

# Week 2: Intro to Snap!

## CMPS10

---

PARDIS “PARIS” MIRI

UNIVERSITY OF CALIFORNIA, SANTA CRUZ

DEPARTMENT OF COMPUTER SCIENCE

# Outline

---

- Part 1: Snap is a web-based program.
  - access, load, sign in, and sign up in Snap!
- Part 2: Coding Examples in Snap
  - “Say” vs. “Say with Wait”
  - Custom blocks.
  - reporters (good for debugging) and coordinate system in SNAP!
  - Multiple Strips, Strip coloring, pen down, and clear
- Part 3: Fancy Kaleidoscope

# Part 1: Snap is a web-base Program

---

- Your code runs in the browser.
- The performance of the program you write is independent of how advanced your computer is.
- Snap is independent of the operating system (Mac, Linux, Windows, etc.) you are running.
- Snap depends on capabilities of your web browser.
  - **Recommendation: Use Chrome. Snap performs best in Chrome.**
  - Note: Snap had some issues in **Safari** and **Internet Explorer** in the past.

# Part 1: How to access SNAP

---

- Open Chrome
- Type in “Snap Berkeley” or directly go to
  - <http://snap.berkeley.edu/snapsource/snap.html>
- Sign up, check your email for password.
- **Sign in**
- Click on the cloud icon and **save** your project with a name.
- Start working!
- \*\*\*Always sign in first and then continue saving as you progress.\*\*\*

# Part 1: How to load Lab 2 Assignment

---

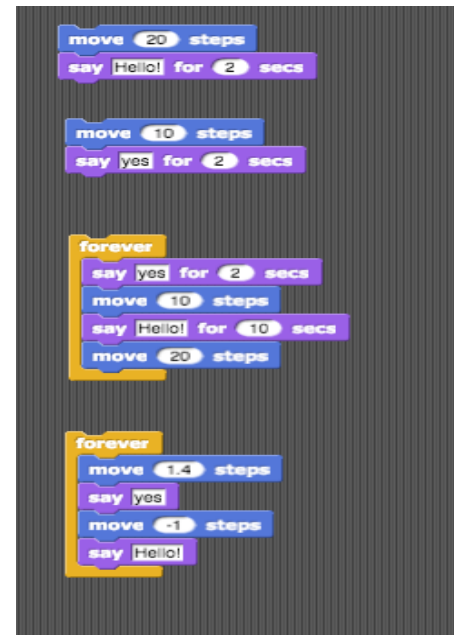
- Lab 2 Assingment: Kaleidoscope
  - When you have completed your Kaleidoscope, be sure and save it in the cloud and then export it as XML so you can turn it in to ecommons.
  - Extra credit is 2 points. Try to be creative.
  - **Due: Jan 18<sup>th</sup>, 8am.**

# Part2: Code examples

---

What happens when say is used without a timer...

Recommendation: Use Say with wait instead of Say.



```
move 20 steps
say Hello! for 2 secs

move 10 steps
say yes for 2 secs

forever
  say yes for 2 secs
  move 10 steps
  say Hello! for 10 secs
  move 20 steps

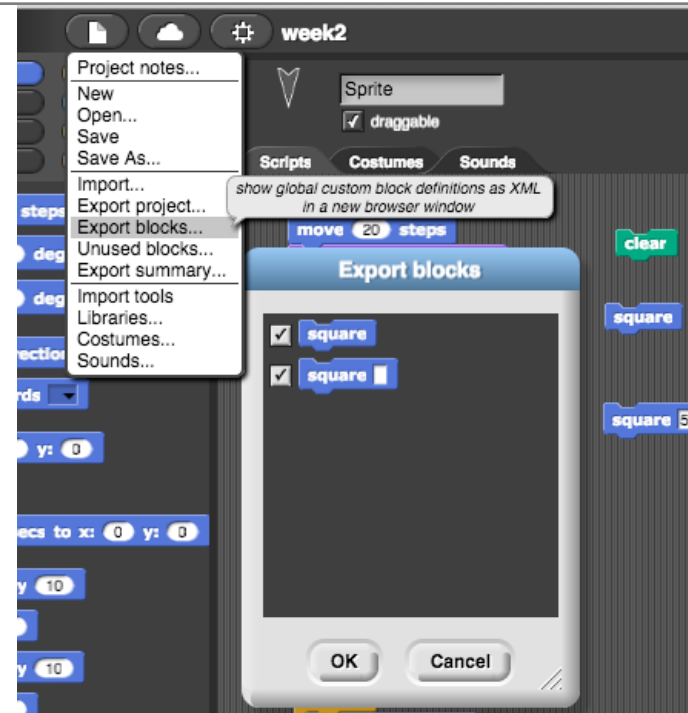
forever
  move 1.4 steps
  say yes
  move -1 steps
  say Hello!
```

# Part2: Code examples

How to move around custom blocks...

Recommendation: Export them as Blocks.

Then import them in your other projects.



# Part2: Code examples

---

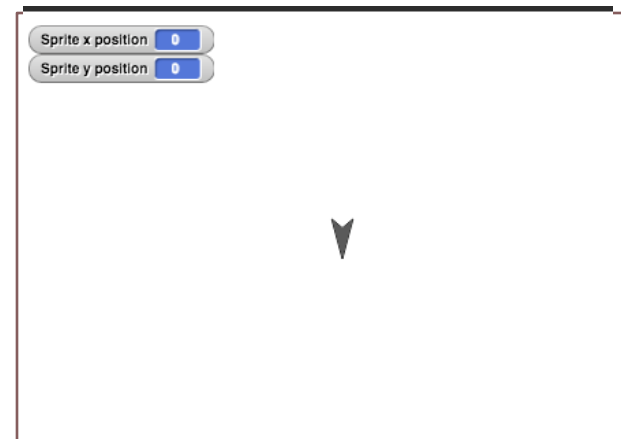
## Coordination System in Snap...

### Recommendation:

use x and y reporter if you ever got confused.

In Snap, (0,0) is always in the middle of the page.

Use “Go to x [0] y[0]” to get an idea where the middle is.

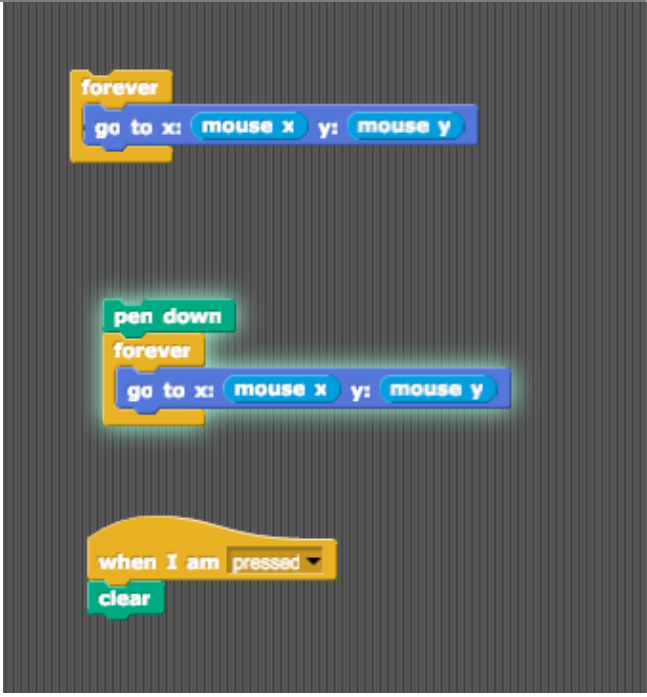




# Part2: Code examples

How to draw as I move around my mouse...

Recommendation: always pen down first, then following the (x,y) coordinate of the mouse.



```
forever
  go to x: mouse x y: mouse y

pen down
forever
  go to x: mouse x y: mouse y

when I am pressed
  clear
```

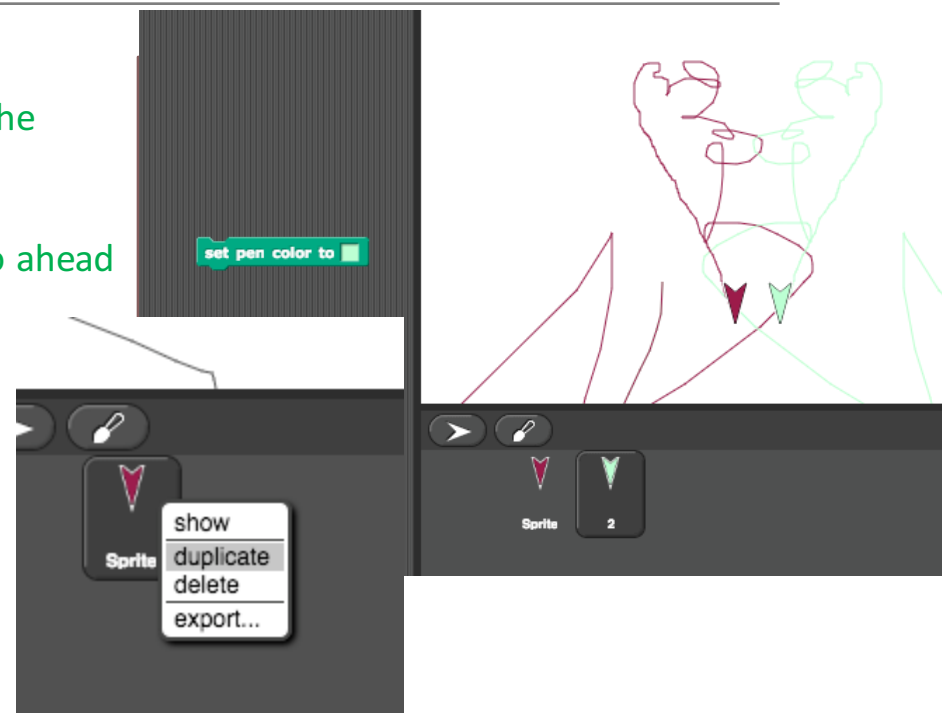
The image shows three separate Scratch code snippets on a dark grey background. The first snippet consists of a yellow 'forever' loop block containing a blue 'go to x: mouse x y: mouse y' block. The second snippet starts with a green 'pen down' block, followed by a yellow 'forever' loop block containing a blue 'go to x: mouse x y: mouse y' block. The third snippet starts with a yellow 'when I am pressed' block, followed by a green 'clear' block.

# Part2: Code examples

## How to get multiple sprites ...

Recommendation: duplicate the first sprite, set the Sprite color.

Note: The code automatically gets duplicated. Go ahead And modify it accordingly.

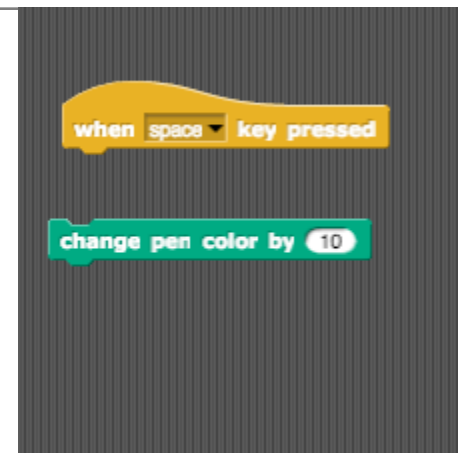


# Part 2: Code examples

---

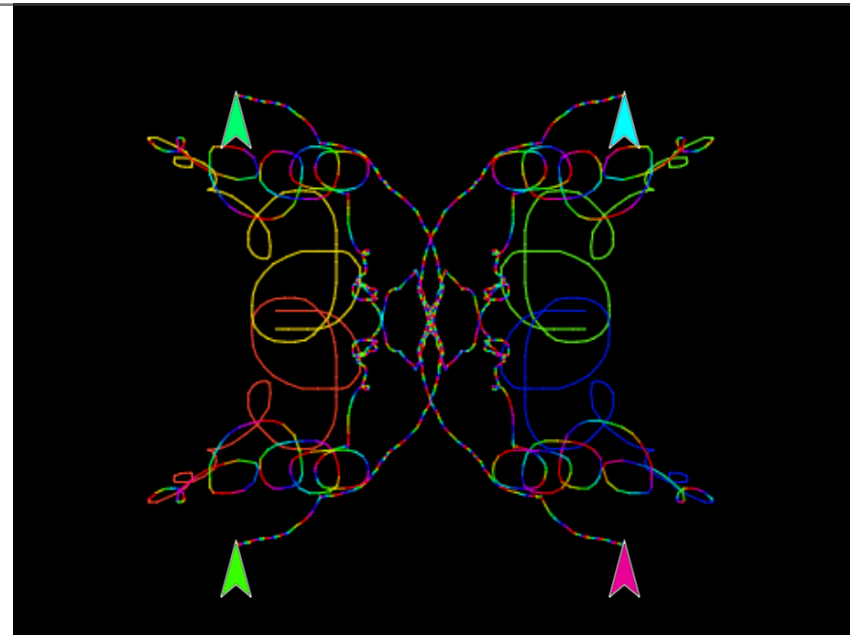
How to make my code fancy...

Recommendation: play around these two items.



# Part 2: Fancy Kaleidoscope

How to make my code fancy...



# Part 3: Week 2 Lab Assignment

---

**Assignment Link:** [ecommons](#)

**Assignment duration:** About two hour

**Assignment:** Complete the following “Intro to Snap!” activities (on your own or via a partner): [Intro to Snap!](#)

**Output:** Export the kaleidoscope and save it as **XML**.

exmaple: **kaleidoscope.xml**

**What to submit?** Submit your xml file as "Attachments" in the submission area.

If you **collaborated** with someone, mention their name and email address in the comment area.

If you are **resubmitting** (after being graded in the lab), mention mention that in the comment area.

**Due: Jan 18<sup>th</sup>, 8am.**

# Part 3: Week 2 Lab Assignment

---

## Grading Criteria:

- If you don't submit as a **XML**, you will receive **zero**.
- Full credit is 10. Two points of extra credit is available for this assignment.
- Make sure all four sprites start from the same point with x and y coordinate of (0,0).

# Part 2: SNAP! : Snap Programming Syllabus

---

- Week 1: Complete your choice of one of the three “Hour of Code” Activities.
- Week 2: “Intro to Snap!”
- Week 3: “Loops and Blocks”
- Week 4: “Conditional and Variables”
- Week 5: “Abstraction and Testing”
- Week 6: “Multiple Sprites” – **mini project**
- **Week 7: “Lists” + midterm (18<sup>th</sup> Feb)**
- Week 8: “Recursion and Fractals”
- Week 9 and 10: “Final Project”

# Questions?

---