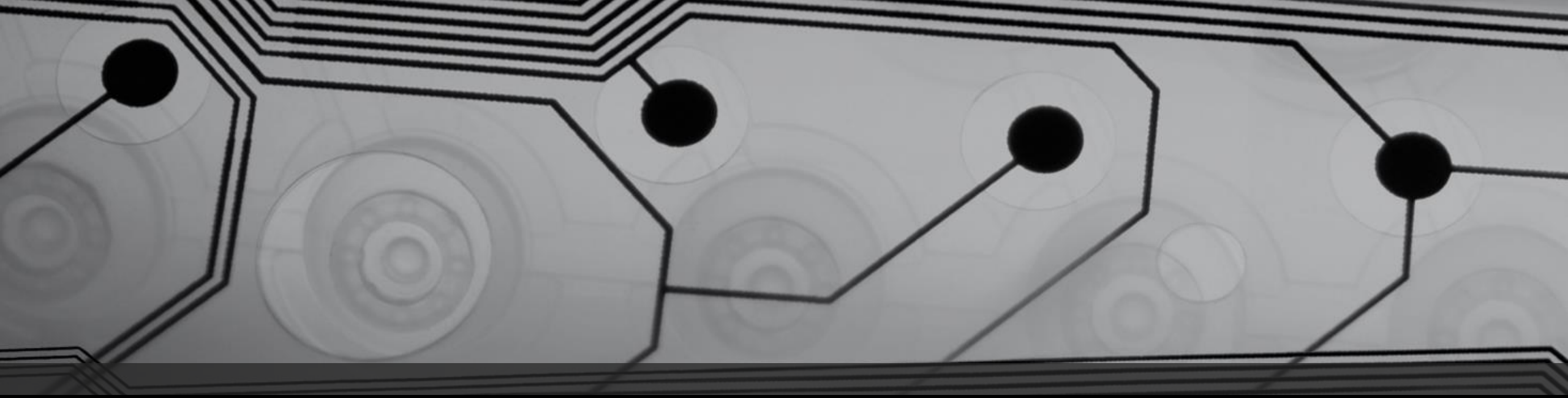


Projects in Python

Marvin G. Fuller, Ph.D.

CTAE RESOURCE NETWORK

python™

Session Overview

- Welcome & Introductions
- Overview of the Day
- Python & Integrated Development Environments (IDEs)
- Understanding “Hello World”
- Break
- Introduction to Python (Introductory Projects)
- **Lunch**
- Projects

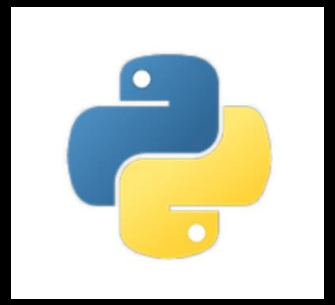
Python?

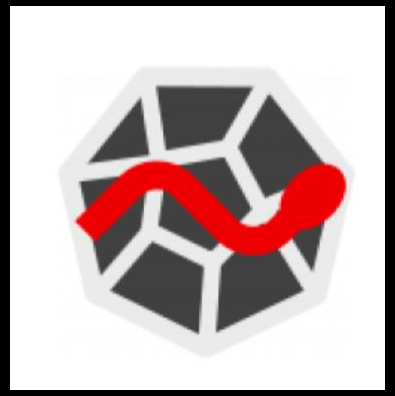
- Python 3.x
- General purpose language
- TONS of specialized libraries making it useful in science applications, web development, game design, cybersecurity, data science, machine learning, etc…
- One of the fastest growing languages (github)
- Easy to learn and use – based off English and simplifies commands

Opensource.com (2018) survey
~10,000 responses

Category	Info	Extra
Age	Majority in 20's & 30's	Representation in all age ranges
Job Types	Over half are developers	Many others uses it in other job roles
Years in Job	Very distributed from < 1 year to 11+ years	~50% full employment and ~50% independent
Money	Mean = $115k	Data science = more $$

Integrated Development Environments
AKA IDEs





Web-based Environments
Repl.it, Anaconda, sourcelair, codeacademy…

Hello World

Our First Program!

Hello World - Analysis

- Initial Setup
- Introduction to the IDE
- IDE Customization
- Introduction to the idea of notes within a program
- Coding conventions
- Exploration
- Write the first program
- Create an error
- Add + explore -> repeat

Variables & Data Types

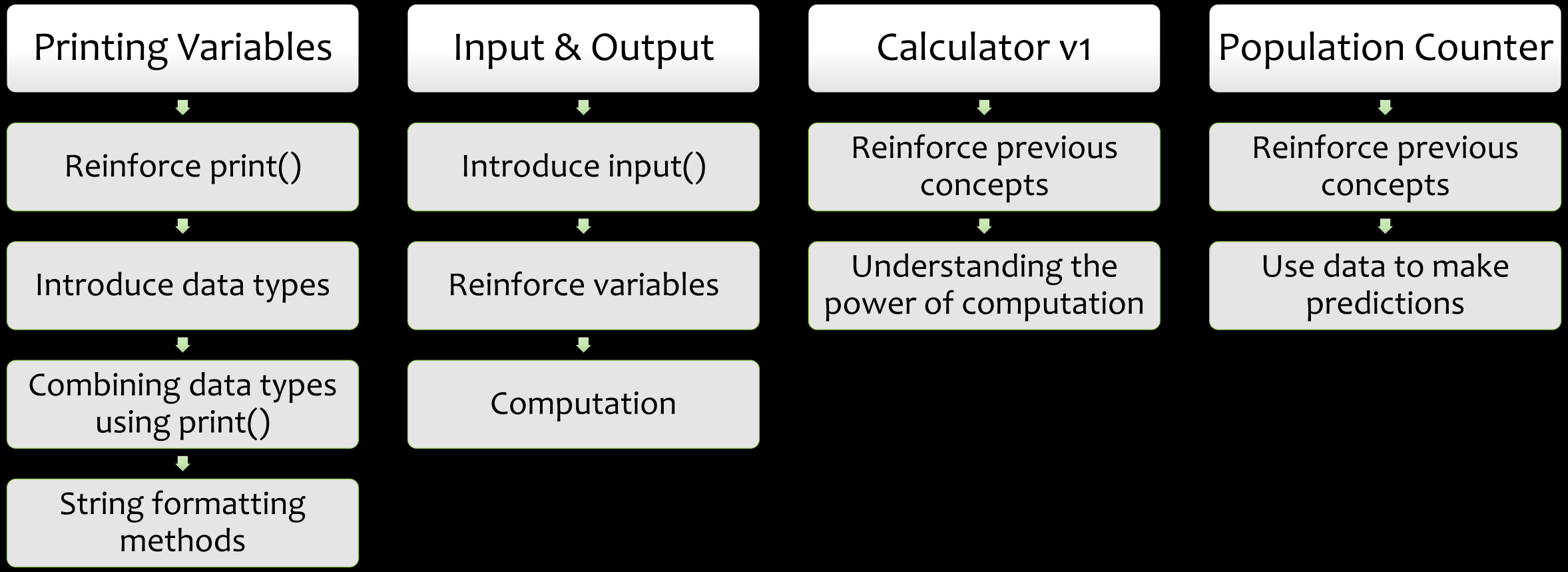
- Variables must be introduced early
- Many ways of teaching the concept of variables in computer science

My Way

- Compare & contrast to math – what do students know about variables and how it is different from math class to computer science.
- Might as well introduce the idea of data types…

Class	Definition	Conversion
String	Any collection of characters and symbols.	str()
Integer	Positive and negative whole numbers	int()
Float	Positive and negative decimal numbers	float()
Boolean	True and False	

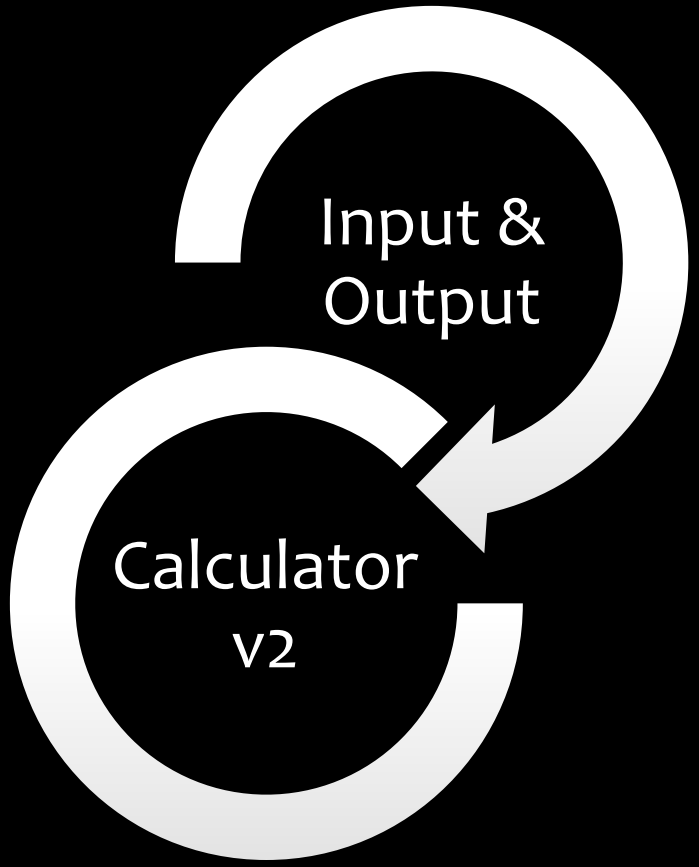
Intro Programs
Sequential Programming



Flow Control
Loops

- Use a for loop to repeat the greeting x times.

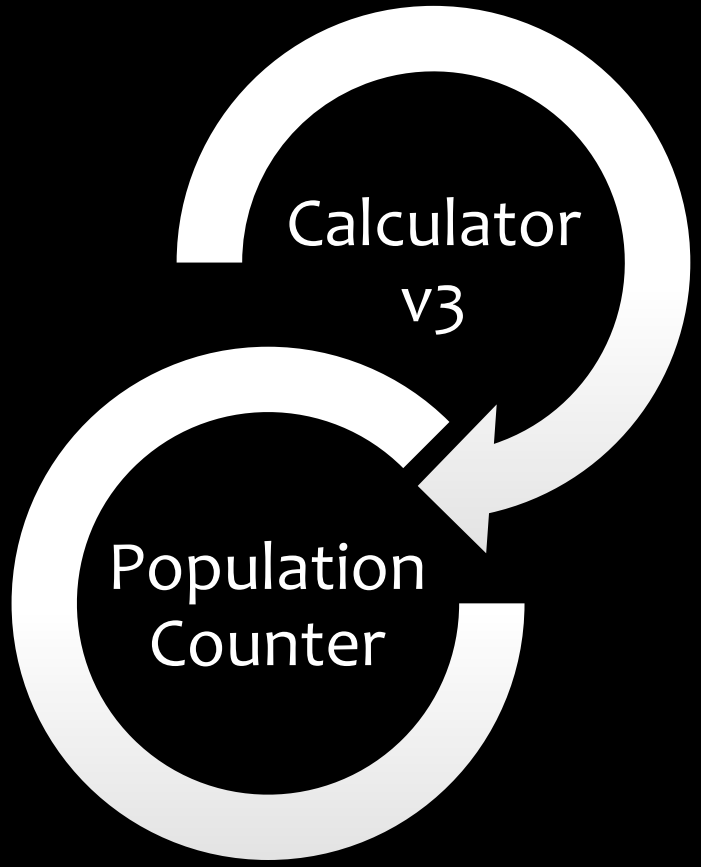
Flow Control
Conditional Statements



- Detect the user & respond appropriately
- Detect if the age is an integer

- Detect if the value entered is a number

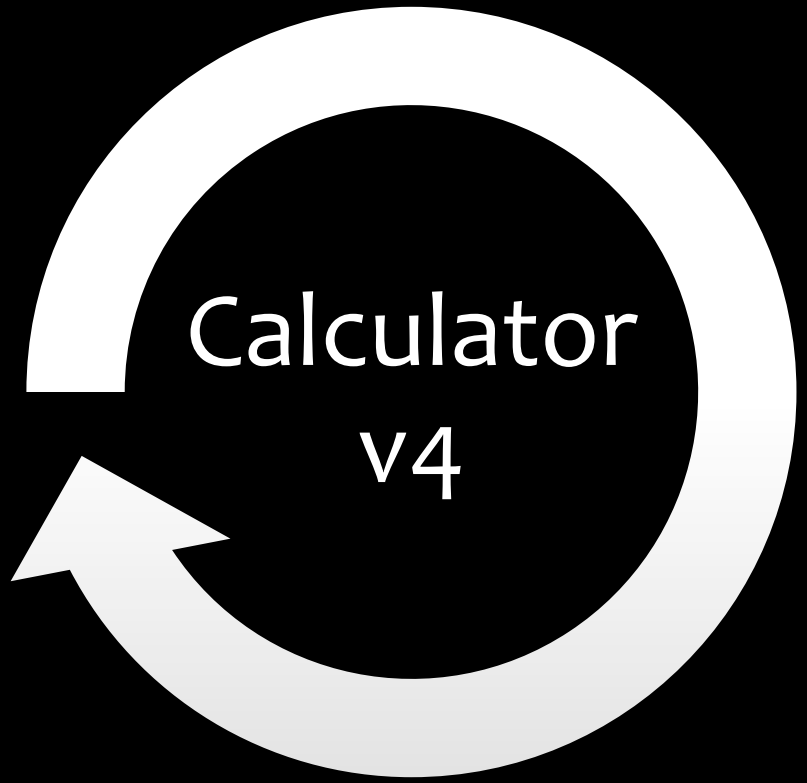
Flow Control
Combining Loops & Conditionals



- Add an infinite loop
- Ask the user if they would like to calculate again

- Same as the calculator
- Practicing the addition of user interaction

Flow Control Functions



- Add a menu function
- Move calculation to a function
- Add multiple calculations to the same program
- Use a loop to repeat until the user quits
- Use conditionals to call the correct function

LUNCH!!!!

Recap & Upcoming

Prior Knowledge

- Basic input and output
- Data types and variable usage
- Conditional Statements
- Loops
- Functions

Coming up next

- Libraries
- String functions
- Data Structures (Lists)
- Deeper Computation
- Error Checking

Random Number Guessing Game

Create a game where a random number is generated, then it allows the user to guess.

- Introduce the idea of a library and use the random library
- Begin with a simple right/wrong
- Discuss what would make the game better. Examples include allowing more than one guess, feedback regarding the guess, allowing the player to play again, and keeping track of total wins.
- Add elements in one at a time to continuously improve the game. This is a great way to demonstrate the software cycle and other development practices.

Password Checker

Takes an password from the user and tells the user if the password is "good". If it is not, tell the user what problem exists. This introduces the idea of functions that can be attached a variable.

- Introduce the len() function
  - Report if the password is too short, OK, or strong
- Introduce string functions
  - .isnumeric() – True if the string is all numbers
  - .isalpha() – True if the string is all letters
  - .islower() – True if the string is all lower-case letters
  - .isupper() – True if the string is all upper-case letters

Mean, Median, and Range

Take in numerical values until the user enters a blank value. Report the mean, median, and range.

- Introduce the idea of a list
- Demonstrate appending to the list
- Ensure each value is a number (error checking)
- Introduce the .sort() list function
- Computation
- May add in mode

Random Lists – A random menu

I am bored with eating the same meals for dinner. Create multiple lists containing food items. Generate a random dinner including a main dish and at least two side dishes. Use the random library to select an item from each of the three lists. Display them appropriately.

- This provides practice with accessing a function in a library as well as an item in a list.

Random Story

Your English teacher wants you to write a short story. Unfortunately, you have too many ideas and are not sure which story elements would work well together. Create a program that randomly selects at least five different elements of your story, and then display the completed story. One random element should be the setting or part of the setting. Ensure that any combination of elements works grammatically.

- Provides practice with accessing a function in a library and items in lists.
- This program is a great way to demonstrate how a computer can do work more efficiently than a human.

A Murder Mystery

Similar to the randomized story, this mystery story includes random elements. However, the story is interactive. The setup for the story, including selecting a random criminal, occurs, and then the story is told. The player then must interact with the story to determine the criminal.

- This is a highly expandable project – the story can be randomized, clues can be given, the criminal may lie, etc.
- Instead of only using lists, Tuples may be introduced here for more organization. We create a tuple for each character containing their name; if they are alive; if they are the criminal; a clue they may give; etc.

MATH!!!

Some basic mathematics functions that not only reinforce programming concepts but also support your math teachers.

- Determine if a number is prime
- Find the factors of a number
- Displaying change (Note – this one can be strange)
- Statistics, especially if you did not hit the basics earlier
- Fibonacci sequence calculation
- Logs, transformations, and any other computation through college!!

Cool Stuff with Libraries

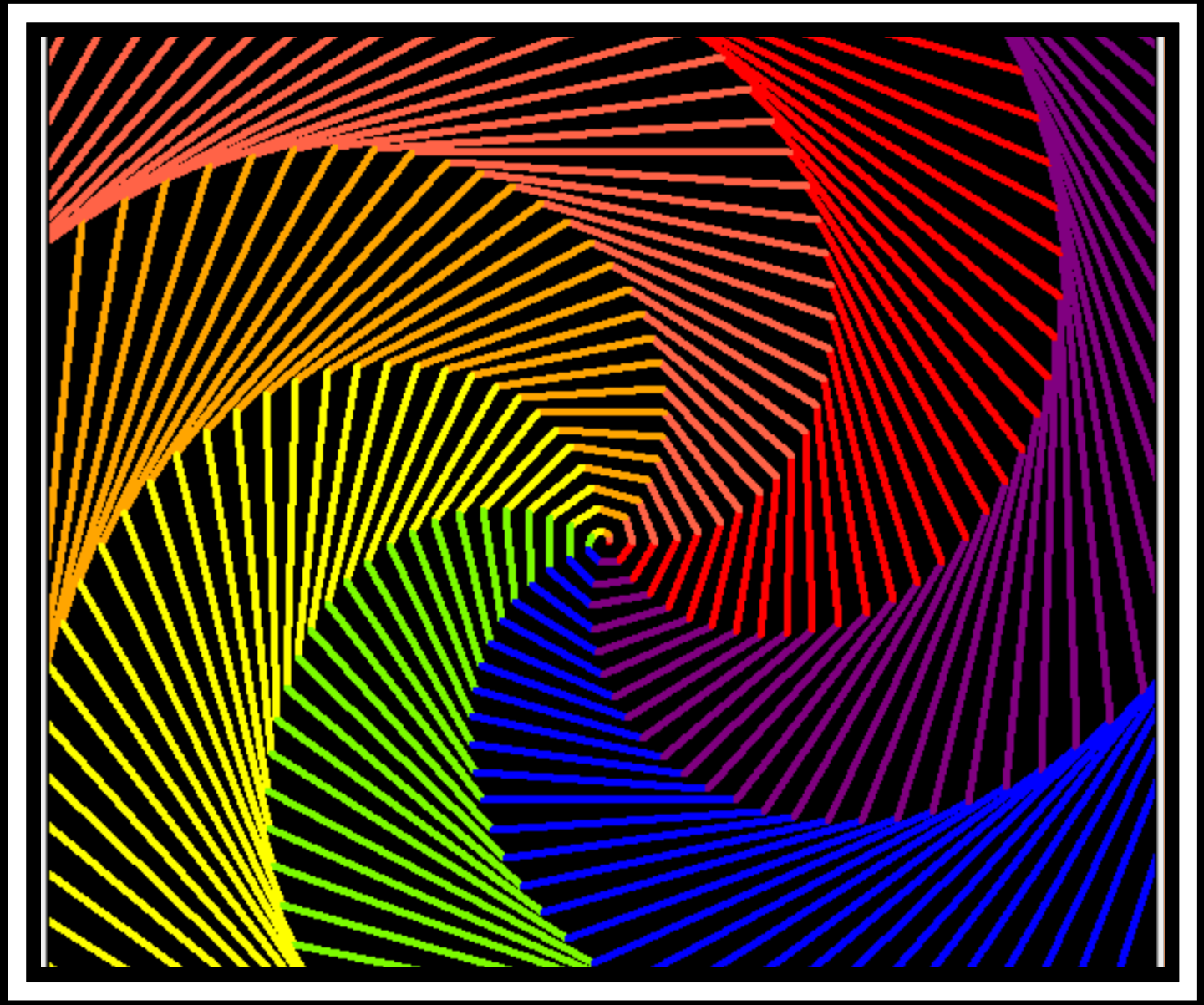
The power of Python exists in all of its libraries. You could easily devote an entire course to a few libraries as well as connect to any area.

Turtles

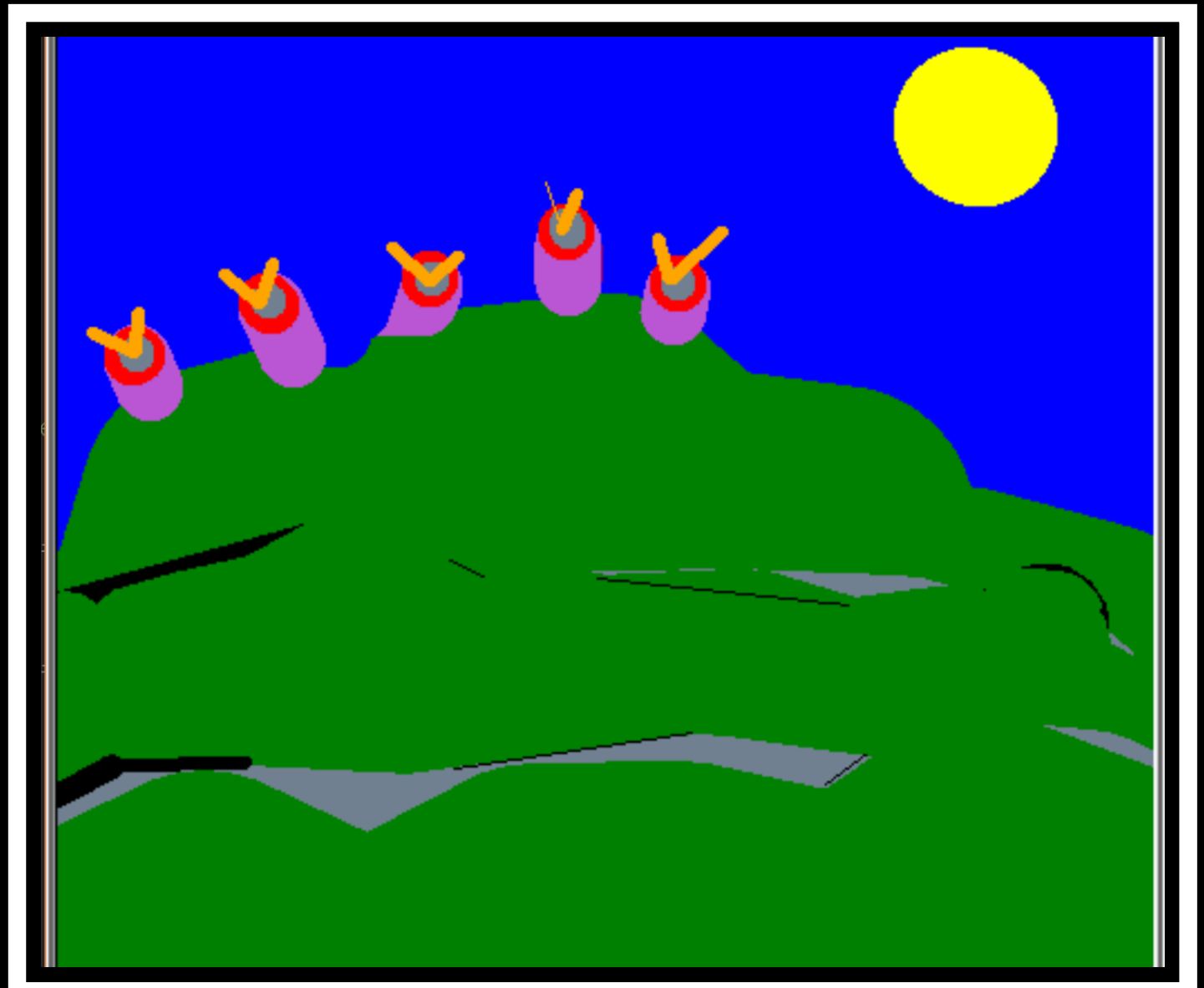
Lots of packaged Python lessons use the turtle library. Why not! Drawing and computer graphics are neat.

Start with some simple drawing commands (lines, circles, dots), then move to more complex.

Students can even make a simple interactive drawing program similar to Paint! We assign a color to each number, use the spacebar to raise and lower the pen, and assign changing the pen width to the arrow keys.



Cool Designs with turtles



Simple
drawing
program

File Manipulation

Students create a menu driven contact list manager.

The program should allow you to load and save the contacts list to a file, as well as allow the user to add and display the contacts list.

Extensions include deleting a contact, searching for a contact, searching for partial matches, including additional information about the contact, etc.

```
My contacts list:


Main Menu:
    A.  Add a new contact
    D.  Display contacts
    O.  Open file
    S.  Save file
    X.  Exit / Quit
Option:
```

Contact List or File Management

```
# Functions
def printMenu ():
    print('1.  List all phone numb
    print('2.  Add a phone number
    print('3.  Delete a phone numb
    print('4.  Lookup a phone numb
    print('5.  Load numbers')
    print('6.  Save numbers')
    print('7.  Exit')
```

Contact List or File Management

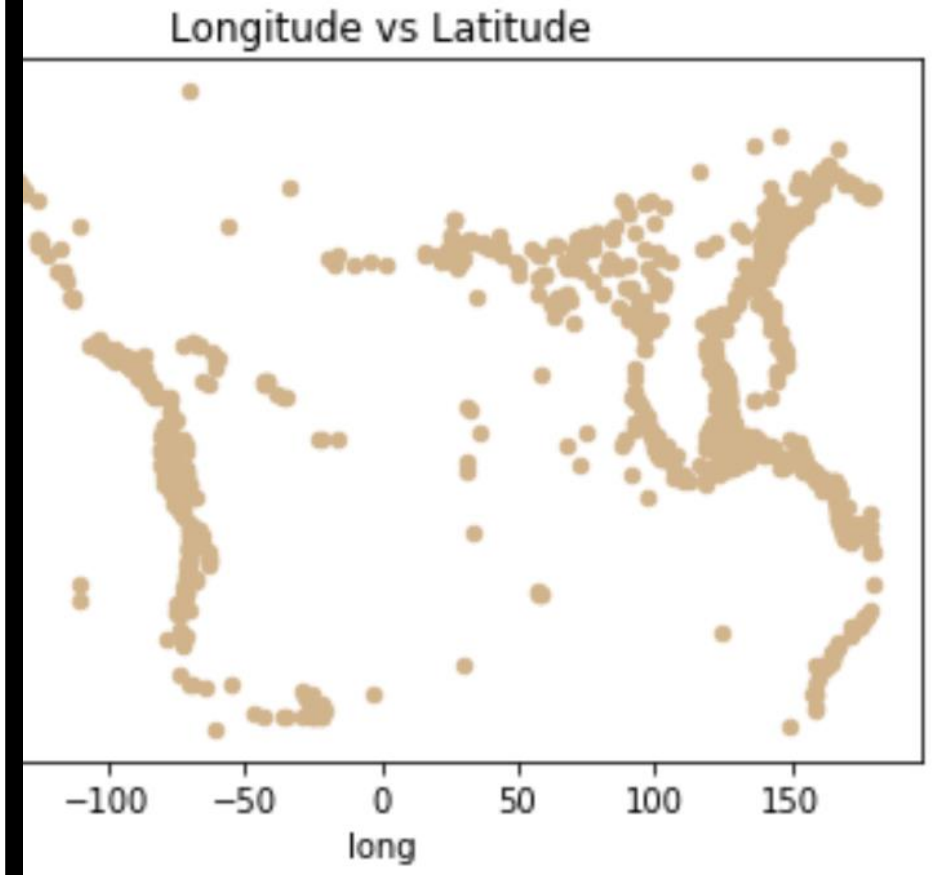
Pandas and matplotlib

Students use a dataset and create multiple visualizations from it.

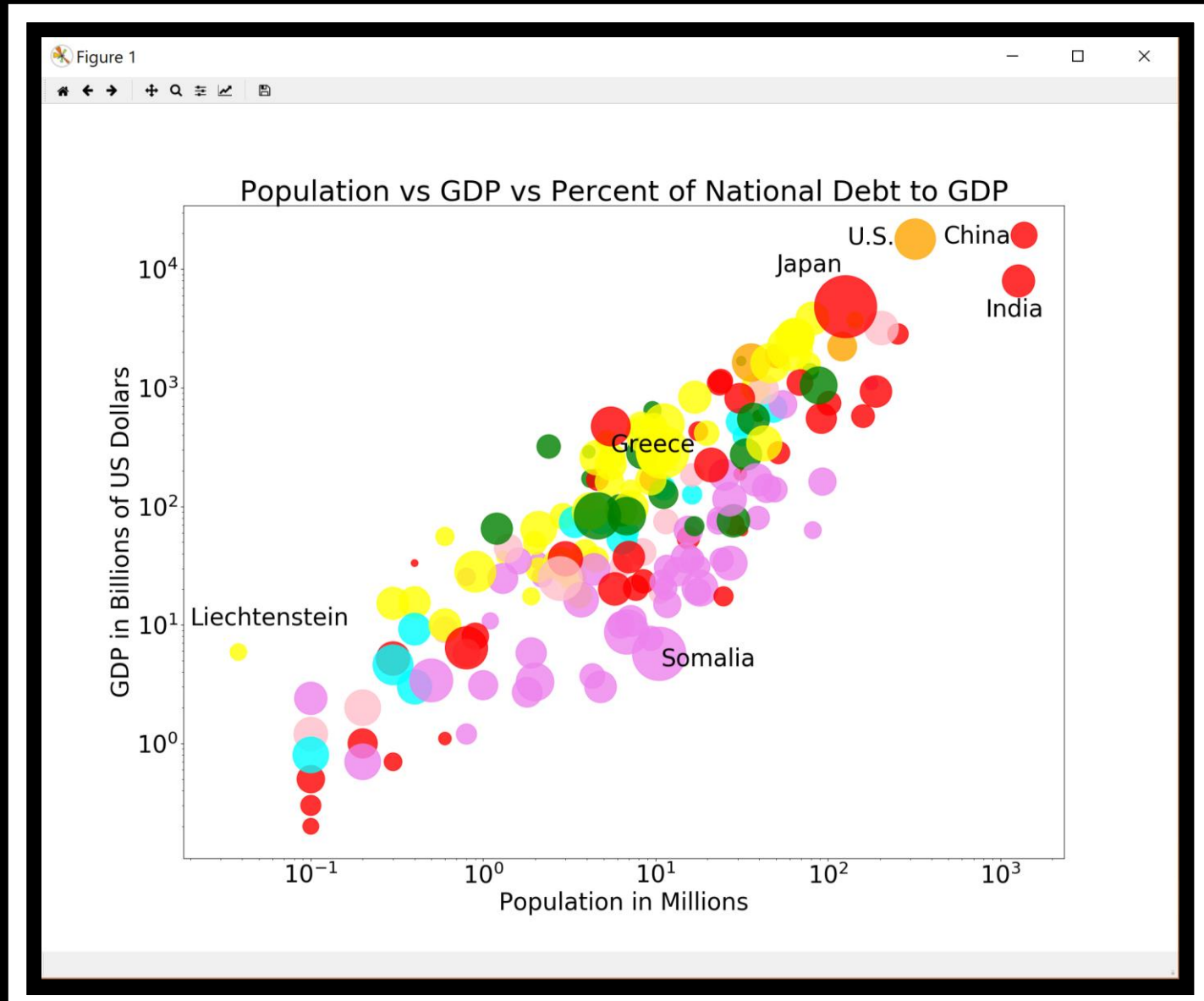
Panads is used to create a better data structure for complex data while matplotlib allows for most graphing needs.

Datasets can be grabbed from ESPN, Socialblade, General Social Survey (GSS), or made up! Use Excel to create a dataset, then export it as a CSV (comma separated values).

```
tter(x='long', y='lat', title = 'Longitude vs Latitude'
```



Using pandas & matplotlib for 2-line graphs



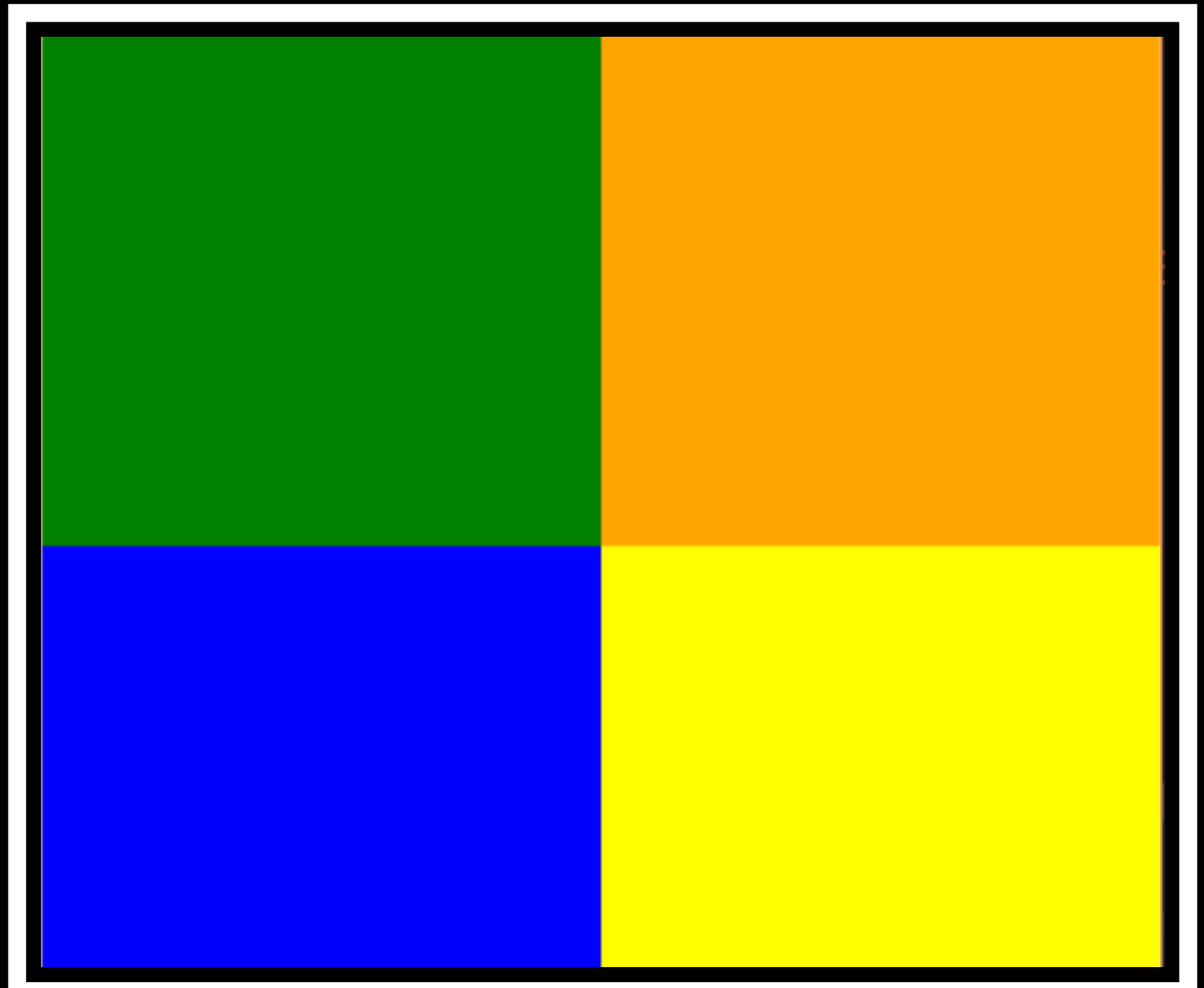
Multiple ways of looking at a dataset

tkinter

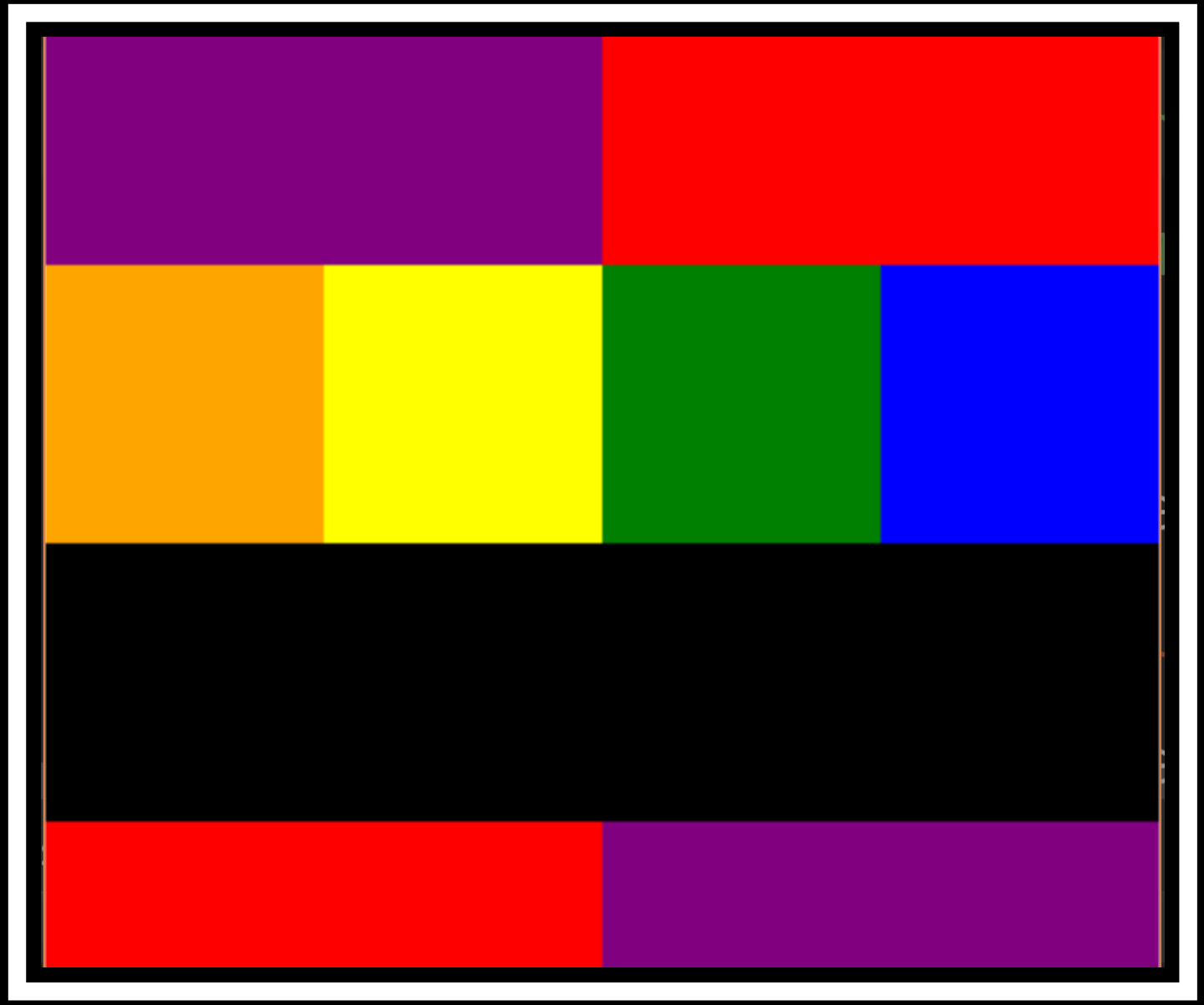
Tkinter is Python's built-in graphical user interface (GUI) system. Students create interactive windows with buttons and input/output boxes.

Any program done before can now be converted to have a graphical interface with buttons, color, and fields.

GUI:
Hello World



Frame
Challenges
Easy



Frame
Challenges
Hard