

Lab No 10: Scratch Basis

Objective:

- To provide the interactive tool for the beginner to learn the programming just like playing games.

CLO's:

- CLO 02

What is Scratch:

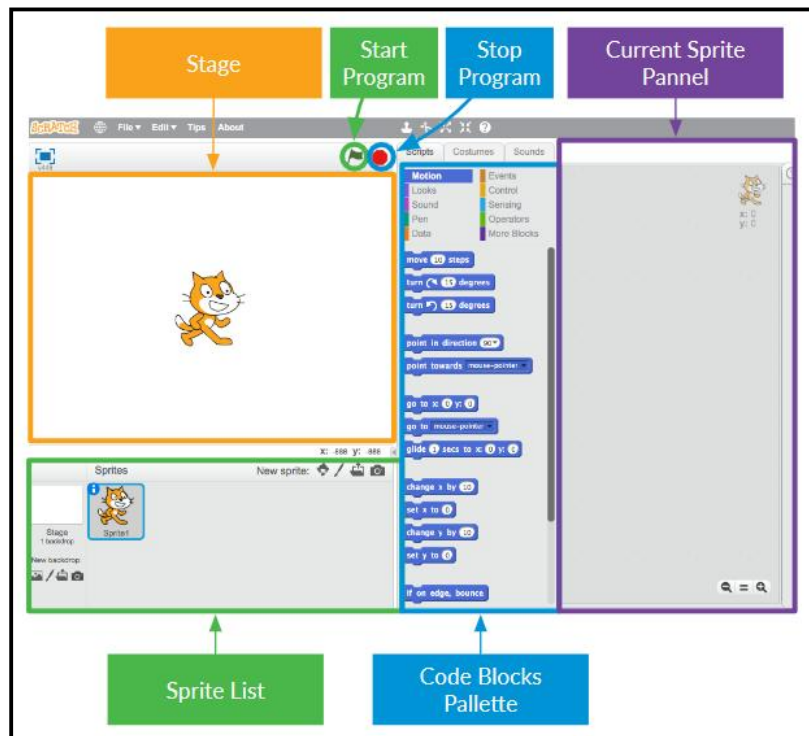
Scratch is a programming language that makes it easy to create your own interactive stories, games, and animations – and share your creations with others on the web. Scratch is developed by the Lifelong Kindergarten research group at the MIT Media Lab (<http://ilk.media.mit.edu>). Our group develops new technologies that, in the spirit of the blocks and finger paint of kindergarten, expand the range of what people can design, create, and learn.

1. These Sushi Cards are going to help you learn to create computer programs in **Scratch**. To do this, you're going to need to have Scratch!

You can either download it and install it on your computer by going to dojo.soy/ download scratch or use it in your web browser by going to dojo.soy/usescratch. Do one of these two and once you're set up, move on to the next step.

2. Once you sign-in to the online Scratch, or load it on your computer, you'll see a screen like the one below.

This screen has a few parts you'll need to remember. They are labelled in the picture below.



3. There are a few words from that picture that need explaining:

The Stage

This is where your Scratch Programs run. It has

- One or more backgrounds (images on the background of the screen)
- Any code blocks associated with it (we'll get to this)

Sprites

Any object that goes on the stage is a sprite. In Scratch, the sprite is:

- The image on the stage
- Any alternative costumes (looks) it has
- Any sounds associated with it
- Any code blocks associated with it

The current sprite is the one that's selected in the sprite list.

Code Blocks

Code in Scratch comes as blocks, that you click together to make programs. You choose blocks from the code blocks palette and drag them into the current sprite panel, then click them together.

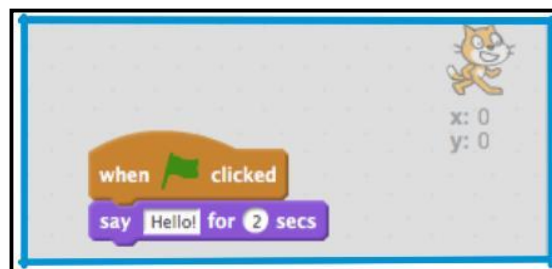
There are 10 categories of blocks, which are color coded, and you can select each category from the list at the top of the code blocks palette.

4. Time to get coding! Go back to Scratch and click on the Scratch Cat in the sprite list. The cat is now the current sprite.

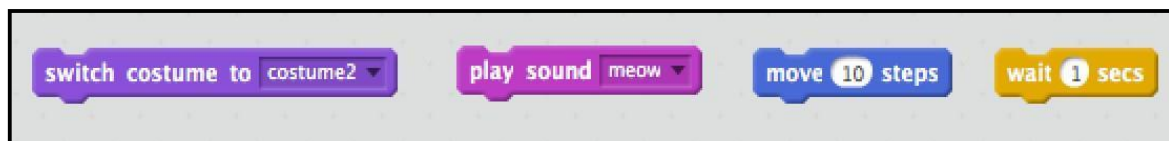
Go to the Events category in the code blocks palette and choose the "when [flag symbol] clicked" event, then drag it onto the current sprite panel.

Then go to Looks in the code blocks palette and choose the "say [Hello!] for [2] secs" block and drag it onto the current sprite panel, connecting it to the end of the other block, like this:

Now click the Start Program button and watch what happens!

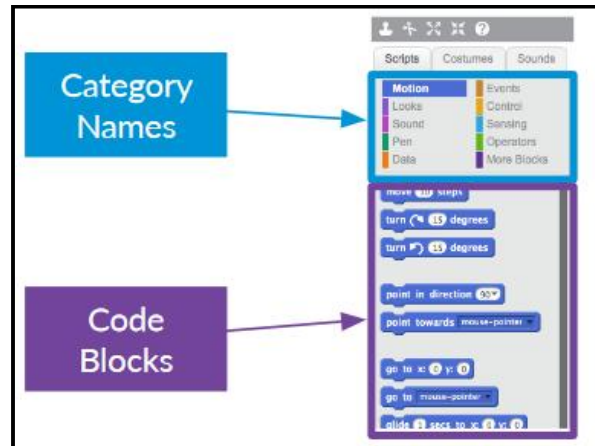


1. Great! You've written your first Scratch program. Time to learn a little more about getting code in and out of Scratch! Scratch code is made up of blocks that you snap together to make programs.



These blocks come from the Code Blocks Pallet where they are broken up into different categories. By clicking on the category names, you can see the blocks in that category. Here, the motion category is selected.

All of the blocks in the selected category are shown in a list. You can pick the one you want, click on it and hold down the mouse button, then just drag it onto the current sprite panel and let go.



2. Once the block is in the current sprite pane you can move it around and snap it to other blocks. If you want to see what a block does, you can double-click on it and it will run! Normally, you want your blocks to run automatically, when something happens. This is why most of your programs will start with a block from the events category. Most often, it will be this one: The code blocks connected to this block will run after the green flag is clicked



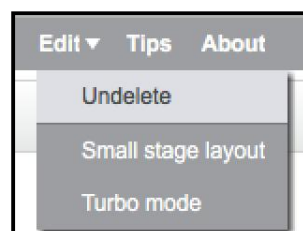
3. Code blocks run from top to bottom, so the order you snap your code together in matters. In this example, the sprite will say "Hello!" before it will play the meow sound.



4. Removing, or deleting, code blocks you don't want in your program is easy! You just have to click on the scissors icon and then click on the code you want to remove.



Be careful though! You will delete all the blocks connected to the one you click on. If you do this by accident and want to get your code back, you can use the edit menu and click on undelete. Try adding, deleting, and undeleting some code blocks now!



5. Now you know how to move code around and make things happen, time to try a simple program: Making the Scratch Cat walk in a circle!



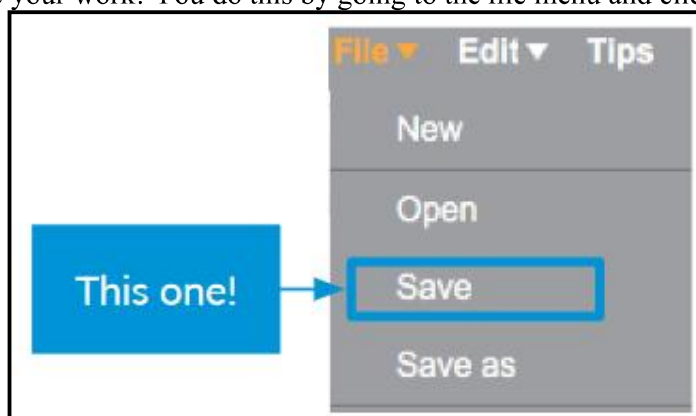
Make sure you have the cat selected in the sprite list and then drag these blocks together. You'll find them in events and motion. Then, click on the green flag above the stage. If you click too many times and the cat walks away, you can drag it back!



That's a cat walking in a straight line... not exactly what you want. Snap this block to the end to make it walk in a circle. It's in motion too. This block makes the cat turn 15 degrees of the full 360 degrees that make up a circle. You can change that number, and the number of steps, by clicking on the number and typing a new value.



6. Now save your work! You do this by going to the file menu and clicking save.



1. Now your scratch cat moves, but wouldn't it be more fun to control it with the arrow keys? On this card, you're going to learn how to do that! As you've probably guessed, you're going to need event and motion blocks again! This time, you're looking for this block:



If you click the little arrow (▼) beside “space” you’ll get a list of all the keys on your keyboard to pick from. You’re going to need four of these blocks and you can connect them to motion blocks like this:



2. Your cat moves back and forwards now, which is pretty cool, but it doesn’t move up or down. Also, if you look through the motion blocks, you’ll see there are no blocks for “up” or “down”. There are a whole bunch of them related to x and y though.

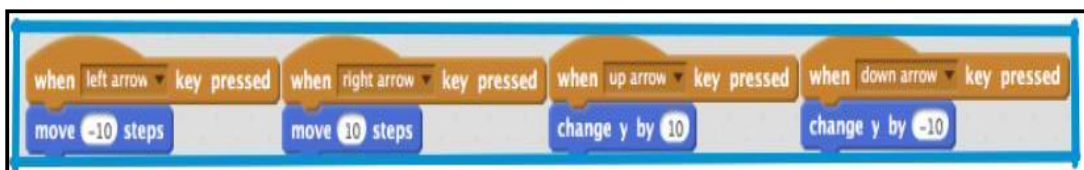
When programmers need to talk about the positions of objects, like sprites, we often use x and y coordinates to describe them. The x-axis runs from left to right, while the y-axis runs from bottom to top.

A sprite can be located by the coordinates of its Centre, written something like (15, -27) where 15 is the x position and -27 is the y position.

To get a feel for how this actually works, take the go to block from motion and drop it onto your sprite panel. You don’t need to connect it to anything.



3. Now you know about x and y coordinates, you can make the cat move up and down! You just need to change its y value. You can update your code like this:

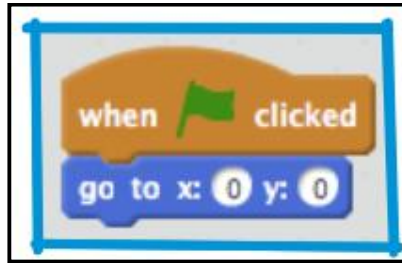


Now when the arrows are pressed, the cat can move all over the stage!

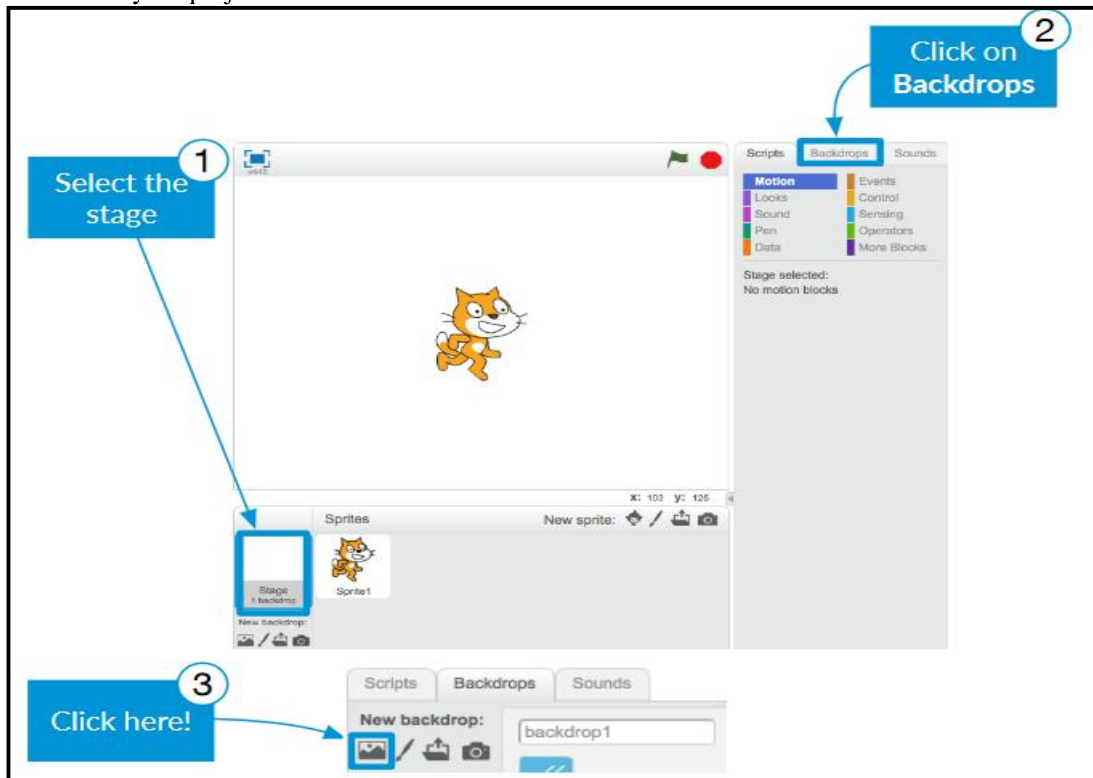
4. The cat moves all over the screen, but imagine this is a game: How do you restart it? You need to get the cat back to its original location when the player starts the game.

In Scratch, they start the game by clicking on the green flag, so you need to change the cat’s x and y coordinates when that happens.

That’s actually pretty easy! The Centre of the stage is (0,0) in x and y coordinates. So all you need is an event block for that green flag and the go to block you’re already using. Set the go to (0,0) and then snap it to the flag event block.

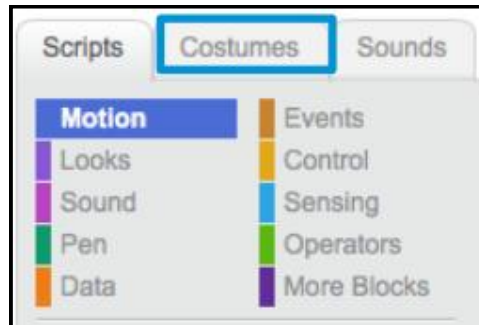


1. So your cat moves up and down, as well as left to right. You should probably put it somewhere sensible for that... like under water! Scratch has a library of backgrounds and sprites for you to use to make your project look awesome.

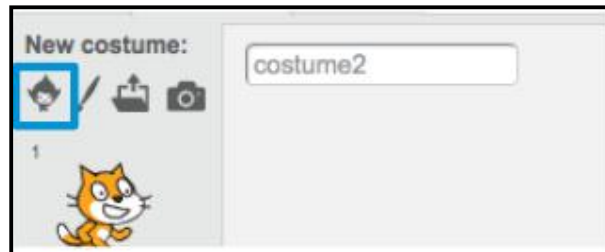


Then pick your favorite underwater background!

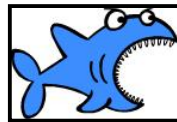
2. Great! Of course, now you have a cat underwater. Cats aren't usually big fans of that, but you can fix that. You can turn the cat into a shark!
First, select the cat and click on the Costumes tab



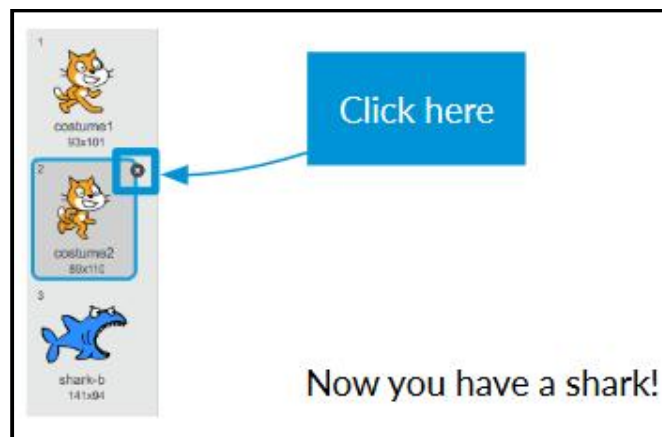
3. Then, click on Choose Costume from Library



4. Pick this shark



5. Now remove the cat costumes by selecting each of them and clicking on the x.



1. Now you have a shark that moves about underwater. Nice! Time to add some fish for it to catch!
Click the new sprite button and then choose a fish from the screen that opens.



That fish is a bit big compared to your shark though. Use grow and shrink to make it the right size.



Click on grow or shrink and then click on the fish to make it get bigger or smaller.

2. Nice! Later, you're going to add some code to make the fish move around on its own, without help from the player. Your player will be the shark, trying to catch the fish. However, it does look a little funny to have that shark swimming backwards. Just like you'd usually turn around rather than walking backwards, the shark would turn around rather than swimming that way. Luckily for you, Scratch has a block for this!

The point in direction block lets you pick the direction your sprite is pointing in. You can type in any number, but it comes with the four you'll need most already in there: up, down, left and right. You'll find it in the motion blocks section. Grab it and snap a couple of them into your shark's code, like this:

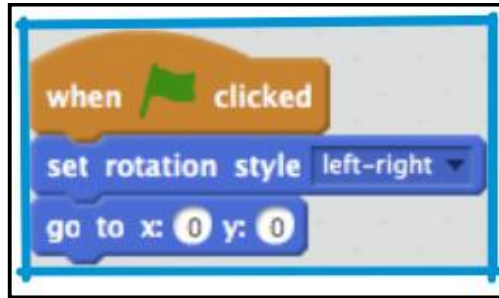


3. If you tried moving the shark around after you added the "point in direction" blocks, you might have noticed something a little strange happening. The shark may not be turning quite right!



The problem here is that the shark sprite started, as all sprites do, with the "all around" rotation style, and what you need it to have is the left-right style.

As usual, there's a block for that and it's in motion! You just need to update your reset code from Card 3 to set the rotation style, like this:



1. Ok, now it's time to make the fish swim on its own. To do this, you're going to need a new kind of block: a control block. These let you do things a certain number of times, or under certain conditions.
Select your fish sprite and drag a "when green flag clicked" event block, a "forever" control block and a "move 10 steps" motion block into the script panel like this:



The fish does whatever is in the "forever" block over and over again, forever. Once it has reached the end it goes back to the top of the block and starts again. Now click the green flag and watch what happens!

2. Well, that fish just crashed into the side of the stage, and it was moving far too fast for your shark to catch. First, you need to slow it down. That's actually pretty easy, you just need it to wait for a little while later it moves those 10 steps.
There's a control block that can help you here:

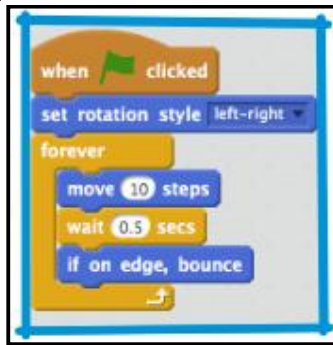


You can set how many seconds you want the fish to wait. For now, try half a second (0.5). You can test out different values later, to see which is the best for the game. Remember you can change the number of steps too!

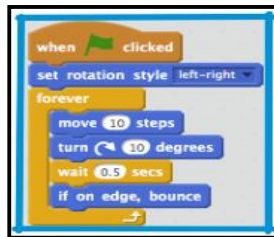


3. The fish moves now, but you need it to bounce off the edge too. Yet again, there's a motion block for this! It's the "if touching edge bounce" block. It checks if the sprite is touching the edge and, if it is, turns left, right, up or down as appropriate. Of course, this will lead to an upside-down fish, so

you need “set rotation style” again.



4. The fish moves back and forward now, but only in a straight line. That’s going to be a bit too easy for the player to catch with the shark. You need to make the fish swim more unpredictably. You already know from Card 2 how to make a sprite turn, so start there: Add a turn into the fish’s swimming and click the green flag.



5. It’s better, but there’s still too much of a pattern. It needs to be more random. Luckily, Scratch can do random for you! You’ll just need a new kind of block, called an operator block. Operators take in one or more values (numbers, text, true/false values) and give back a single value. You can tell the kind of value it will give back by the shape of the block: round ends give numbers or text; pointy ends give true/false.



You need the “pick random” operator block, and you need to plug it into the “turn degrees” motion block by clicking and dragging it into the field where you set the number of degrees. You can change the minimum and maximum numbers it will pick, but the default values (1 and 10) are pretty good for this game, so you can just leave them.

Update the fish code to this and then run it by clicking the green flag:

