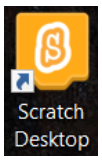
Creating an Application with Scratch

Scratch is a simple programming environment that allows you to create programs using blocks of code rather than text code. It is often used to create games, although it is perfectly capable of creating a wide range of other practical applications. We will use Scratch to create just such an application in this exercise.

Suppose your school library was going to set up a small information kiosk that allowed users to find information about the various services that were available in the school. Scratch could be used to create an interface / menu system for the kiosk.

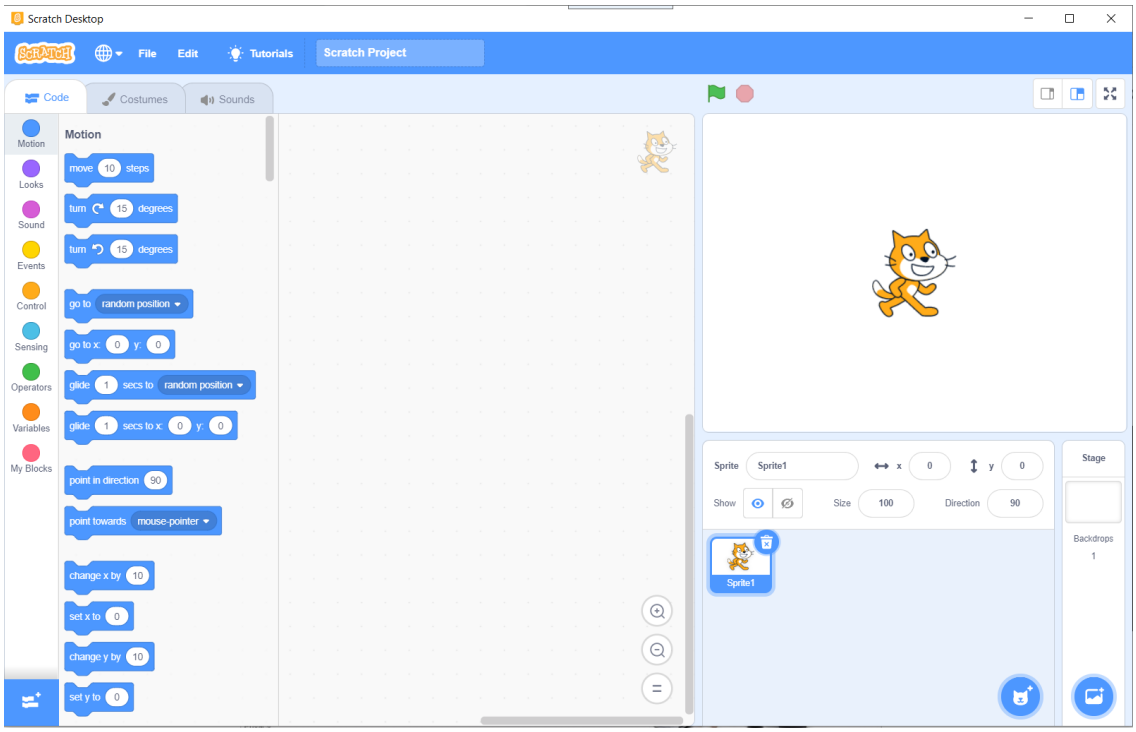
Note you can use the online version or the desktop version of Scratch. There are only a few small differences. The desktop version is used in these exercises for simplicity in saving files as well as removing the need for an Internet connection while you work. There is also no need to have a log in account with the desktop version

1. Open **Scratch** by selecting it from the start menu or from the desktop.



Many of you have probably used Scratch before but if you haven't, it's pretty easy to use.

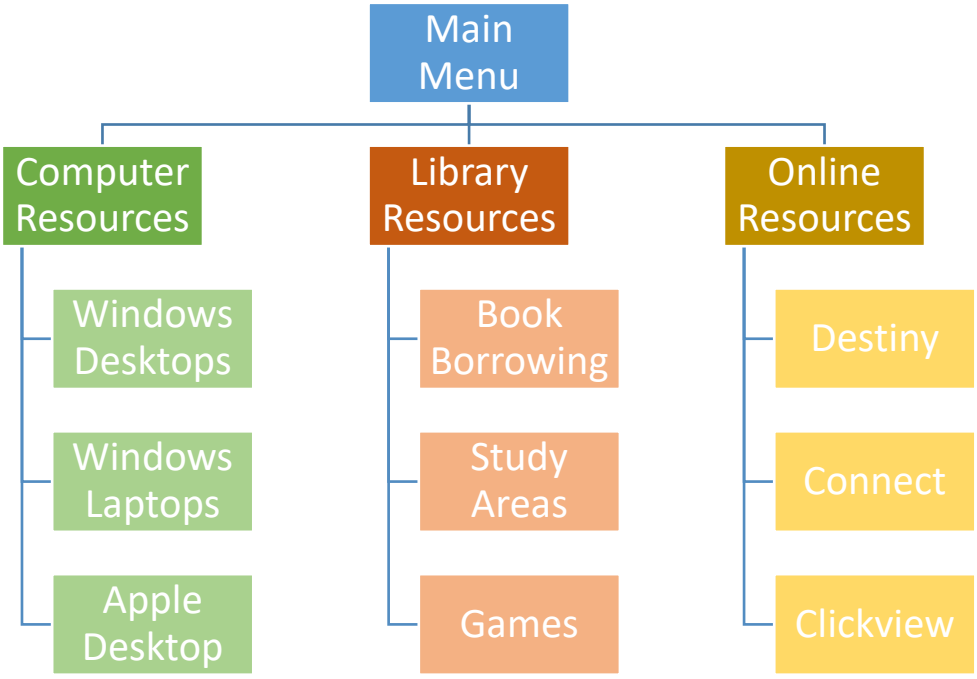
- The section to the right with the cat in the middle is the **stage**. This is where the program happens.
- The section below that is where your **sprites** are organised. Sprites are the various objects in your program (such as characters in a game or menu items in an interface).
- The section on the left is where you place your **Script** blocks. A selection of script blocks you can choose from is shown next to it with coloured categories along the side to choose from.



Planning

Firstly a bit of planning. It's always a good idea to plan a program before creating it. Proper planning can help you to avoid logic errors and other problems. In this case, since we're creating a system of menus, we've made a plan that shows the hierarchy of menus in our system.

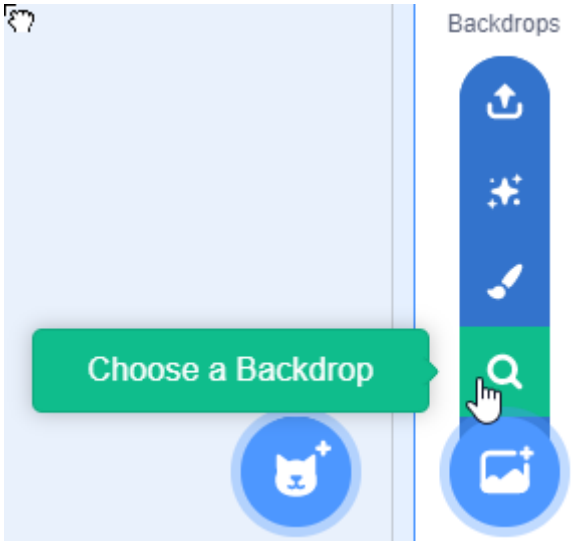
We will have a main menu, with options that lead to 3 further menus.



Menu Backgrounds

First we'll select backgrounds for each of our menus.

1. Move your mouse over the Backdrop icon in the bottom right corner of the screen and then click on the **Choose a Backdrop** option when it appears.



2. Pick a suitable background such as Room 1 (the book shelves make it suitable for a library main menu)

3. Select 3 more backgrounds. The ones shown below are suggestions.

Each time you select one it will look like you have replaced the one you already had but the other ones are still there.

Note You can also use the Upload Backdrop option to import a picture to be used as a backdrop.

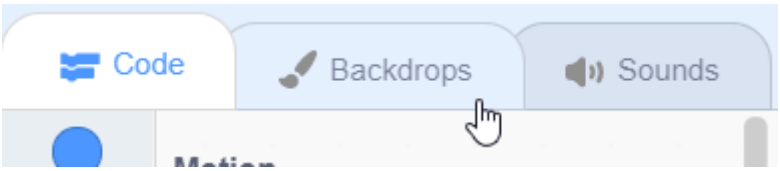

Room 1


School


Chalkboard


Spaceship

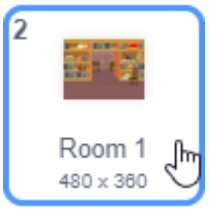
4. Click the **Backdrops** tab at the top of the code section.



You will now be able to see all of the imported backdrops listed. Each one will have a name based on the name of the picture you imported. When we want our program to change to a particular backdrop we will refer to them by these names. To make it easier, we will change the names to reflect the menus they will be used for.

The first one in the list is the blank backdrop we started with. Since we won't be using it, we could delete that one but it won't matter if we leave it there. When you click a backdrop, a small bin icon in the corner allows you to delete it.

5. Click on the first of the imported backdrops.

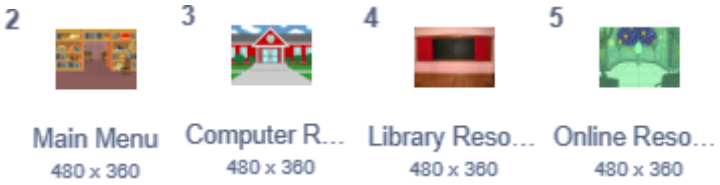


This opens that backdrop in an editor with some drawing tools which allow you to modify the backdrop. If you had the original blank backdrop selected, you could use these tools to draw your own backdrop.

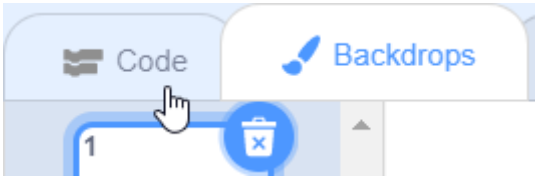
6. Above the editor is the current name of this backdrop, next to the *Costume* label. Change the name to **Main Menu**.

7. Press **Enter** after typing the new name and the new name will appear under the backdrop's icon.

8. Change the name of your other three menus to **Computer Resources**, **Library Resources** and **Online Resources**.

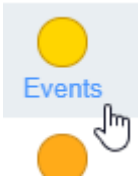


9. Click the Code tab when they are all renamed.



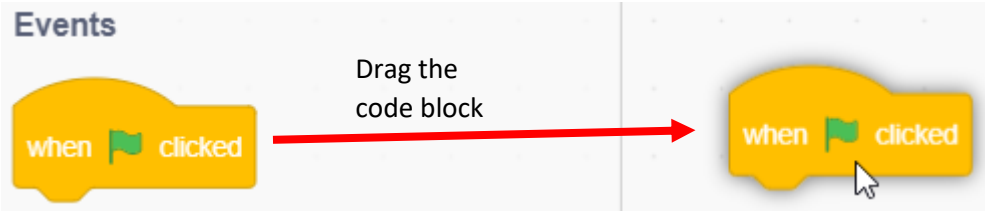
We want to make sure that the backdrop for the main menu is the first one that shows when our program starts so we will add our first bit of script.

10. Select the **Events** category.



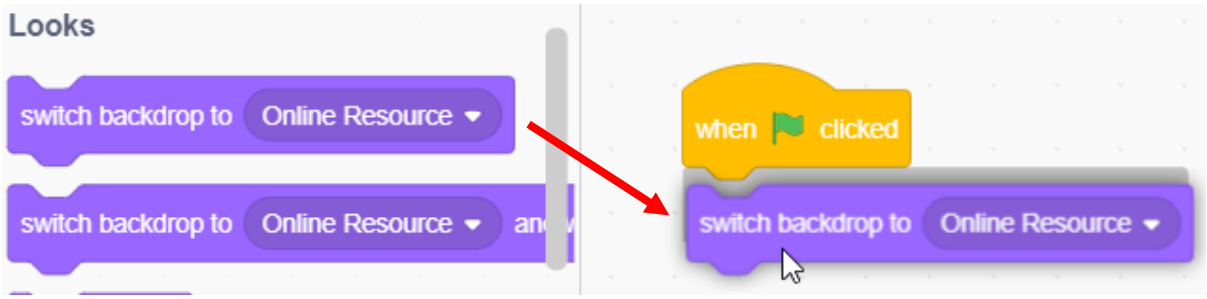
In Scratch a program starts when the Green flag icon  at the top is clicked. We will add some code blocks that tell the program to show the main menu backdrop as soon as the green flag is clicked to begin the program.

11. Select the **when** **clicked** code block and drag it to the script area.

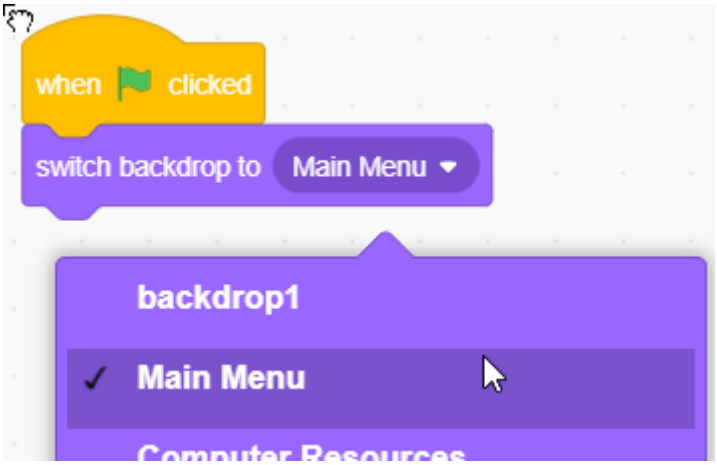


12. Click the **Looks** category.

13. Drag the **switch backdrop to** code block so that it snaps on to the bottom of the block we already added.



14. Change the selected backdrop to Main Menu.



Test our Code

1. Click the Backdrops tab and select one of the backdrops other than the main menu to make it display on the stage.
2. Click the **green flag** icon at the top of the stage to run the program.

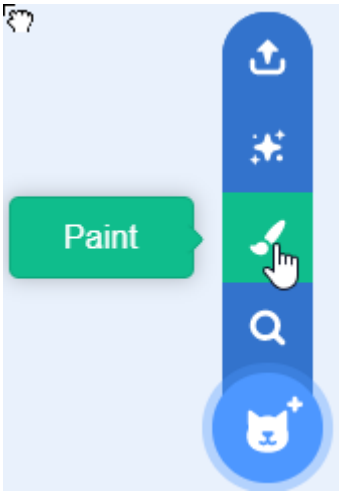
The code will make the main menu backdrop display in the stage area. We've now made sure that this is always the first backdrop to display when the program begins.

Setting up our main menu

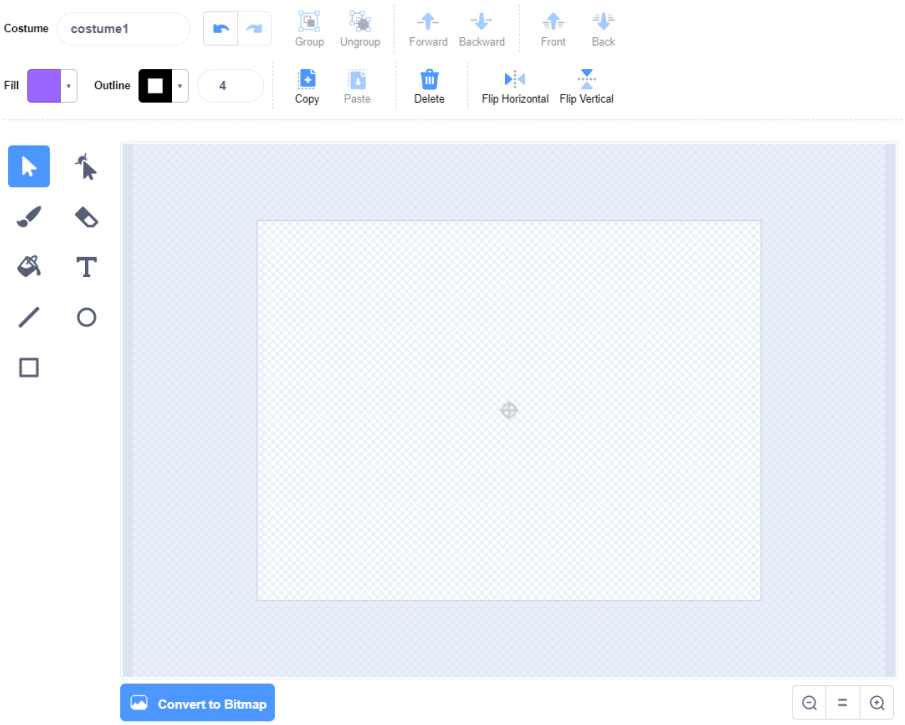
Our menu will consist of a few menu options. Clicking each of these options will take the user to a different menu. Each of the menu options will be created by adding a new sprite that contains text.

Creating menu options

1. In the sprites area, move your mouse over the new sprite icon and then click on the Paint option.

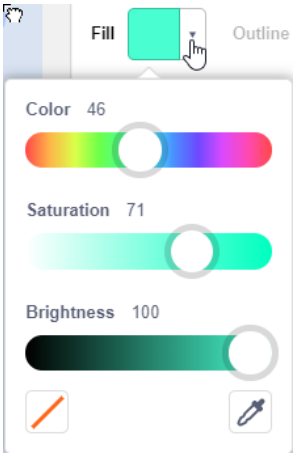


The main part of your screen will now be taken up by an editing area. The edge of this area contains numerous tools that allow you to draw and edit sprites.



2. Click the **T** icon on the side of the sprite editor. This allows us to place text in the sprite area.

3. Click the **Fill colour** icon and then use the **Colour**, **Saturation** and **Brightness** options to select a colour for your text. Remember it should be a colour that will be visible when the text is in front of the backdrop.

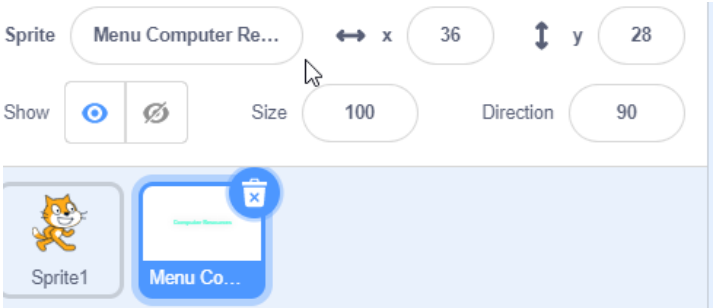


4. Click in the sprite editing area and type **Computer Resources**.

When you create a sprite it is usually a good idea to make sure the contents of the sprite are in the middle of the sprite editing area. This ensures that Scratch will correctly position the sprite when needed.

5. Click the **Select** icon next to the editing area.
6. With the select tool activated, drag the text so that its middle lines up with the small cross in the middle of the editing area.

7. At the top of the sprite area, change the name of the sprite to **Menu Computer Resources**.

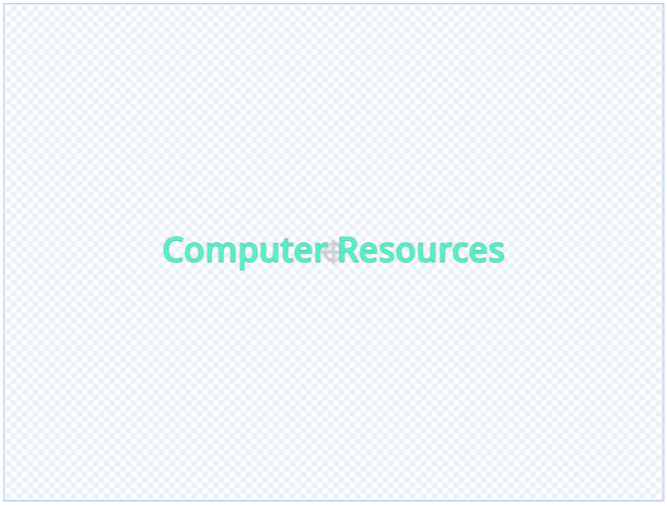


You can use your mouse to reposition the text sprite on the stage. Depending on what colour you chose, you might find it hard to read the text on the background. One way to make it easier to see is to add a thin outline around the text.

8. Make sure the sprite is selected with the Select tool in the sprite editing area.

9. From the options at the top of the sprite editing area, change the outline thickness from 0 to 1. You can also select a different colour for the outline if that helps.

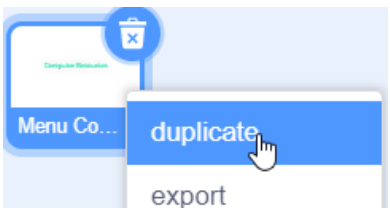
10. Resize your text to make it larger and reposition it in the centre of the editing area.



11. Drag the sprite on the stage so that it is toward the left and middle. You might need to drag the cat sprite out of the way (we will use the cat sprite later).

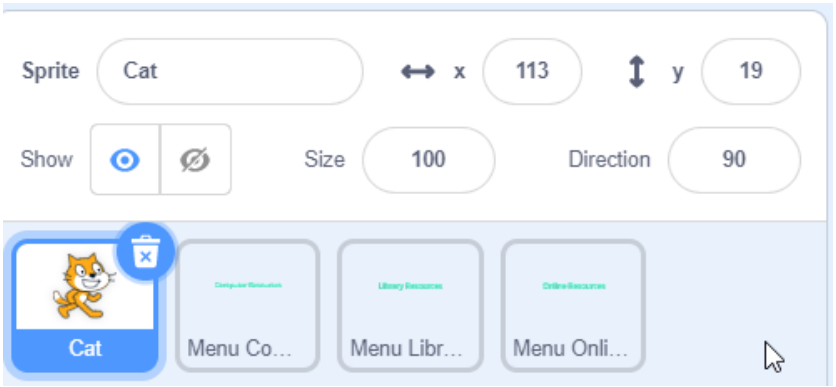
12. Repeat steps 1 to 0 to Create a new sprite that shows the text **Library Resources**. Rename the new sprite to **Menu Library Resources**.

Tip	If you want your other 2 sprites to look the same apart from what the text says, you can right-click a sprite in the sprites area and select Duplicate. Then you only need to change the text and change the name of the sprite.

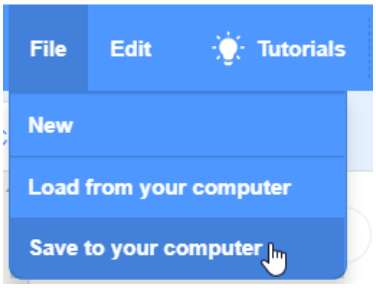


13. Repeat steps 1 to 0 to Create a new sprite that shows the text **Online Resources**. Rename the new sprite to **Menu Online Resources**.
14. Position the two new menu items on the stage as shown in the example below. Don't worry about getting the positioning exact at we will give them more precise positioning later with code.

15. You should have 4 sprites including the cat sprite that was there when you opened Scratch. Rename the Cat sprite so that it is called **Cat** instead of Sprite 1.



16. From the File menu select **Save to your computer**.



17. Save your scratch project in a suitable location with the filename **Library Interface Menu**. Save it regularly from now on.

Adding Animation to the menu

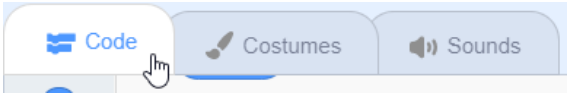
Moving things around the scratch stage is based on coordinates. Click one of the sprites in the sprite area and along the top you will see x and y number. The x coordinates are for horizontal (sideways) position. The y coordinates are for vertical (up and down) position. 0 is right in the middle so that for the x coordinates, positive numbers are to the right while negative numbers are to the left. For the y axis positive numbers are toward the top while negative numbers are toward the bottom.

1. Move your cat sprite so that he is in the doorway. Don't worry if he overlaps the text sprites.

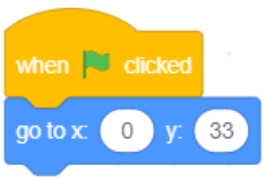
We'll set this as the starting position for the cat. You will notice that the x and y values above the sprite show its current coordinates on the stage.

Tip you can enter x and y values to make the cat move to exact coordinates on the stage.

2. Make sure your Code tab is selected.



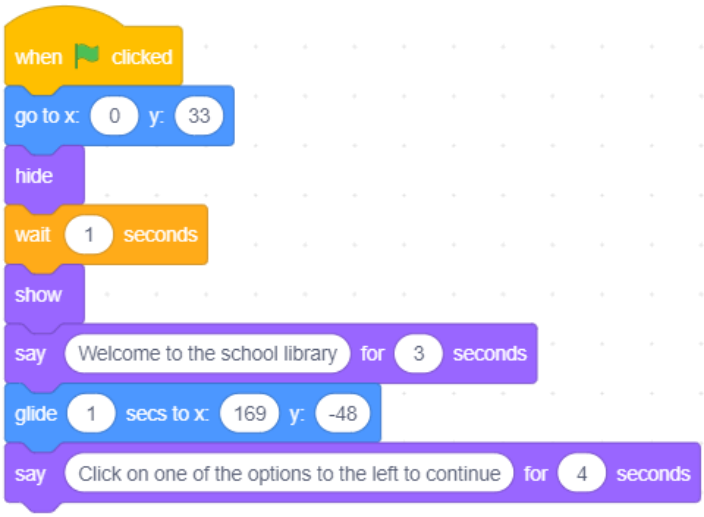
3. From the **Events** category add a **when**  **clicked** code block to the code area.
4. Select the **Motion** category. Here you will find all the code that is to do with positioning and movement.
5. Add a **go to x and y** code block to your start block. You will notice that the go to block has the current coordinates of the sprite already entered which saves you having to type them yourself.



Now we will add code blocks that will do the following as soon as the program begins.

- Hide the cat sprite and then wait for one second.
- Make the cat appear and then say a welcome message (in a speech bubble).
- Move to the right of the stage.

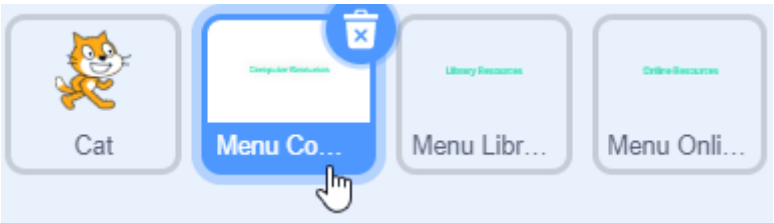
6. From the **Looks** category add a **hide** code block.
7. From the **Control** category add a **wait 1 seconds** block (the number can be changed for different amounts of seconds when needed)
8. From the **Looks** category add a **show** code block. Later we'll add some code to make the appearance more gradual.
9. From the **Looks** category add a **say Hello! For 2 seconds** block.
10. Change the text so that instead of "Hello!" it says "Welcome to the school library" and change the number to **3** seconds.
11. Drag the cat to the right side of the stage (in front of the chair).
12. From the **Motion** category. Notice that all of the code blocks with coordinates have now updated to the new position.
13. Add a **glide 1 secs to x y** block.
14. From the **Looks** category add another **say Hello! For 2 seconds** block.
15. Change the text to "Click on one of the options to the left to continue" and change the number of seconds to **4**.
16. Your code should look like the example to the right.
17. Click the **green flag** icon to test your code.



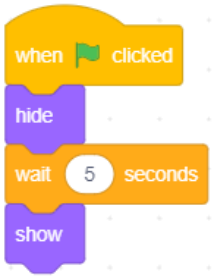
Now we will add code to each text sprite that will:

- Make it hidden when the program starts.
- Wait until the cat has moved.
- Gradually fade in to view.

18. Click on the **Menu Computer Resources** sprite to select it.

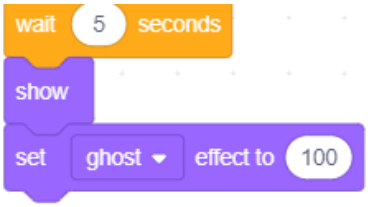


19. Add the following code blocks to the sprite.



This will make the sprite go from hidden to showing instantly after waiting 5 seconds but we will add a few blocks that will do it in a more gradual way.

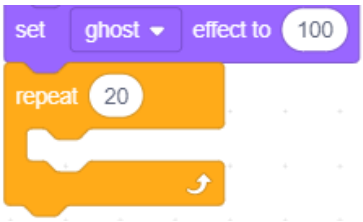
20. Add a **set effect to** block.
21. Change the options for the effect block so that the effect is ghost (transparent) and the effect amount is 100 (completely see through).



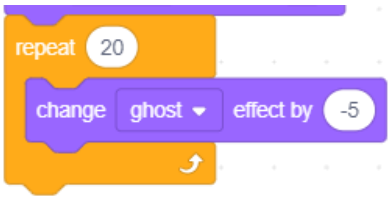
Now we want to add some blocks that will gradually reduce the ghost amount. We could add several change effect blocks like this. 20 blocks that each reduce the ghost effect by 5% would eventually turn off the see through effect.

This would be a really long way to do it though. A better way would be to simply but in one of these blocks and add another block that tells the computer to do it 20 times.

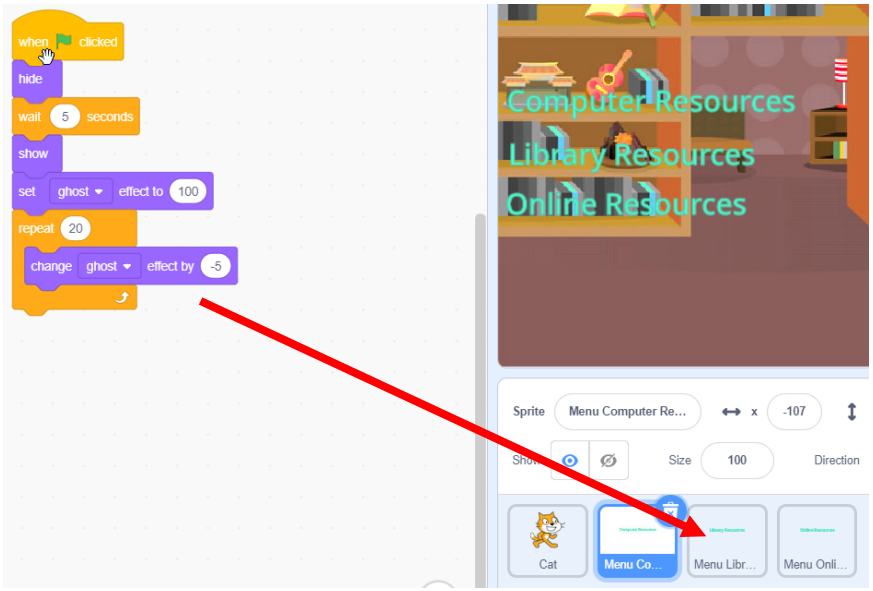
22. From the **Control** category add a **repeat 10** block and change the number to **20**.



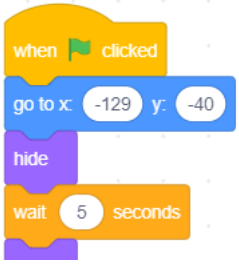
23. From the **Looks** category add a **change effect by** block with the effect type set to **ghost** and the amount set to **negative 5**. Place it inside the repeat block.



We need to add the same code to the other 2 text sprites. Instead of doing it all again we can simply copy it all.



24. Drag the first block in your code (the click green flag block) to move the entire group of blocks. Drag the code on top of your Menu Library Resources sprite.
25. Drag it again on to your Menu Online Resources sprite.
26. Click on either one of those sprites and you will see that the code has been copied in to each of them.
27. Click the green flag icon to run your program and test the code.
28. Lastly we will add a block to each of the text sprites to make sure it starts in the same position.
29. Add a go to x and y block to each of the text sprites.
30. Test the program again to make sure they are each starting in the right position. Make adjustments to the x and y starting position if needed.



Bonus Add some code blocks to the Cat sprite so that it fades in to view just the same as the text sprites.

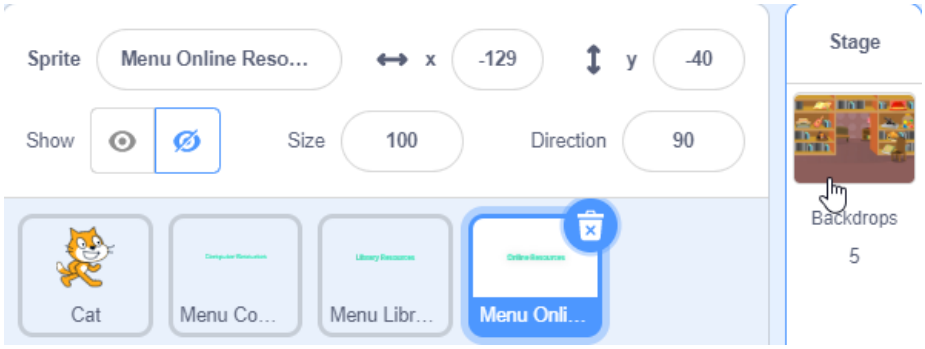
Creating our other menus

Each of the options on the main menu will open another menu when clicked. It will do this by changing to a different backdrop, hiding the sprites on the main menu and displaying previously hidden sprites that will appear on the new menu.

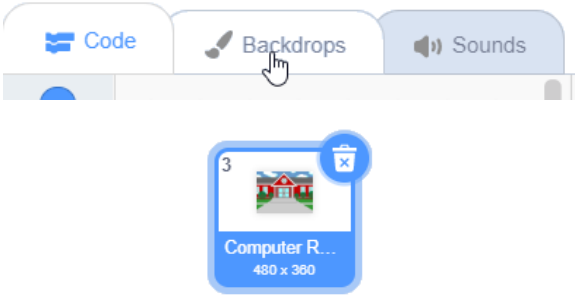
First, we'll hide the text sprites from the main menu so that they aren't in the way when we are working on our next menu.

1. Click on the **Menu Computer Resources** sprite.
2. Click the option above the sprite to change it to hidden.

3. Do the same to hide the **Menu Library Resources** and **Menu Online Resources** sprites.
4. Click on **Backdrops** to the right of the sprites so that the backdrop is selected.



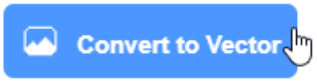
5. Click the Backdrops tab above the code blocks and then click the Computer Resources backdrop to make that the visible one.



We will modify the backdrop to add some text to it. We could do this by adding another text sprite but since the text won't be anything other than decoration, we'll add it to the background. Sprites are best used for part of the program that needs to be active in some way, such as menu items you will click on.

Currently our backdrop is a Bitmap image meaning it is made up out of Pixels (dots). If we convert it to a Vector backdrop so that we will have more editing options, particularly for text.

6. Click the Convert to Vector button at the bottom of the backdrop editing area.



7. Click the **T** (Text) icon.
8. Add text which says "Computer Resources" in a suitable location near the top of the backdrop.
9. Use what you learned earlier to modify the text so that it makes a suitable menu heading.

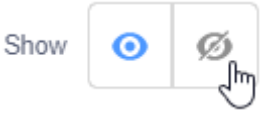
10. Create 3 new text sprites as follows

Sprite Name	Sprite Text
Menu Windows Desktops	Windows Desktops
Menu Windows Laptops	Windows Laptops
Menu Apple Desktops	Apple Desktops

11. Arrange them in suitable positions on your stage

We want this menu and its sprites to appear as soon as the "Computer Resources" options is clicked on the main menu. We will add some code to make that happen. First we will hide these ones so that they are not showing when the program starts.

12. Select the "Menu Windows Desktops" sprite and change its property to hidden.



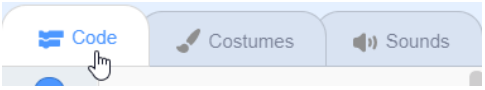
13. Do the same to the Windows Laptops and Apple Desktop sprites so that they are also hidden.
14. Run your program so that the main menu and its sprites are showing.

Getting our menu working

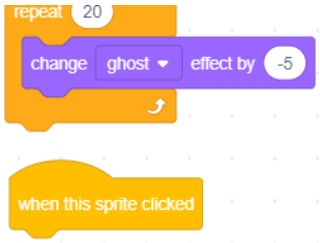
Now that we have our Windows Desktops menu working, we will add some script to the sprite on the main menu so that when it is clicked, it will show that menu. This will mean that when the **Menu Computer Resources** sprite is clicked, several things will happen.

- When then Menu Computer Resources sprite is clicked, it will put out a message to the rest of the objects in our program (called a broadcast) so that other objects are ready for whatever actions need to take places when that sprite is clicked. Those actions will include:
    - Change the backdrop from the main menu backdrop to the Computer Resources backdrop.
    - Hide the menu sprites from the main menu.
    - Display the menu sprites for the Computer Resources menu.

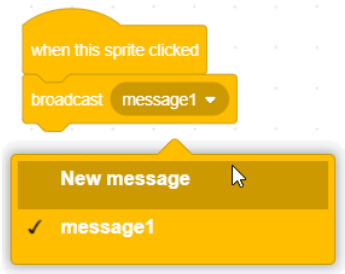
1. Select the **Menu Computer Resources** sprite in the sprites area.
2. Click the **Code** tab to ensure that code blocks are showing.



3. Go to the **Events** category and add a **when this sprite clicked** code block. It can go under or next to your existing code blocks. It doesn't really matter where you put them in the code area. As long as it's easy to see them and make changes when you need to. Try to avoid having code blocks overlapping other code blocks as that can look confusing.

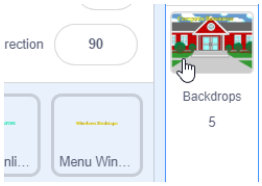


4. While still in the **Events** category, add a **broadcast** block.
5. Click on the name of the broadcast (message1) and then select **New message**. You can call the broadcast message whatever you want but it's helpful to give it a name that describes what it will be used for.



6. Enter **Show Computer Resources** for the broadcast name and click **OK**.

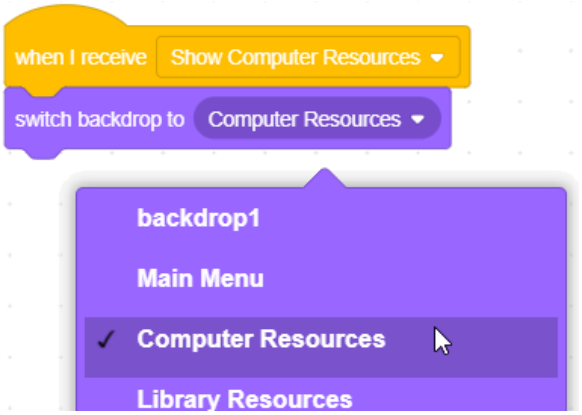
Now that we have created a broadcast, we can add code to other parts of the program that tell them what to do when they receive that broadcast. We'll start with the backdrop.



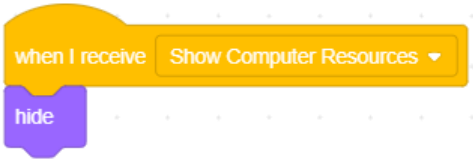
7. Select the backdrop in the sprites area.

Your backdrop already has code telling it which backdrop to show when the program begins.

8. From the **Events** category add **when I receive Show Computer Resources**.
9. From the **Looks** category add **switch backdrop to** and change the backdrop to **Computer Resources**.



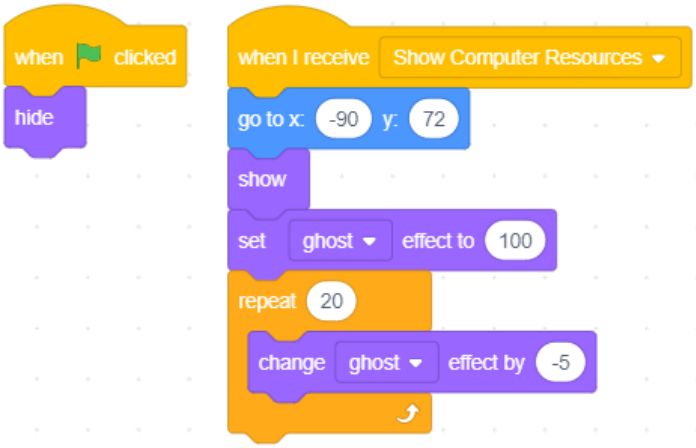
10. Select the **Menu Computer Resources** sprite.
11. Add the following code.



12. Add the same code to the other two sprites on that menu. Remember that you can drag the code on to those sprites to copy it.

Lastly we'll add code to make the menu options from the computer resources menu appear. We will also add code to ensure they are hidden when the program begins.

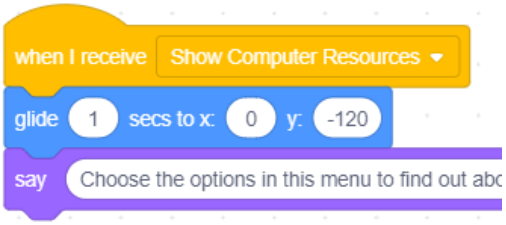
13. Select the **Menu Windows Desktops** sprite.
14. Add the following code. The **go to** position should be the position you want it to be in when it appears. When you test the program you can adjust the position if it doesn't look right.



15. Copy that same code to the **Menu Windows Laptops** and **Menu Apple Desktops** sprites, making sure each one has the right **go to** positions.
16. Start your program and then click the **Computer Resources** menu option to test it.

We'll make the cat move and give a bit of an explanation to this section of the system.

17. Move the cat sprite to a point near the bottom of the stage. The **say for 2 seconds** block should say something like "Choose the options in this menu to find out about the school's computer resources." It should show for 4 seconds.



18. Test the program again to check the cat's movement when this menu appears.

Returning to the main menu

The last thing we will do with the Computer Resources menu is add an option that will return to the main menu.

1. Create a new sprite.
2. Add the text **Main Menu** and format it.

3. Click the **Rectangle** tool. Draw a rectangle that is bigger than the text and a colour that is different from the text. We will use this as a background for the main menu button.
4. Click the **Select** tool and move the rectangle so that it covers the text. Resize it if needed.
5. With the rectangle still selected, click the Backward button at the top to move the rectangle behind the text.

This makes the sprite look more like a button. It also makes the text easier to see on the screen and makes it easier to click on. You may have noticed with your other menu sprites that they work when you click on the letters but not when you click the gaps between letters. This removes that problem.

Tip	You could add a shape behind each of your other text sprites to turn them in to similar looking buttons rather than just text alone.

6. Drag the **Main Menu** button to a suitable location on your stage. It will stay in this same position on each menu except the main menu. Change the name of the sprite to **Menu - Main**.

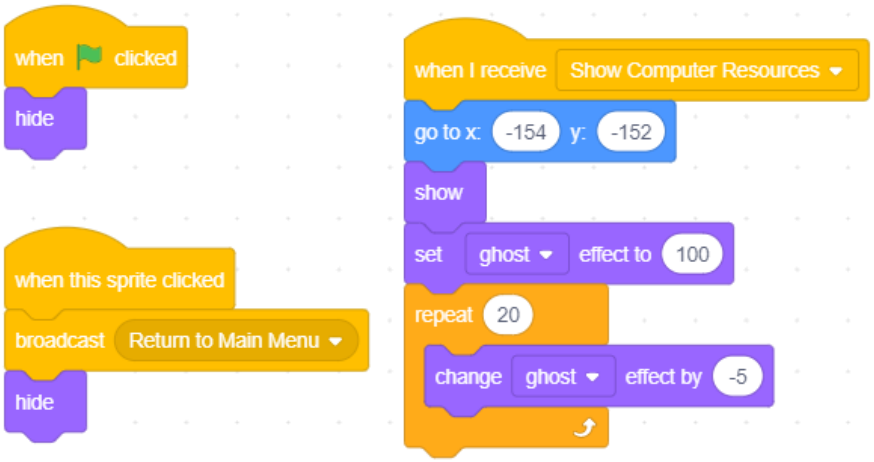
Now we will add some code that does the following:

- Hides the main menu button when the program starts
- Makes the main menu button appear when it receives the **Show Computer Resources** broadcast.

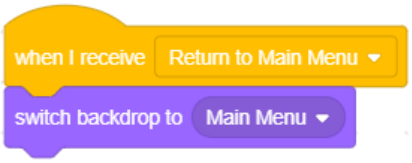
When the main menu button is clicked it will:

- Change the backdrop to the main menu
- Display all of the main menu sprites
- Hide all of the computer resources sprites

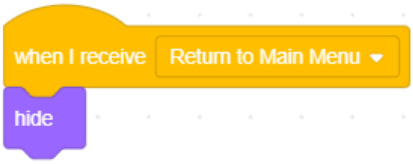
7. Make sure the **Menu - Main** sprite is still selected.
8. Add the following code.



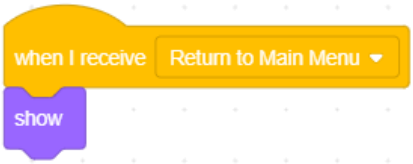
9. Add the following code to the **Backdrop**.



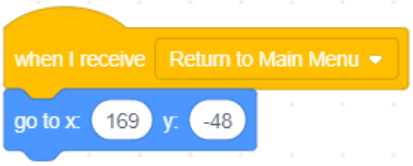
10. Add the following code to the **Menu Windows Desktops**, **Menu Windows Laptops** and **Menu Apple Desktops** sprites.



11. Add the following code to the **Menu Computer Resources**, **Menu Library Resources** and **Menu Online Resources** sprites.



12. Add the following code to the cat sprite. Make sure the coordinates are the same as the glide to coordinates in the **when**  **clicked** block.



13. Click the  to start the program to test it.
14. Click the **Computer Resources** option to move to that menu.
15. Click the **Main Menu** button to return to the main menu.

Create the remaining menus

You can see that creating menus is mostly a case of hiding and displaying different sprites when a menu option is clicked.

16. Use what you have learned to do the following.

- Create text menu sprites for the **Library Resources** menu and the **Online Resources** menus.
- Add code so that when the **Library Resources** option or the **Online Resources** option are clicked on the main menu it will display the correct menu with the correct sprites being displayed and hidden.

When you are finished you should have a complete menu system. All that would remain now if we were to complete the system is to add code for when each individual menu option is clicked.