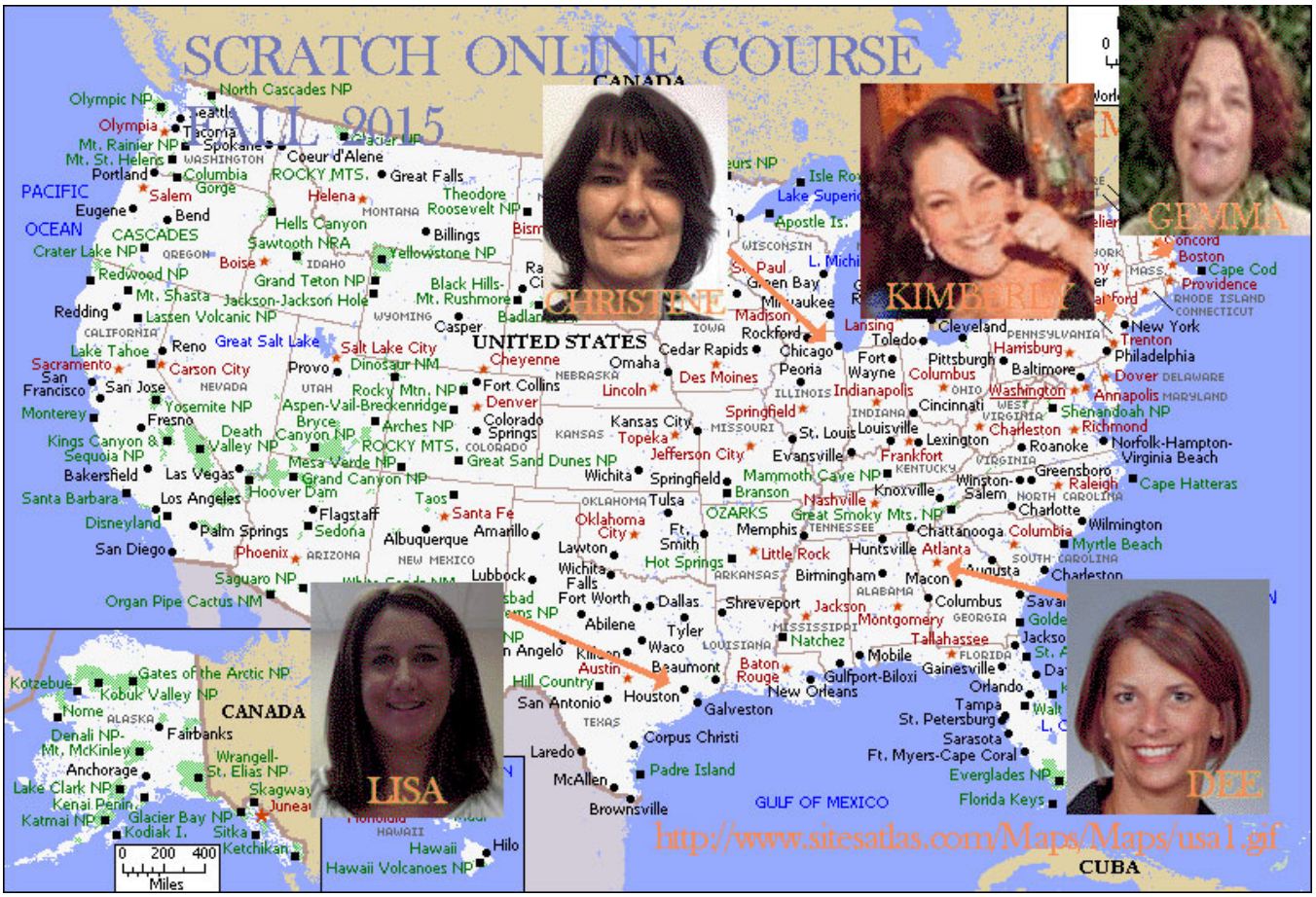
Scratch • Lesson 2 page 1 • Oct 5

All lessons stored at http://teachingcompany.com/F15

Who's Who in our Scratch Online Course:

Our shared document is at http://tinyurl.com/f15coding
HW 2.11 on page 16 involves this document

Our web page with assignments is http://teachingcompany.com/f15



Reminders
Lesson 3 will be on Monday 10/19 two weeks from now.
Please send me an email when you want me to check your work for the week and give you feedback.

Scratch • Lesson 2 page 2 • Oct 5

All lessons stored at http://teachingcompany.com/F15

Monday 10/5

Hi Chris, Dee, Gemma, Kimberly and Lisa,
Hope you had a good first week in our course.

We begin Lesson 2 with an important program that I have nicknamed The Big Kahuna. Please watch this first 6 min video at http://youtu.be/EPQ-QnlsZ5k and study it thoroughly, making it your own. This might mean watching it twice!

I would like you to be able to add up the numbers from 1 to any specified number and re-create this program without notes. Yes, there was a 1999 movie called The Big Kahuna and one source on Wikipedia explains Kahuna as a "*Hawaiian word, defined as a priest, sorcerer, magician, wizard, minister, expert in any profession*."

The C variable is the COUNTER. It is set to ZERO to begin and each time thru the loop it increases by 1.

The T variable is doing the running TOTAL. Each time thru the loop the current value of T is increased by whatever C is. So after 3 times thru the loop C now becomes 4 and that is added to the current value of T which is 6 (1+2+3).

The SAY statement contains the fancy JOIN command which we will get to later in lesson 2.

Got it? C for counter and T for total!
Repeat loop for all the action and a HEADER where the variables are initialized.

HW 2.1 REQUIRED
CHOOSE one of these two challenges (2.1A or 2.1B) and SHARE IT

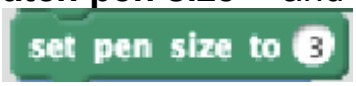
Using my program as a sample, modify it so that it finds the sum of the first 10 odd numbers (1,3,5 thru 19). Your program will look like mine above but what should C start with? And do you change C by 1? By the way, the first 10 odd numbers add up to 100 which is 10 squared. I wonder if that is a coincidence ;-)
If you do this one, please name your program "Exercise 2.1A Sum of First 10 Odd Numbers" and share it.

or choose this one HW 2.1B
Using my program as a sample, modify it so that it finds the sum of the numbers from 17 to 49 (17, 18, 19 thru 49). Your program will look like mine above but what should C start with? And do you change **repeat 100** to something else? By the way, the 33 numbers from 17 to 49 add up to 33*33 or 1089. Maybe that is why I like 33 so much!
If you do this one, please name your program "Exercise 2.1B Sum of the Odd Numbers from 17 to 49" and share it.

Next, we build on the commands from Lesson 1 and add a few!

Here are four commands from Lesson 1 most of which should trigger some recognition from you before we begin doing new things. Please make sure you are good to go with all four. Reminder that SHOW means SHOW THE COSTUME which might mean the CAT and/or the SAY or THINK statements.

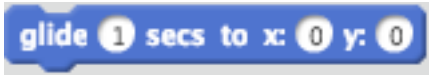
This next command is useful but we did not talk about it much. The pen thickness of 3 is a little better visually on the screen. The default size is 1. The pen gets insanely thick as you go to size 10 or 20 or beyond. Whenever you forget this info, you can just google these 3 words -- **scratch pen size** -- and the info is right there!



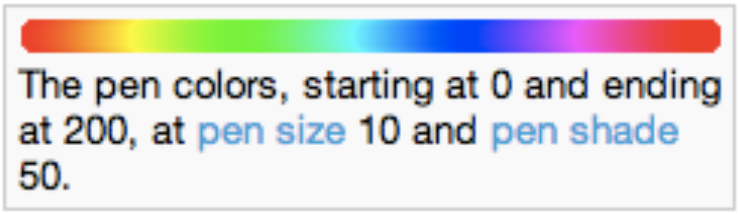
This next command is also useful. It makes the CAT face a certain direction where 0 degrees means FACING UP and 90 degrees means FACING RIGHT.

Here is another one we did not talk about much. It makes the CAT move to a specific location on the screen. Remember that it will LEAVE A TRAIL unless the pen is up.

Another powerful one we did not talk about at all makes the CAT travel to a specific location on the screen and lets you specify the time it takes which can be a decimal as .5 or .1 -- wonderful command! You will love GLIDE if you have not used it yet.

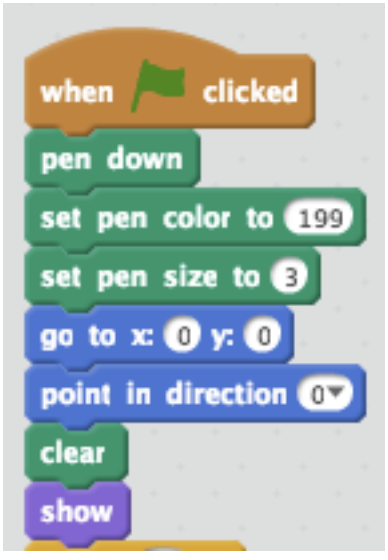


Finally, I have told many of you to use SET PEN COLOR to a NUMBER and not the command that SETS to a COLOR. The color chart for each number from 0 to 200 is below on the right. Whenever you forget this info, you can just google these 3 words -- **scratch pen color** -- and the info is right there!

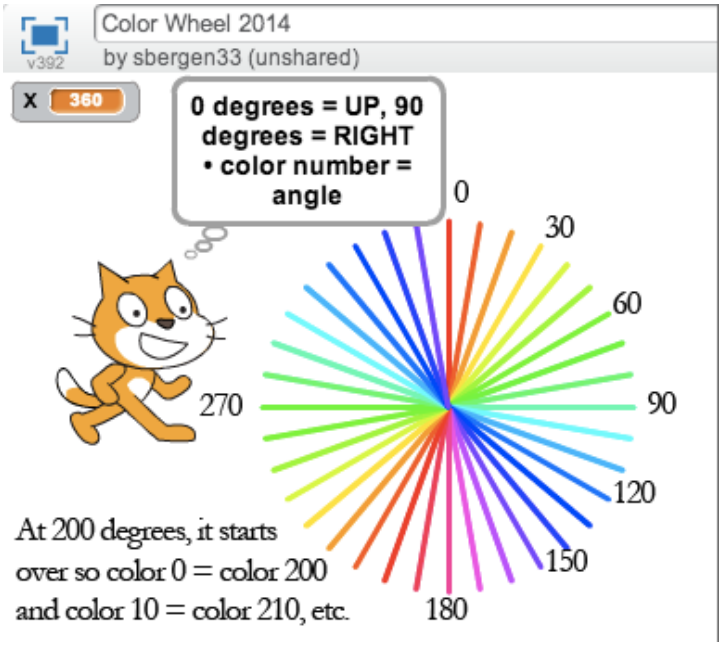


Some of the above commands are so common that I have pulled them together into this mini program that I call my STARTER SET which is on the next page I keep this in one program in my library and then load it up. I change the name with SAVE A COPY and begin programming each time I am doing a project. Feel free to grab this program from my library (sbergen33) and REMIX so you can save it as your own. REMIX is an important button once you grab a program that someone else has shared. REMIX moves the program into your own library.

HW 2.15 **Second video to watch:** http://youtu.be/HRxh2yLNm8Y
This 3 minute video explains one way of using my STARTER SET efficiently at the top of every program. *The video refers to "Lesson 3 template" but this simply reflects an anomaly from when this was used in a previous course*. You will find the file in my library as **2014 Lesson 2.0 Template** or **2014 Lesson 3.0 Template**.
Here is the direct URL -- http://scratch.mit.edu/projects/16772661 -- for you.

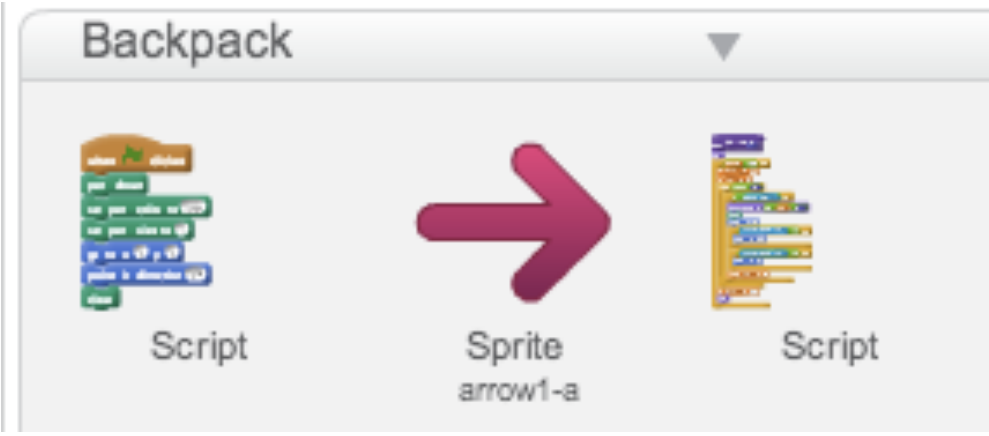


By the way, here is a color wheel to help you visualize the color numbers better. I also put a link to this color wheel on our shared collaborative doc.



An alternative to my STARTER SET is to use the BACKPACK

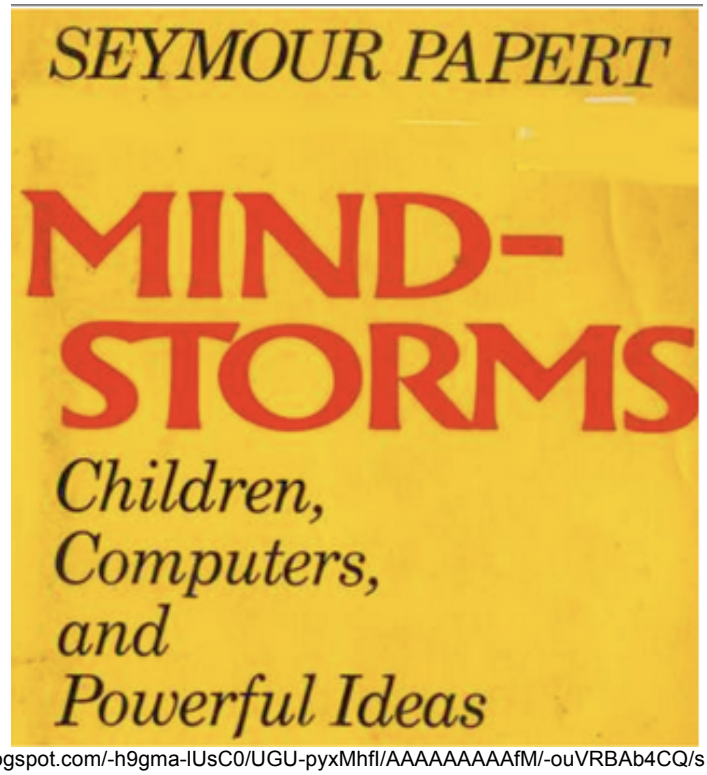
A different way to do this involves the BACKPACK. After you get my code or you change it (since you might prefer for the CAT or ARROW to face right) you then drag the code to your BACKPACK located at the bottom of the SCRATCH screen. This is also shown in one of the subsequent videos. With this technique, you don't get the arrow each time however, but you can also store the ARROW in your backpack.



You can open or close your Backpack with the triangle button.

You can delete items from your Backpack by clicking with the right mouse button (or control-click on a Mac).

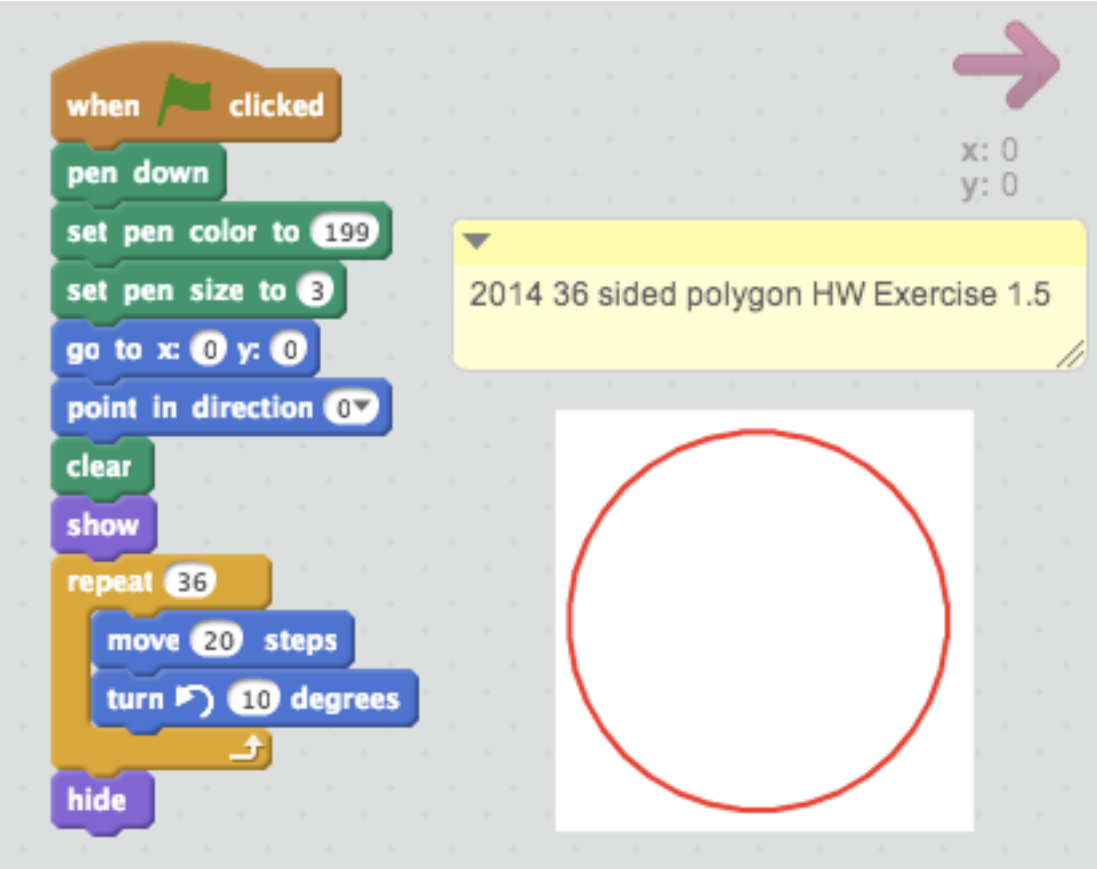
Pedagogical Side Bar: In teaching coding, it is really important for you as teacher to create an environment when everyone can be successful. The reason that I advocate having students draw LETTERS of the alphabet is that it creates an environment for success and allows for differentiated learning and levels to use the popular phrase. Seymour Paper, creator of Logo and author of the book MindStorms in the 1970s talked about programming as both "body syntonic" and "ego syntonic." By ***body syntonic*** *he explained that when you program the cat or arrow to draw a shape, you become the cat or arrow and you turn your head or body as you think through how to draw a shape. By* ***ego syntonic****, he explained that you identify with the cat or the arrow and you get a tremendous sense of personal satisfaction and gratification when you accomplish the challenge that you attempted.*



from http://1.bp.blogspot.com/-h9gma-lUsC0/UGU-pyxMhfI/AAAAAAAAAfM/-ouVRBAb4CQ/s1600/Picture+2.png

Here is one solution to HW Exercise 1.5 from last week.

HW Exercise 1.5 -- Draw a polygon with 36 sides.
You will need to keep the length of each side small.



HW 2.16 VIDEO 3: Please watch the 7 minute video called Backpack, Comments and Other Tricks http://youtu.be/3iWRz-KyLVY
in which I explain a few other tricks as well as
• getting my first 7 STARTER commands and using your BACKPACK
• the yellow COMMENT feature for notes of your choice

One goal of Lesson 2 is to gain more skills about variables that we just got a taste of last week. I also want to build on the ASK/ANSWER dialogue that we just began last week. Finally I want to introduce you to RANDOM NUMBERS.

Next required HW problem for Lesson 2 is to choose HW 2.2A or 2.2B

HW 2.2A Modify the 36 sided polygon which really draws a circle. Change it to draw small circles or large circles allowing the user to specify the size of the circle. Name your program Exercise 2.2A VARIABLE CIRCLES and share it.

HW 2.2B Modify the 36 sided polygon which really draws a circle. Change it to draw a fraction of a circle allowing the user to specify semicircle or quarter circle. Or allow the user to give a number that specifies what fraction of a circle he/she wants. You decide on the user interface dialogue which of course affects the coding. Name your program Exercise 2.2B PARTS OF A CIRCLE and share it.

Both programs above require the use of

and the

variable. Reminder that the result of the ASK command puts the info typed into the VARIABLE made by Scratch called ANSWER. The checkmark to the left of ANSWER either displays ON or OFF the value of ANSWER on the Scratch stage.

Next, I would like you to have the skill to take any program that MAKES A LETTER and be able to CONVERT it to a program that make the letter variable. You can pick the letter you created in assignment 1. Let us do a letter such as T and then a more complicated one with a curve.

HW 2.25 VIDEO 4 to watch on Programming Variable Sized T (9 min)
http://youtu.be/a69EtSwMAdQ

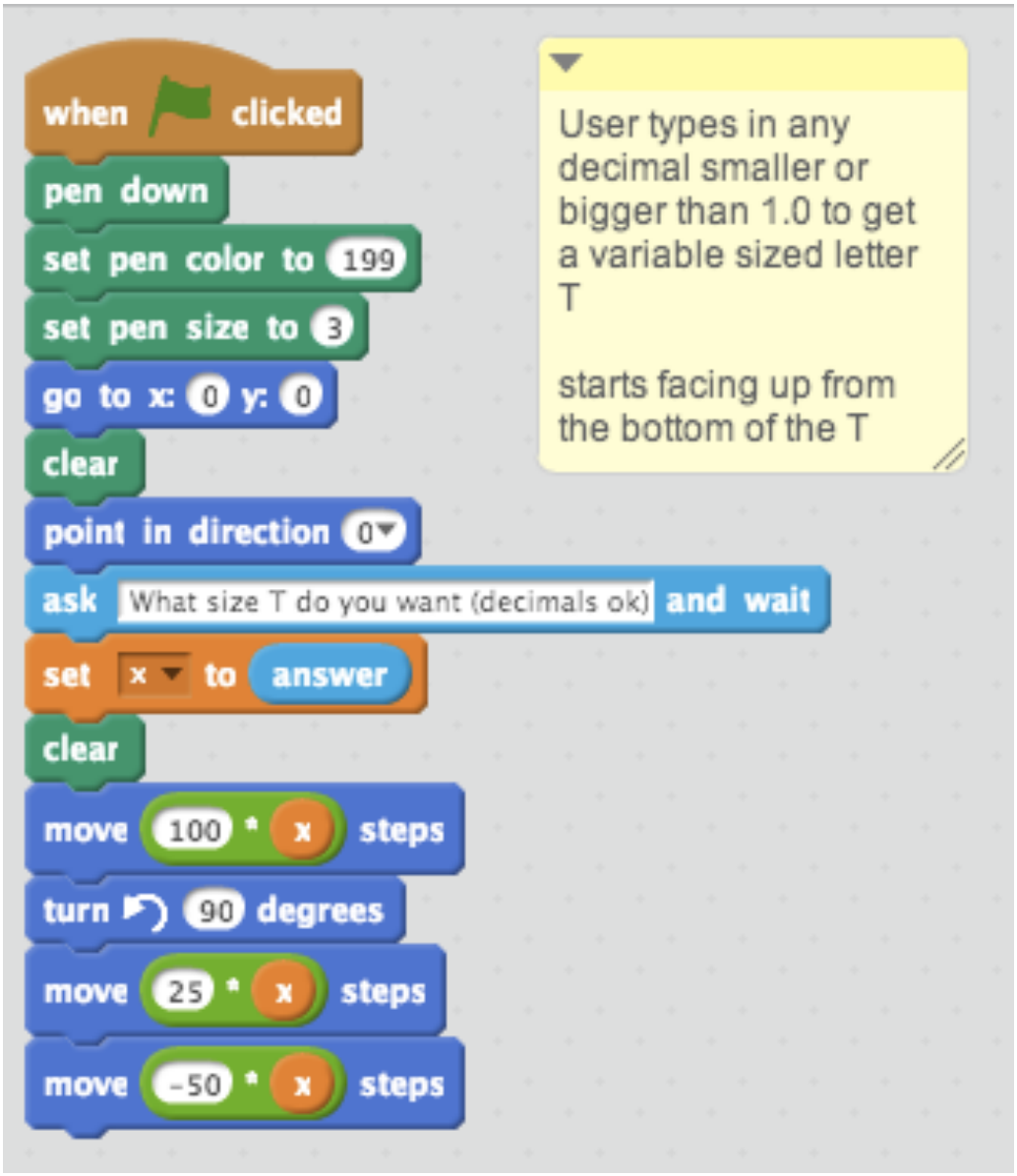
To add a variable we need to create one using the DATA category. Then we will need to multiply each of the numbers by that variable -- the 100, the 25 and the -25. To create a variable we go to DATA and choose to MAKE A VARIABLE. As a math teacher, I am choosing "x" but you can choose "size" instead. Reminder that once you do so, you get 4 new and free commands

The checkmark means to display the variable on the left side of the stage (where the cat is) and it is the same as clicking the command SHOW VARIABLE. The first two on the left are very important and we will use frequently. The first one called SET assigns a number to the variable (because you don't always want to bother the user to type it in). The second one called CHANGE will be used when we learn to add the numbers from 1 to 100.

HW 2.3 Pick any letter you created in Lesson 1 and modify it so that it makes that letter to be variable sized. Name your program Exercise 2.3 VARIABLE SIZED LETTER and share it.

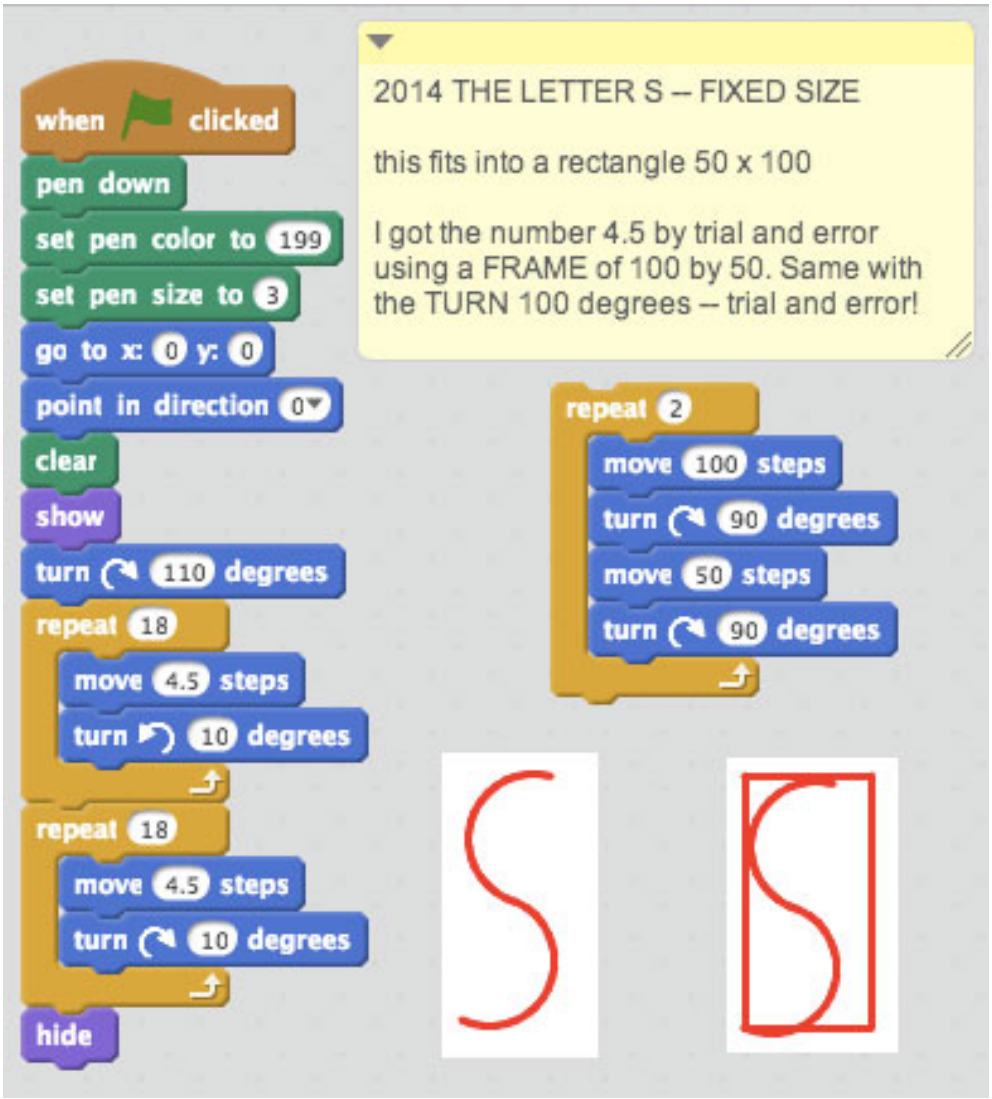
Below you will see my code for making a "T" to be variable sized. Note that I use the lime green OPERATORS component that has two circles multiplied. You will see the manipulations and explanations on the video.

Note: if you used the GO TO command in Lesson 1 to draw the uppercase letter, you will need to modify it first before you start 2.3. Why? Because a command like GO TO 40, 100 references an exact location on the screen and that makes it impossible or at least very complicated to re-program as a variable sized letter.



Now you need to do the same for the letter S that I have written. You can find this program in my library sbergen33. The program is called 2014 THE LETTER S -- FIXED SIZE because you do not want to have to retype it.

Here is the direct URL -- http://scratch.mit.edu/projects/16722994 -- for you.

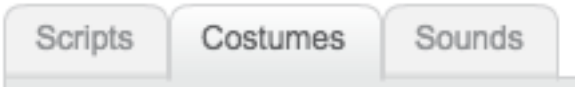


HW 2.4 OPTIONAL Modify the above program (or you can use your own "S" if you made one) so that it makes the letter S of variable size. After you get my program called 2014 THE LETTER S -- FIXED SIZE and REMIX it and do the work, change its name to be Exercise 2.4 VARIABLE SIZED S and share it.

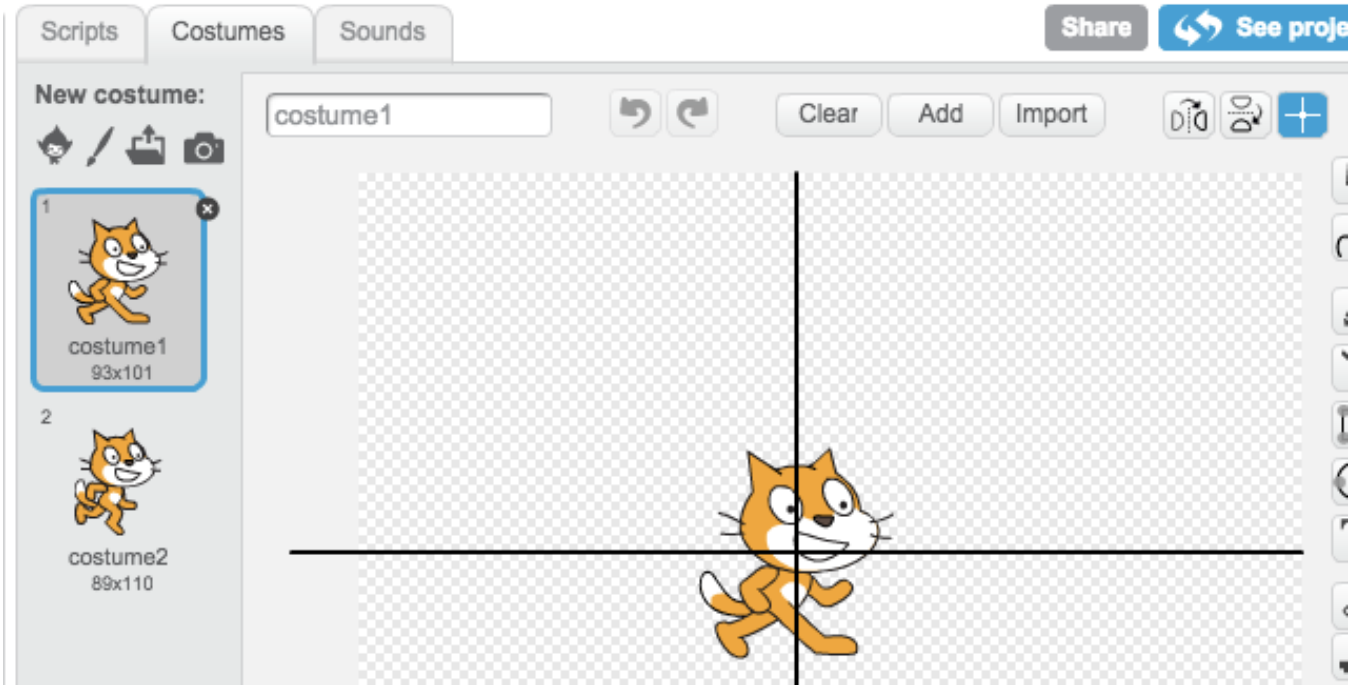
HW 2.5 OPTIONAL Pick any 3 letters of the alphabet -- perhaps your initials or any 3 letter word like CAT or THE or HAM or TEA. Write a program to make this word be on the screen using the structure of each letter being approximately 100 tall and 50 wide. Make sure there is a bit of space between the letters. Name your program Exercise 2.5 THREE LETTER WORD and share it.

Next topic -- Setting the Zero Point of a Graphic

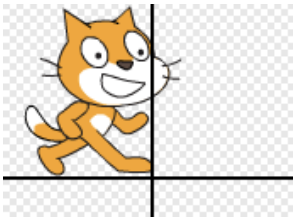
We saw in lesson 1 how the ink comes out of the back of the cat's neck. What I want to show you in this lesson involves how you adjust the zero point of a graphic.



When you click on COSTUMES at the top of the screen -- right between SCRIPTS and SOUNDS -- you can change costumes or use the graphic editor to modify the existing graphic.



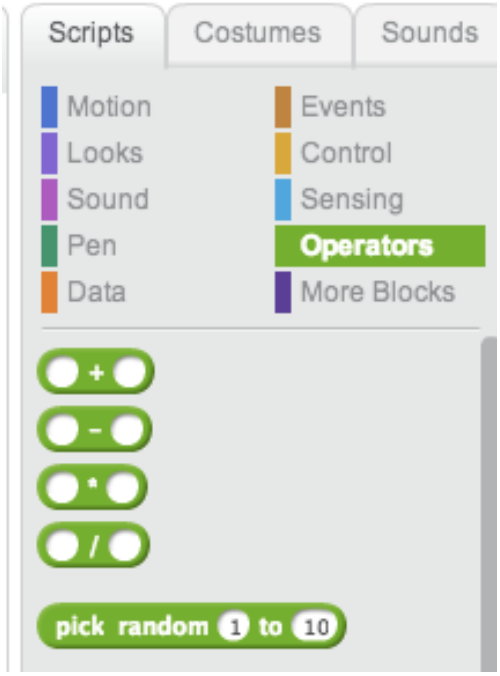
Look at the upper right corner where you see the BLUE PLUS SIGN. This icon allows you to see where the ZERO POINT currently is or you can change it. By zero point, I am referring to the location on the graphic where the ink comes out. You can pick up the zero point (what math teachers call the origin) and move it anywhere you want on the screen, e.g. to the cat's toe like this



Next topic -- An Introduction to Random Numbers

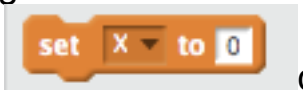
Finally, we introduce the RANDOM command and build on all the ways we can use it with graphics and variables. You will find the random command as the fifth command in the lime green section called OPERATORS.

Please note that this OPERATORS section contains options for adding, subtracting, multiplying and dividing two things. This is the key for making one variable be based on another as you will see in the video(s).



All of these commands in lime green can be put into the "white rectangle"

of the orange SET command as some of you have discovered already. When you grab a lime green operator and drag it into the

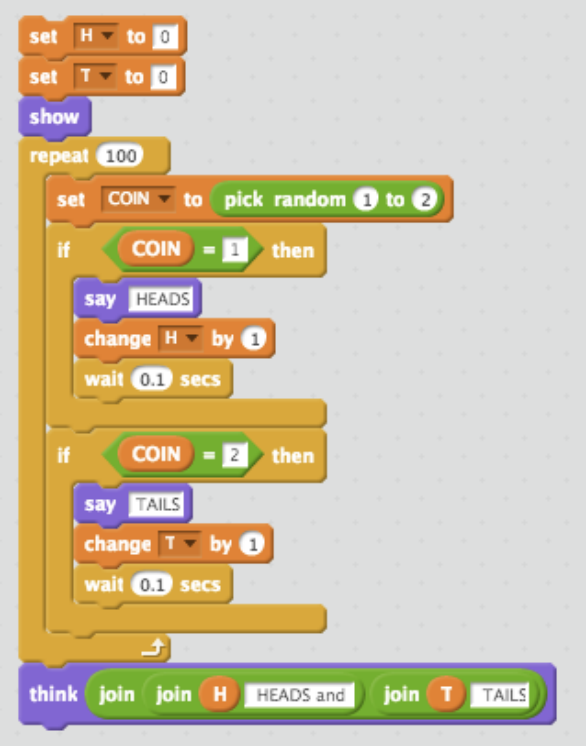


command,
you use the LEFT EDGE of the lime green object as the piece you drop.

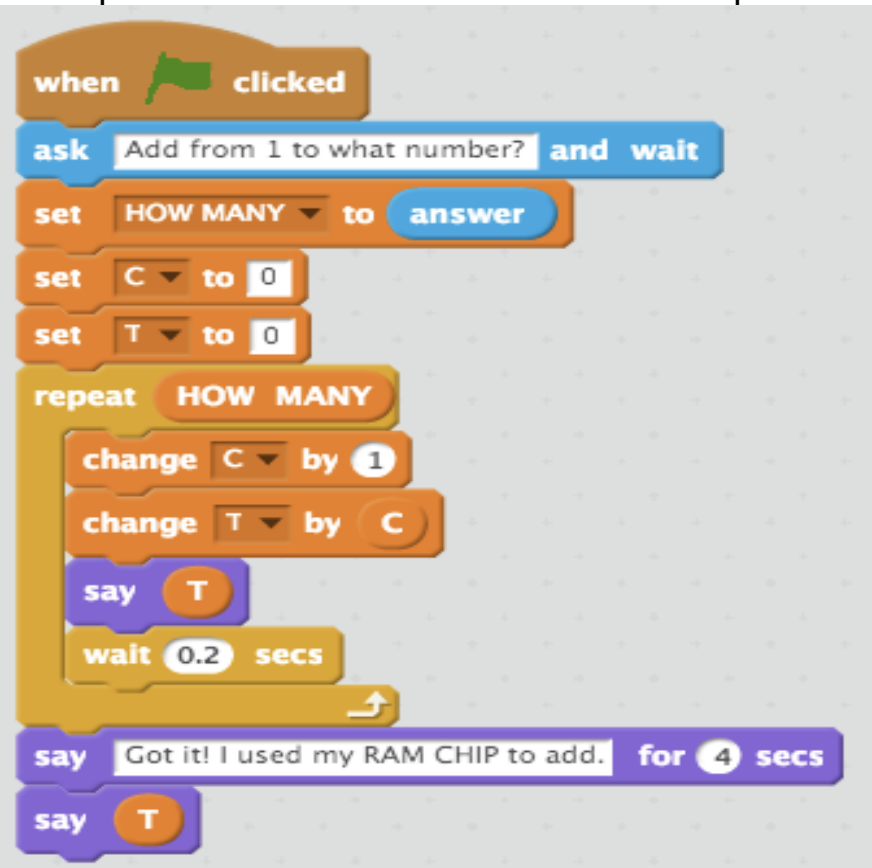
So let us write a program to FLIP A COIN 100 times and count up how many heads and how many tails.

What are the elements?
Repeat 100 times? -- we need a REPEAT loop
Random number? -- we need the RANDOM command
Visual? We will keep it simple and just display the word Head or Tail on the screen.
Next week, we will get fancier.

VIDEO 5
http://youtu.be/fsc9A19kavs Flip Coins
This 8 minute video covers
• a first example of the random number command
• the use of the JOIN command
• the use of COUNTERs to keep track of things



We end by using variables to extend the BIG KAHUNA problem to now add up the numbers from 1 to a specified number. Scratch is a real computer language!



Above is the program, Lesson 2.3 Add from 1 to Specified Number this time with two variables C and T that show you how to add up the numbers from 1 to any specified number. C is the counter and T is the running total. The third variable called HOW MANY is assigned by the user from the ASK command using the same technique as before with the important temporary transfer variable called "answer" in lowercase which is created by the system (not us).

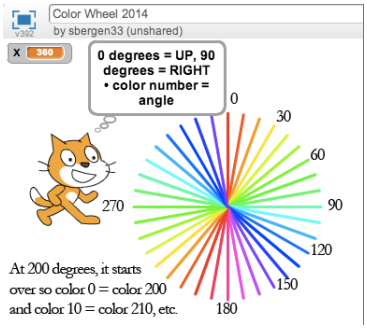
The important two new steps involve the word CHANGE. When we say CHANGE C BY 1, we are using a classic programming technique to COUNT. When we say CHANGE T BY C, we are using a classic programming technique to SUBTOTAL.

Believe it or not, I have told hundreds of students and adults to memorize these two techniques, study them and make them yours ever since 1974 when I started teaching programming in BASIC in Charlottesville Virginia. I was a math teacher and graduated college in 1973. In my first year of teaching, the computer teacher told me that he was going to take kids on the Wilderness Survival elective in the Blue Ridge mountains and that he asked me to teach the course!

That's it for this week. If you find everything above relatively easy and have extra time, below there are optional challenges.

REQUIRED -- choose one below. Name in our standard way. Share it.
OPTIONAL? Do more than one!
But if you are spending more than two hours per week, I am not responsible for your health or for any time implication to your regular job or away from your family. Yes programming can be addictive ;-)
HW 2.6 Use the variable sized hexagon and the random number command to make a tessellation of hexagons (a popular pattern of bathroom tiles).

HW 2.7 Compute the interest on $100 deposited in a bank at 7% interest for 13 years.

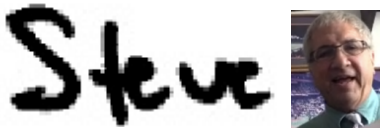


HW 2.8 Write the code to create the COLOR WHEEL.

HW 2.9 Modify your code in HW 2.5 so that your THREE LETTER WORD is variable sized.

HW 2.10 Write a program to figure out if 1001 is a prime number or not. More challenging? Write a program to ask the user for any number and then the computer figures out if prime or not. Most challenging? Write a program to find all the prime numbers under 1000.
Samples to help you? See https://scratch.mit.edu/projects/55668018/
or see https://scratch.mit.edu/projects/55669190

☑ HW 2.11 REQUIRED Please find 3 Vocab words or concepts in our Shared Doc. Please provide a definition or explanation on our shared doc. Use a unique color. Also fill in the info on the chart on page 1 of our shared doc.



781-953-9699 skype name = stevebergen (no spaces) sbergen33@gmail.com

Scratch • Lesson 2 page 17 • Oct 5

All lessons stored at http://teachingcompany.com/F15

INDEX of HIGHLIGHTS can be found on the last page

☑ HW Watch the 6 min video on The Big Kahuna (page 2)

☑ Choose either 2.1A or 2.1B -- sum of the first 10 odd numbers or the sum of the numbers from 17 to 49 (pages 2-3)

☑ Build on the Commands from Lesson 1 and adding a few more (page 3)

☑ Pen Color varies from 0 to 200 (page 4) color wheel (page 5)

☑ HW 2.15 Watch the 3 min video on my STARTER SET (page 4)

☑ How to use Your Backpack (page 5)

☑ HW 2.16 Watch the 7 min video on Backpack, Comments (page 7)

☑ HW Choose either 2.2A or 2.2B -- modify the 36 sided polygon (page 8)

☑ HW 2.25 Watch the 9 min video on Variable Sized T (page 9)

☑ HW 2.3 Pick any letter you created in Lesson 1 and modify it so that it makes that letter to be variable sized. Name your program HW 2.3 VARIABLE SIZED LETTER and SHARE it so I can then see it. (page 9)

☑ HW 2.4 OPTIONAL Modify the above program (or you can use your own "S" if you made one) so that it makes the letter S of variable size. (page 11)

☑ HW 2.5 OPTIONAL Pick any 3 letters of the alphabet. (page 11)

☑ Setting the Zero Point (page 12)

☑ An Intro to Random Numbers (page 13)

☑ HW Watch the 8 min video on Flipping Coins (page 14)

☑ Choose one of these four programs (page 16)

☑ HW 2.6 Use the variable sized hexagon and the random number command to make a tessellation of hexagons (a popular pattern of bathroom tiles). (page 16)

☑ HW 2.7 Compute the interest on $100 deposited in a bank at 7% interest for 13 years. (page 16)

☑ HW 2.8 Write the code to create the COLOR WHEEL. (page 16)

☑ HW 2.9 Modify code in HW 2.5 so that 3 LETTER WORD is variable sized. (p 16)

☑ HW 2.10 Optional programs with prime numbers (page 16)

☑ HW 2.11 Three Vocab words or concepts per week on our Shared Doc. Please choose any 3 that have not been done and provide a definition or explanation on our shared doc. (page 16)