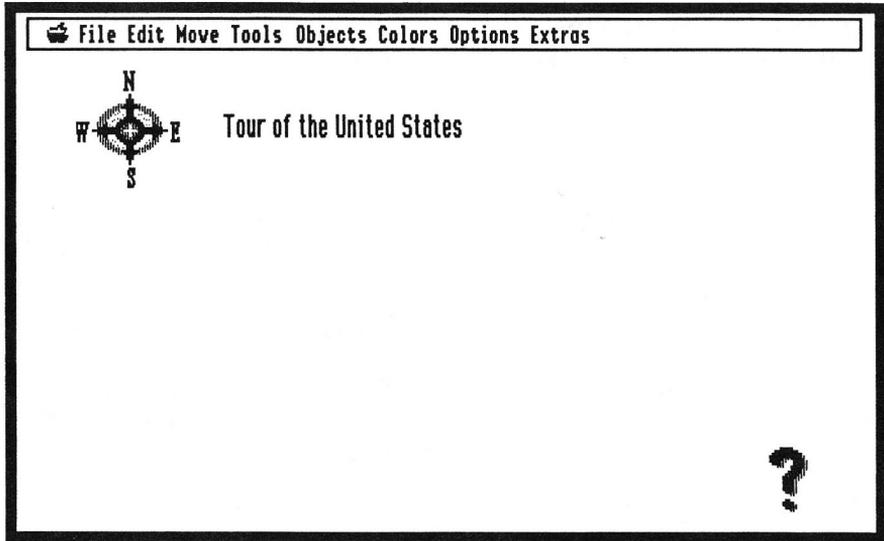
• Choose the “Rain” screen transition effect.

• Click “OK” on the screen transition page.

• Click “Done” on the lower-right section of the Button Actions screen.



Your screen should look like this:



- Insert your data disk and choose *Save stack* or press *Apple-S*.

That's all there is to it! Try it out by clicking on the question mark.

|| Use this picture for scanning when you are using the
|| LightningScan or Quickie sections.



Digitizing Images with LightningScan

In this activity, you will scan an image with a LightningScan hand-held scanner. By rolling the scanner across a surface, the image is translated to the computer as a series of dots. This process is called digitizing. If you are using the Quickie scanner, please use the instructions in the next section.

Setting the Scanner

Before we begin, you must run the LightningScan software. To do so, boot the LightningScan software disk.

The first step is to set the switches to their proper settings. On the left-hand side of the scanner is the **dithering switch**. It is next to the Start button.

Move the **dithering switch** to the “*Letter*” setting. It is the position to the far right. This setting is used for scanning line art. The other settings are used for scanning photographs.

Set the **resolution switch** to the *200 DPI* setting. The resolution switch determines the size and amount of detail in the image. This switch is on the back of the scanner, and has markings from 1-4 on it. *Set the switch to 2.*

The **brightness dial** is on the back of the scanner next to the resolution switch. It is a dial marked “light” and “dark”. In the middle of the dial is a vertical, black bar. *Place this bar in the middle.* The brightness dial determines the brightness of the scanned image. It works just like the darkness control on a copy machine works.

In summary, the switches should have the following settings:

- **Dithering Switch:** “Letter” setting
- **Resolution Switch:** 2
- **Brightness Dial:** halfway point

If you are the first person using the scanner, you will need to choose *Set Slot* from the Scanner menu. The scanner will be in slot two.

Now we are ready to scan the image. We will add a clip-art image of a globe. Later, you will import this into the title screen of your stack.

- Pull down on the Scanner menu. Click *New Scan*.
- Click “OK” to accept the default setting of “Display during scan?” and “16 Greys”.
- Place the picture of the globe on a flat surface. It is very important that the image you’ll be scanning is on a completely flat surface! Otherwise you will have distortion in the image.
- Position the scanner (on the paper) about one inch above the globe.

- Press the “*Start*” button that is located on the left side of the scanner. Slowly and steadily, drag the scanner across the image, from the top to the bottom. If you move the scanner too quickly, the computer will make a low-pitched clicking sound to warn you that part of the image may have been lost.

As you are scanning, you will see the picture you are scanning on the computer screen. Seeing the picture is useful since it allows you to see if you’re scanning straight.

- When you reach the end of the image, press *Return* on the keyboard.
- Once the image shows up on the screen, use the scroll bars to see the entire image. If, for some reason, you did not get a clear copy of the image, go back and scan the image again.
- Pull down on the **File** menu and select “*Save As*”.
- Type `Globe` as the name of the picture.
- Click next to the **Apple Preferred** button and next to `640`.
- Insert your data disk.
- Click the “**Disk**” button until the name of your data disk, “`Data`”, appears as the destination.

⇒ **Hard Drive or Network**

Note to hard drive or network users: be sure your data disk is selected. Press the **TAB** button, if necessary, until the name of your data disk appears in the top of the dialog box.

- Click **Save**.

Importing Scanned Images into HyperStudio

Now we will add the image you just scanned to your HyperStudio stack.

- Launch the HyperStudio program again.
- Choose *Open Stack* from the **File** menu.
- Place your data disk in the drive.
- Click the **USA** stack.

Next, we will add the image to the title screen, the first card you made.

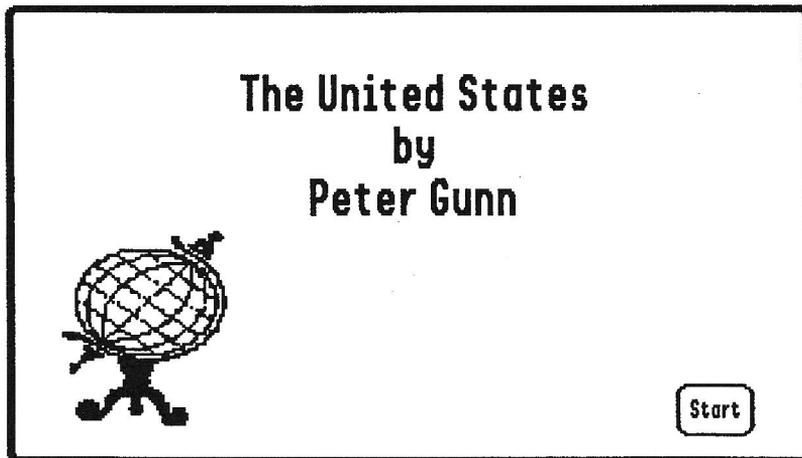
- Choose “*Add clip art*” from the **File** menu.
- Choose `Globe` as the file to add and click “**Open**”.
- Draw a box around the globe by clicking just outside the upper-left section of the globe. Drag the mouse (holding the button down) to the lower-right section of the globe. When there is a box completely around the globe, click “**OK**” at the top of

the screen. (If part of the globe is not inside the box, click outside of the box and do this step again.)

After the above step, you'll be returned to your title screen with the globe in the middle of the screen.

- Place the cursor in the middle of the box.
- Press down on the mouse and drag the box to the lower-left corner of the screen.
- Click outside of the box to place the graphic.
- Choose the *Fill Tool* from the **Tools** menu.
- Choose *yellow* from the **Colors** menu.
- Click the pointed tip of the *Fill Tool* in the white area surrounding the globe.

Here is an example of how your screen should look:



That's it! You have now scanned an image and imported it into your stack!

- Insert your data disk into the disk drive.
- Press *Apple-S* or choose *Save Stack* from the **File** menu.

Note that we saved all of our pictures in this exercise in 640-mode. The LightningScan software has an option to save as 320-mode, which frequently produces better looking images. HyperStudio can use 320-mode clip art, but it will cause the entire card to become a 320-mode card. The result is that text in buttons and text items will become fuzzy. One solution to this problem is to restrict your use of 320-mode graphics to cards which only contain invisible buttons. This way, a user can look at a 320-mode graphic and you may branch to different locations in a stack based on what invisible buttons are clicked.

Digitizing Images with the Quickie Scanner

In this activity, you will scan an image with a Quickie hand-held scanner. By rolling the scanner across a surface, the image is translated to the computer as a series of dots. This process is called digitizing.

Setting the Scanner

Before we begin, you must run the Quickie software. To do so, boot the Quickie software disk if it isn't already booted.

- Click the Quickie button on the title screen.
- Pull down on the Display menu and make sure that the last item says "Switch to 320 mode". If it doesn't, choose it by highlighting it with the mouse.

The next step is to set the switches to their proper settings. On the left-hand side of the scanner is the dithering switch. It is next to the Start button.

Move the dithering switch to the "Letter" setting. It is the position to the far right. This setting is used for scanning line art. The other settings are used for scanning photographs.

Set the resolution switch to the 200 DPI setting. The resolution switch determines the size and amount of detail in the image. This switch is on the right side of the scanner, and has markings from 1-4 on it. *Set the switch to 2.*

The brightness dial is on the back of the scanner next to the resolution switch. It is a dial marked "light" and "dark". In the middle of the dial is a vertical, black bar. *Place this bar under the "T" of the word "Light"*. The brightness dial determines the brightness of the scanned image. It works just like the darkness control on a copy machine works.

In summary, the switches should have the following settings:

- **Dithering Switch:** "Letter" setting
- **Resolution Switch:** 2
- **Brightness Dial:** underneath the "T" in "Light"

Now we are ready to scan the image. We will add a clip-art image of a globe. Later, you will import this onto the title screen of your stack.

- Pull down on the Quickie menu. Choose "Grey Smoothing."

The next screen shows the Smoothing dialog box. You have three color options for your scan mode. For line art, black and white should be chosen. The other options would be appropriate if you were scanning photographs and other images that contain grey shading.

- Choose *pseudo-gray*.
 - Click “OK” to accept your choice.
 - Pull down on the **Quickie** menu. Make sure that “portrait” and “blend lines” have a check mark next to them. If they don’t, click them.
 - Place the picture of the globe on a flat surface. This is very important! If you aren’t scanning a flat image, you will have a distorted picture.
 - Position the scanner (on the paper) about one inch above the globe.
 - Choose *Scan Image* from the **Quickie** menu.
- A dialog box will appear that says, “Ready to Scan”.
- Click “OK” to continue.
 - Press the “*Start*” button that is located on the left side of the scanner. Slowly and steadily, drag the scanner across the image, from the top to the bottom.

As you are scanning, on the computer screen you will see the picture you are scanning. Seeing the picture is useful since it allows you to see if you’re scanning straight.

- When you reach the end of the image, press *return* on the keyboard.
- Once the image shows up on the screen, use the scroll bars to see the entire image. If, for some reason, you did not get a clear copy of the image, go back and scan the image again.
- Pull down on the **File** menu and select “*Save As*”.
- Type **Globe** as the name of the picture.
- Click next to the **Apple Preferred** button.
- Insert your data disk.
- Click the “**Disk**” button until the name of your data disk, “**Data**”, appears as the destination.

⇒ **Hard Drive or Network**

Note to hard drive or network users: be sure your data disk is selected. Press the **TAB** button, if necessary, until the name of your data disk appears in the top of the dialog box.

- Click **Save**.

Importing Scanned Images into HyperStudio

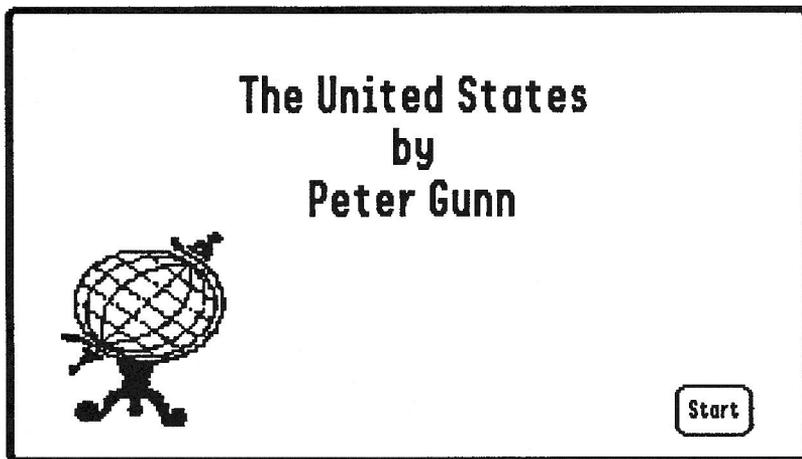
Now we will add the image you just scanned to your HyperStudio stack.

- Launch the HyperStudio program again.
- Choose *Open Stack* from the **File** menu.
- Place your data disk in the drive.
- Click on the **USA** stack.

Next, we will add the image to the title screen, the first card you made.

- Choose “*Add clip art*” from the **File** menu.
- Choose **Globe** as the file to add and click “**Open**”.
- Select the globe by clicking just past the upper-left portion of the globe. While pressing the mouse button, drag the mouse to just past the lower-right portion of the globe. A dotted box should enclose all of the globe. If it doesn’t, click outside the box and try again.
- Click “**OK**”.
- Place the cursor in the middle of the dotted box.
- Press down on the mouse and drag the box to the lower-left corner of the screen.
- Click outside of the box to place the graphic.
- Choose *Standard Palette* from the **Options** menu to restore the palette to its original colors.
- Choose the *Fill Tool* from the **Tools** menu.
- Choose *yellow* from the **Colors** menu.
- Click the pointed tip of the *Fill Tool* in the white area surrounding the globe.

Your screen should look like this:



That’s it! You have now scanned an image and imported it into your stack!

- Insert your data disk into the disk drive.
- Press *Apple-S* or choose *Save Stack*.

Note that we saved all of our pictures in this exercise in 640-mode. The Quickie software has an option to save as 320-mode, which frequently produces better looking images. HyperStudio can use 320-mode clip art, but it will cause the entire card to become a 320-mode card. The result is that text in buttons and text items will become fuzzy. One solution to this problem is to restrict your use of 320-mode graphics to cards which only contain invisible buttons. This way, a user can look at a 320-mode graphic and you may branch to different locations in a stack based on what invisible buttons are clicked.

Digitizing an Image from a Video Camera with ComputerEyes

In this activity, you will videotape your partner. You will then add the digitized picture of him/her to a card in your stack.

Setting Up ComputerEyes

- Boot the ComputerEyes software. If a dialog box appears informing you that the ComputerEyes card isn't in Slot 1, check the connections in your machine and be sure that your Control Panel has slot 1 set to "Your card."
- Pull down on the Image menu and choose **Capture Settings** to set up ComputerEyes for best results with HyperStudio. Make sure the following settings have been chosen. Change the settings to match these, if necessary:
 - Dithering should be "on"
 - Capture speed should be "normal"
 - Palette should be "unfrozen"
 - Auto-calibrate should be "on"
 - Color/levels = 16
 - Color separation = 2
- Choose "OK" after verifying these settings.
- Choose *View Video Input* from the **Image** menu to monitor the image coming from the video camera or laserdisc player.
- Choose *Capture B/W* from the **Image** menu to begin the digitizing process. This will take approximately six seconds.

If you're capturing a live subject, he/she should hold as still as possible during the capture process. You'll get best results with a solid color background, preferably with good contrast to your subject. Make sure you have good lighting, also. Position the camera for a video snapshot of your partner's face, with his/her face filling about 1/4 of the total screen area.

- When the capture is finished, a new window will appear with a 320-mode graphic version of the image on the screen.
- To view the picture in the 640-mode used by HyperStudio, choose *View Special* from the **Image** menu. A dialog box will appear allowing you to choose in which mode the image should be displayed.
- Choose "Super Hi-Res 640 B&W," and "Paint from raw data". This latter choice tells ComputerEyes to build the 640 image from the raw data gathered from the video image. If you were loading a 320 mode graphic scanned earlier where only the final graphic was available, this "raw data" option would not be offered.
- Click "OK" to see the image.
- Click the mouse when you're done viewing the image.

- To save the 640 image to disk, choose *Save Special* from the **File** menu. As before, choose “Super Hi-Res 640 B&W,” and “Paint from raw data”.
- Insert your data disk into the disk drive.
- Click the “Disk” button until the name of your data disk, “Data”, appears as the destination.
- Enter the name you wish to use for this image (you can use `Test . 640` for this example if you wish). Be sure the name you use contains only letters, numbers, and/or periods, and doesn’t have any spaces!
- Click “Save”.

Using A ComputerEyes Image in HyperStudio

Now, let’s use the captured image within HyperStudio.

- Run the HyperStudio program again.
- From the **File** menu, choose *Open Stack*. Insert your data disk and load the USA stack.
- Choose *New Card* from the **Edit** menu.

There are two ways of adding digitized images. One is as clip-art, where the image is “painted” onto the background. The other is as a graphic object, where your image will remain separate from the background. We’ll add it as clip-art for this activity.

- Choose “*Add Clip-Art*” from the **File** menu.
- Insert your data disk into the disk drive.
- Load the file with the video picture (`Test . 640` if you used the earlier suggestion), by clicking on the name, and then “Open”.

A window will appear with the digitized image in it.

- Use the mouse, with the *Selector Tool*, to cut out the part of the image that you’d like to use in your card (about 1/2 of the screen at most).
- Click “OK” at the top of the window.

The window will disappear, and you’ll see the graphic you selected floating on the card.

- Put the mouse on the image, press the mouse button, and drag the image to the upper-right corner of the screen.
- Click outside the box to drop the image in position. That’s it! The image is now part of the background graphic
- Insert your data disk and press *Apple-S* or choose *Save Stack* from the **File** menu.

Adding a Text Item

This new card the we are creating is going to be an “About the Author” card that is accessed from the menu card (#2). To the left of the picture you digitized, you are going to add a text item with some autobiographical information in it.

- Choose *Text style...* from the **Options** menu.
- Choose Century 18, Bold as your *text style*.
- Click the black square under the words “Text Color” to set the color, then click “OK” to leave the Text Style dialog.

- Choose *Add a text item* from the **Objects** menu.
- Read the dialog box and click “OK”.

You will see a box that is very similar to the *Selector Tool's* box. It is sized and moved in the same way, too.

- Place the cursor inside the box. Do not click outside the box! If you do, HyperStudio will think that you are finished placing this text item!
- While pressing down on the mouse button, drag the box to the upper-left corner of the screen.
- Place the tip of the arrow on the very edge of the lower-right corner of the box.
- While pressing down on the mouse button, drag the box down so that it extends to the bottom of the screen. If you accidentally click outside the box while doing this, simply click “Cancel” on the next screen and try again.

- Click outside the box when you have sized it correctly.

You will now see a dialog box that details the attributes your text item may have.

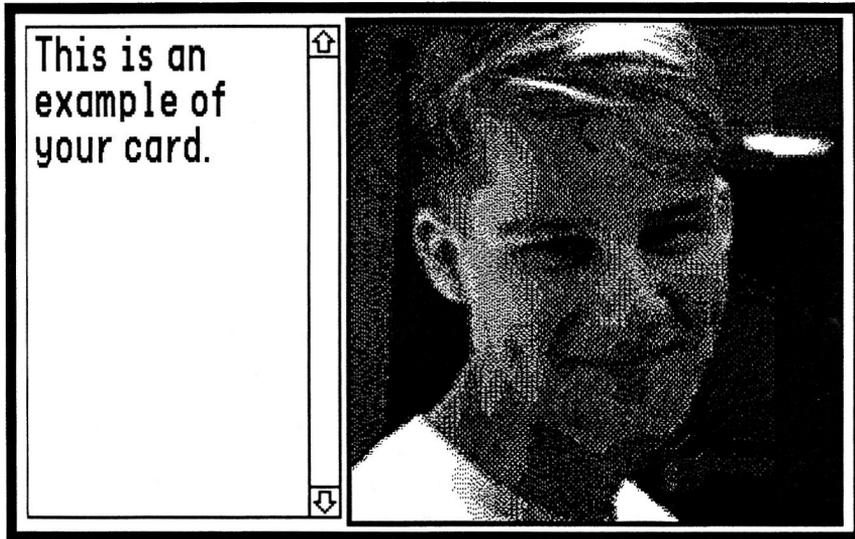
- Click next to “Draw scroll bar”.
- Click the “Add new text” button.

Now you may type in some information about the person whose picture is on this card.

- Click inside the text box to set the insertion point for new text.

Type a little bit about yourself. You may type as much information as you wish. Notice that as you reach the end of the box, HyperStudio automatically scrolls up another line. If you wish to see the beginning of your text, move the scroll bar on the right with your mouse.

Your screen should look like this:



- Insert your data disk into the disk drive.
- Press *Apple-S* or choose *Save Stack* from the **File** menu.

Button Connections

Now we need to add a button to this card so that you can return from this card to the menu. We will also add a button to the menu card that accesses this card.

- Choose *Add a button* from the **Objects** menu. Choose the shaded rectangular button. Type **Back** and click “Position...”
- Place the cursor inside the button, click the mouse, and drag the button to the lower-right portion of the screen.
- Click outside the button to place it.
- Click “Connect to another card.”
- Select *Jump to Card...* (in the **Move** menu) and type 2, to move to Card #2.
- Click “OK”.
- Choose “Venetian Blinds” as the visual effect.
- Click “OK”.
- Click “Done”.
- Insert your data disk and press *Apple-S* or choose *Save Stack* from the **File** menu.
- Click the button you’ve just created to go to the menu card.

Now we will add a button to the menu card that accesses the card with the ComputerEyes digitized picture that you just made. This button will be a graphic item.

- Make sure that you are at the menu card.
- Choose *Add clip art* from the **File** menu.

* **One-Drive System**

Note to one-drive system users: insert the /HS.Art disk into the disk drive.

☺ **Two-Drive System**

Note to two-drive system users: remove your data disk and insert the /HS.Art disk into the disk drive.

⇒ **Hard Drive or Network**

Note to hard drive or network users: locate the /HS.Art disk.

- Load the *Education.1* file.
- Select the picture of the teacher in front of the black board.
- Position the picture underneath the compass.
- Choose the *Text* tool from the **Tools** menu.

* **One-Drive System**

Note to one-drive system users: you may be prompted for the /HyperStudio disk at this step or on the next step. If so, insert it as directed and click "OK".

- Choose *text style* from the **Options** menu.
- Choose Geneva 12, Bold as the font and style.
- Type **About the author**
- Use the *Selector Tool*, if necessary, to align the text.
- Choose *add a button* from the **Objects** menu.
- Click the invisible rectangular button style.
- Click "Position..."
- Size the box around the teacher icon.
- Click outside the box.
- Click "Connect to another card".
- Use the *Move to next card* from the **Move** menu to move to the ComputerEyes digitized picture card.
- Click "OK" when the proper card is displayed.
- Choose "Right to left" as the transition.
- Click "OK" and then "Done".
- Insert your data disk into the drive and press *Apple-S* or choose *Save Stack* from the **File** menu.

That's it! You have successfully digitized a video image with ComputerEyes, imported it as clip art into HyperStudio, and added a Text Item!

Typical uses for digitized pictures would include putting students' pictures in their reports, a stack on local history with illustrations captured from a videotape, or an

“open house” school map stack with pictures of students and staff on various cards.

Note that we saved all of our pictures in this exercise in 640-mode. The ComputerEyes software has an option to save as 320-mode, which frequently produces better looking images. HyperStudio can use 320-mode clip art, but it will cause the entire card to become a 320-mode card. The result is that text in buttons and text items will become fuzzy. (You can also use the *Load Background* option, in the File menu, to load an entire 320-mode card background.) One solution to the problem of fuzzy text is to restrict your use of 320-mode graphics to cards which only contain invisible buttons. This way, a user can look at a 320-mode graphic and you may branch to different locations in a stack based on what invisible buttons are clicked.

Creating an Animation

There are two types of animation in HyperStudio: the “classic” form of animation, accessed through the “Activate animation...” option in the Button Actions dialog box, and the “new” type of path animation, accessed through the Animator NBA (New Button Action).

The “classic” type of animation displays several frames repeatedly to give the illusion of motion, while the “new” type of animation allows you to specify a path for a still object to follow on-screen.

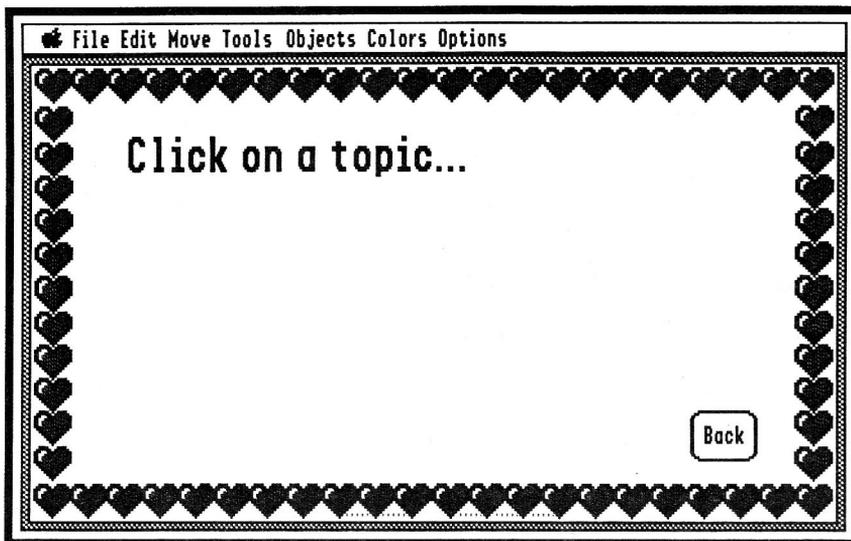
In this activity, you will make a new help screen with a blinking light bulb, using the “classic” form of animation. (Information on the “new” type of animation can be found in appendix D of the HyperStudio Reference manual.) Instead of making the light bulb appear to move on the screen, we’re going to have it stay in one place and blink on and off.

Since your old help screen has a text item on it, and we want to place the animation where the text item is, we will first need to delete the text item.

- Go to the help screen.
- Choose the *Pointer* from the Tools menu. (This allows you to edit all objects.)
- Click inside the text item.
- Press the Delete key.

The Text

- Choose the *Text Tool*.
- Choose red from the **Colors** menu.
- Choose *Text Style* from the **Options** menu.
- Choose Shaston 16 Bold as your *text style*, then click “OK” to exit the Text style dialog box.
- Position the cursor in the top one-third of the screen on the left side.
- Type **Click a topic...**
- If necessary, position the text with the *Selector Tool* so that your card looks like the picture on the next page.



The Animation—Creating the “Movie”

Creating the frames for the animation, and then accessing it with a button are two completely independent actions. First, you will use the current card as a temporary screen to make the frames of the “movie”—that is, exactly what will be played back when the user clicked the button. After that, you will restore the original card image and then create a button to play the animation.

Our animation is going to be of a light bulb blinking on and off.

- Choose *Add Clip Art* from the **File** menu.

★ | One-Drive System

Note to one-drive system users: insert the /HS.Art disk into the disk drive. Click “OK” on the “no volume on-line” dialog box message.

✎ | Two-Drive System

Note to two-drive system users: insert the /HS.Art disk into the second disk drive.

⇒ | Hard Drive or Network

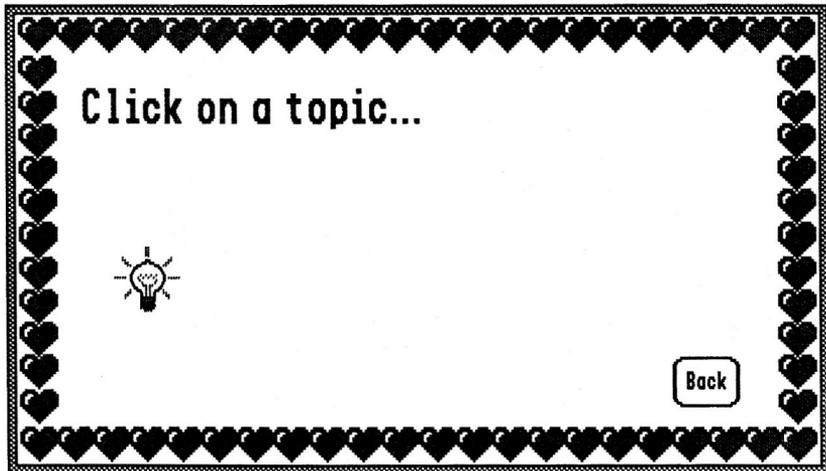
Note to hard drive or network users: locate the /HS.Art disk.

- Click the **I**cons file to select it as the file you want to view.
- Click “Open”. The **I**cons screen will be displayed.

- Select the light bulb by positioning the cursor just past the upper-left portion of the light bulb, pressing the mouse button, and dragging the mouse diagonally down and to the right, until there is a box around the light bulb.
- Click “OK”. You will be returned to your card.
- Position the light bulb on the left-hand, lower section of the screen by positioning the cursor inside the box, pressing the mouse button, and dragging the light bulb.

If there are any stray marks on the screen, you can erase them with the *Eraser Tool*.

Your card should look like this:



★ | One-Drive System

Note to one-drive system users: insert your data disk into the disk drive.

☞ | Two-Drive System

Note to two-drive system users: remove the /HS.Art disk and insert your data disk into the disk drive.

⇨ | Hard Drive or Network

Note to hard drive or network users: in the next step, **be sure** to choose your data disk as the destination! Press the TAB key until the name of your data disk appears in the top of the dialog box.

This is the new background for this card, and also the first frame for our movie.

- Choose *Save Screen* from the **File** menu. Name the file `Bulb1`
Your file must be named exactly like this, with no spaces!

Now you've saved the first frame of your animation: the light bulb on the screen. The second frame of our animation will be with the light bulb "out". We'll "turn the light bulb off" by erasing the "beams of light" around the outside.

- After the screen has been saved, use the *Eraser Tool* to erase the "beams" coming out of the light bulb. Be careful that you don't erase the light bulb itself! Remember, you can select *Undo* (from the **Edit** menu) to undo your most recent action, so if you accidentally erase part of the light bulb, select *Undo* immediately.
- After you have all of the "beams" erased, choose *Save Screen* from the **File** menu and save it as `Bulb2`

You have now created a small, two frame "movie" on the disk. Our next step is to restore the card background.

- Choose *Load Background* from the **File** menu to restore the original screen.
- Click `Bulb1` to select it.
- Click "Open".

Adding a Button to Activate the Animation

At this point, the separate frames of our "movie" is on the disk and the only task remaining is to create a button that triggers the animation.

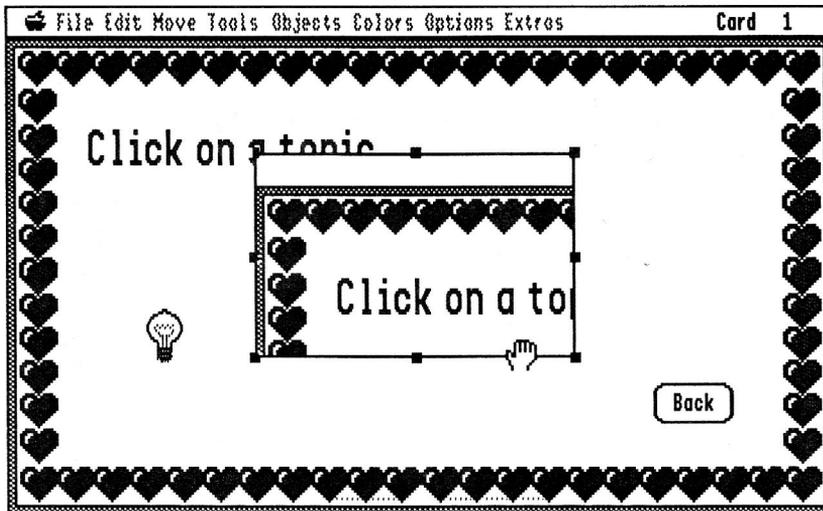
- Choose *Add a button* from the **Objects** menu.
- Choose the rounded style by clicking in it.
- Press the Clear key and type `See the animation!`
- Place the cursor on the button and drag it to lower-left side of the screen.
- Click outside the button to place it.
- Click next to "Play Animation..." on the Button Actions screen.
- Choose `Bulb1` as the animation to attach to this button.

What you need to do now is to create a "window" that will show the moving light bulb sequence you just created. You will place this "movie" of the moving light bulb over the original light bulb image that is already on the screen.

A rectangle will appear in the middle of the card. This rectangle is the window in which your animation will appear. You will see the upper-left portion of the screen in the window.

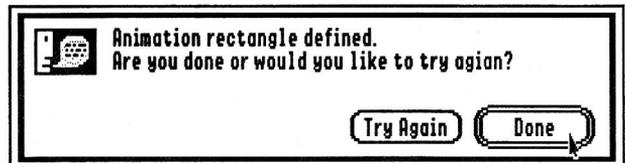
(The original light bulb is still "behind" this rectangle; the rectangle simply displays the first frame of your movie and allows you to set it's position.)

Your card will look like this:



- Place the hand inside the rectangle and press on the mouse button.
- Slowly move the mouse upwards and to the left until the light bulb appears in the rectangle.

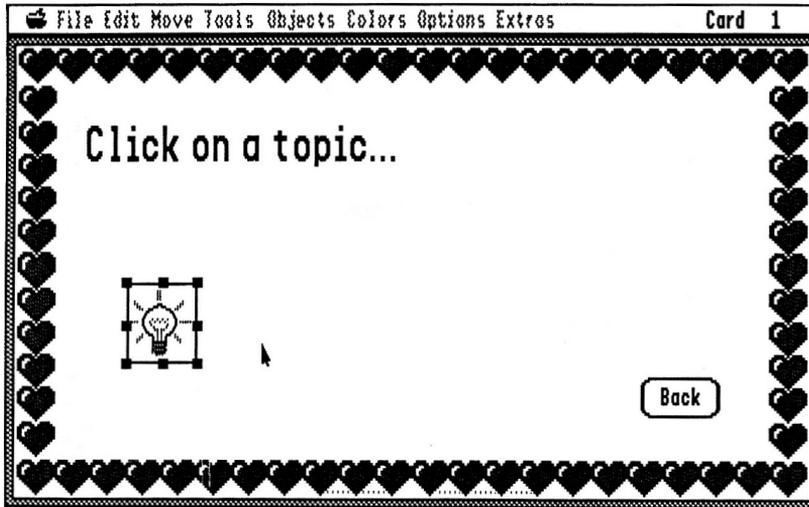
If you accidentally click outside of the rectangle, HyperStudio will think you are finished and display the dialog box to the right. If you're not finished, clicking the "Try Again" button will let you continue.



Now we will move the animation window to the bottom of the screen so that your "movie" will appear on the bottom of the screen.

- Move the hand onto any part of the frame. When the hand is on the edge of the frame, the cursor will show as a cross-hair. The cross-hair indicates that dragging with the mouse will allow you to move and size the frame.
- Making sure that the cursor shows as a cross-hair, press the mouse button and drag the window so that it is over the original light bulb image.
- Place the cross-hair on one of the "handles" (small dark squares) and drag to make the rectangle about two inches high vertically and as wide as the clear area of the screen. Make sure that your rectangle doesn't overlap any part of the buttons or border.

Your screen should look like this:



- As you change the size of the rectangle, the light bulb may occasionally move out of view. If it does, drag the interior of the window with the hand until the light bulb reappears.
- Align the light bulb in your window so that it is on top of the light bulb that is already on the screen. If you don't align the two graphics, when your animation "movie" starts, the first frame will "jump" or flicker, since the moving animation is in a different place than the stationary light bulb.

Aligning the Graphic

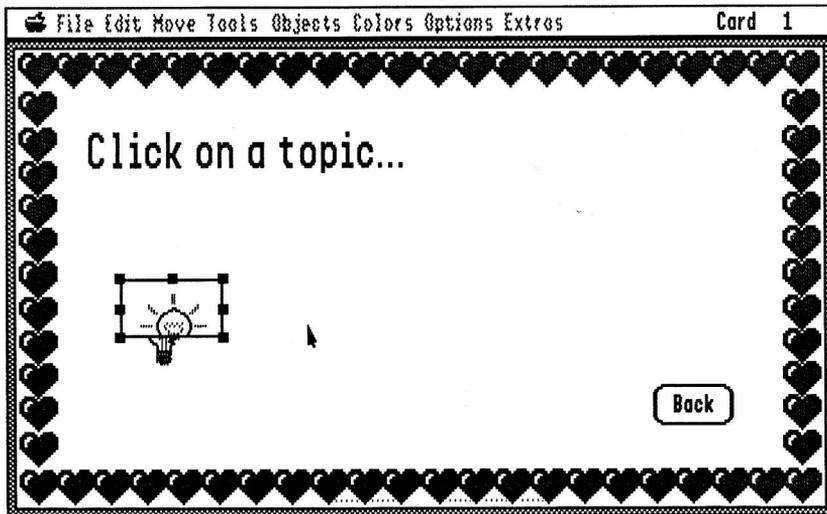
Align the two light bulbs by moving the light bulb with the hand cursor, or adjusting the frame, as needed. You can tell if the two images are aligned by clicking with the hand cursor on the rectangle's frame. If the light bulb jumps or otherwise moves at all, they are not aligned.

(Clicking the edge of the rectangle briefly displays the image underneath the animation rectangle. You want both light bulbs to be right on top of each other.)

There are two techniques for aligning the graphics. Use the technique you like best.

- Move the light bulb in the window in the same direction that the image jumps when you click the mouse. Continue moving the light bulb until it no longer flickers when you click the mouse.

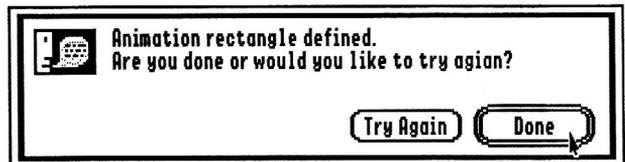
• Use the mouse to pull up the bottom of the window until you can see top half of the light bulb in the window, and the bottom half of the other light bulb on the screen. The screen will look like this:



(Then, adjust the animation window until the two images line up perfectly. Pull down the bottom of the window until it covers the screen image of the light bulb.)

Once you have the light bulb in the animation rectangle aligned with the light bulb on the card,

• click outside the rectangle. The dialog box shown to the right will appear, verifying that you have set up the animation as you want it.



• Click “Done.” HyperStudio will build your animation from the frames on the disk and then return to the Button Actions dialog box.

- Click in the Rate box and delete the default setting by pressing the Clear key.
- Enter a rate of 25. (This is how fast your animation will move across the screen. A rate of 5 is fast, and 45 is slow.)
- Click in the Repeat Count box and delete the default setting.
- Enter a repeat count of 6. (This sets how many times your animation will repeat.)
- Click “Done.”
- Insert your data disk and choose *Save stack* or press *Apple-S*.

That’s it! Click the button to see your animation!

In summary, the animation you created exists as clip-art files on the disk. When activated by a button, only the “mini-movie” you created is embedded in the stack.

Animations are not played from the disk or made by “flipping” between cards; HyperStudio attaches the animation sequence to the button. In fact, you can copy and paste the button and the animation will follow!

For more information on the Animator NBA (which allows you specify a path that an object will follow), see appendix D of the HyperStudio Reference manual.

Using SimpleScript

SimpleScript is the name of the programming language that comes with HyperStudio. It is designed to be extremely easy to use, but not at the expense of power—the language allows you to manipulate graphic and text items in HyperStudio, in addition to offering variables, structures (such as For–Next loops) and complete mathematical functions. (If you’ve ever written a program in AppleSoft BASIC or HyperTalk, you’ll notice similarities to each in SimpleScript. However, SimpleScript is not exactly the same as either of those.)

To introduce you to SimpleScript, this short tutorial will take you through the steps of creating three short programs. Along the way, you’ll learn how to use the SimpleScript editor and how to work with HyperStudio objects from within SimpleScript.

This section assumes that you’ve been through the previous parts of this tutorial and that you’re comfortable with the material covered above. You should be familiar with the basic operations of HyperStudio (such as loading and saving a stack, adding clip-art, and navigating within a stack). In particular, you need to be sure you understand how to create a button and a graphic object, and how to use the HyperStudio Tape deck to record sounds.

Your First Program

Let’s dive right in to SimpleScript by writing a program. This first program will simply move to a specified location on the screen, set a font and text color, and then draw your name on the screen.

- Start up HyperStudio 3.0, then pick *New Stack* from the **File** menu. We want to start with a clean white background.
- Select *Add a Button...* from the **Objects** menu, and make a visible button in the lower right-hand corner of the screen, about two inches up from the bottom and two inches from the right. The button title should be “Who?”.
- When you get to the button actions screen, click “Scripting language...” (If HyperStudio has more than one programming language for you to choose from, it will present you with a list—in that case, click “SimpleScript” and then “OK”.)
- Now you are in the SimpleScript editor. Here, you can build an entire program without ever touching the keyboard: by simply selecting the SimpleScript commands from the menus, your program is written for you.

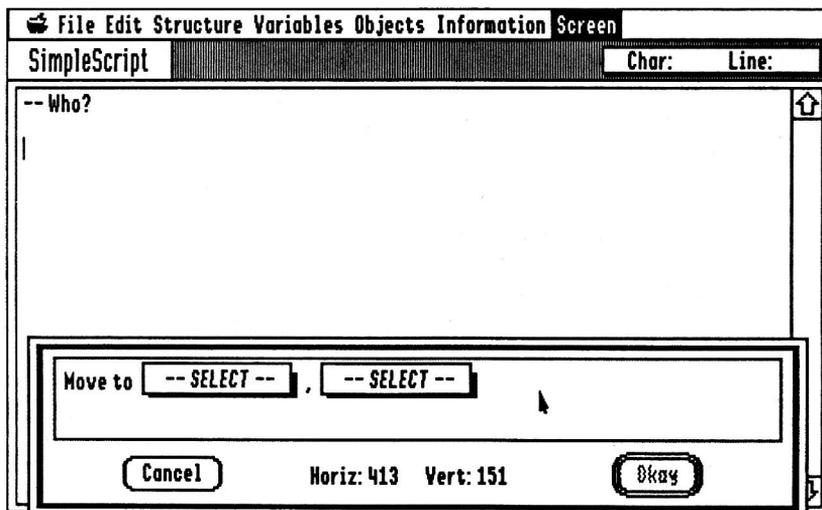
Additionally, having the commands always just under the menu bar provides a sort of “online help” facility—if you know the command that you want to use but don’t remember the exactly syntax, you can select it from the appropriate menu and instantly have it on the screen in front of you.

The program that we're going to write is going to be four lines long: one to set the position of text that will be drawn, one to set the font that the text will be drawn in, one to set the color for the text, and the last line to finally draw the text.

The First Line: Setting the Position

The first two lines of your program are already typed in for you: the first line is a comment line (which says "-- Who?") and then a blank line below that. The comment line does nothing when the program is run—it's just there to remind you the name of the button that you're writing a script for. You can enter more comment lines at any time by simply typing them in: when a SimpleScript program is running, if a line starts with two dashes, it is ignored.

- To write the first line of the program, choose *Move to* from the Screen menu.



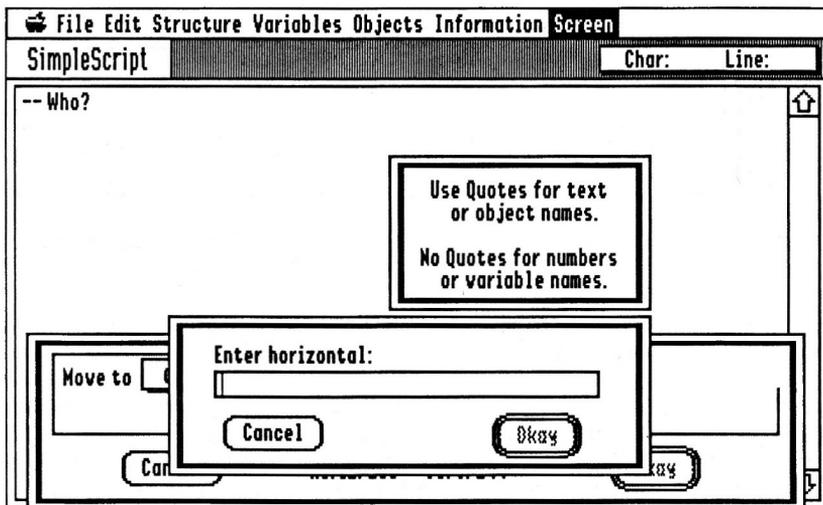
- The “Move to” command dialog box will appear, as shown above. You’ll notice two pop-up menus, both labeled “--SELECT--”. This is where you can select the different parameters for the “Move to” command.
- By looking in appendix B of the HyperStudio Reference (which has a list of all of the SimpleScript commands), or just by using SimpleScript for a bit, you’ll find that the “Move to” command requires two parameters: a horizontal and vertical position to locate the mouse cursor.
- As you move the mouse cursor around on the screen, you’ll notice the “Horiz:” and “Vert:” values constantly being updated. This way, you can determine the appropriate coordinates by simply moving the mouse.

We need to select a location almost anywhere on the screen that isn’t in the lower right corner (since that’s where the “Who?” button is), and that isn’t within an inch of the edges of the screen. The text will be drawn above and to the right of the point

we specify—that is, the point we specify will be used as the lower-left corner of the first character of the text. As a result, if you choose a point too high on the screen or too close to the right edge, the text we draw will simply run right off the screen.

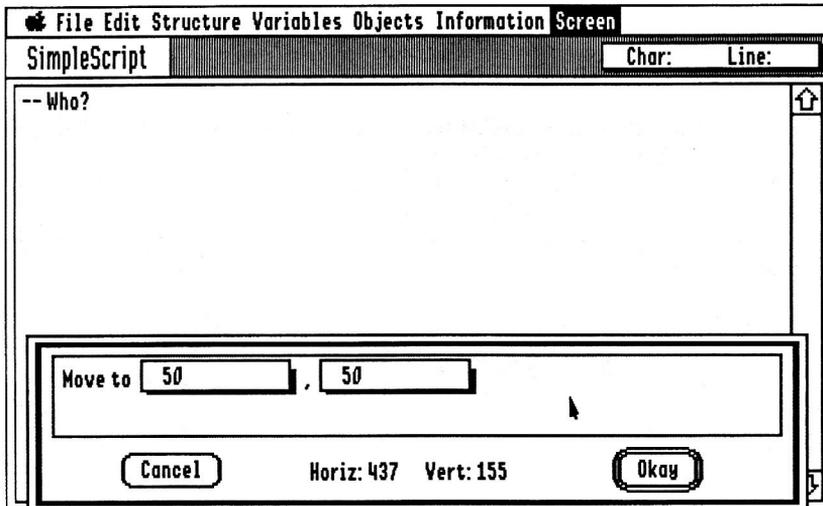
For this sample program, we'll use coordinates of 50, 50.

- To specify the horizontal coordinate, select “(type)” from the left pop-up menu. You'll notice one other option in the pop-up menu: “It”. This is a default variable in SimpleScript and is discussed more, below.
- After selecting “(type)”, two new dialog boxes will appear: one with instructions and a larger dialog box with two buttons (“Cancel” and “OK”), prompting you to enter the horizontal coordinate.



- Type “50” (without the quote marks) and click “Okay.”
- You are returned to the dialog box with both pop-up menus, and the first has been changed to “50” (the horizontal coordinate that you just entered). Repeat the process for the second pop-up menu, and enter “50” for the vertical coordinate.

- When you're done, the dialog box should look like this:



If it doesn't, click "Cancel," then go back to "The First Line," above, and try again.

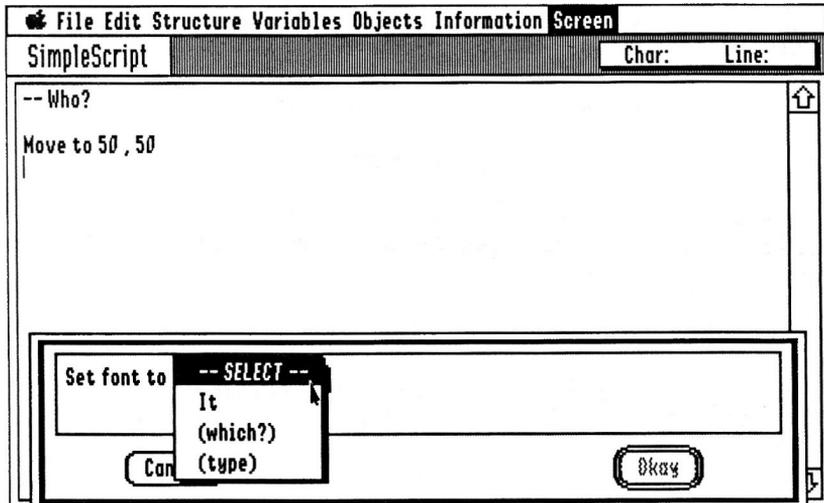
If everything looks okay, click "Okay." The dialog box will close and in the editor, the line `Move to 50, 50` will be typed out for you.

Note that, if you had wanted to, you could have simply typed out `Move to 50, 50`. But what if you didn't remember if the command was "Move to" or "Move over to" or maybe even just "Move"? This is why all of the SimpleScript commands are located under the menus at the top of the screen—and why SimpleScript is the programming language that you can use without ever touching a reference manual.

The Second Line: Setting the Font

The second line of our first program will actually set the font that the text will be drawn with.

- Select *Change font* from the **Screen** menu.
- A dialog box with a "--SELECT--" pop-up menu will appear—much like the one that appeared when we created the "Move to" line above.



If you look in the pop-up menu, you'll notice the same two options that "Move to" had: "It" and "(type)". You'll also notice one more option: "(which?)".

The "(which?)" option appears when there is a special way to specify what information should be used at that point in the statement. For example, if we were choosing a color instead of a font, the "(which?)" option would bring up a dialog box containing all of the colors in the system, and we could just click on the one we liked.

Selecting "(which?)" from the pop-up menu in this case, however, will bring up the system font selection dialog box.

- Select "(which?)" from the pop-up menu. The standard system font selection dialog box will appear. Click "Geneva" in the font name list, and click "12" in the font size list. Then, click "OK" to leave the dialog box.

You'll notice that "Geneva 12" has been placed in the pop-up menu, so that the entire line in the dialog box reads, "Set font to Geneva 12". Since this is what you wanted it to say, click "Okay" to continue, and, like before, you will be returned to the SimpleScript editor with the new line in your script. (If the pop-up menu

doesn't say "Geneva 12," try selecting "(which?)" from the pop-up menu again, or, just click "Cancel" and start again at "The Second Line," above.)

That's two lines down, and two to go.

The Third Line: Setting the Color

The next statement that we want to add to our program is one to change the text color.

For this line, instead of using the pull-down menus, let's try typing it directly into the editor.

- Position the cursor below the line that reads "Set font to Geneva 12".
- Type "Set text color to 4" and press Return.

"4" is the number value for red. (You can find the number values for all of the colors in appendix B of the HyperStudio Reference manual, "SimpleScript Reference.")

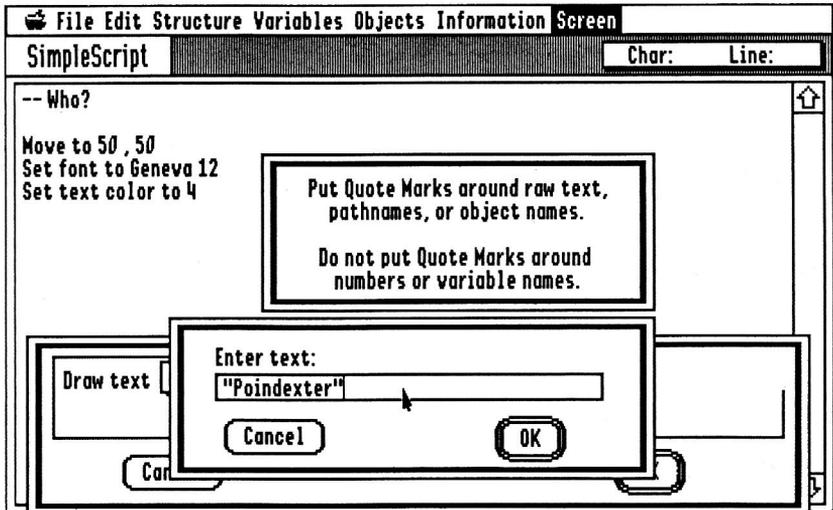
The Final Line: Drawing the Text

Finally, now that we've set the font, color, and screen position, we're ready to draw the text on the screen.

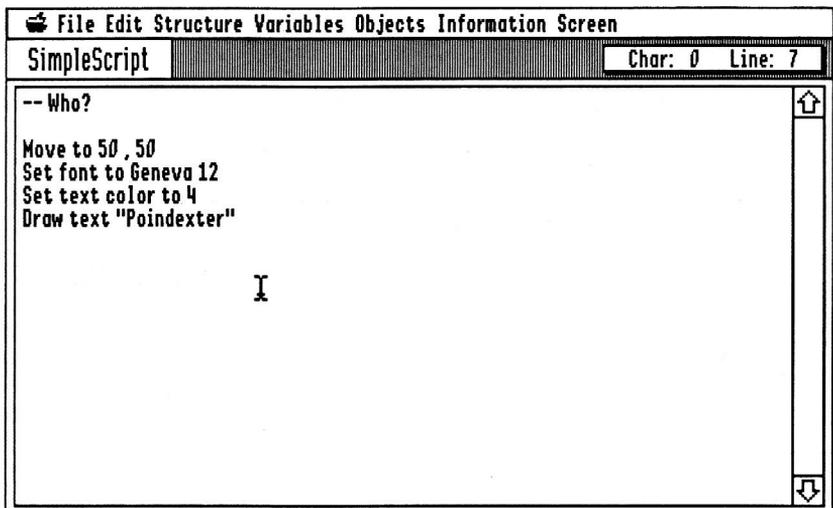
- Select *Draw text* from the **Screen** menu.

When the dialog box appears with the "--SELECT--" pop-up menu, select "(type)" and in the resulting dialog, type your first name *surrounded by quote marks*. It's very important that you use double quote marks—if you don't, HyperStudio will think that you're trying to create a new variable.

After entering your name, the screen will look like this:



Click "OK" to exit this dialog box, then "OK" to return to the editor.



Running the Program

At this point, our program is completely written. Double-check that the screen looks like the picture above, then,

- Pick *Save and Quit Editor* from the **File** menu.

- Click “Done” to tell HyperStudio that you are done defining your button. (Note that the checkbox next to “Scripting Language...” is now marked.)
- Finally, click the “Who?” button, and your name should be written in red, with the Geneva font (at 12 point size) at location 50, 50 on the screen!

If you made a typo in your program, you will automatically enter the SimpleScript editor, where the offending line will be highlighted. You can make corrections here and then select *Save and Quit Editor* from the **File** menu to return to your stack.

Looping and Variables

Now that you’ve got the basics of using SimpleScript out of the way, let’s try something a little more complex—a program that plays a sound as many times as you specify. The program will ask how many times it should play the sound (which we will record ahead of time).

Recording the Sound

- Leave your “Who?” button in place. (Don’t delete the button or select *New Stack*.)
- Make a new button named “sound”. Place it below the “Who?” button on the screen. At the Button Actions dialog box, click “Play a Sound...”

We need this button to play a sound, but, before it can play the sound, we have to record the sound and store it in HyperStudio. We’ll do this by using the HyperStudio Tape deck and recording a sound in a button that won’t actually be used. (This way, that button will “hold” the sound, but we’ll make a new button to play the sound with SimpleScript.)

- You should be at the HyperStudio Tape deck. Click the “Record” button and say your name firmly into the microphone.
- Press Return stop recording. Click “Play” to hear your sound and if you’d like to re-record it, simply click “Record” again.
- When you are satisfied with the name of the sound, type “me” as a name for the sound and then click “OK.”

This will store the new sound in the “sound” button.

- Click “Done”. You’re done making the “sound” button — if you click the button, it should play the sound you just recorded, called “me,” and do nothing else.

Writing the Program

Now, we'll make a button that uses the "me" sound.

- Make a new button named "Say It". Place it below the "sound" button on the screen. At the Button Actions dialog box, click "Scripting Language..."

We're going to type the entire program in the editor without using the pull-down menus.

- Enter the following program (the first line and the blank line below it should already be there). Note that you don't need to indent the fifth and sixth lines; they will automatically be indented for you.

```
-- Say It

Ask question "How many times?" with It
For Loop = 1 to It
    Play sound "me"
    Pause 45 ticks
Next
```

Be sure there are no typos in your program; check to be certain that what you have entered matches the listing above.

Before stepping through this program line-for-line, try running it:

- Select *Save and Quit Editor* from the File menu.
- You should be back at the Button Actions screen with "Scripting language..." marked. Click "Done."
- Finally, try your button out: click the "Say It" button. You should see a dialog box, prompting you for the number of times it should play a sound. Enter a single digit and click "Okay" (or press Return).

The sound that you recorded earlier will be played, as many times as you have specified.

How Does It Work?

Let's take a look at each line of the program and determine how it works.

```
-- Say It
```

This first line is simply a comment. It is the name of the button, and is ignored when the program is run. You can erase this line, or change it to say anything you like—as long as it starts with two dashes, it's considered a comment and won't affect how the program acts.

```
Ask question "How many times?" with It
```

Ask question is a SimpleScript statement. According to the SimpleScript reference, the exact syntax is Ask question *expression* with *variable*. This means that you have to fill in *expression* (which can be either a literal string, as we used here, or a variable, such as It) and *variable* (which must be the name of a variable).

Ask question takes the *expression* you specify and places it in a dialog box, with an edit line below the *expression*. You can then enter text, and click "Okay." When you do, the text entered is placed in the *variable*.

So, this line of the program (Ask question "How many times?" with It) asks the user how many times he or she wants the sound played, then places their response in the variable It.

Variables are simply holders for strings. So, in this case, if the user typed "4", that "4" would be stored in the variable It. (It is the "stock" variable that is always available in SimpleScript. You can make a new variable by simply starting to use one: for example, you could have changed the statement to Ask question "How many times?" with response and it would work fine, defining the value of the new variable response along the way.)

```
For Loop = 1 to It
```

This line is the beginning of a For - Next loop, which is a special statement that allows you to repeat other statement(s) as many times as you specify.

For Loop = 1 to It instructs SimpleScript to count from 1 to the value stored in the variable It. It uses a new variable, Loop, as the "counter." (While inside of the For - Next loop, you can examine the variable Loop to see how far along the loop is. The first time through the loop, Loop will equal 1, second time Loop will equal 2, and so on.)

This will be explained further, below. Suffice to say for now that the For Loop = 1 to It statement begins a loop.

```
Play sound "me"
```

This statement is pretty simple! It just plays the sound in stack named "me". (Note that, if SimpleScript can't find a sound in the stack named "me", it'll try to play a sound file *on disk* called "me". If that isn't successful, no sound is produced and SimpleScript simply continues along to the next line.)

```
Pause 45 ticks
```

Another easy line: one tick is $\frac{1}{60}$ th of a second; this statement simply waits for three-quarters of a second to pass before it allows the program to continue.

This statement is used because the `Play sound` statement, above, only *starts* a sound playing—it doesn't wait for the sound to stop before continuing. By inserting this statement, to wait for a just a moment, the sound may play most of the way (or completely) before the loop continues.

Next Loop

This is the end of the `For - Next` loop, started above. When SimpleScript reaches this line in the program, it adds one to the loop counter (stored in the variable `Loop`), then checks to see if it's reached the end of the loop by comparing the loop counter to the end of the loop (stored in the variable `It`). If the value stored in `Loop` is less than or equal to the value stored in `It`, SimpleScript branches back to the start of the loop (the `Play sound` statement) and runs all of the statements inside of the loop again.

If `Loop` is larger than `It`, the loop has ended, and SimpleScript simply continues past the `Next Loop` statement—to the end of the program.

Modifications

This program, though very simple, introduced you to variables and looping with SimpleScript. Try some of these modifications:

- Change the `Pause 45 ticks` line to have the computer pause for less time (try 15 ticks) or more time (you could change it to `Pause 3 seconds`, if you like). By varying the pause, the effect of the sound playing several times inside of the loop can cause a “stuttering” effect.
- Remove the `Pause 45 ticks` line and see how the program works. (To quickly “remove” the line without actually cutting it out of the source code, insert two dashes at the beginning of the line. This will cause the line to become a comment, which SimpleScript will ignore when the program is run.)
- Change the program to play a different sound. If SimpleScript can't find the sound that you specify in the stack, it will look for the sound file on disk (in the same directory as the stack).
- Add one line to the end of the SimpleScript program attached to the “Who?” button: `Go to button "Say It"`.

This simple statement does just what you would imagine: transfers control from one button to another. By adding it to the “Who?” button, when the “Who?” button is clicked, your name will be drawn and then spoken by the computer.

The Quiz Stack

The last program you will be writing will quiz the user about their knowledge of vehicles. Specifically, it will present three pictures: a boat, a truck, and an airplane. Then, the stack will ask the user which one of those three they're most likely to find in the air. When the user points to the airplane, the screen border will flash and the program will let them know they got it right.

To do this, we need to set up the graphic objects before actually writing the SimpleScript program that controls the game.

The layout of the screen is simple: in the middle of the screen, about an inch from the bottom, will be a button named "Quiz." This is the button that starts the game. In the center of the screen, from left to right, will be the three graphic objects (the boat, the airplane, and the truck). Finally, the top part of the screen will be blank: this is where the SimpleScript program will draw instructions for the user.

Internally, the SimpleScript program that controls the game is almost as simple: SimpleScript repeatedly examines the coordinates of the mouse and checks to see if they're within the coordinates of the airplane graphic object. If they aren't, nothing happens; otherwise, it means that the user has moved the mouse pointer over the airplane and that the game is won.

Note that the coordinates of an object are often referred to as the "rectangle of the object" or just the "rect". These coordinates are expressed with four numbers: $x1$, $y1$, $x2$, and $y2$. The two $x1$ and $y1$ coordinates are the screen coordinates of the upper-left hand corner of the rectangle, and the $x2$ and $y2$ coordinates are the screen coordinates of the lower-right hand corner. (Another way to think of this is that $x1$ is the left side, $y1$ is the top, $y2$ is the bottom, and $x2$ is the right.) By checking the mouse coordinates and seeing if they're inside of the airplane's "rect," we can determine if the user is pointing to the airplane or not.

Preparing the Stack

In order to get ready to write the SimpleScript program, we're going to place the three graphic objects in place.

- Choose *New Stack* from the **File** menu. If you want to save the "Who?" stack created in the previous section, click "Save" and give the stack a name; otherwise, click "Don't Save".
- The first thing to add to the new stack is the boat graphic object. So, select *Add a Graphic...* from the **Objects** menu. All three of the graphic objects we're using come from a screen full of clip art on the `HS.Art` disk, called `Output`.

★ | One-Drive System

Note to one-drive system users: insert the `/HS.Art` disk into the disk drive. Click "OK" on the "no volume on-line" dialog box message.

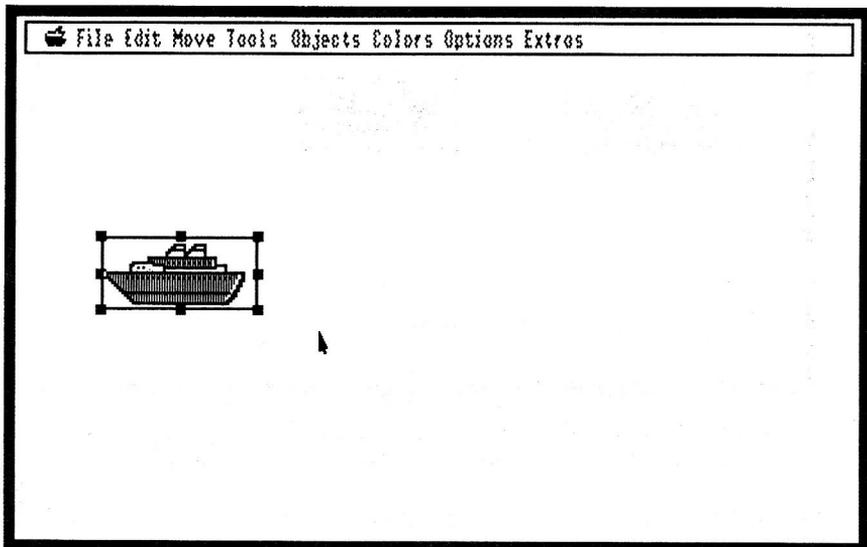
⌘ | Two-Drive System

Note to two-drive system users: insert the /HS.Art disk into the second disk drive.

⇨ | Hard Drive or Network

Note to hard drive or network users: locate the /HS.Art folder. It should be inside your HyperStudio folder.

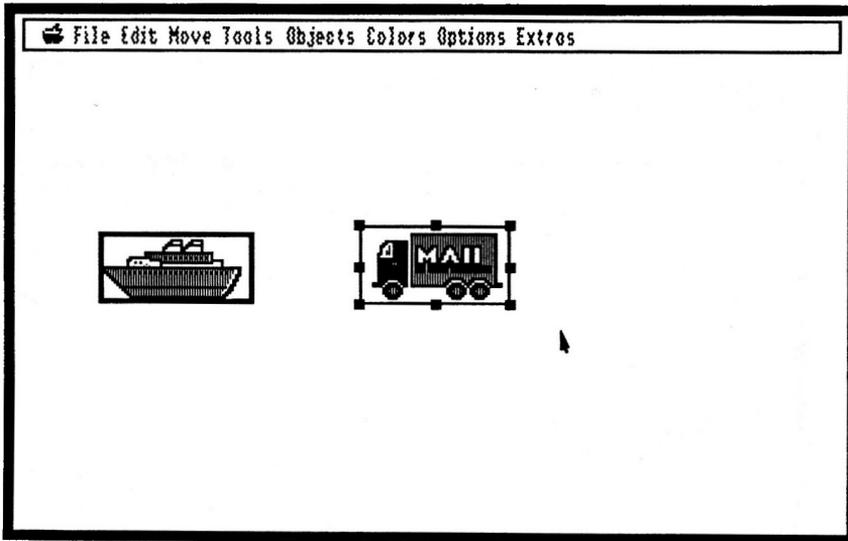
- Double-click the Output file to select it and open it.
- Crop the graphic so that you have only the boat within the graphic object window. The boat graphic object is a little to the left of the center of the Output screen.
- By dragging the frame around the window, move the boat to the middle of the screen, then place it about an inch from the left side of the screen. Your screen should look like this:



- Click outside of the graphic object window to place the graphic object on the screen. You will be presented with the Graphic Info dialog box.
- Type "boat". By naming the graphic, it's a lot easier to work with in SimpleScript. (If we didn't name the graphic, we would have to refer to the object's card position value, which isn't nearly as easy as just referring to it by name.)
- Click "OK".

Now that you've got the boat graphic object in place, let's add the airplane and truck graphic objects.

- Select *Add a Graphic...* from the **Objects** menu. Again, double-click the Output file to open it.
- Crop the graphic so that you have only the mail truck within the graphic object window. (The mail truck is on the left side of the Output screen.)
- By dragging the frame around the window, move the truck to the center of the screen, just to the right of the boat. Leave a little bit of a gap between them—about an inch. Your screen should look like this:



- Click outside of the graphic object window to place the graphic object on the screen. You will be presented with the Graphic Info dialog box.
- Type “truck”. Again, by naming the graphic, it's a lot easier to work with in SimpleScript.

There's one graphic object left: the airplane.

- Select *Add a Graphic...* from the **Objects** menu. Once again, double-click the Output file to open it.
- Crop the graphic so that you have only the airplane within the graphic object window. (The airplane graphic is in the center of the Output screen.)

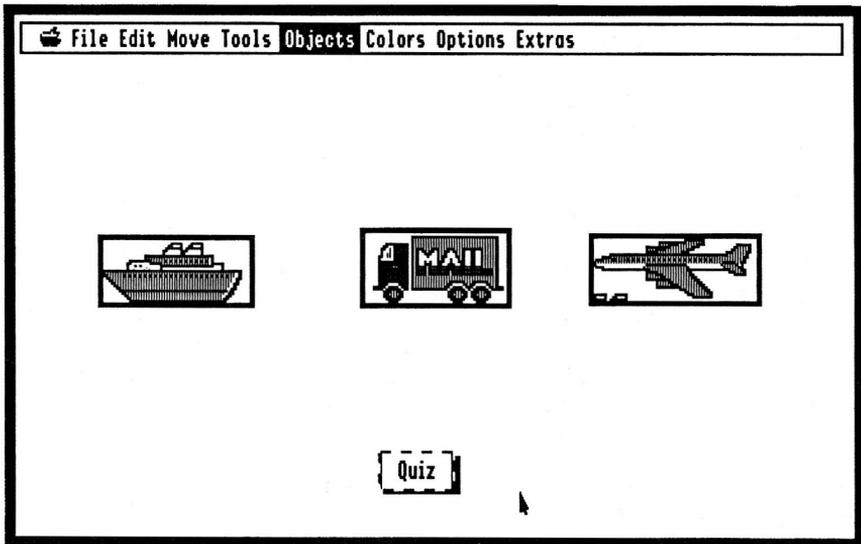
- By dragging the frame around the window, move the airplane to the center of the screen, and to the right of the truck. Leave a little bit of a gap between the airplane and the truck—about an inch.
- Click outside of the graphic object window to place the graphic object on the screen. You will be presented with the Graphic Info dialog box.
- Type “plane”.

Now that we have the stack set up, we can go ahead and create the button which will play the game.

Creating the Button

- Choose *Add a Button...* from the **Objects** menu. Make a shadow button—the second button type down on the right side of the eight button types. It looks like this: 

- Name the button “Quiz”. Position it directly below the three graphic objects, at the bottom center of the screen. Your screen should look like this:



- At the Button Actions screen, click “Scripting language...” and then enter the following program in the SimpleScript editor. Like before, the first two lines will already be in the editor, and you don’t need to worry about indenting lines—that will be handled automatically.

```
-- Quiz

-- draw instructions
Move to 50, 50
Draw text "Point to the object you find in the air."

-- watch mouse to see if it's over the plane graphic
Get rect of Graphic "plane" into x1 , y1 , x2 , y2
Repeat until (mx > x1 and mx < x2) and (my > y1 and
    my < y2)
    Read mouse position into mx , my
End repeat

-- erase instructions
Fill flag is On
Set line color to 15
Draw box from 50, 30 to 500, 100

-- play with the border color
Get border color into It
For J = 1 to 5
    Set border color to J
    Pause 5 ticks
Next J
Set border color to It

-- let the user know they won!
Beep
Show message "Right!" with icon Caution
```

- Double-check your program against this listing, being sure that you have it all correct. (If you have any mistakes, it could lead to unexpected results or simply an error message from SimpleScript.)
- Choose *Save and Quit Editor...* from the File menu to return to the Button Actions screen. Click “OK” to finish creating your button.
- Now, click the button to try the program out!

How it Works

Let’s look at the program, line for line, to see how it works.

```
-- Quiz
```

This first line is simply a comment, and identifies the button that this program is attached to. Remember that any lines that start with two dashes are simply ignored by SimpleScript.

```
-- draw instructions
Move to 50, 50
Draw text "Point to the object you find in the air."
```

This section of code draws the instructions for the user in the top part of the screen. It's very similar to the first example program, above: SimpleScript simply prepares to draw on the screen at location 50, 50, then draws the text string "Point to the object you find in the air."

```
-- watch mouse to see if it's over the plane graphic
Get rect of Graphic "plane" into x1 , y1 , x2 , y2
Repeat until (mx > x1 and mx < x2) and (my > y1 and
    my < y2)
    Read mouse position into mx , my
End repeat
```

This is the main loop of the quiz and the part that does the "dirty work." It first gets the rect of the graphic object "plane" into four variables: *x1*, *y1*, *x2*, and *y2*.

After that, a special type of loop (called a Repeat - Until loop) is started. For now, let's skip the remainder of the Repeat until... line and examine the rest of this section of code.

The next statement, which reads the mouse position and stores them into the variables *mx* and *my* (for mouse X position and mouse Y position), is indented—a clue that it is *inside* of the Repeat until loop. The following line is the other clue: End repeat. This marks the end of the Repeat until loop.

Let's take a closer look at this line:

```
Repeat until (mx > x1 and mx < x2) and (my > y1 and
    my < y2)
```

This line instructions SimpleScript to continually run all of the statements inside of the loop (which is only the Read mouse instruction) until several specific rules are met. In order to explain them, here's that same line in English:

```
Repeat until the mouse's horizontal location is
    within the airplane's horizontal area, and until
    the mouse's vertical location is within the
    airplane's vertical location.
```

Another way to look at it is:

```
Repeat until the mouse's X coordinate is larger than
the airplane's left edge and less than the
airplane's right edge, and until the mouse's Y
coordinate is larger than the airplane's top
edge and smaller than the airplane's bottom
edge.
```

Transforming that a little further into SimpleScript gives us:

```
Repeat until (mouse's X coordinate > airplane's left
edge and mouse's X coordinate < airplane's right
edge) and (mouse's Y coordinate > airplane's top
edge and mouse's Y coordinate < airplane's
bottom edge)
```

Does it seem a little clearer now? The Repeat until... loop will continue until the mouse pointer is inside of the rectangle surrounding the airplane.

```
-- erase instructions
Fill flag is On
Set line color to 15
Draw box from 50, 30 to 500, 100
```

Since the Repeat until loop runs until the user actually points to the airplane, by the time we reach these lines, we know that they've won the game. As a result, the first thing to do is erase the instructions painted on the screen earlier. This is done by simply drawing a white box over the text.

```
-- play with the border color
Get border color into It
For J = 1 to 5
    Set border color to J
    Pause 5 ticks
Next J
Set border color to It
```

This small program fragment cycles the screen border by using a For - Next loop, like the one in the "Say it" program. The fragment first gets the current border color and stores that value in the variable *It*. This way, after we get done changing the border color, we can reset it back to what the user originally had. (This is *always* a good idea, when programming: leave things the way you found them!)

After saving the border color, a short loop is run that simply changes the border color to the current value of the loop counter (in the variable *J*). After changing the border color, we pause briefly (for 5 ticks—remember, 60 ticks makes one second,

so this is a very short wait), and then continue. The result is a flashy color show in the screen border.

The last step is to restore the old border color, saved in the variable `It`. If we left this line out, the screen border color would be left at dark gray (color #5, the last border color change before the loop ends).

```
-- let the user know they won!  
Beep  
Show message "Right!" with icon Caution
```

Finally, we let the user know that they did win by beeping the speaker and showing a message in a dialog box. The caution icon seemed ideal for this message: it's an exclamation mark inside of a triangle.

Suggested Changes

- Try changing the program so that it asks for the object most likely to be found in the water. This is actually one of the easiest changes to make to the program. Just get the rect of the “boat” object instead of the “plane,” and be sure to change the instructions!
- What happens if you remove the `Set border color to It` line? (Remember, the easiest way to “remove” a line is to just place two dashes at the start of the line.)
- It's also a simple change to make the program check for the mouse over a text field or button. You could have three text fields with words in them, and quiz the user for the misspelled word, for example.
- Try building on to the stack by adding another card which has three more objects and uses the same program to quiz the user.

Where To Go From Here?

There's a lot of SimpleScript that isn't covered here because there's an enormous amount of power and flexibility in any programming language, limited only by your imagination. Hopefully, this tutorial has given you a taste of what SimpleScript can do, and now you may explore on your own.

Conclusion

By following this tutorial, you've gotten a taste of the power and extreme ease-of-use that HyperStudio offers. You probably already have an idea of some of the stacks that you're going to create to share with others, and you may be eager to explore further.

To see more of what HyperStudio can do, you may wish to examine the sample stacks included with the HyperStudio package (on the disks `/Samples.1` and `/Samples.2`). Another way to get ideas is to simply page through the HyperStudio Reference manual and see what interests you.

Regardless of how you decide to dive into the program, we hope you enjoy the multimedia experience for multi-talented minds, HyperStudio.

