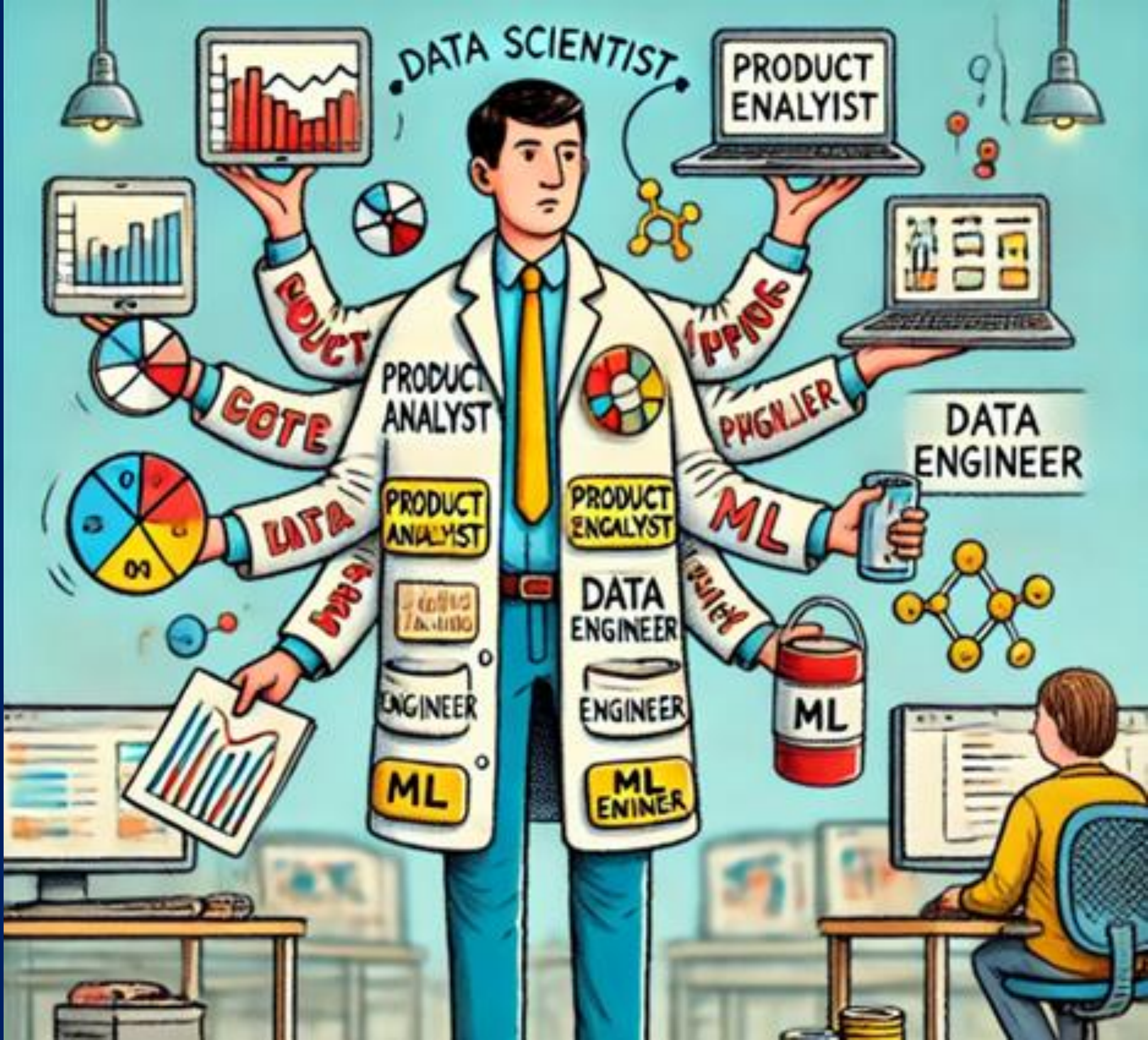
CS 51 Computer Science Principles

Module 3: Data, Internet, Computer and Programming

Unit 3: Programming and Algorithms

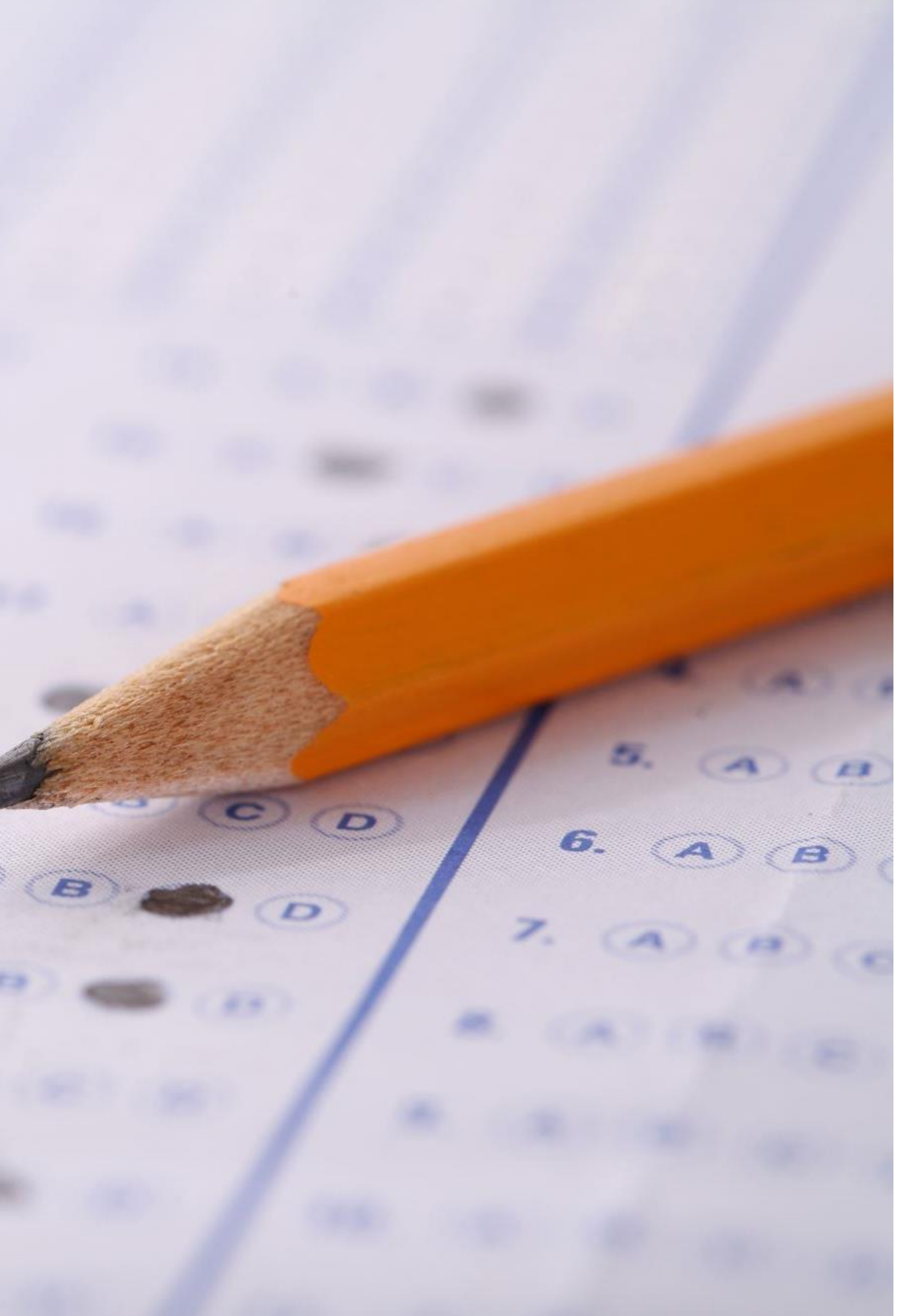
LECTURE 5 APP LAB OVERVIEW

DR. ERIC CHOU
IEEE SENIOR MEMBER



Objectives

- This Big Idea covers the vast majority of the code you'll see on the AP test in the spring. It describes basic components of most programming languages such as variables, lists, and procedures.
- Unlike AP Comp Sci A, which only teaches Java, there's no programming language specification for AP CSP. Your teacher could use a block-based language like Scratch or a text-based language like Python. In order to accommodate for these differences, The AP CSP test uses a basic **Pseudocode**, or a simplified programming language.
- The College Board's Pseudocode shares many similarities with the coding language Python, which is used to help write examples across this guide.
- All photos of Pseudocode come from the Exam Reference Sheet on page 214 of the CED, found here.



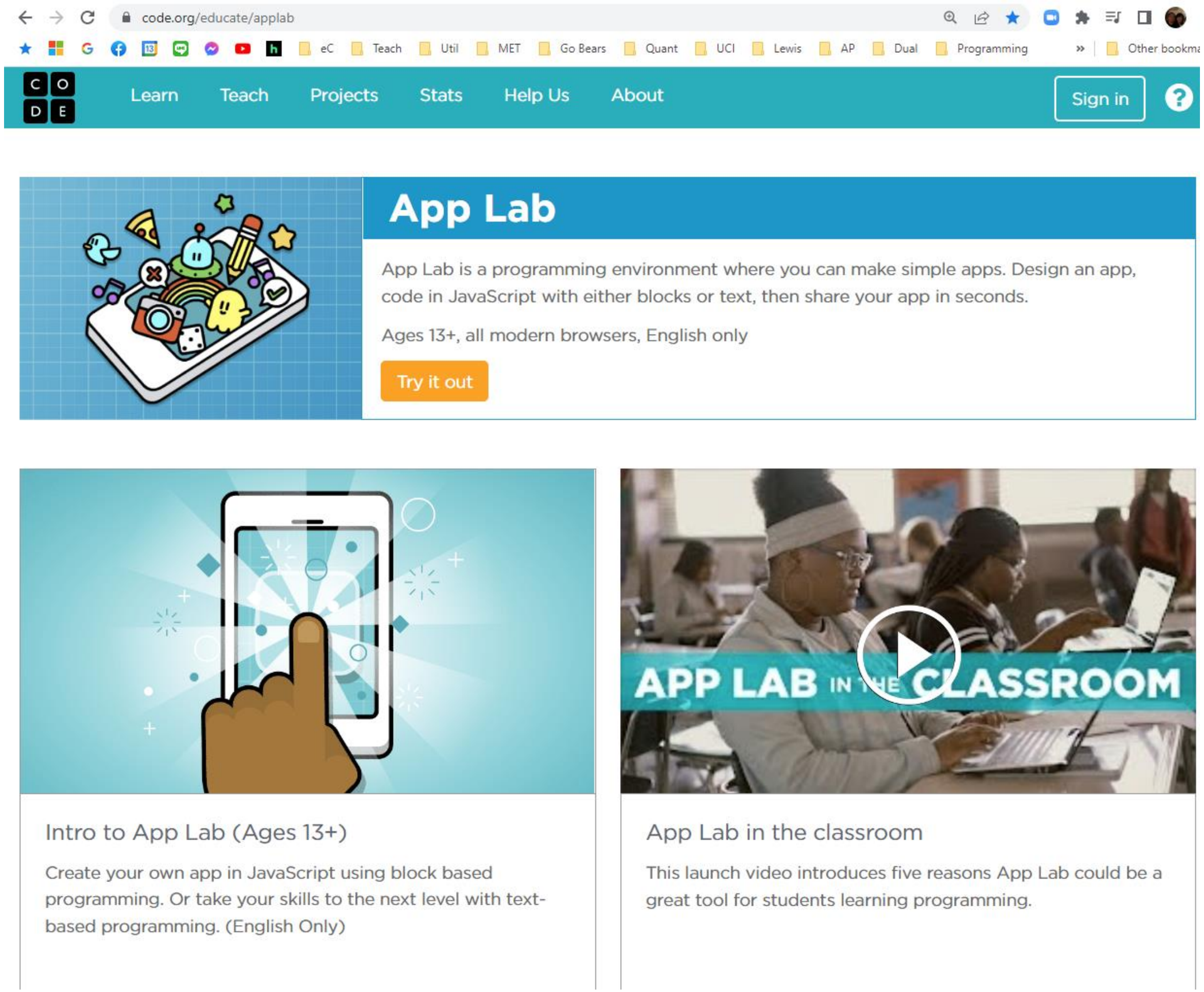
Unit Overview

Exam Weighing:

- 30-35% of the AP Exam
- Practically, this translates to a good portion of the questions on the test. This unit also makes up the bulk of your final Create project. It's a big part of this course.

Code Studio

FINISH UP IN CODE STUDIO WITH ASSESSMENTS AND REFLECTIONS



App Lab

App Lab is a programming environment where you can make simple apps. Design an app, code in JavaScript with either blocks or text, then share your app in seconds.

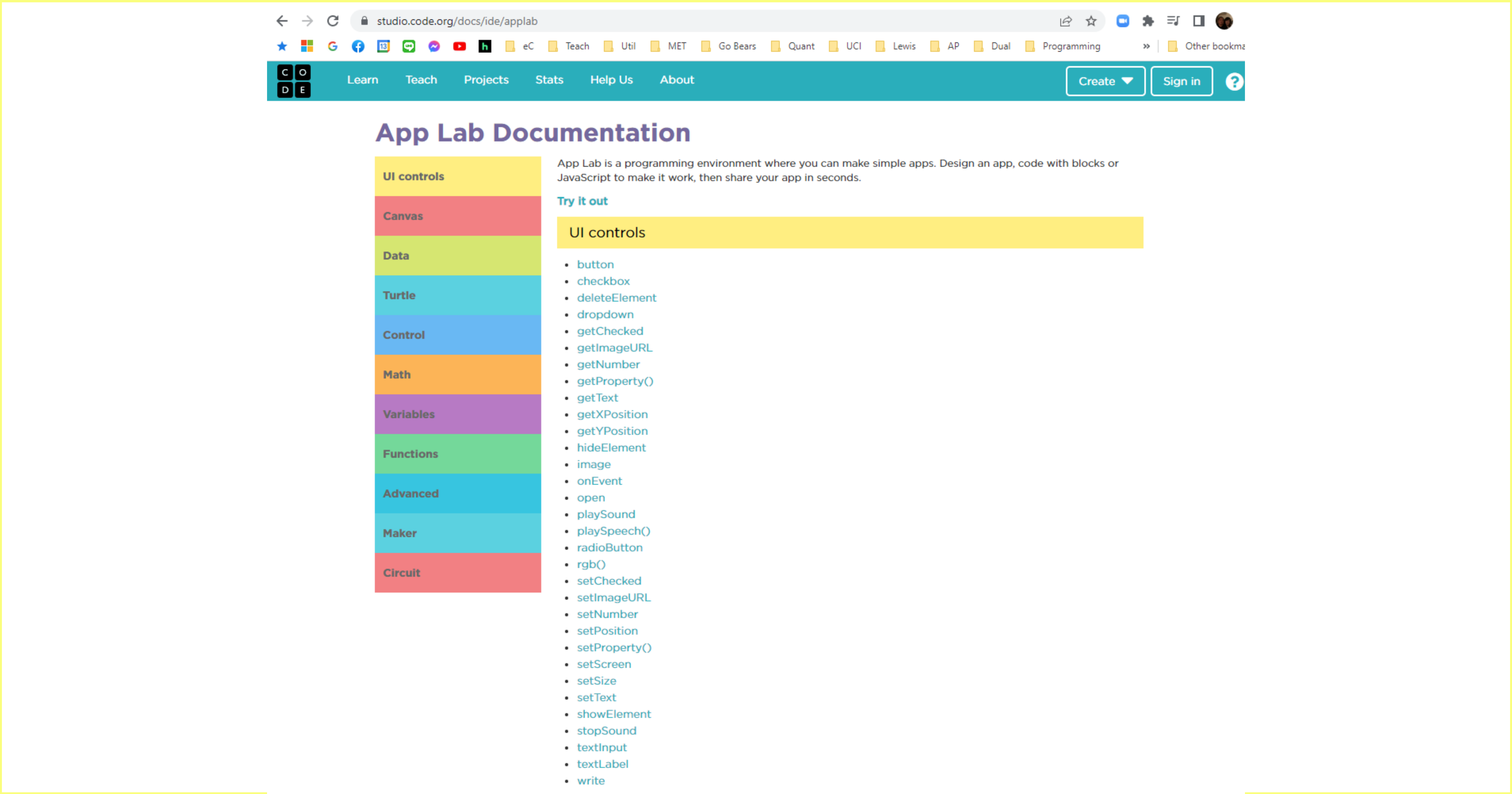
Ages 13+, all modern browsers, English only

Intro to App Lab (Ages 13+)

Create your own app in JavaScript using block based programming. Or take your skills to the next level with text-based programming. (English Only)

App Lab in the classroom

This launch video introduces five reasons App Lab could be a great tool for students learning programming.



App Lab Documentation

App Lab is a programming environment where you can make simple apps. Design an app, code with blocks or JavaScript to make it work, then share your app in seconds.

Try it out

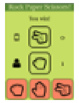
UI controls

- button
- checkbox
- deleteElement
- dropdown
- getChecked
- getImageURL
- getNumber
- getProperty()
- getText
- getXPosition
- getYPosition
- hideElement
- image
- onEvent
- open
- playSound
- playSpeech()
- radioButton
- rgb()
- setChecked
- setImageURL
- setNumber
- setPosition
- setProperty()
- setScreen
- setSize
- setText
- showElement
- stopSound
- textInput
- textLabel
- write

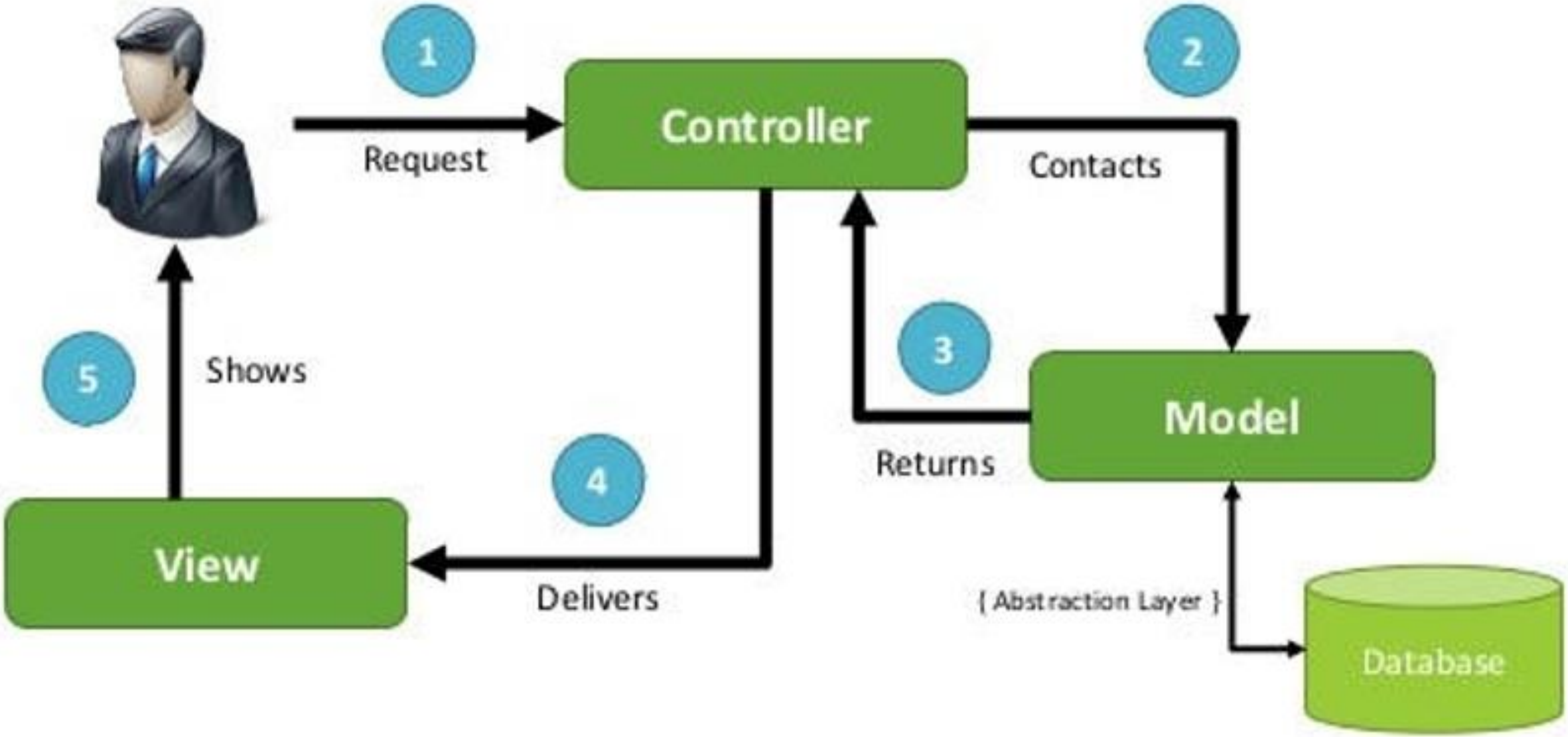
Introduction to App Lab

LECTURE 1

Computer Science Principles units that use App Lab

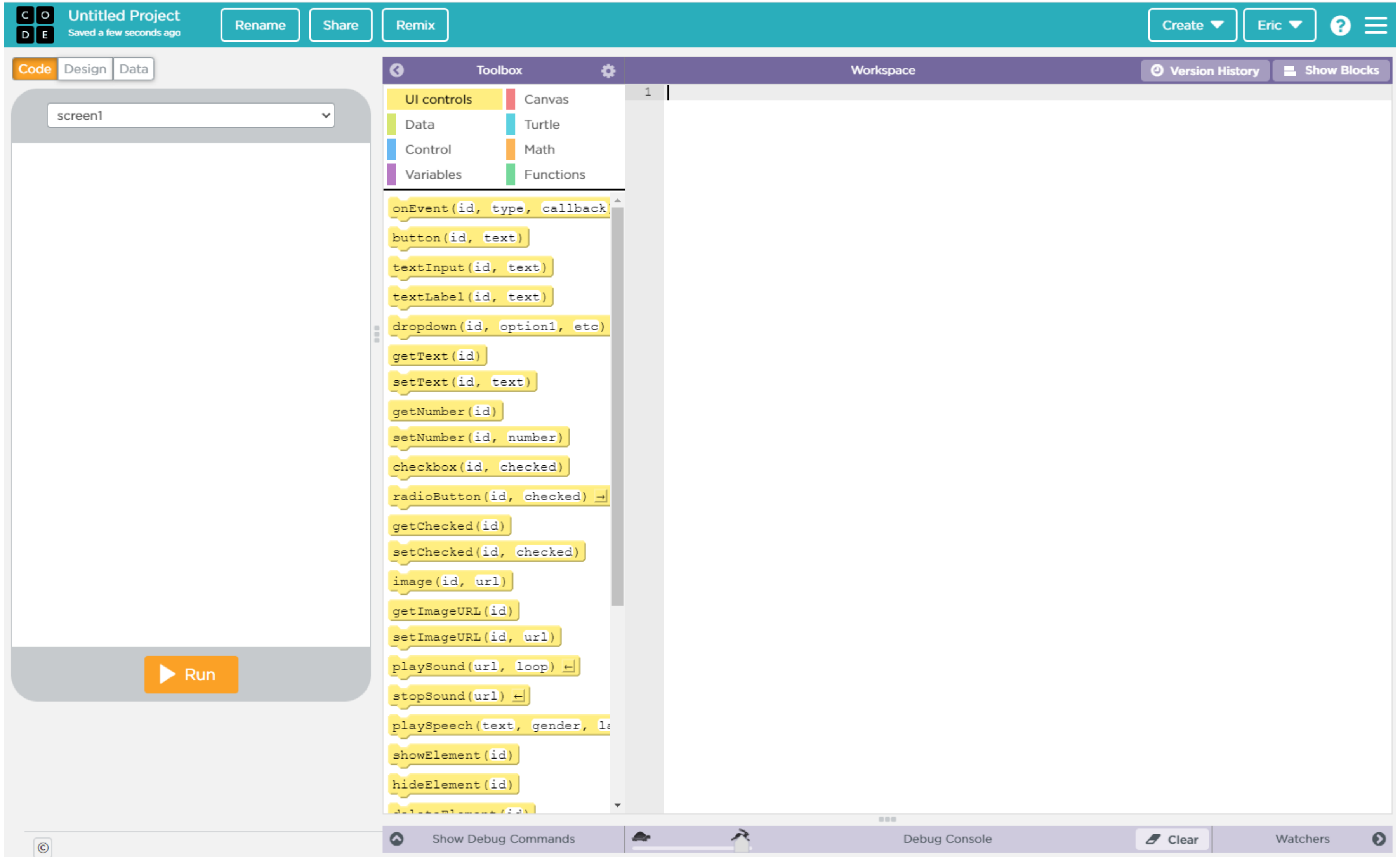
Link	Unit description
 Unit 3	Intro to App Design Students design their first app while learning both fundamental programming concepts and collaborative software development processes. Students work with partners to develop a simple app that teaches classmates about a topic of personal interest. Throughout the unit, they learn how to use Code.org's programming environment, App Lab, to design user interfaces and write simple event-driven programs. Along the way, students learn practices like debugging, pair programming, and collecting and responding to feedback, which they will be able to use throughout the course as they build ever more complex projects. The unit concludes with students sharing the apps they develop with their classmates.
 Unit 4	Variables, Conditions, and Functions Students expand the types of apps they can create as they learn how to store information (variables), make decisions (conditionals), and better organize code (functions). Each programming topic is covered in a specific sequence of lessons that ask students to 'Explore' ideas through hands-on activities, 'Investigate' these ideas through guided code reading, 'Practice' with sample problems, and apply their understanding as they 'Make' a one-day scoped project. The entire unit concludes with a three-day open-ended project in which students must build an app that makes a recommendation about any topic they wish.
 Unit 5	Lists, Loops, and Traversals Students learn to build apps that use and process lists of information. Like the previous unit, students learn the core concepts of lists, loops, and traversals through a series of EIPM lesson sequences. Later in the unit, students are introduced to tools that allow them to import tables of real-world data to help further power the types of apps they can make. At the conclusion of the unit, students complete a week-long project in which they must design an app around a goal of their choosing that uses one of these data sets.
 Unit 7	Parameters, Return, and Libraries Students learn how to design clean and reusable code that can be shared with a single classmate or the entire world. In the beginning of the unit, students are introduced to the concepts of parameters and return, which allow for students to design functions that implement an algorithm. In the second half of the unit, students learn how to design libraries of functions that can be packaged up and shared with others. The unit concludes with students designing their own small library of functions that can be used by a classmate.
 Unit 9	Data Students explore and visualize datasets from a wide variety of topics as they hunt for patterns and try to learn more about the world around them from the data. Once again, students work with datasets in App Lab, but are now asked to make use of a data visualizer tool that assists students in finding data patterns. They learn how different types of visualizations can be used to better understand the patterns contained in datasets and how to use visualizations when investigating hypotheses. At the conclusion of the unit, students learn about the impacts of data analysis on the world around them and complete a final project in which they must uncover and present a data investigation they've completed independently.

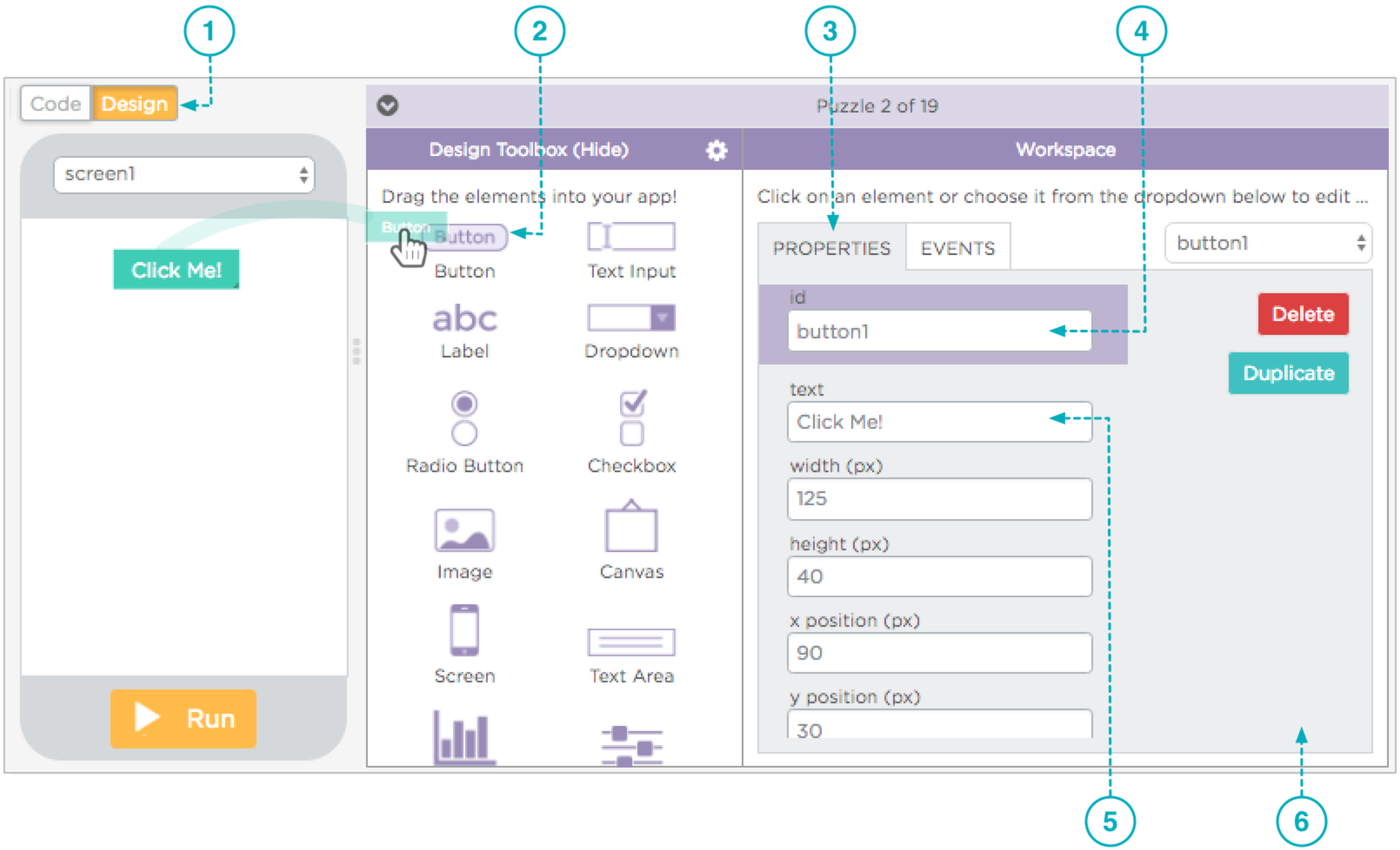
How it works


1
Request
Controller
2
Contacts
3
Returns
Model
{ Abstraction Layer }
Database
4
Delivers
View
5
Shows

Introduction to Design Mode

LECTURE 2


C O D E
Untitled Project
Saved a few seconds ago
Rename
Share
Remix
Create
Eric
Code
Design
Data
screen1
Run
Toolbox
Workspace
Version History
Show Blocks
UI controls
Canvas
Data
Turtle
Control
Math
Variables
Functions
onEvent(id, type, callback
button(id, text)
textInput(id, text)
textLabel(id, text)
dropdown(id, option1, etc)
getText(id)
setText(id, text)
getNumber(id)
setNumber(id, number)
checkbox(id, checked)
radioButton(id, checked)
getChecked(id)
setChecked(id, checked)
image(id, url)
getImageURL(id)
setImageURL(id, url)
playSound(url, loop)
stopSound(url)
playSpeech(text, gender, la
showElement(id)
hideElement(id)
1
Show Debug Commands
Debug Console
Clear
Watchers



id: button1

text: Click Me!

width (px): 125

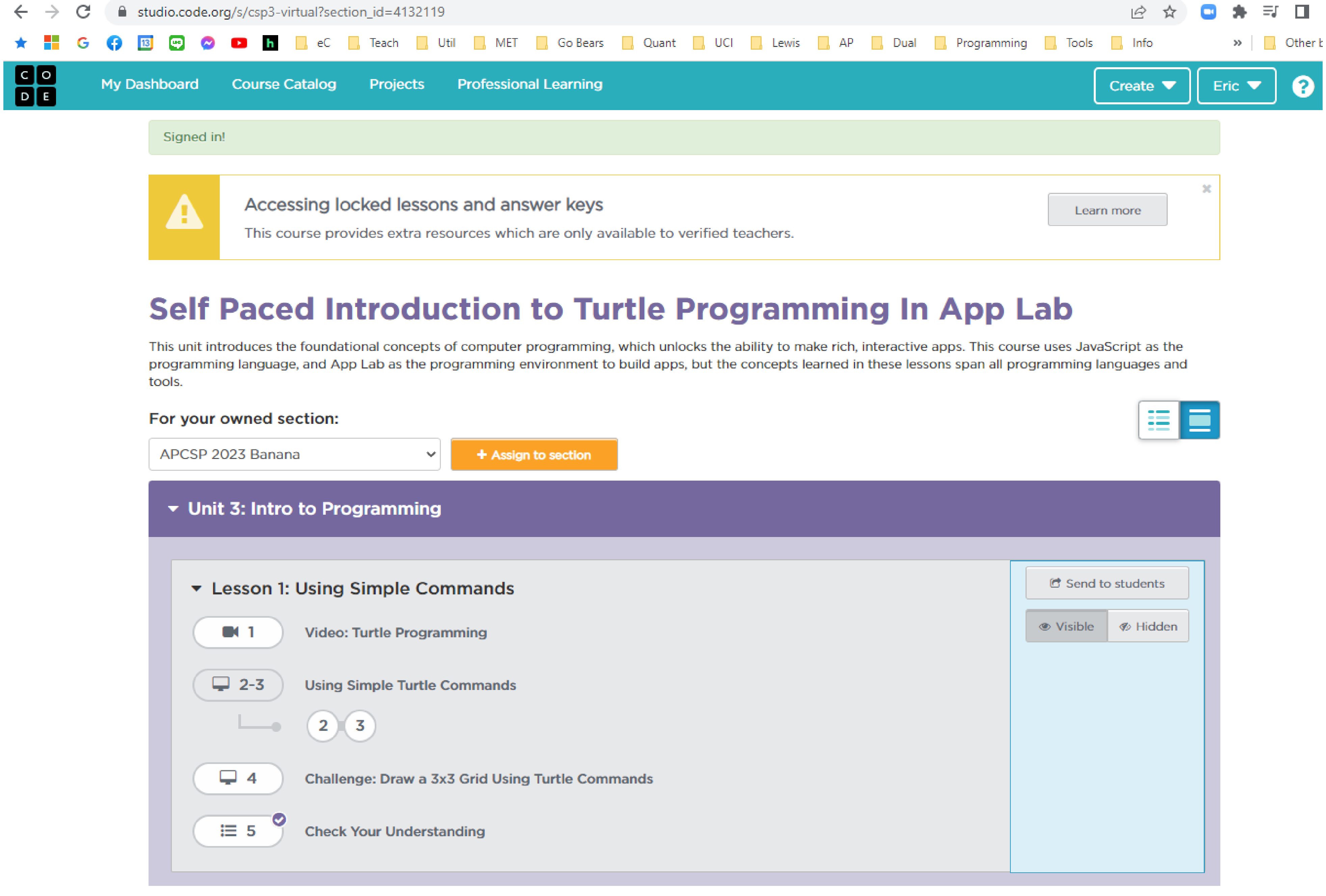
height (px): 40

x position (px): 90

y position (px): 30

Intro to Programming

LECTURE 3



Signed in!

Accessing locked lessons and answer keys

This course provides extra resources which are only available to verified teachers.

Self Paced Introduction to Turtle Programming In App Lab

This unit introduces the foundational concepts of computer programming, which unlocks the ability to make rich, interactive apps. This course uses JavaScript as the programming language, and App Lab as the programming environment to build apps, but the concepts learned in these lessons span all programming languages and tools.

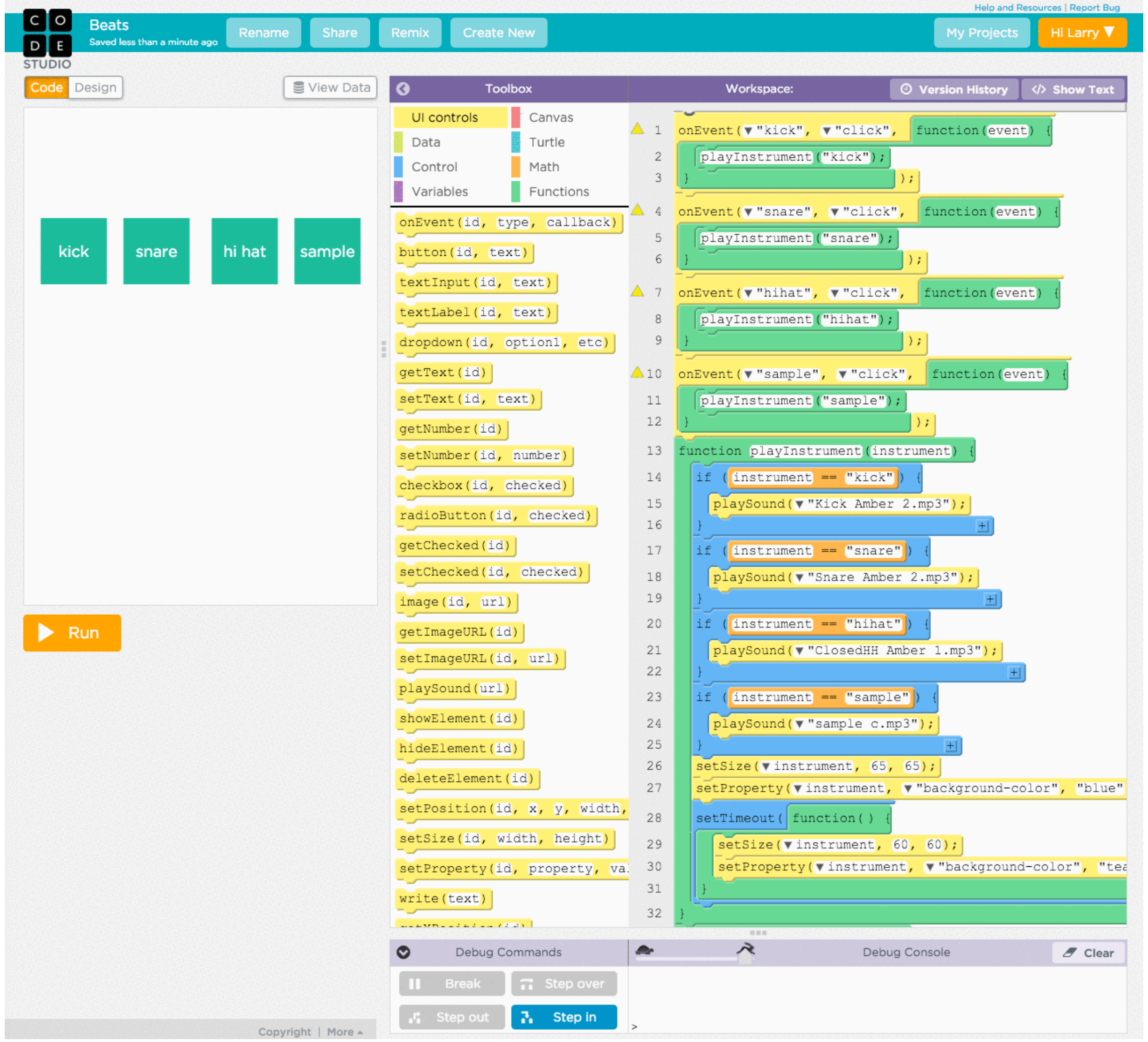
For your owned section:

APCSP 2023 Banana

Unit 3: Intro to Programming

Lesson 1: Using Simple Commands

- 1 Video: Turtle Programming
- 2-3 Using Simple Turtle Commands
  - 2 3
- 4 Challenge: Draw a 3x3 Grid Using Turtle Commands
- 5 ✓ Check Your Understanding

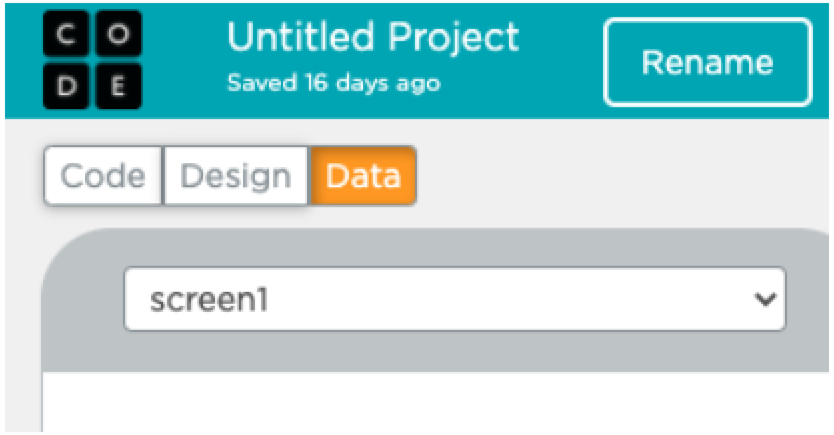


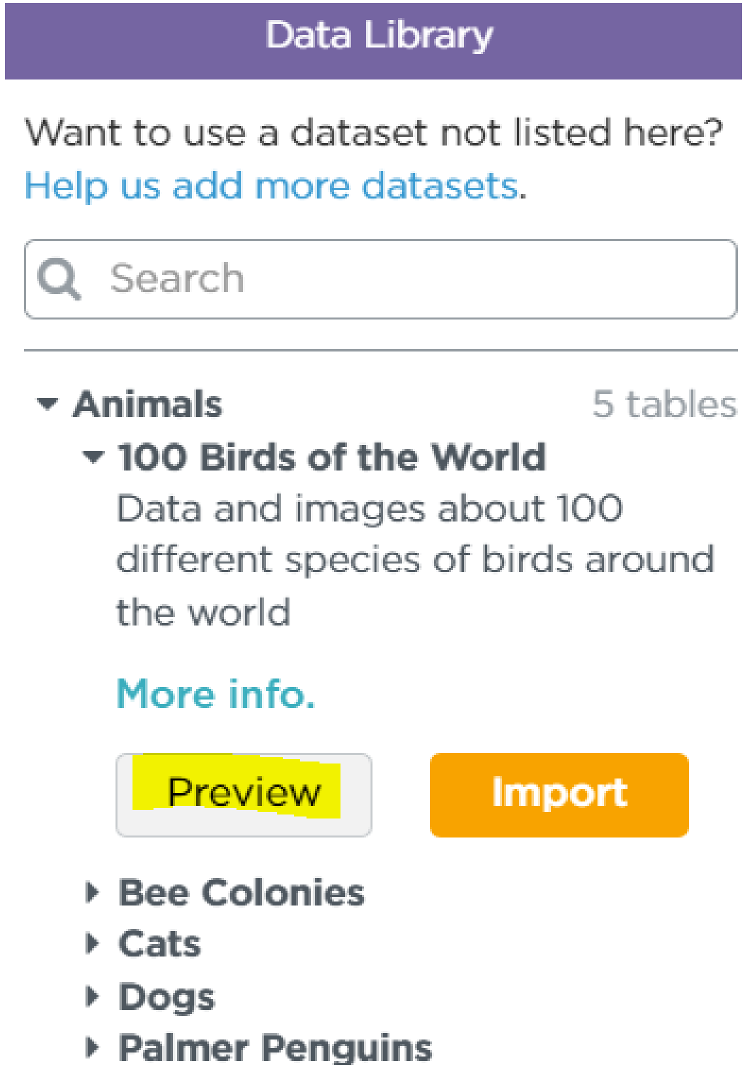
```
onEvent(▼"kick", ▼"click", function(event) {
  playInstrument("kick");
});
onEvent(▼"snare", ▼"click", function(event) {
  playInstrument("snare");
});
onEvent(▼"hihat", ▼"click", function(event) {
  playInstrument("hihat");
});
onEvent(▼"sample", ▼"click", function(event) {
  playInstrument("sample");
});
function playInstrument(instrument) {
  if (instrument == "kick") {
    playSound(▼"Kick Amber 2.mp3");
  }
  if (instrument == "snare") {
    playSound(▼"Snare Amber 2.mp3");
  }
  if (instrument == "hihat") {
    playSound(▼"ClosedHH Amber 1.mp3");
  }
  if (instrument == "sample") {
    playSound(▼"sample c.mp3");
  }
  setSize(▼instrument, 65, 65);
  setProperty(▼instrument, ▼"background-color", "blue"
  setTimeout(function() {
    setSize(▼instrument, 60, 60);
    setProperty(▼instrument, ▼"background-color", "tea
  }
}
```

Database

LECTURE 3

Dataset Design





Data Library

Want to use a dataset not listed here?
Help us add more datasets.

▾ Animals 5 tables

▾ 100 Birds of the World

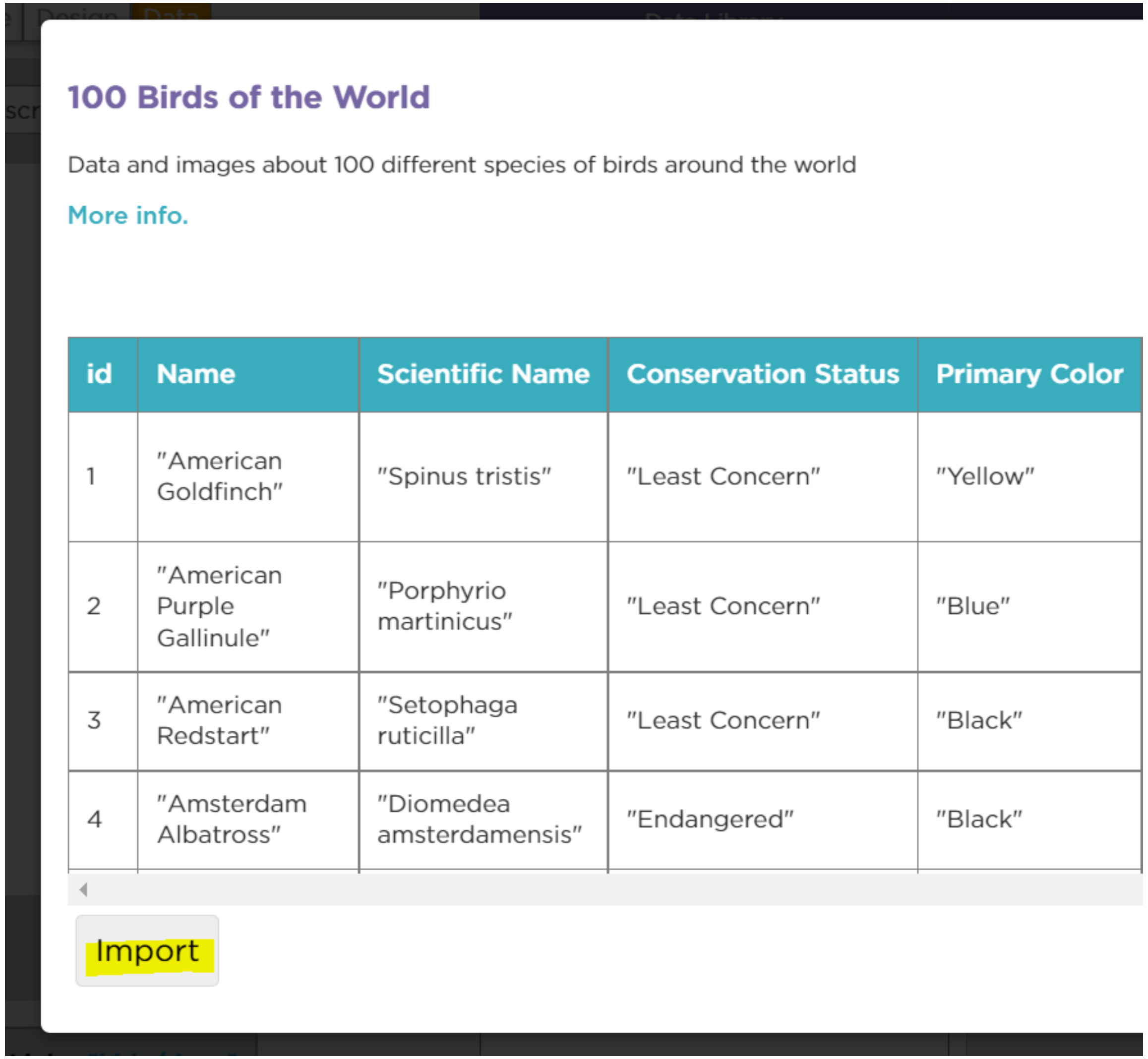
Data and images about 100 different species of birds around the world

More info.

▸ Bee Colonies
▸ Cats
▸ Dogs
▸ Palmer Penguins

Data Library

Importing Data

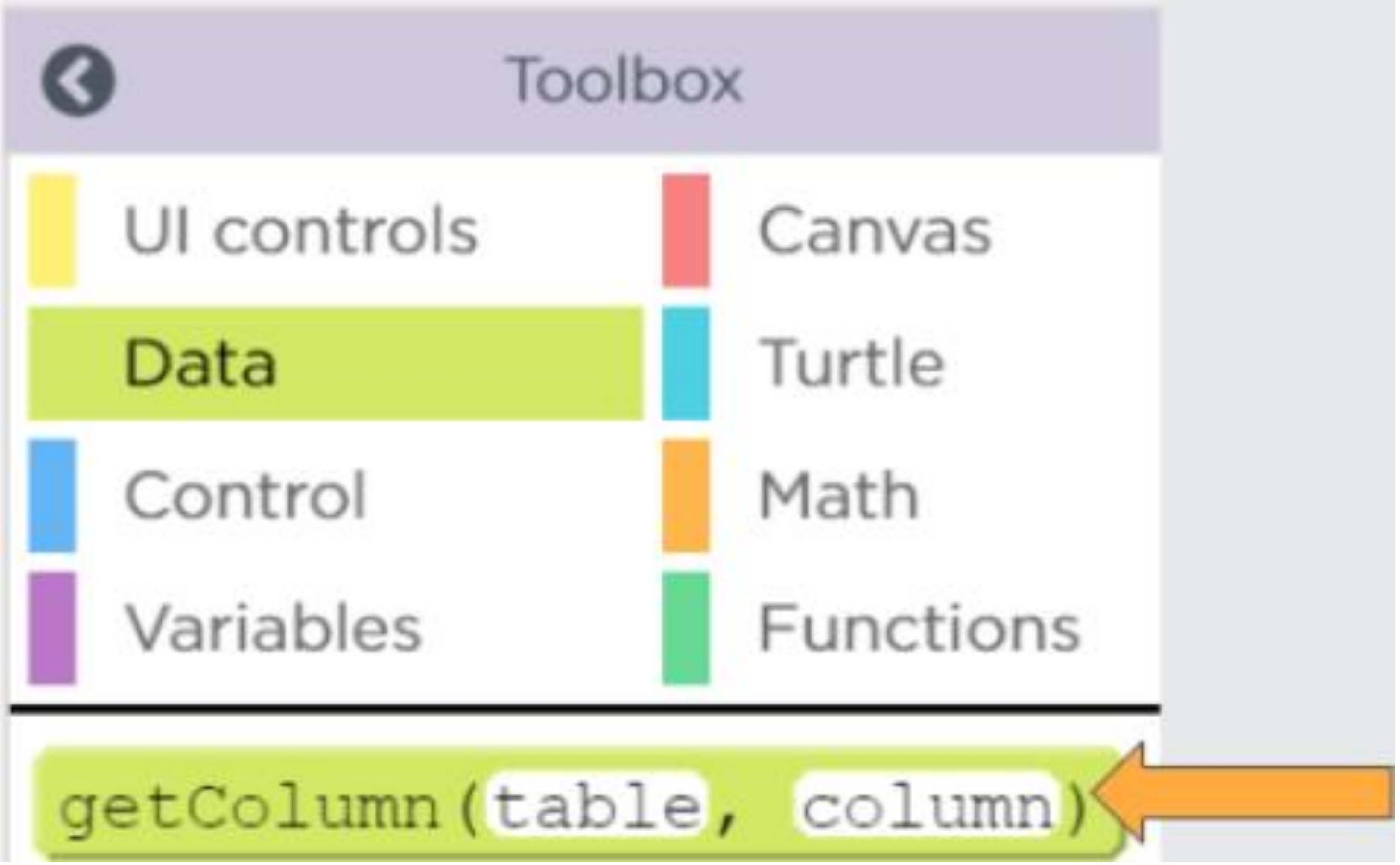


100 Birds of the World

Data and images about 100 different species of birds around the world

More info.

id	Name	Scientific Name	Conservation Status	Primary Color
1	"American Goldfinch"	"Spinus tristis"	"Least Concern"	"Yellow"
2	"American Purple Gallinule"	"Porphyrio martinicus"	"Least Concern"	"Blue"
3	"American Redstart"	"Setophaga ruticilla"	"Least Concern"	"Black"
4	"Amsterdam Albatross"	"Diomedea amsterdamensis"	"Endangered"	"Black"

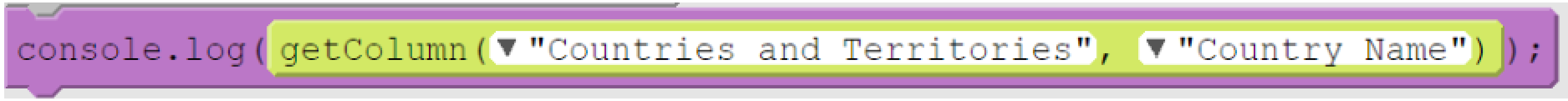


```
getColumn(table, column)
```

Get Column

Data Retrieval

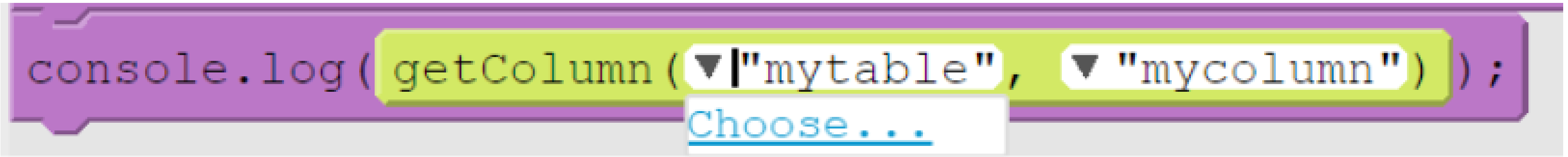
- Your first simple program can be to print out the contents of one of the columns in your table:



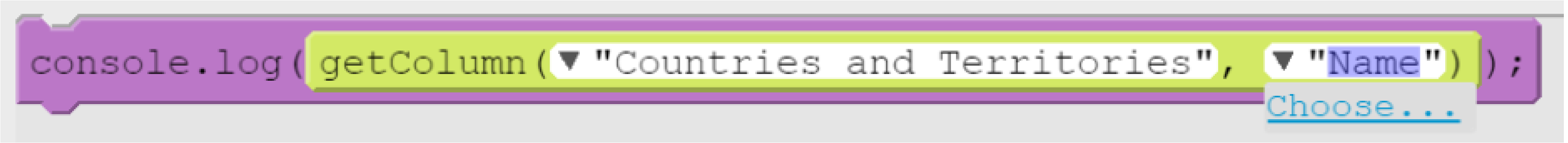
```
console.log(getColumn(▼"Countries and Territories", ▼"Country Name"));
```

Data Retrieval

- Make sure to update the name of the table to the one that you imported and the column name that you want to use (tip: every table has its own unique list of column names):



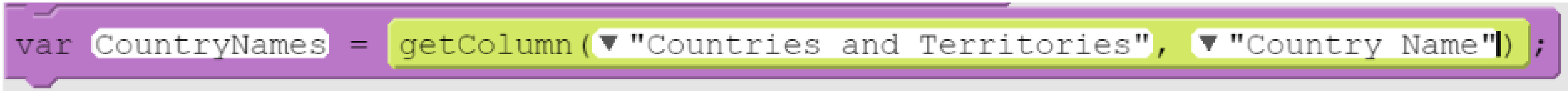
```
console.log(getColumn("mytable", "mycolumn"));
```



```
console.log(getColumn("Countries and Territories", "Name"));
```

Data Retrieval

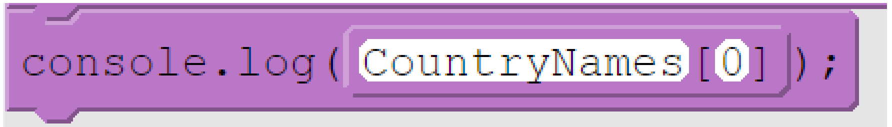
- You can also store the content of the column of data in your table in a variable so you can reference it in other parts of your program:



```
var CountryNames = getColumn(▼ "Countries and Territories", ▼ "Country Name");
```

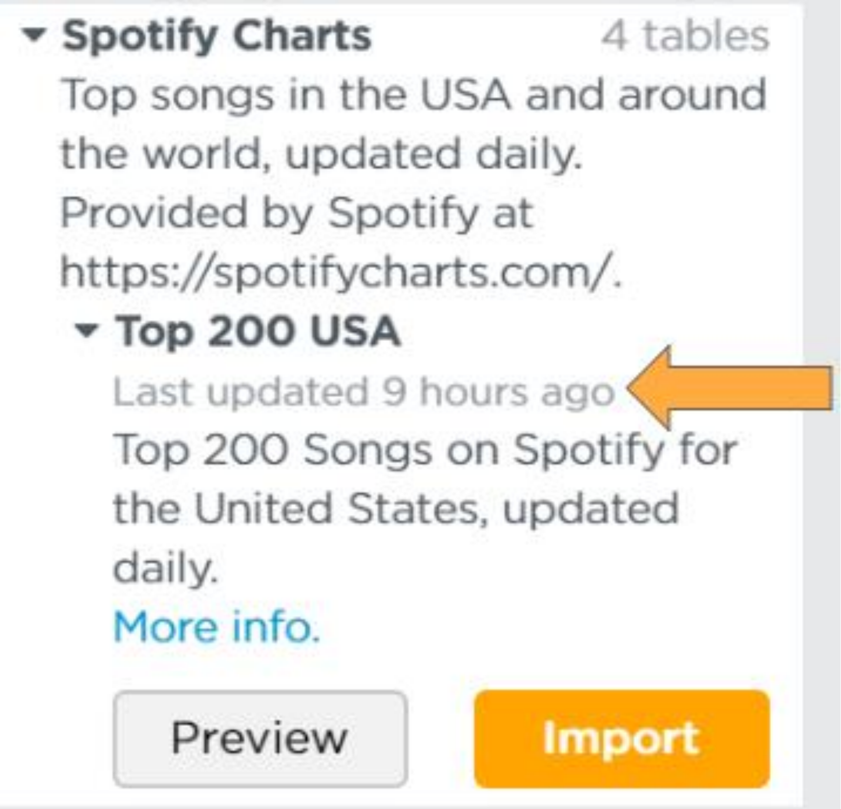
Data Retrieval

- Here, we are accessing and printing the first item in the "CountryNames" variable (which contains a list of all the Country Names in the "Countries and Territories" table):



```
console.log(CountryNames[0]);
```

Additional Information



Spotify Charts 4 tables

Top songs in the USA and around the world, updated daily. Provided by Spotify at https://spotifycharts.com/.

Top 200 USA

Last updated 9 hours ago

Top 200 Songs on Spotify for the United States, updated daily.

More info.

- Some of the datasets are "live" and are updated on a regular basis. You can see when a live dataset was last updated by the timestamp. If there is no timestamp included, this dataset is not being updated regularly:

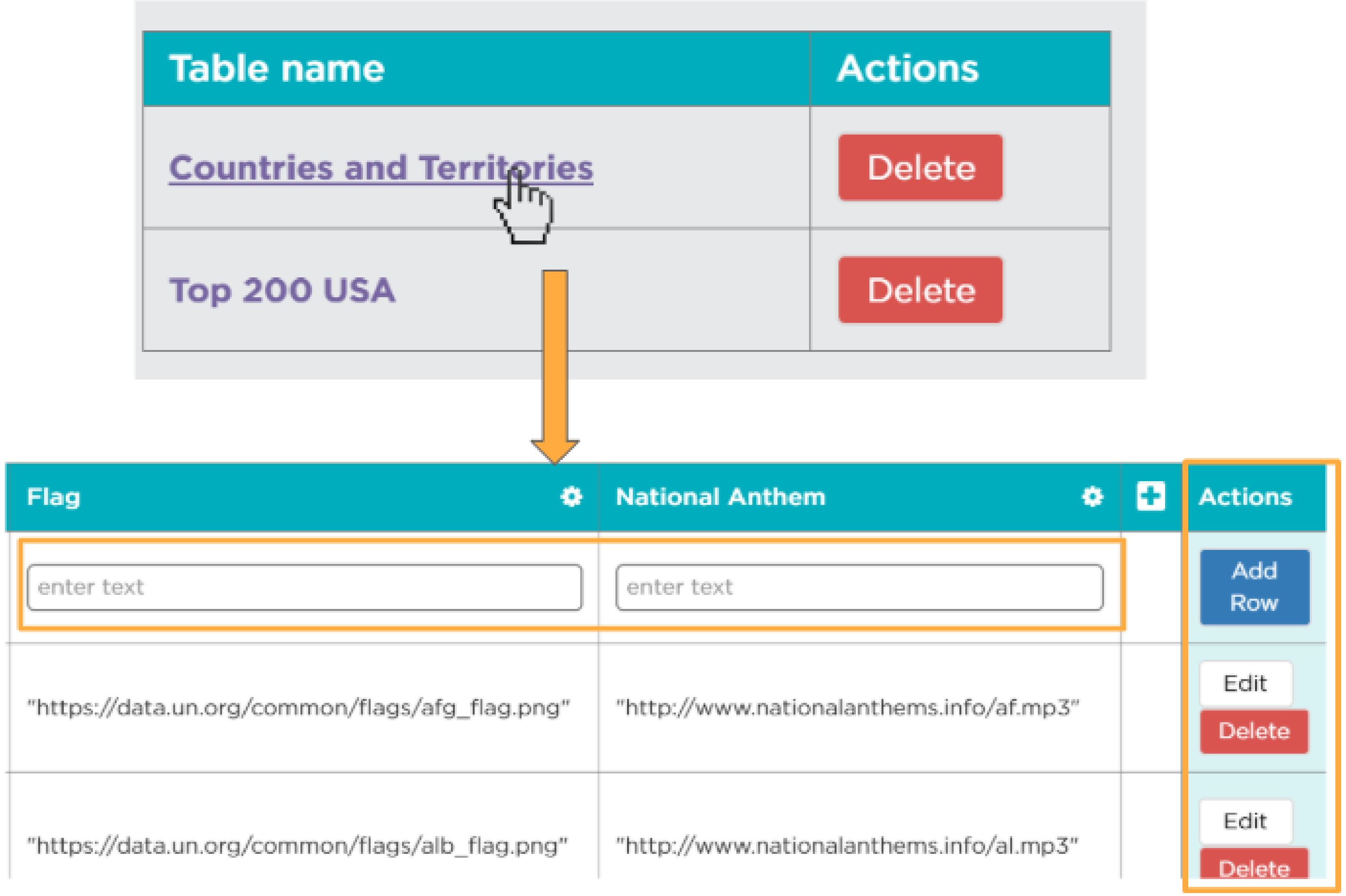


Table name	Actions
Countries and Territories	Delete
Top 200 USA	Delete

Flag	National Anthem	Actions
enter text	enter text	Add Row
"https://data.un.org/common/flags/afg_flag.png"	"http://www.nationalanthems.info/af.mp3"	Edit Delete
"https://data.un.org/common/flags/alb_flag.png"	"http://www.nationalanthems.info/al.mp3"	Edit Delete

Additional Information

- Once you import a dataset, you can edit its content (such as: delete rows, add rows, rename columns, edit individual rows). Note: you cannot edit "live" datasets since these are being updated on a regular basis from an outside source:

Additional Information

- In addition to using the "getColumn" block, you can use "readRecords" to access data in your table. Here is information on using **ReadRecords()**

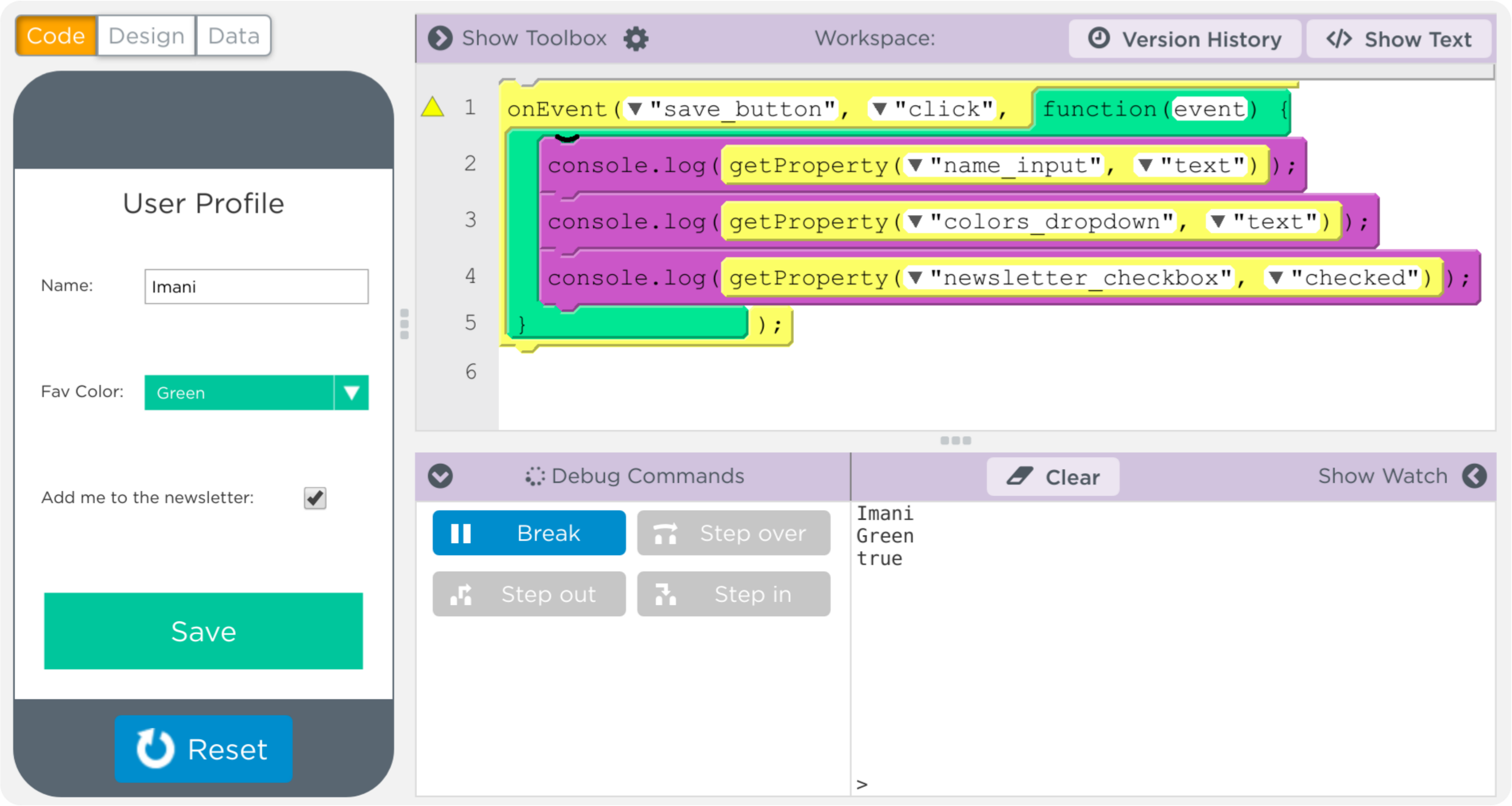
- If you want to learn more about leveraging Lists and Traversals to use data in your app, **this CS Principles lesson** is a good introduction!

- If you have ideas of **additional datasets** you think we should include in App Lab, please let us know!

How to build Larger Database on Your Web Page

Debugging

LECTURE 4



User Profile

Name: Imani

Fav Color: Green

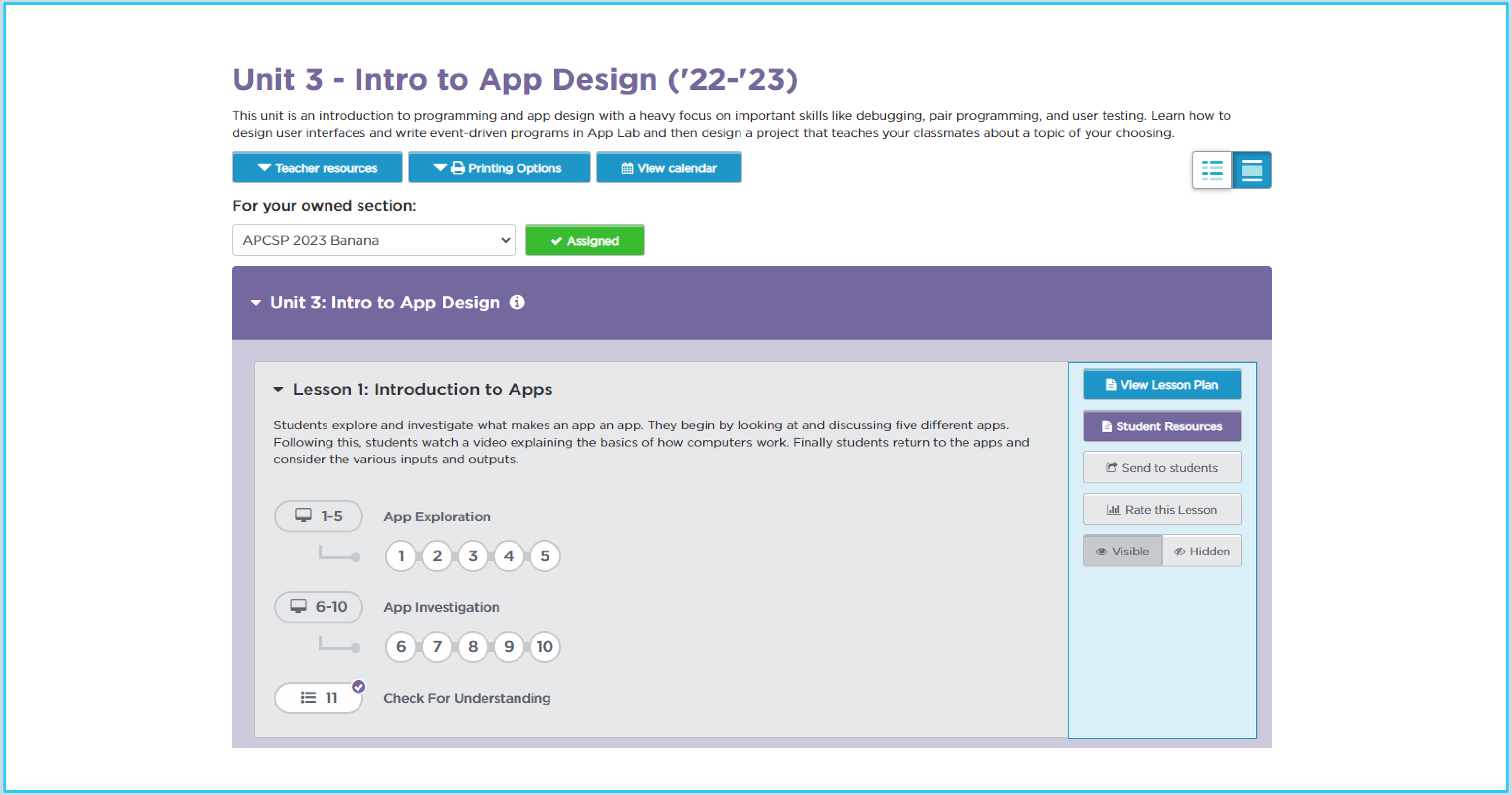
Add me to the newsletter: ☑

```
onEvent("save_button", "click", function(event) {
  console.log(getProperty("name_input", "text"));
  console.log(getProperty("colors_dropdown", "text"));
  console.log(getProperty("newsletter_checkbox", "checked"));
});
```

```
Imani
Green
true
>
```

Introduction to App Design

LECTURE 5



Unit 3 - Intro to App Design ('22-'23)

This unit is an introduction to programming and app design with a heavy focus on important skills like debugging, pair programming, and user testing. Learn how to design user interfaces and write event-driven programs in App Lab and then design a project that teaches your classmates about a topic of your choosing.

For your owned section: APCSP 2023 Banana

Unit 3: Intro to App Design

Lesson 1: Introduction to Apps

Students explore and investigate what makes an app an app. They begin by looking at and discussing five different apps. Following this, students watch a video explaining the basics of how computers work. Finally students return to the apps and consider the various inputs and outputs.

1-5 App Exploration

1 2 3 4 5

6-10 App Investigation

6 7 8 9 10

11 Check For Understanding