Teaspoons of Computing for All Students in All Classes

Mark Guzdial,
Program in Computing for the Arts and Sciences

https://bit.ly/cs4MI2024-PCAS

Today's Story

- History: Computer science was created to be taught to ***everyone*** in ***all*** subjects,
  and had ***nothing*** to do with jobs.

- We might need to re-invent some things to get there.

- What we are teaching at the University of Michigan in the Program in Computing for the Arts and Sciences (PCAS)
  - All teachers *outside* of computer science and information
  - Teaching students across arts, humanities, and sciences.

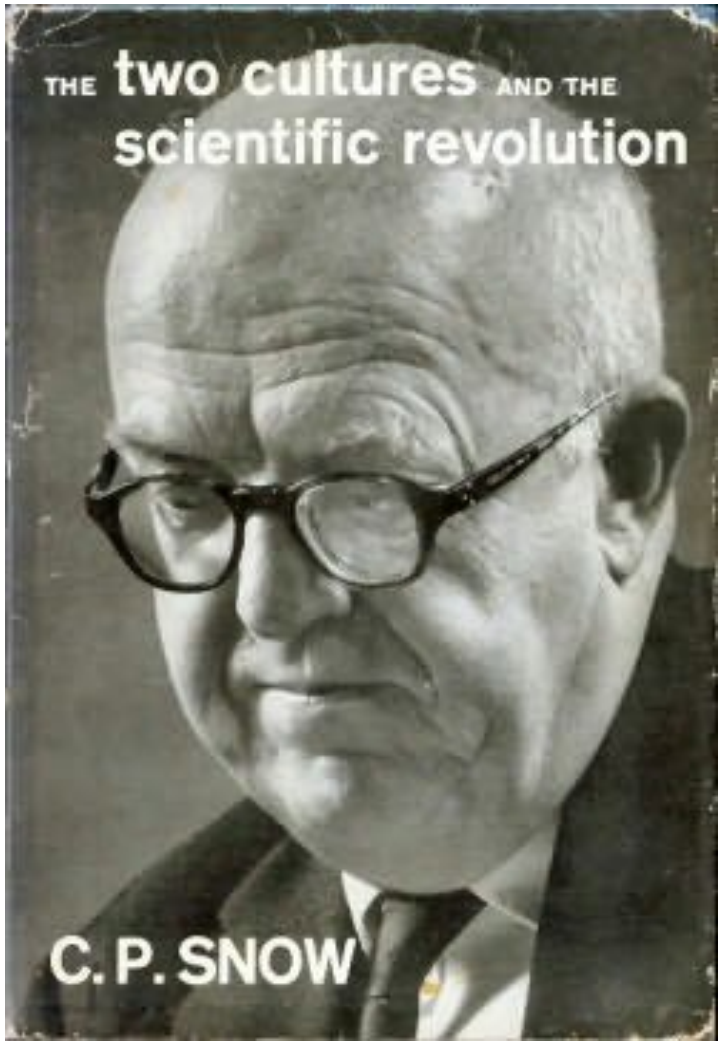
- Question for you:
  Could this be a model in your school or district?

George Forsythe

1961

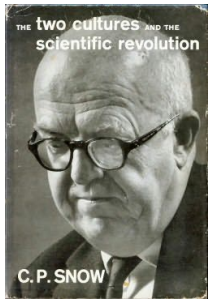
The computer is a necessary tool for learning science, mathematics, or engineering

1968



1961

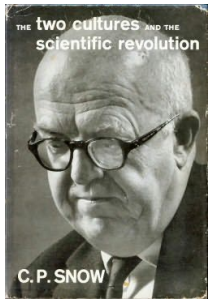
"A handful of people, having no relation to the will of society, having no communication with the rest of society, will be taking decisions in secret which are going to affect our lives in the deepest sense."



Peter Naur

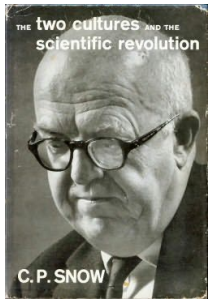
1967

"This is a necessary condition for humankind's supremacy over computers and for ensuring that their use do not become a matter for a small group of experts, but become a usual democratic matter, and thus through the democratic system will lie where it should, with all of us."



Alan Perlis
1961

Programming changes how we understand



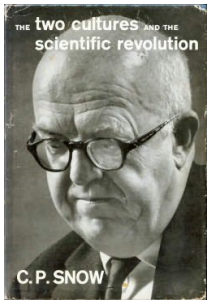
J.C.R. Licklider
1961

“Perhaps better poetry will be written in the language of digital computers of the future than has ever been written in English.”

Math and Meteorology

Astronomy

Chemistry and Math



Chemistry and Physics

Psychology and physics

First published definition of Computer Science

"The study of computers and all the phenomena surrounding them."

***Science*, 1967**

Alan Perlis

Herb Simon

Alan Newell

This is broader than how most people define computer science today. Let's call this *Computing*

Definitions of Computer Science

"The study of computers and all the phenomena surrounding them." (Perlis, Newell, & Simon, 1967)

Computer Science is the study of computers and computational systems. (Encyclopedia Brittanica)

Computer science is the study of computers and algorithmic processes, including their principles, design, implementation, and impact on society. (Tucker, 2006 - K-12 CS Framework)

Computer science is the foundational discipline with an emphasis on discovery related to programming, algorithms, and data structures. (ACM/IEEE Computing Curriculum 2021)

President Obama “CS for All”

2016

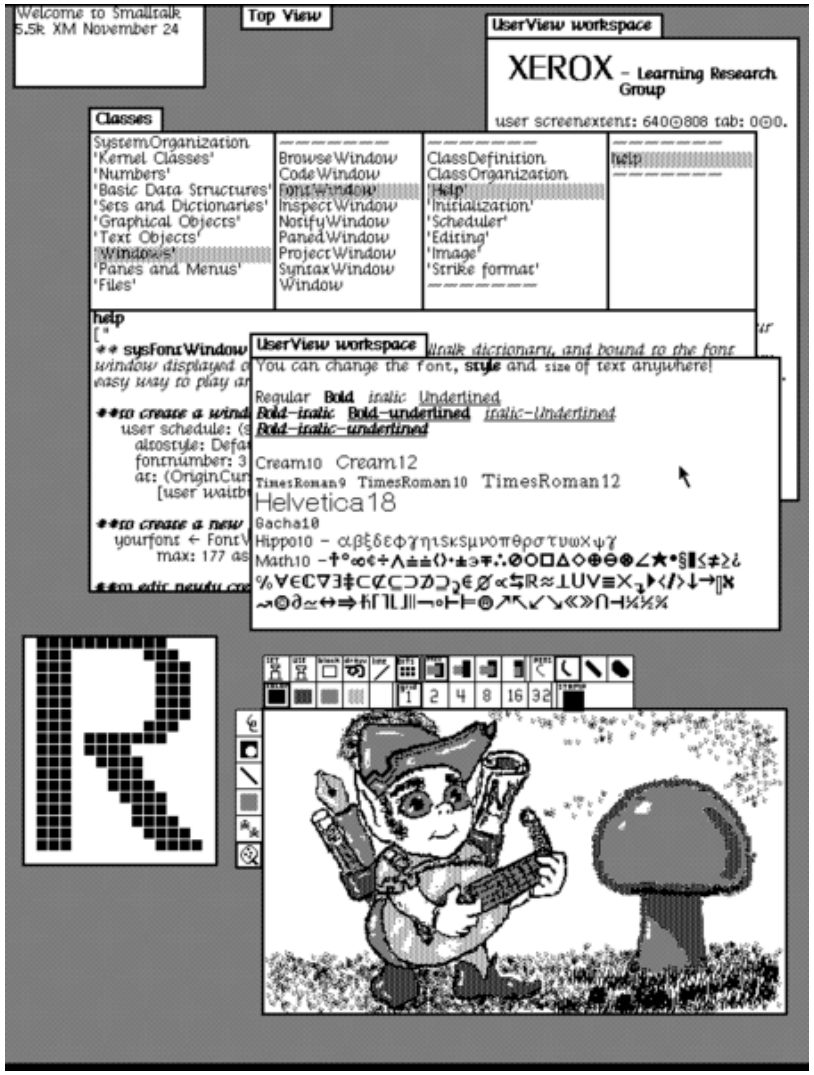
Computer science (CS) is a “new basic” skill necessary for economic opportunity and social mobility.

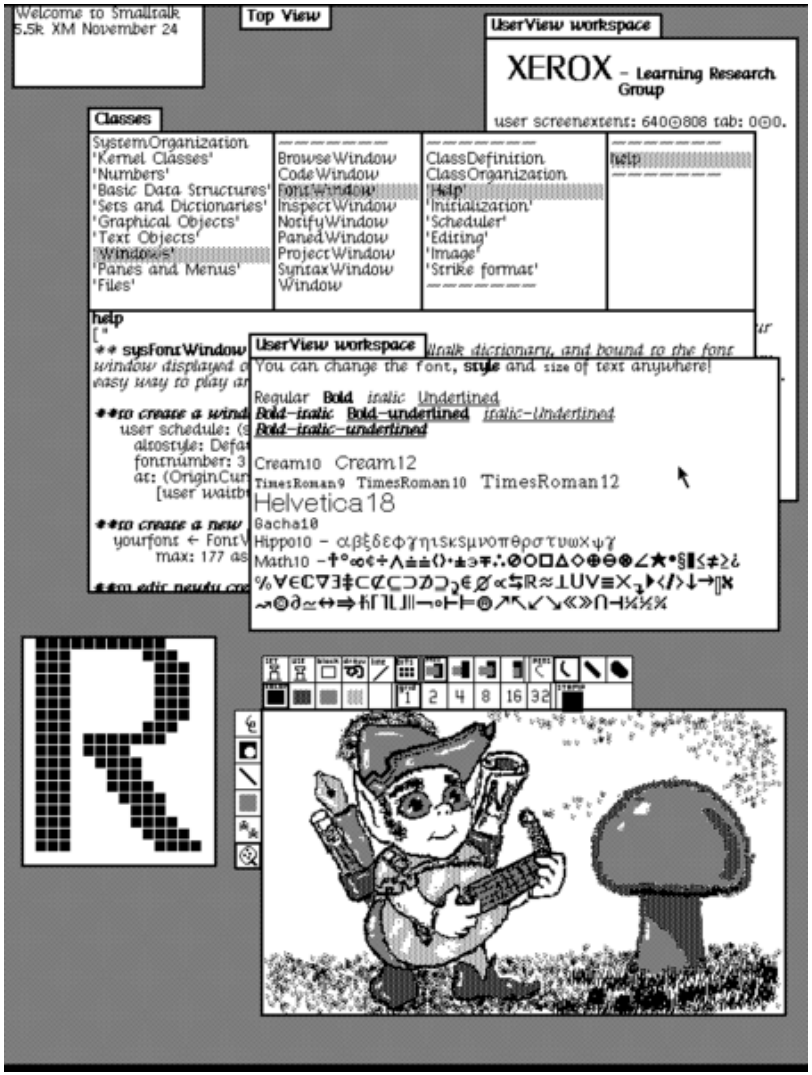
Photo: ObamaWhitehouse.archives.gov

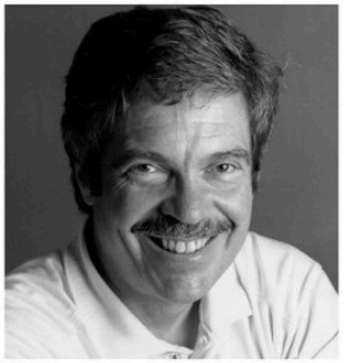
When did this become about “economic opportunity”?

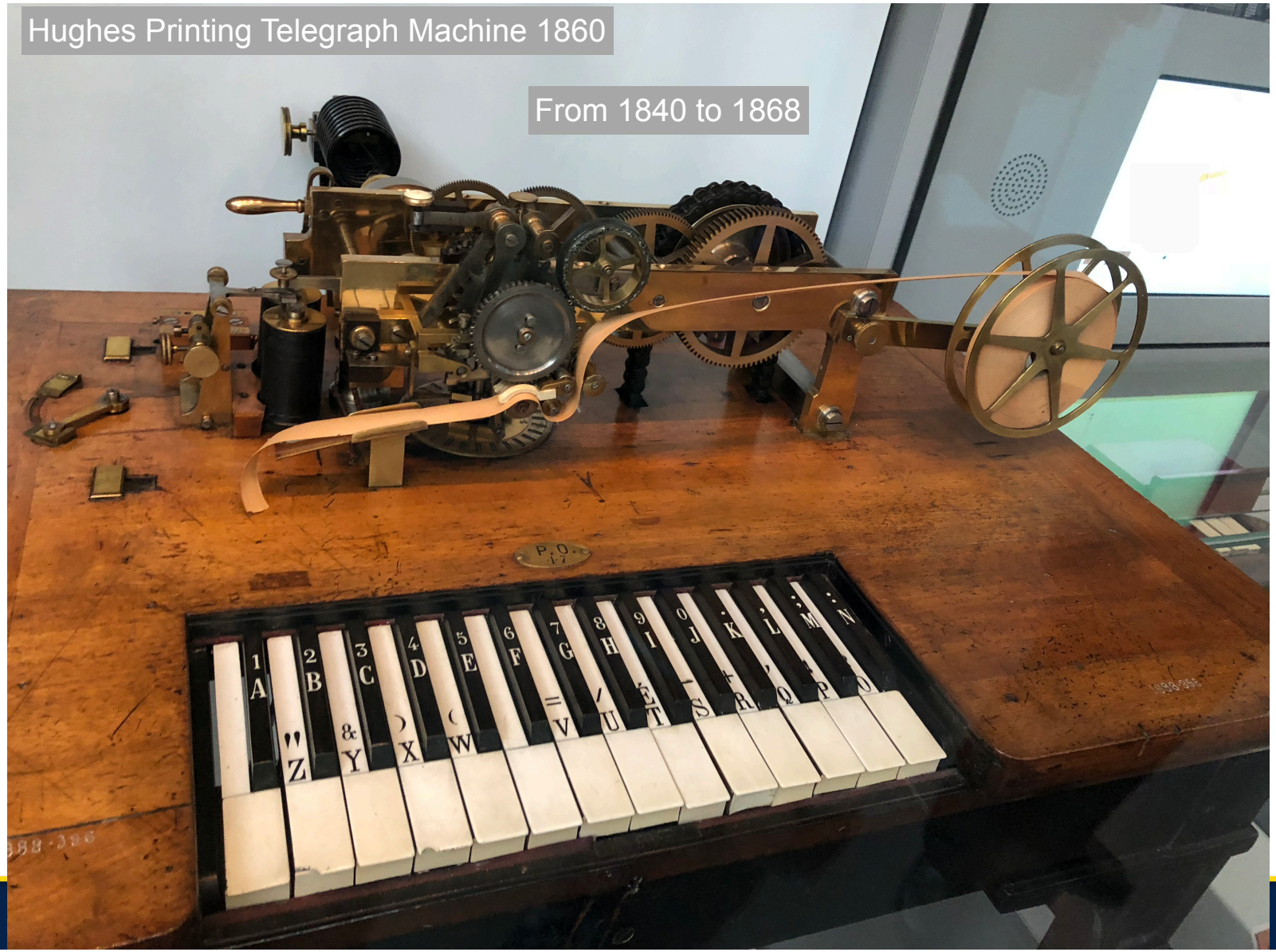
- Forsyth, Perlis, Snow, Naur, Simon, and Newell were all arguing for computing education for *everyone* ***years*** before Silicon Valley was created.
  - It’s important for STEM learning
  - It’s important for democracy
  - It creates new forms of expression

Changing Computing Education to Reach Everyone


Welcome to Smalltalk
5.5k XM November 24
Top View
UserView workspace
XEROX - Learning Research Group
user screenextent: 640@808 tab: 0@0.
Classes
SystemOrganization
'Kernel Classes'
'Numbers'
'Basic Data Structures'
'Sets and Dictionaries'
'Graphical Objects'
'Text Objects'
'Windows'
'Panes and Menus'
'Files'
BrowseWindow
CodeWindow
FontWindow
InspectWindow
NotifyWindow
PanedWindow
ProjectWindow
SyntaxWindow
Window
ClassDefinition
ClassOrganization
'Help'
'Initialization'
'Scheduler'
'Editing'
'Image'
'Strike format'
help
help
UserView workspace
You can change the font, style and size of text anywhere!
Regular Bold italic Underlined
Bold-italic Bold-underlined italic-Underlined
Bold-italic-underlined
Cream10 Cream12
TimesRoman9 TimesRoman10 TimesRoman12
Helvetica18
Gacha10
Hippo10 - αβξδεφγηιsκsμνοπθρστυωxψγ
Math10 -†°∞¢÷∧≐≙()·±∋∓∴⊘○□Δ◇⊕⊖⊗∠★•§∥≤≠≥¿
%∀∈ℂ∇∃‡⊂⊄⊆⊃∄⊋}€∅∝⇆ℝ≈⊥UV≡×₊▸‹/›↓→▯ℵ
⇝◎∂≃↔⇒ℏΓ℩⅃∥¬◦⊢⊨◎↗↖↙↘«»∩⊣¼½¾
1
2
4
8
16
32


Welcome to Smalltalk
5.5k XM November 24
Top View
UserView workspace
XEROX - Learning Research Group
user screenextent: 640@808 tab: 0@0.
Classes
SystemOrganization
'Kernel Classes'
'Numbers'
'Basic Data Structures'
'Sets and Dictionaries'
'Graphical Objects'
'Text Objects'
'Windows'
'Panes and Menus'
'Files'
BrowseWindow
CodeWindow
FontWindow
InspectWindow
NotifyWindow
PanedWindow
ProjectWindow
SyntaxWindow
Window
ClassDefinition
ClassOrganization
'Help'
'Initialization'
'Scheduler'
'Editing'
'Image'
'Strike format'
help
UserView workspace
You can change the font, style and size of text anywhere!
Regular Bold italic Underlined
Bold-italic Bold-underlined italic-Underlined
Bold-italic-underlined
Cream10 Cream12
TimesRoman9 TimesRoman10 TimesRoman12
Helvetica18
Gacha10
Hippo10 - αβξδεΦγηιϛκσμνοπθρστυωχψζ
Math10 -
1 2 4 8 16 32




Hughes Printing Telegraph Machine 1860
From 1840 to 1868

For 30 years, this was the common keyboard

We may still be waiting for our QWERTY keyboard.

We need to find what makes great ideas of computing accessible.

Maybe it's not Python.

Our Research Focus: Task-Specific Programming

Goal: Integrate programming* to enhance learning in K-12 school and university non-CS classes.

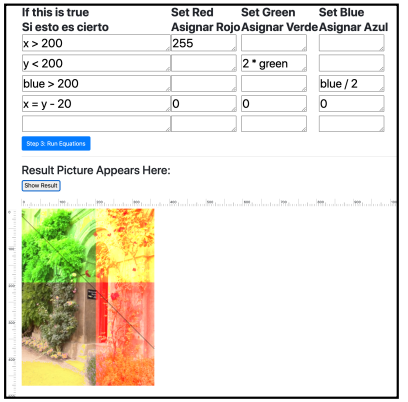
- Using participatory design with teachers to result in *adoptable* programming.

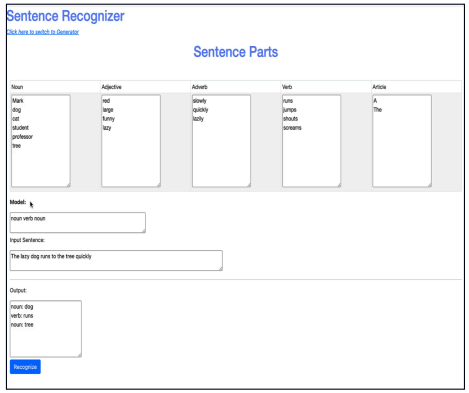
- Building task-specific programming environments to be *highly-usable*.

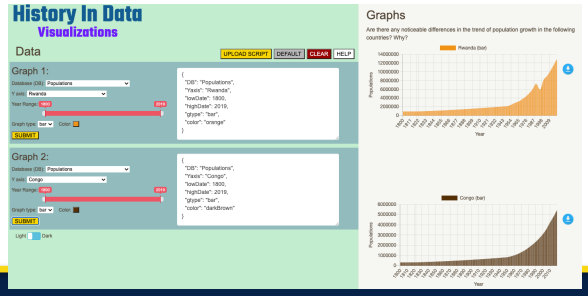
- *TSP* Languages => Teaspoon Languages
  Putting a Teaspoon of Computing in other subjects

What is a Teaspoon Language?

- They can be used by students for a task that is useful to a teacher.
- They are programming languages, i.e., a notation for defining a computational process.
- They can be learned within 10 minutes, so students can learn and use them within a single class session.

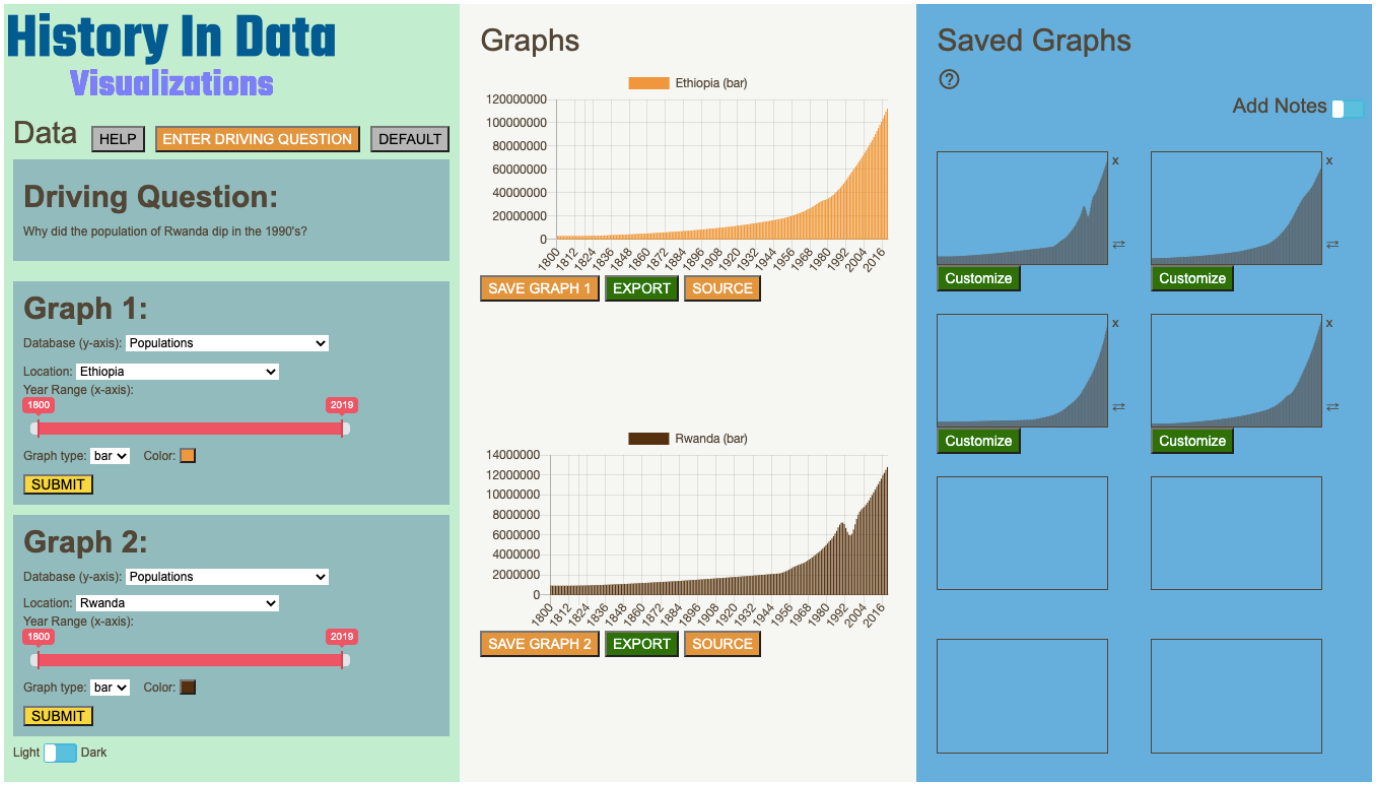






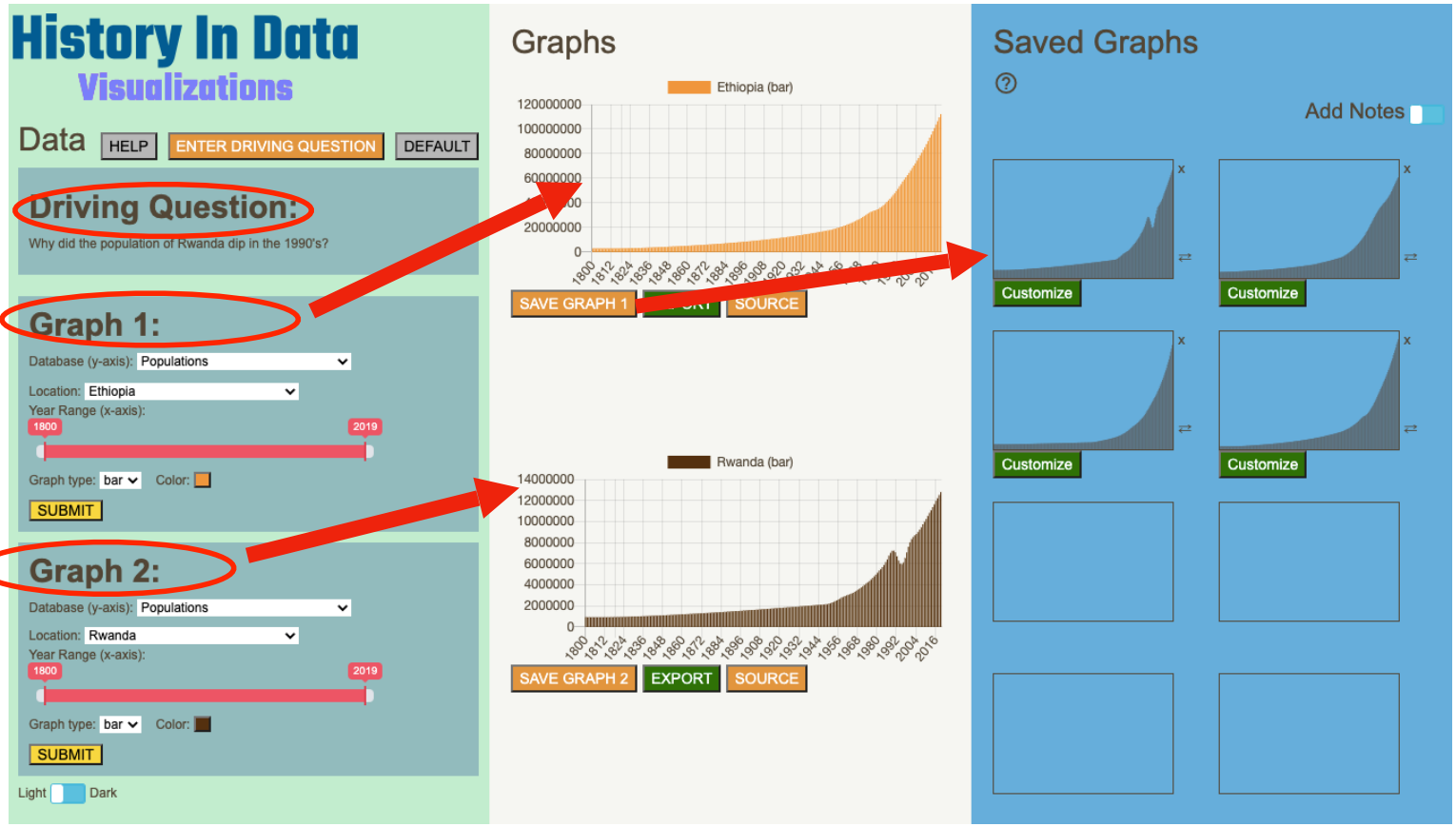
DV4L: Data Visualization for Learning

For History Courses



Collaboration with Drs. Tammy Shreiner and Bahare Naimipour

https://bit.ly/cs4MI2024-PCAS



Collaboration with Drs. Tammy Shreiner and Bahare Naimipour

Collaboration with Drs, Tammy Shreiner and Bahare Naimipour

History In Data

Visualizations

Data

UPLOAD SCRIPT | DEFAULT | CLEAR | HELP

Graph 1:

Database (DB): Populations

Y axis: Rwanda

Year Range: 1800 — 2019

Graph type: bar Color:

SUBMIT

```
{
  "DB": "Populations",
  "Yaxis": "Rwanda",
  "lowDate": 1800,
  "highDate": 2019,
  "gtype": "bar",
  "color": "orange"
}
```

Graph 2:

Database (DB): Populations

Y axis: Algeria

Year Range: 1800 — 2019

Graph type: bar Color:

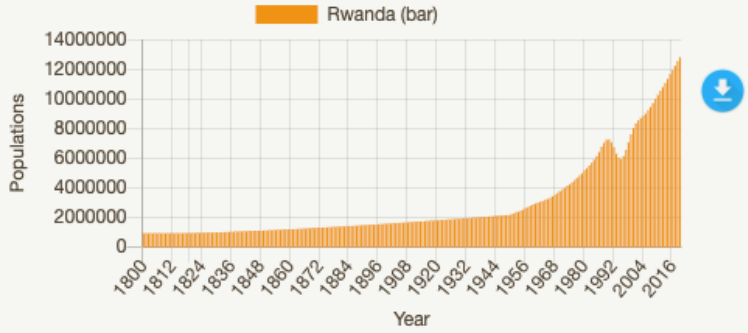
SUBMIT

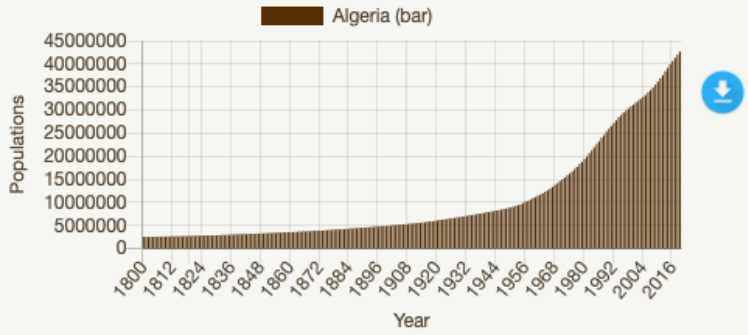
```
{
  "DB": "Populations",
  "Yaxis": "Algeria",
  "lowDate": 1800,
  "highDate": 2019,
  "gtype": "bar",
  "color": "darkBrown"
}
```

Light Dark

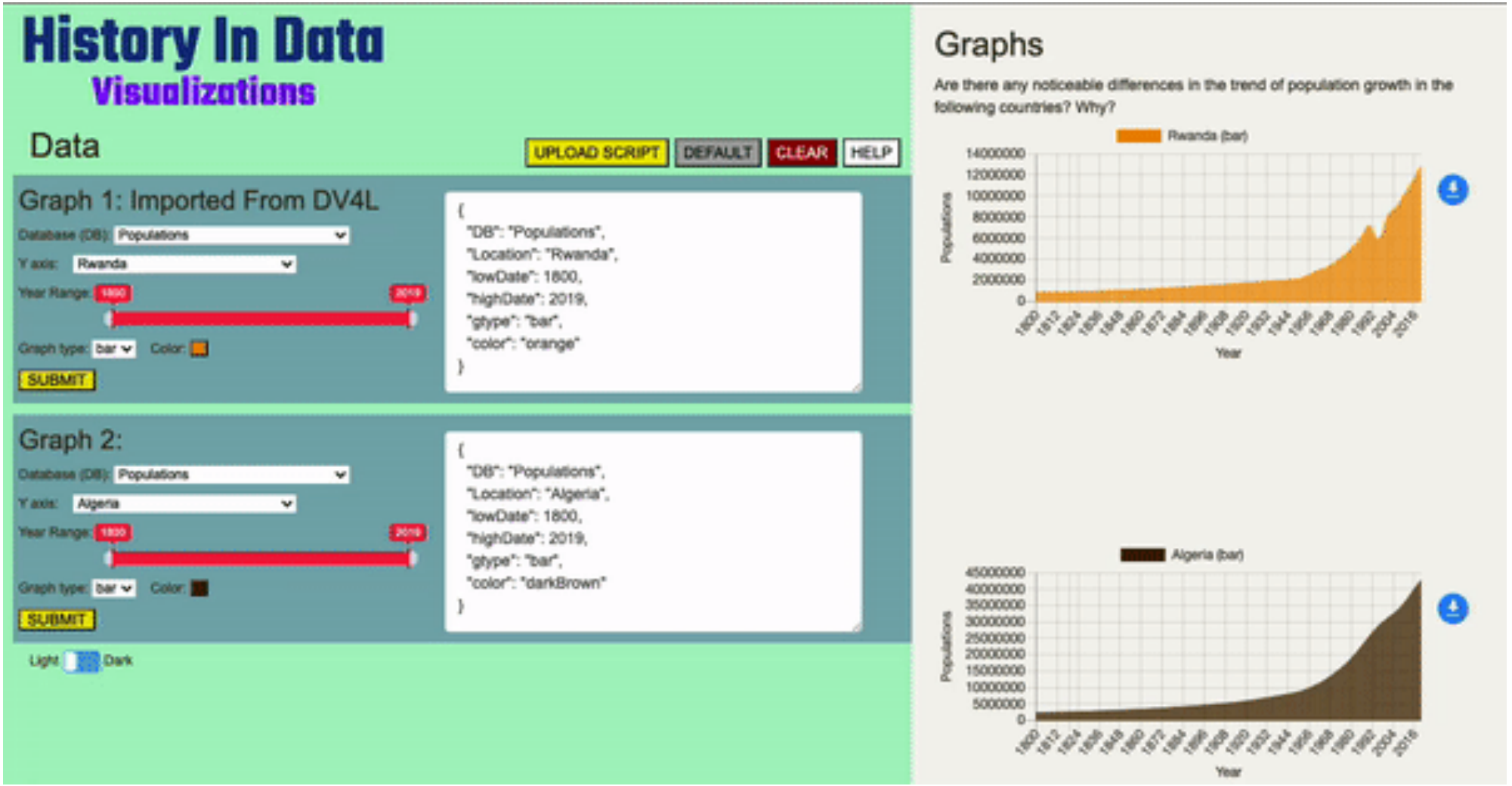
Graphs

Are there any noticeable differences in the trend of population growth in the following countries? Why?





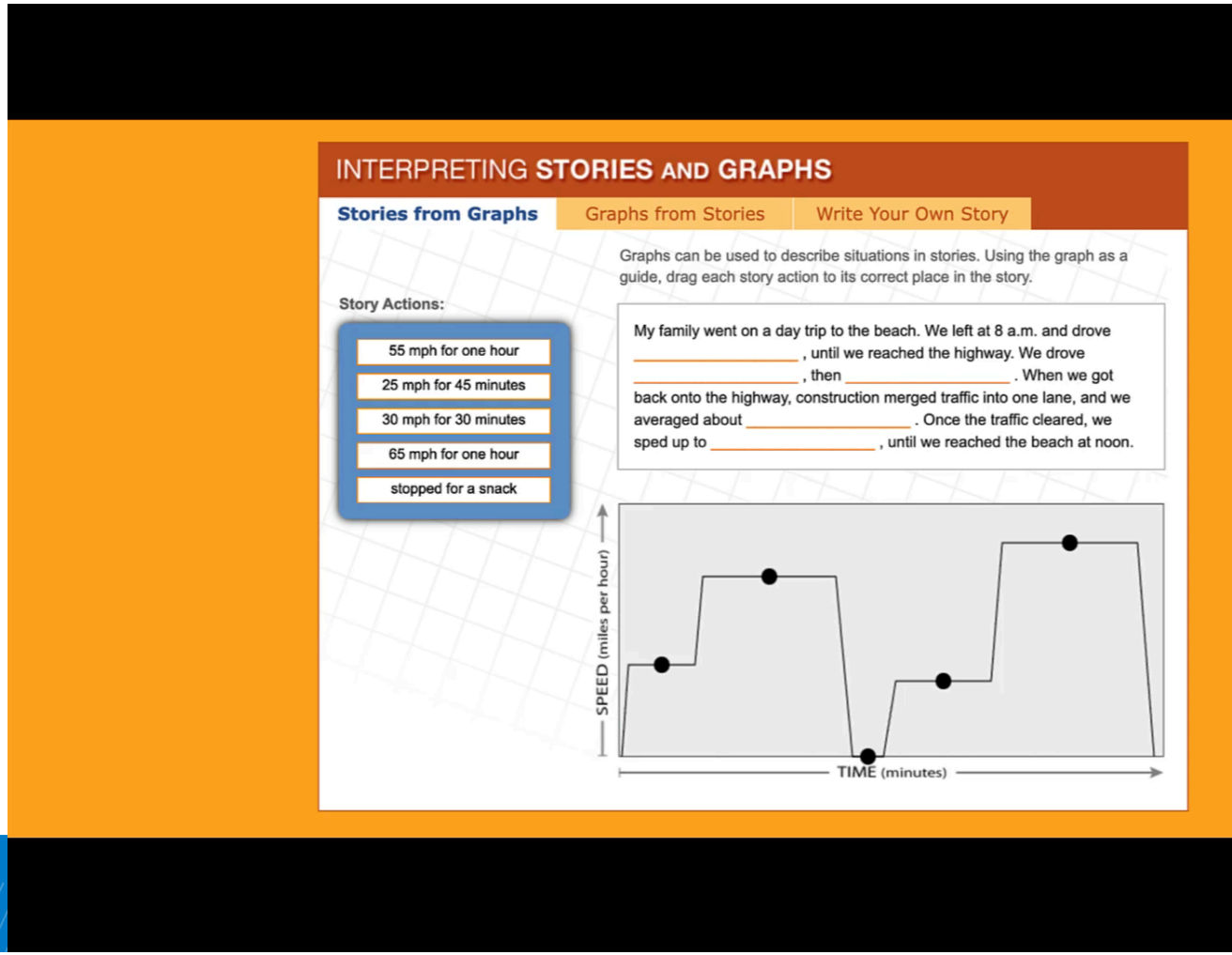
Collaboration with Drs, Tammy Shreiner and Bahare Naimipour



Collaboration with Drs, Tammy Shreiner and Bahare Naimipour

What an elementary school teaspoon language might look like

- Interpreting Stories and Graphs from PBS



Could you use these to address Michigan CS Standards?

- 1A-DA-06 Collect and present the same data in various visual formats.
- 1A-DA-07 Identify and describe patterns in data visualizations, such as charts or graphs, to make predictions.
- 1A-AP-09 Model the way programs store and manipulate data by using numbers or other symbols to represent information.
- 1B-DA-06 Organize and present collected data visually to highlight relationships and support a claim.
- 1B-DA-07 Use data to highlight or propose cause-and-effect relationships, predict outcomes, or communicate an idea.
- 1B-DA-06 Organize and present collected data visually to highlight relationships and support a claim.

- *1A-AP-10 Develop programs with sequences and simple loops, to express ideas or address a problem*

Broadening access and participation in computing

WHAT WE'RE TRYING AT MICHIGAN

Program in Computing for the Arts and Sciences

Mark Guzdial, mjguz@umich.edu

https://bit.ly/cs4MI2024-PCAS

PCAS Story

University of Michigan is creating a **Program in Computing for the Arts and Sciences** in the College of Literature, Science, and the Arts (LSA).

- No connection to our Computer Science & Engineering Division nor our School of Information.

Key Points:

1. A process of self-study: How is computing education for LSA different from what CS is offering?
2. A scaffolded series of computing activities for making computing accessible to everyone

https://bit.ly/cs4MI2024-PCAS

LSA Computing Education Task Force

What does LSA need in Computing Education?

Dean Anne Curzan and Associate Dean for Undergrad Ed Tim McKay charged the Computing Education Task Force 2020-2021

- What do LSA students need to know about computing?
- What classes and programs already exist?
- Where should we be going?

- Conducted dozens of interviews, reviewed hundreds of courses, surveyed over 100 LSA faculty.

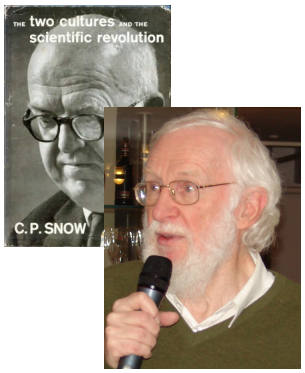
- Final report is available:

https://bit.ly/cs4MI2024-PCAS

3 Themes for Computing Education in LSA

- ***Computing for Discovery***: Computational science enables new discoveries across natural and physical sciences.

- ***Computing for Expression***: Computing has changed how we communicate and engage with others, from social media to Pixar to AR/VR.



- ***Critical Computing,* or *Computing for Justice*:** Computers and applications are pervasive in our daily lives, and thus have immense cultural, social, and political influence. Who is supported by computing, who is oppressed, and how can we create better models?

Creating PCAS

Program in Computing for the Arts and Sciences

Launched Summer 2022 - me and Gus Evrard, a first-generation computational cosmologist.

Teachers in our program from Music, American Culture, Anthropology, Physics, and Linguistics.

Goals:

- To **meet the needs of all LSA students** to learn about computing, especially programming.
- To create **new computing courses** around the themes of justice, expression, and discovery.
- To **create new credentials** to enhance majors and provide computing-centric minors in all divisions

What we currently offer

COMPFOR 101/HONORS 202: **The Transistor Disruption: How a Tiny Tool Transforms Society and Science**

COMPFOR 250/LING 321: **Alien Anatomy: How ChatGPT Works**

Discovery

COMPFOR 131/BIO 131/BIOPHYS 117: **Python Programming for the Sciences**

COMPFOR 150/LING 123: **The ABC's of Python: Language, Mind, and the Nature Of Programming**

COMPFOR 303/ANTHRBIO 369: **Fundamentals of scientific computing in R**

Expression

COMPFOR 121: **Computing for Creative Expression**

COMPFOR 221: **Digital Media with Python**

COMPFOR 304: **Synthesis to Streaming: Music in Digital Culture**

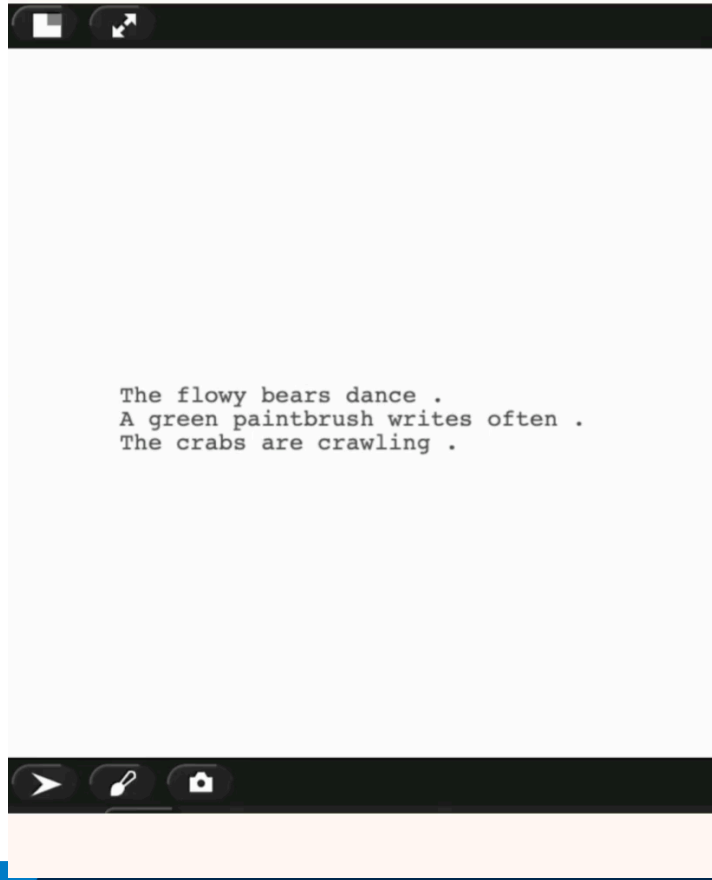
Justice

COMPFOR 111: **Computing's Impact on Justice: From Text to the Web**

https://bit.ly/cs4MI2024-PCAS

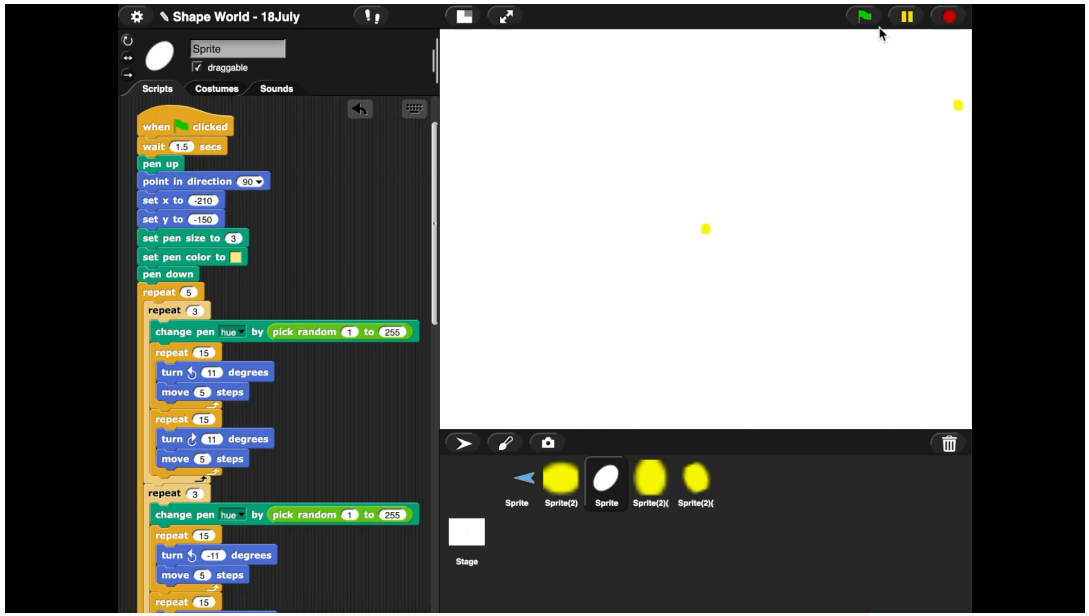
COMPFOR 111 *Computing's Impact on Justice: From Text to the Web*

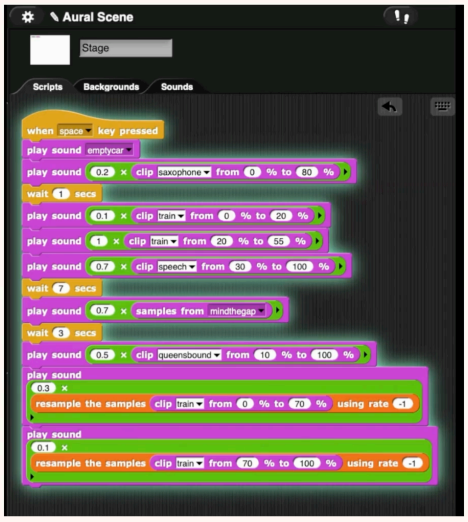
- How computers understand text and language.
  - It's English-first and mostly English-only
  - How info bots are made. **Build a haiku generator. Build a Chatbot (in multiple languages)**
- How the Internet works: From Text to the Web
  - HTML. **Generating Web pages: From blocks, from Twine, from databases**.
- From the Web to Data
  - How search engines work. Security, privacy, and GDPR.
  - Analyzing what the Web records about you. **Analyzing server log files.**
  - How to Visualize data. **Making arguments with visualizations.**
- Artificial Intelligence and Machine Learning
  - Kinds of ML. **Build a gesture recognizer**
- Limitations of technology: Security, Blockchain and NFTs, and DALL-E



COMPFOR 121 “Computing for Creative Expression”

- Building art from shapes
- How image filters work
- How to manipulate digital sounds
- Interactivity
  - Building a drawing program that uses the microphone or camera
- Side-scroller video game
- Text:
  - Build a chatbot with graphical and audio representations
  - Build webpages with styles and embedded interaction





COMPFOR 302: Alien Anatomy: How ChatGPT Works

Topics and Driving Questions of the Course

- History of AI — the earliest chatbots
- What changed with AI such that we could build ChatGPT?
- What does it mean to "train" an ML system? How was ChatGPT trained?
- What is a neural network? How does a neural network work?
- How do we use a neural network to understand language? What does it mean to generate language via probabilistic terms?
- How do we use a neural network to understand images?
- How do professionals create ML systems?
- How do humans learn and understand language?
- Is ChatGPT intelligent? Could it be?
- What can't ChatGPT do? What are the costs of ChatGPT?
- Ethical and Explainable AI

Grades will be based on participation in class, reflections on readings, student homeworks, bi-weekly quizzes (lowest score dropped), and a team final project. There will be no midterm or final exams.

Student Homeworks

Most of the homework will be in Snap, as in the **SnapGPT** project and **Music-GPT** project.

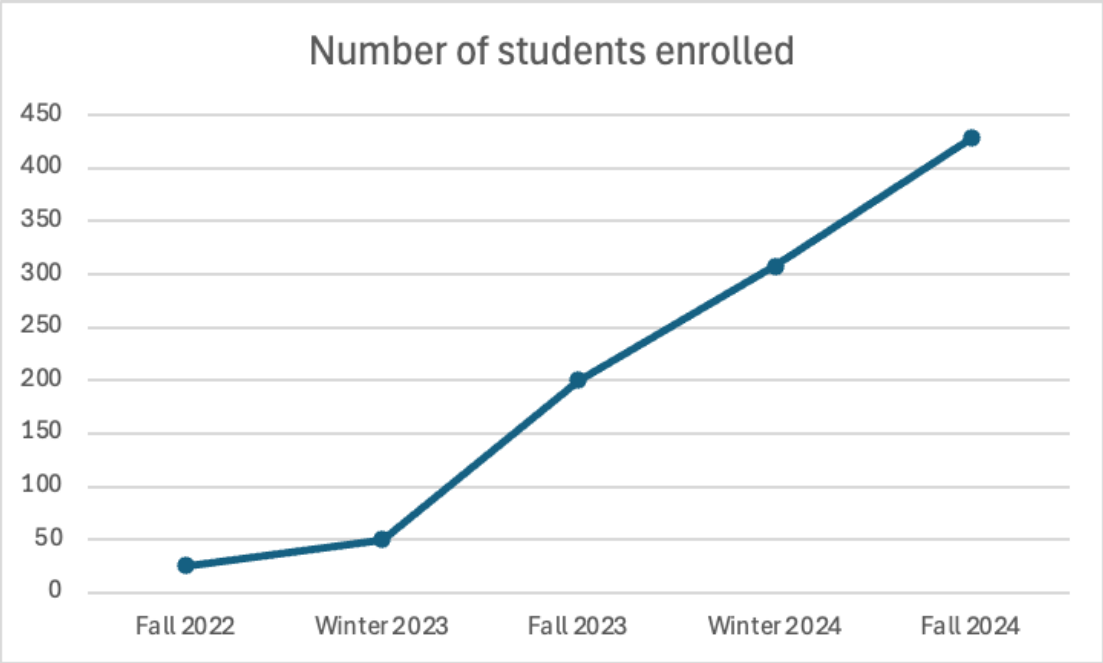
- Build a chatbot out of production rules
- Build a classifier for processing images of hand gestures
- Build a gesture recognizer
- Build a simple neural network
- Build a text generator based on a statistical language model
- Build a neural network in Python using Jupyter notebooks

Co-taught with Steve Abney

https://bit.ly/cs4MI2024-PCAS

Growth in the program

- In Fall 2022, we had 11 students in Expression and 14 students in Justice (total: 25)

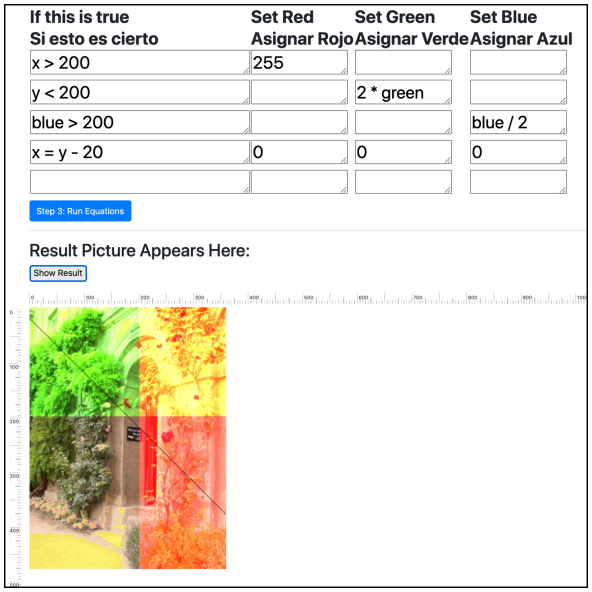


428 this semester

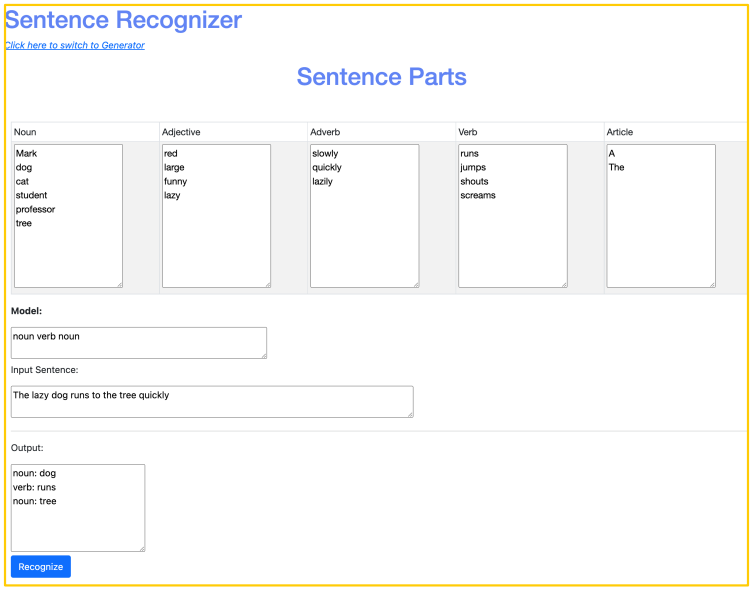

Scaffolded Structure to introduce Programming
in Expression and Justice
COMPFOR Courses

Learning Programming in Context, Step 1

Students start learning computational concepts using task-specific programming languages, “teaspoon languages.”



For *Creative Expression*, learning how image filters work.



For *Computing’s Impact on Justice,* learning how computers recognize sentence parts.

https://bit.ly/cs4MI2024-PCAS

Pixel Equations

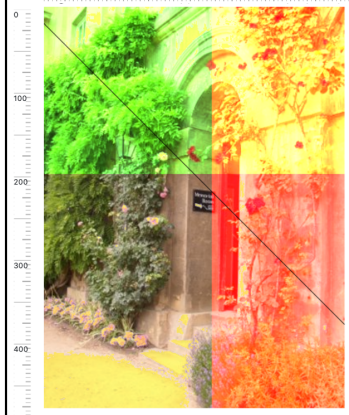
Designed For High School Math and Engineering classes

If this is true Si esto es cierto	Set Red Asignar Rojo	Set Green Asignar Verde	Set Blue Asignar Azul
x > 200	255		
y < 200		2 * green	
blue > 200			blue / 2
x = y - 20	0	0	0

Step 3: Run Equations

Result Picture Appears Here:

Show Result



Collaboration with Emma Dodoo

https://bit.ly/cs4MI2024-PCAS

Pixel Equations

Select your preferred language

◉English

Idioma/Language

Step 1: Pick your input picture

Which picture would you like to use?

◉ File named: arch.jpg 

○ File named: Bayamon.jpeg 

○ File named: beach.jpg 

○ File named: dog.png 

○ File named: san-juan.jpeg 

○ File named: TSM-Map.png 

○ File named: detroit.jpg 

○ File named: DetroitSkyline.jpg 

which will select all pixels where the `x` coordinate is greater than the `y` coordinate.

Then write equations for how to change red, green, and blue (rojo, verde, y azul) for the selected pixels. You can invert each color by subtracting from 255 (e.g., set red/rojo to `255-red` (o `255-rojo`)).

If this is true Si esto es cierto	Set Red Asignar Rojo	Set Green Asignar Verde	Set Blue Asignar Azul
x > 200	255		

Step 3: Run Equations

Result Picture Appears Here:

Show Result

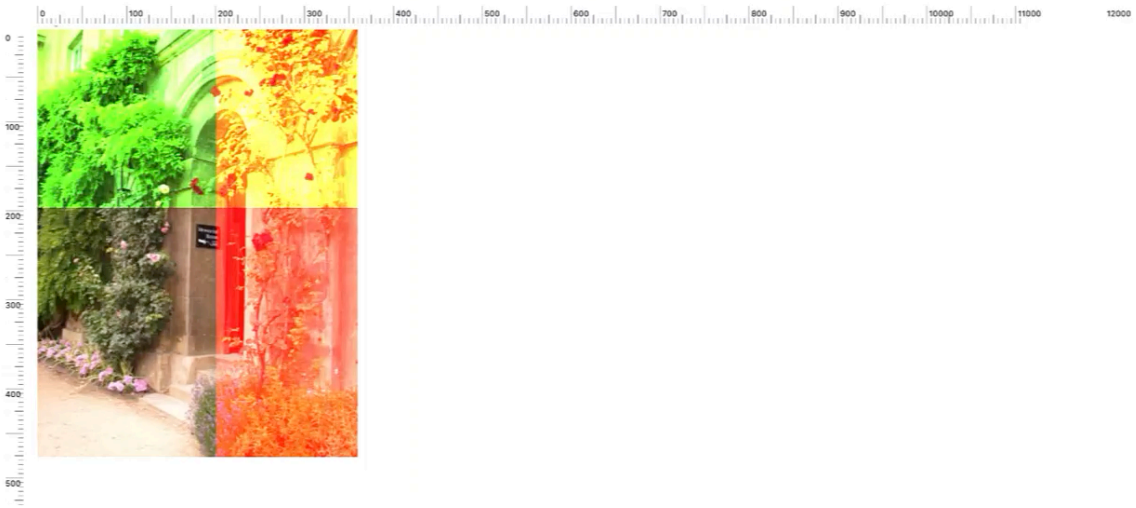
set red/rojo to 255-red (o 255-rojo)).

If this is true Si esto es cierto	Set Red Asignar Rojo	Set Green Asignar Verde	Set Blue Asignar Azul
x > 200	255		
y < 200		2 * green	

Step 3: Run Equations

Result Picture Appears Here:

Show Result



Pixel Equations

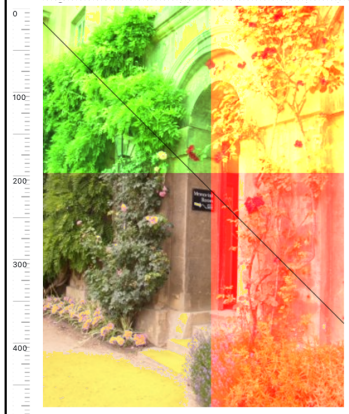
Designed For High School Math and Engineering classes

If this is true Si esto es cierto	Set Red Asignar Rojo	Set Green Asignar Verde	Set Blue Asignar Azul
x > 200	255		
y < 200		2 * green	
blue > 200			blue / 2
x = y - 20	0	0	0

Step 3: Run Equations

Result Picture Appears Here:

Show Result



Sentence Recognizer /Generator

Sentence Recognizer

Click here to switch to Generator

Sentence Parts

Noun	Adjective	Adverb	Verb	Article
Mark dog cat student professor tree	red large funny lazy	slowly quickly lazily	runs jumps shouts screams	A The

Model:

noun verb noun

Input Sentence:

The lazy dog runs to the tree quickly

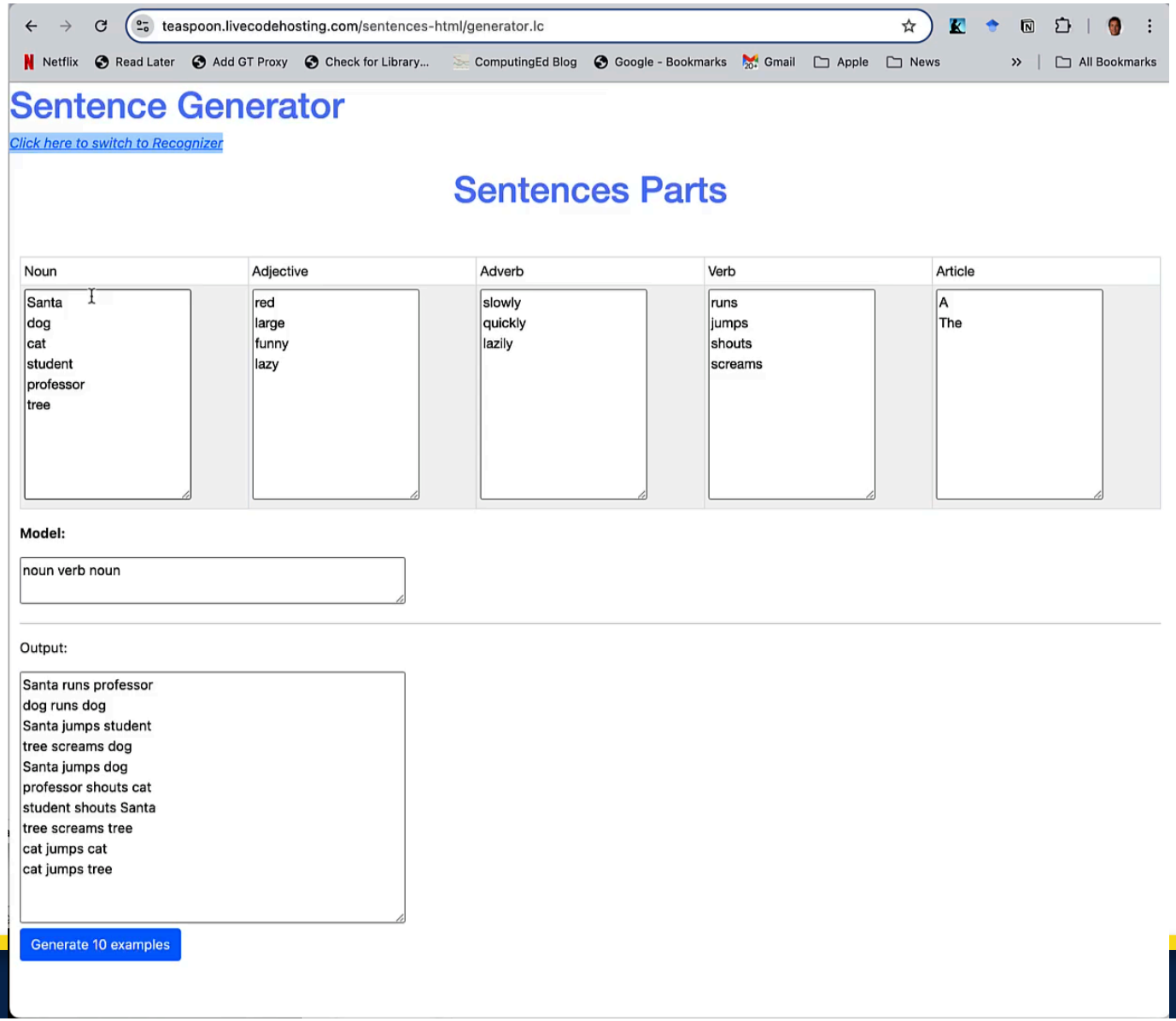
Output:

noun: dog
verb: runs
noun: tree

Recognize

https://bit.ly/cs4MI2024-PCAS

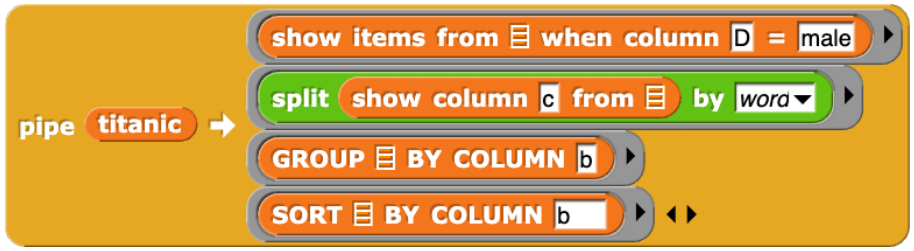
Sentence Generation with Weights

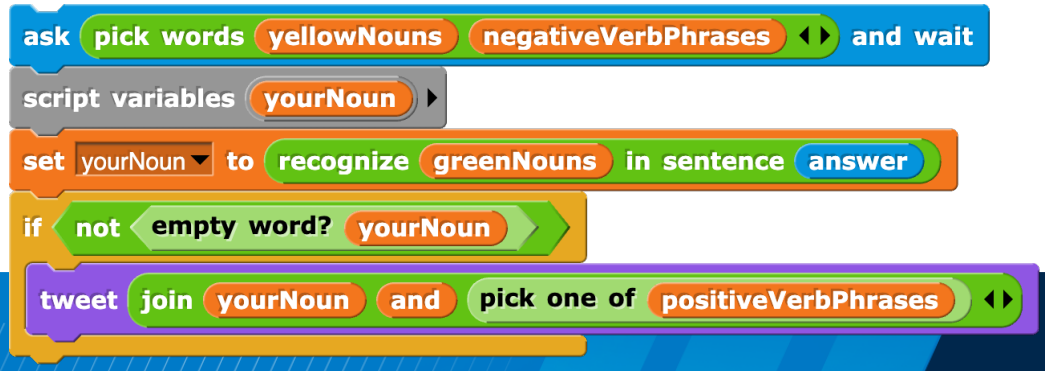


Learning Programming in Context, Step 2

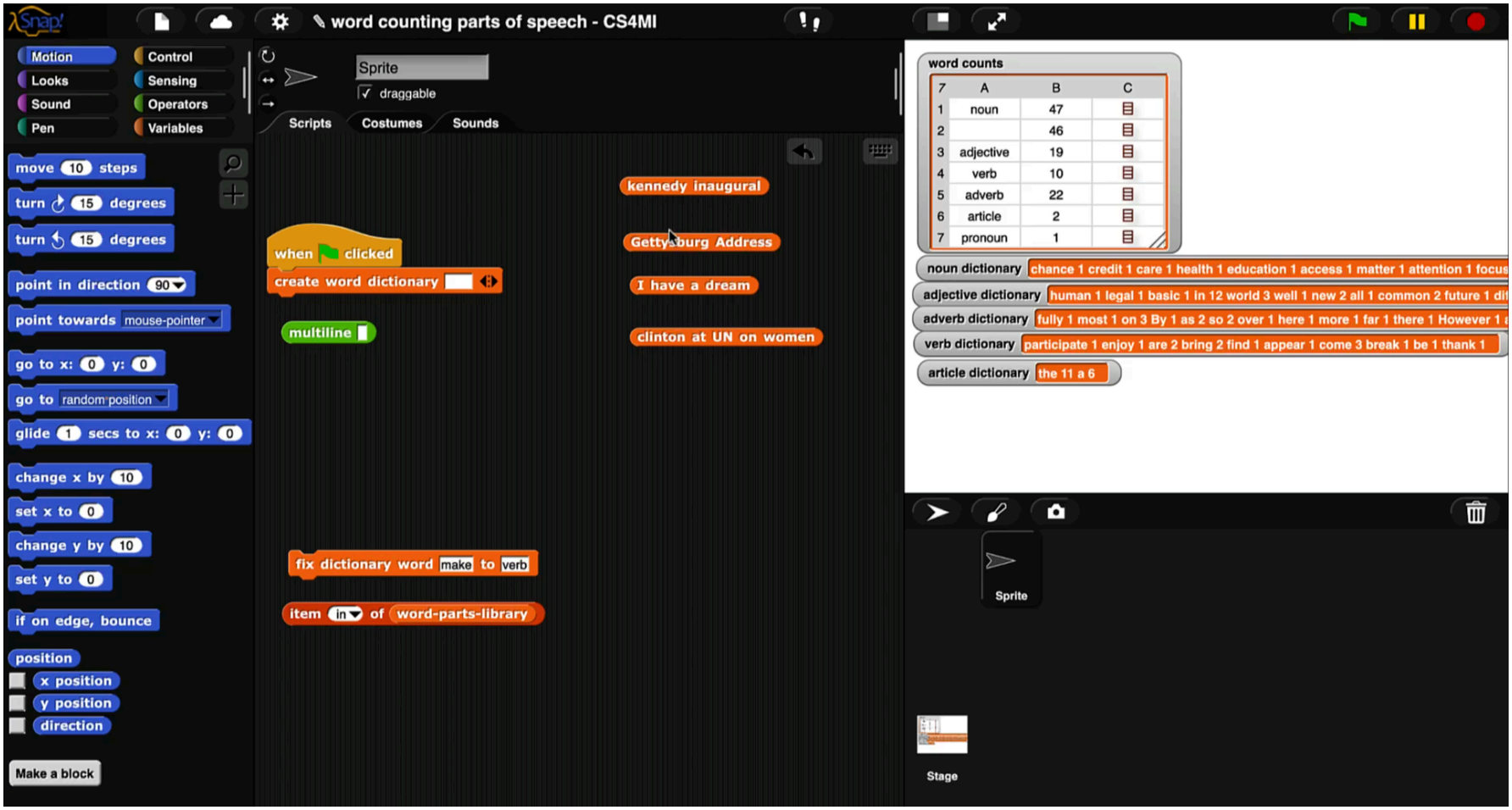
Students build their own programs using *Snap*, a block-based language developed at the U. California-Berkeley. Examples:

- For *Creative Expression*, building their own image and sound filters.
- For *Computing's Impact on Justice,* building biased chatbots to understand how misinformation spreads, and building database queries.





Sentence Generation with Weights



Sentence Generation with Weights

Sentence Generator

Click here to switch to Recognizer

Sentences Parts

Noun	Adjective	Adverb	Verb	Article
God 1 nation 5 resolve 1 devotion 2 measure 1 cause 1 task 1 work 1 note 1 power 1	human 1 legal 1 basic 1 in 12 world 3 well 1 new 2 all 1 common 2 future 1	fully 1 most 1 on 3 By 1 as 2 so 2 over 1 here 1 more 1 far 1	participate 1 enjoy 1 are 2 bring 2 find 1 appear 1 come 3 break 1 be 1 thank 1	the 11 a 6

Model:

article adjective noun verb adverb

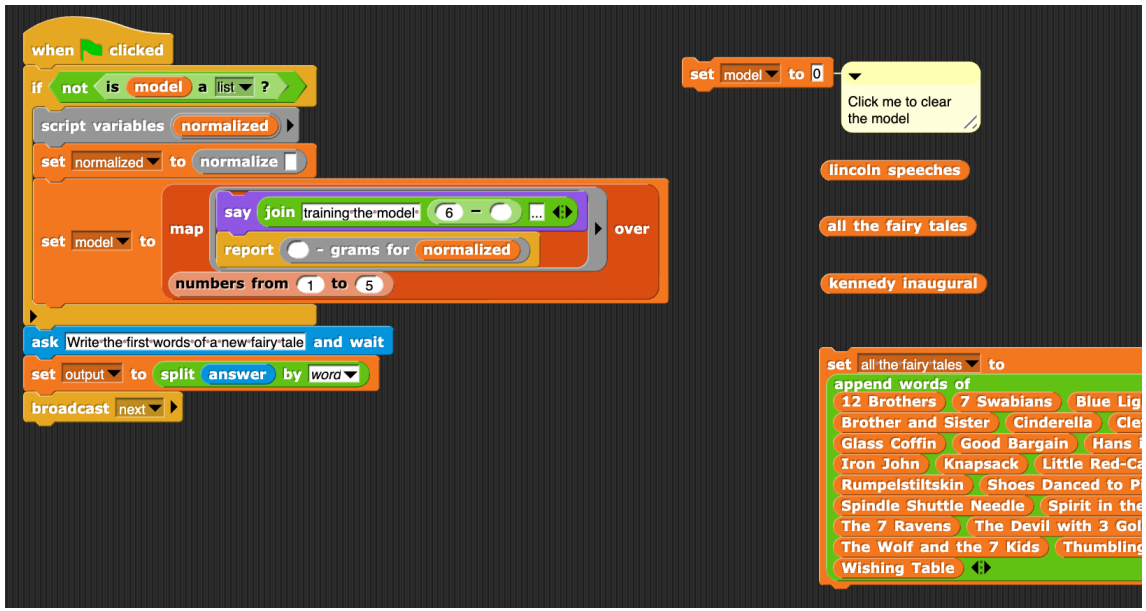
Output:

a common cause more
the world war come again
the every Liberty bring part
the United nation come truly
the new resolve bring about
the common nation appear way
a every sense However
the men be as
the living men thank Fourth
a in portion come By

Generate 10 examples

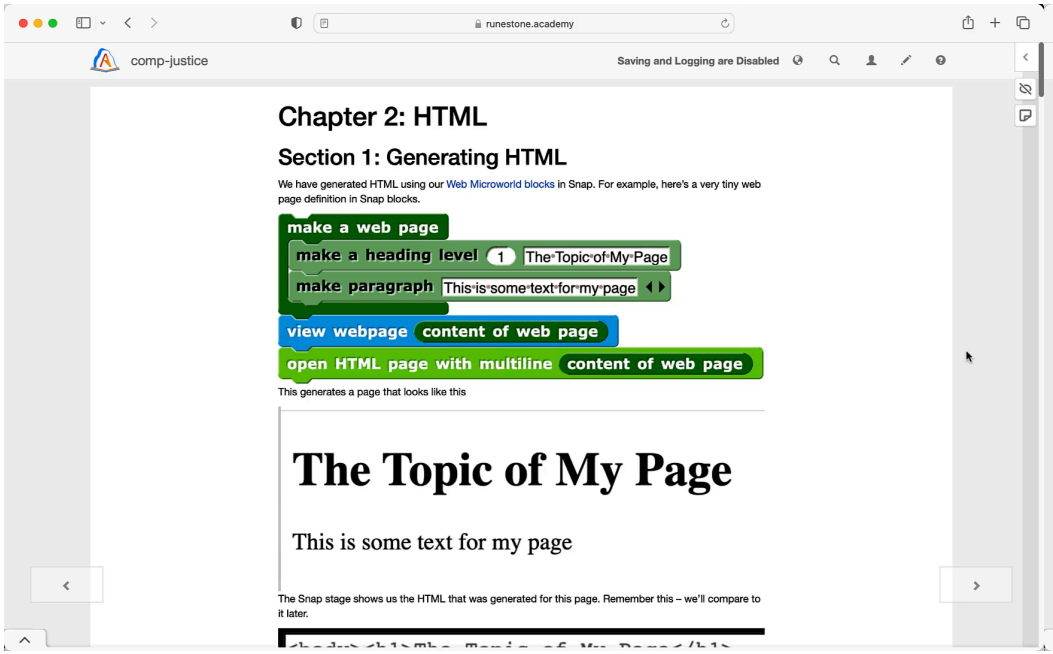
Leading to building a SnapGPT

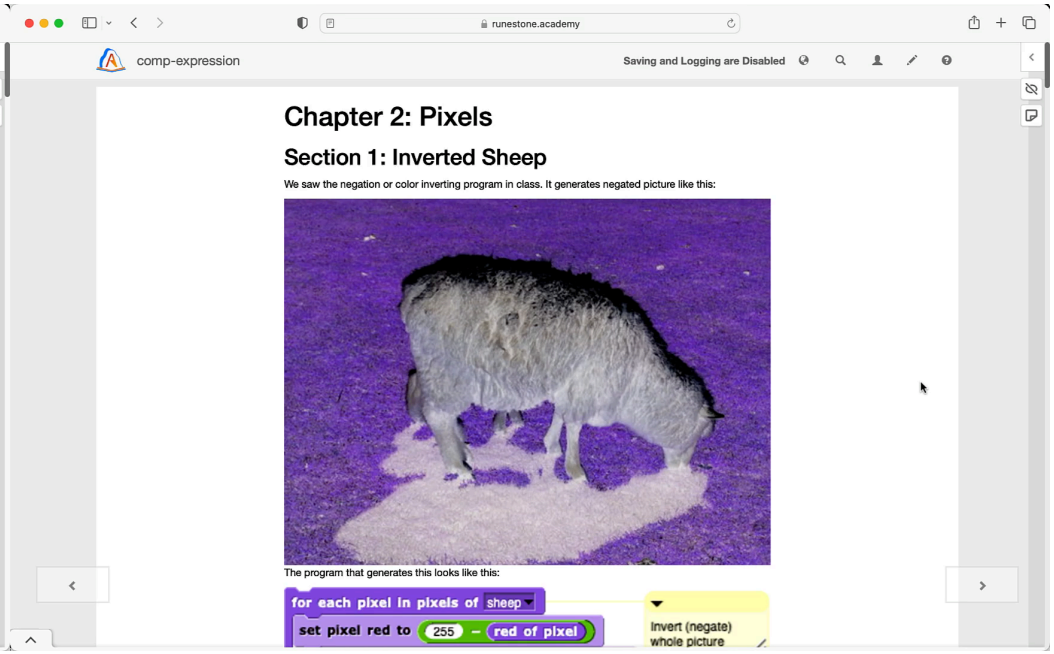
- A chatbot that can write new fairy tales,
  Or new speeches by Abraham Lincoln,
  John F. Kennedy…



Learning Programming in Context, Step 3

At the end of each unit, students complete activities in an ebook where they see Snap code they've used before, then Python or Processing programs that do the same things, then answer questions about the textual code, which sometimes invites them to change the code.





A Participatory Design Process: Describing One Path through Computing for Justice

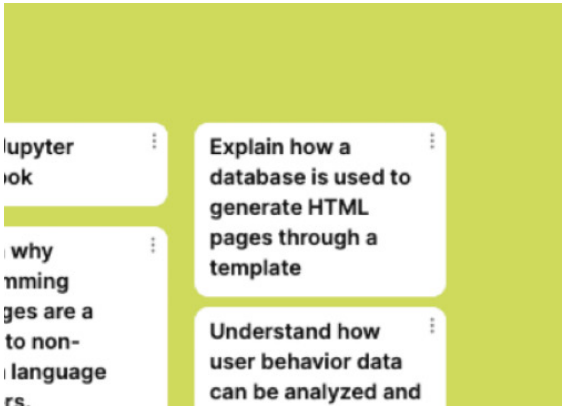
What do Justice Scholars want Students to Know about the Internet?

History Professor, LaKisha Simmons gave me this list:

1. There are things called databases.

2. That databases, if they are designed well, are easy to index and to find information in.

3. Databases could be used to automatically generate Web pages.



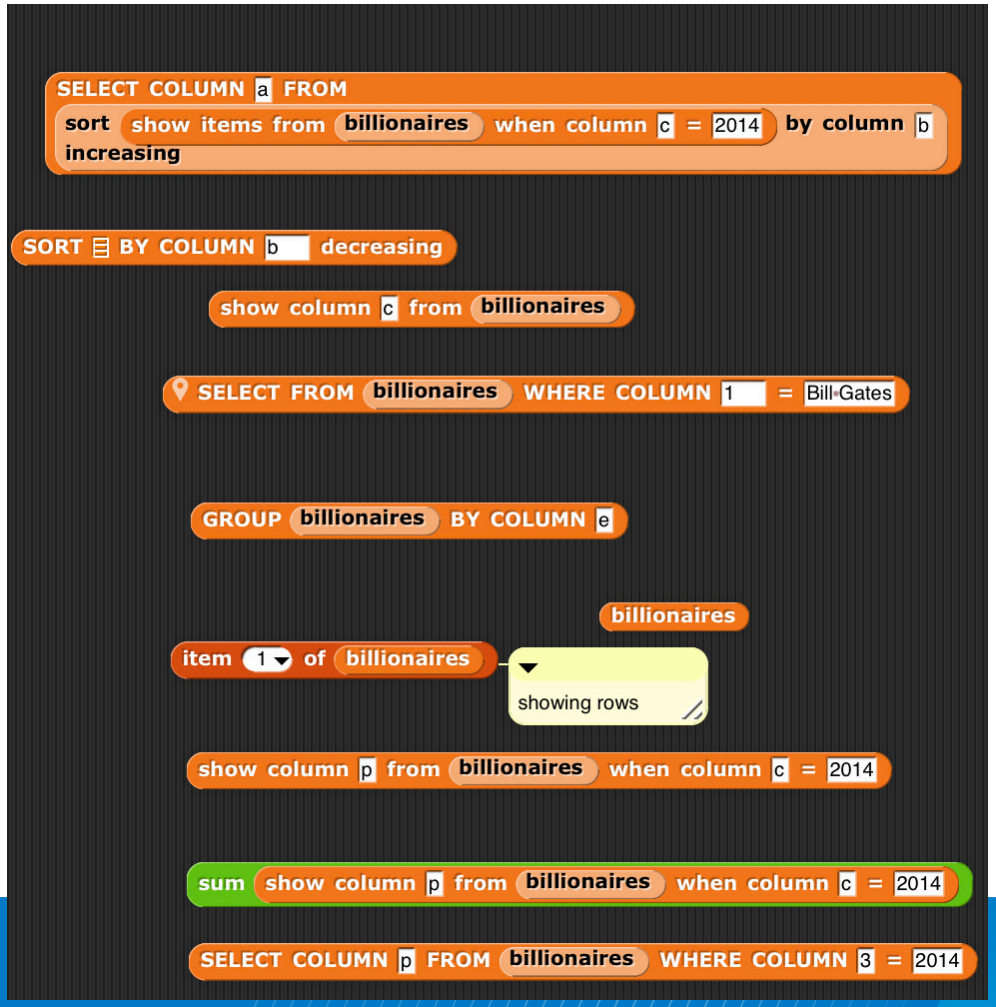
These are typically “advanced” topics

- None of the Michigan nor CSTA CS standards have the word “database” in them.

- Few undergraduate CS programs ***start*** at databases and Web programming.
  - Typically, we start with introductory programming, then data structures and algorithms, then…

- But we don’t ***have*** to, and ***this*** is what humanities fields want.

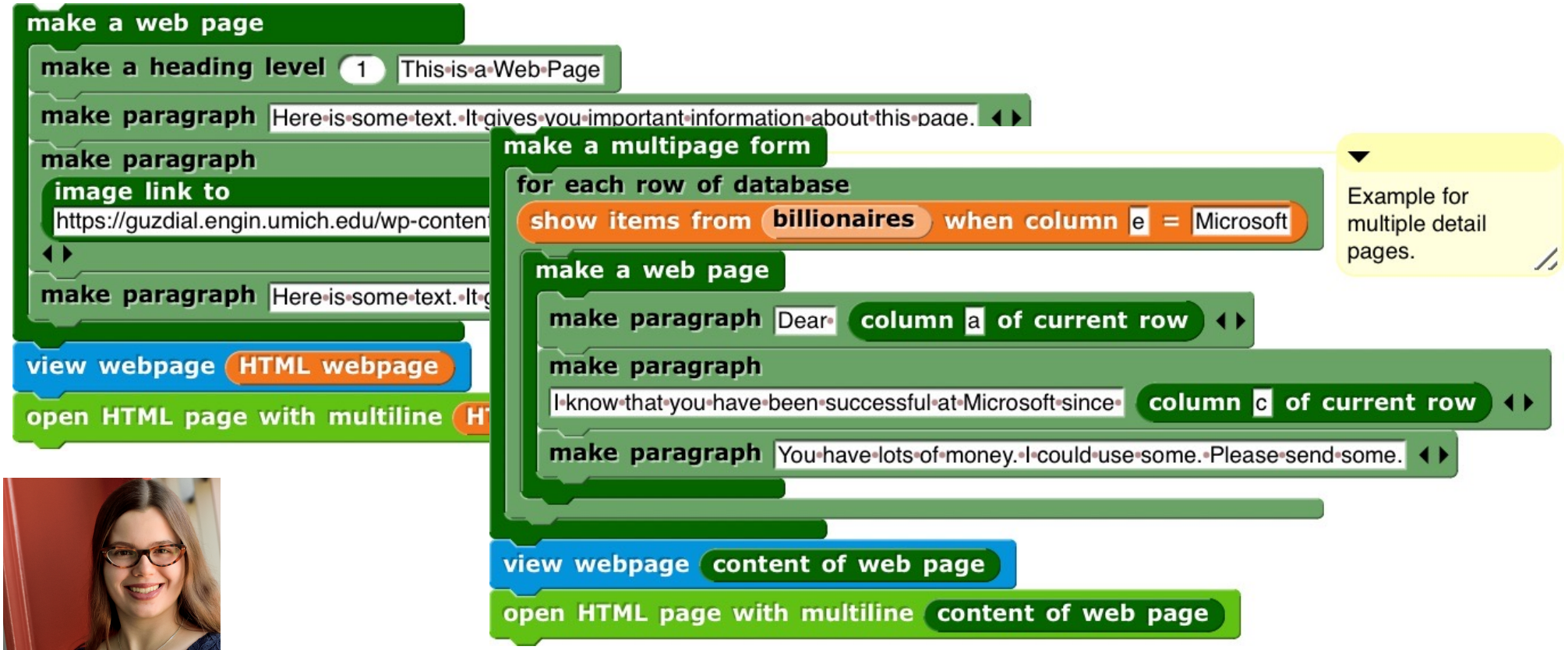
Supporting CSV files as databases



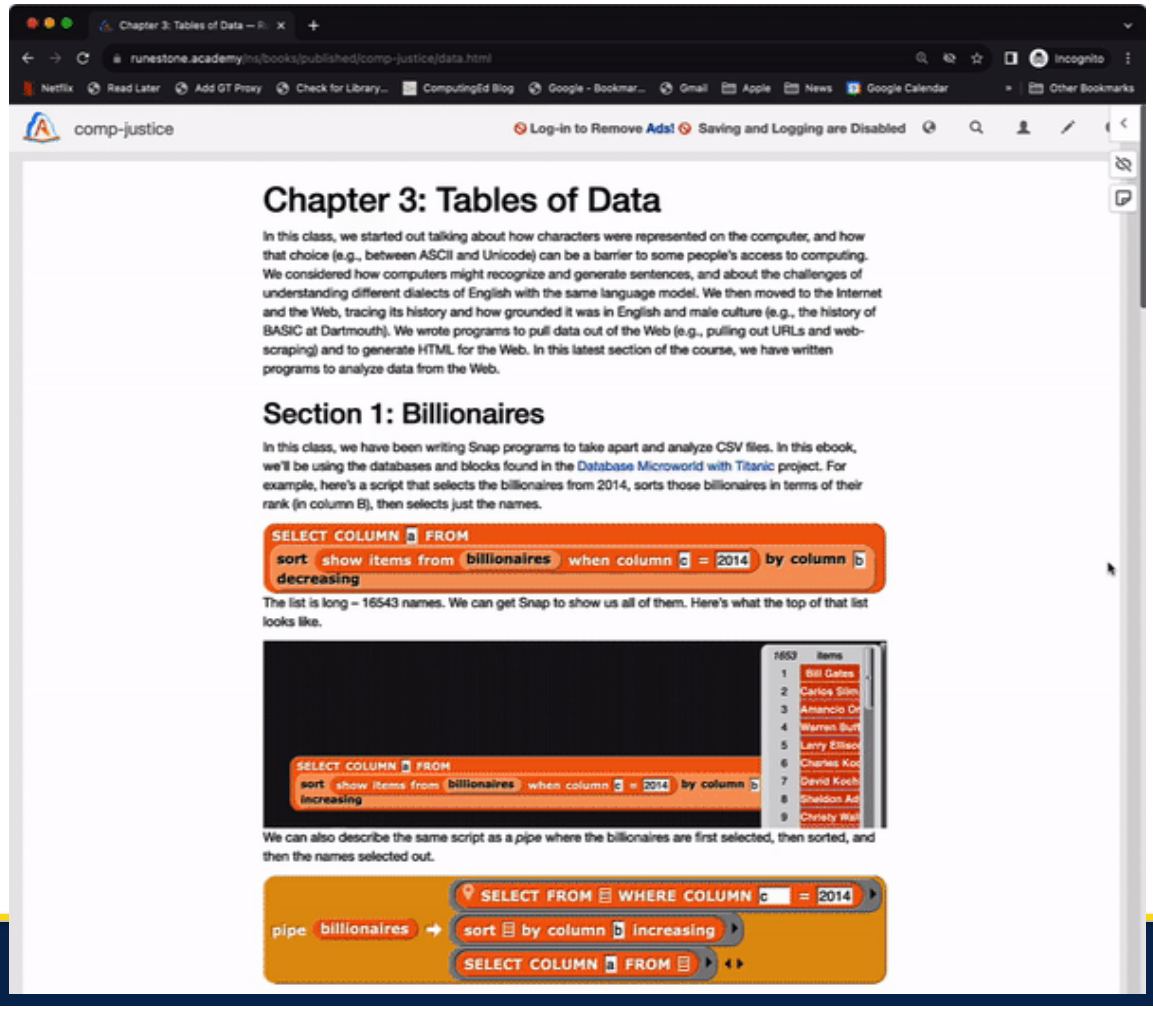
Thanks to Fuchun Wang

But how are we going to do Web pages?

Custom Snap Blocks!



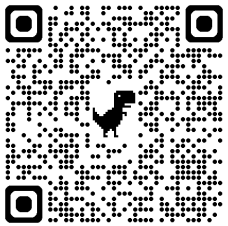
Connecting to SQL in EBook



Summary

Summary

- The founders of computing proposed programming as a tool for learning for ***everyone***.
  - But somewhere along the line, it narrowed to being about getting jobs
- Computing education for ***everyone*** connects to social studies, English/Language Arts, AI, science, mathematics, and art.
  - Will likely include a wide range of programming experiences, including Snap and teaspoon languages
  - Is this an approach for CS that might work in your school?



https://bit.ly/cs4MI2024-PCAS

Workshop after this,
Survey for more information

https://bit.ly/cs4MI2024-PCAS

Programming can be a Tool for Learning Anything

WE NEED TO MAKE COMPUTING ACCESSIBLE TO EVERYONE

Some of the Collaborators on This Work

- Barbara Ericson, Gus Evrard, Kelly Campbell, Miranda Parker, Kathryn Cunningham, Amber Solomon, Bahare Naimipour, Tamara Nelson-Fromm, Emma Dodoo, Tammy Shreiner, Elise Lockwood, Adaline de Chenne.
- Undergraduate researchers: Aryan Bannerjee, Alexandra Rostkowycz, Erin Shi, Brandon Geng, Jessica Zhang, Ben Steinig, Fuchun Yang, Aoife Harte, Chloe Nguyen, Kashmira Reddy, Kristen Taurence, Angela Li, Derrick White, Jessie Houghton.

Thank you!

Workshop after this,
Survey for more information