API CAN CODE

COLLEGE OF EDUCATION

Unit 2: Computational Foundations of Data Science

Lesson 2.3: What is the role of Programming in Data Science?

In this lesson, students will be introduced to EduBlocks. They will work with the Mario Kart dataset and learn how to manipulate and filter the data.

Duration: 90 minutes

Objective: By the end of this lesson, students will know how to run, clone, and create an EduBlocks program and how to filter the data.

Lesson Walkthrough: Unit 2 Lesson 3 - Teacher Walkthrough

CSTA Standards in this Lesson

Identifier	Concept	Subconcept	Standards
HS-ALG-PS-02	Algorithms	Problem Solving	Optimize the design of an algorithm using procedural abstraction and control structures.
HS-ALG-PS-03	Algorithms	Problem Solving	Evaluate algorithms for efficiency, correctness, and clarity, using metrics or test cases.
HS-DAT-DC-23	Data & Analysis	Data Processing	Use a digital tool to clean and organize text-based data.
HS-PRO-RD-17	Programming	Reading & Documenting Code	Analyze how a segment of code works, including the role of parameters, return values, and data structures.

Lesson activities

Warmup - Conditional Statements (5 min)

(CSTA standards in this activity: 3A-AP-15)

- Students look at a dataset organized in a table and refresh their memory of index values and conditional statements.

Introduction to EduBlocks (15 min)

(CSTA standards in this activity: 3A-AP-23)

This work was made possible through generous support from the National Science Foundation (Award # 2141655). API Can Code is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0) License

- Introduce students to EduBlocks with the available slides.
- It's very important that students create an account and sign into it so that they can save their work.
- Make sure to note the **clone, save** and **rename** features which they should use for every program they work with today!

Introduction to JSON format (15 min)

(CSTA standards in this activity: 3A-DA-09, 3A-AP-13)

- JSON is a structure for storing data.
    - syntax is comprised of pairs of **keys** and **values**
    - **import json** block can be used to ready EduBlocks to use JSON-format data
    - the general format '[{"key":"value", "key2":"value2"}, {"key":"value3", "key2":"value4"}]' is reviewed in the slides with a Mario Kart example!
- Individual observations, or values within those observations, can be pulled with different print blocks as shown.
    - Students modify a given program to print car names instead of driver names.

Mario Kart Drivers Program 1 - Use & Modify (20 min)

(CSTA standards in this activity: 3A-AP-17, 3A-AP-18)

- Students clone a program and use it to print the names of drivers. Some questions for class discussion:
    - **What does this print out?** Why do you think it prints out what it does?
    - **What is the variable "i"?** Do you think it matters what this variable is called? What if we named it something different?
    - **How many times does the loop run?** Why do you think it runs this many times?
- Now, attach the larger chunk of code starting with "sum = 0" below the for loop.
    - **Run your program.** What does this part of the program seem to do?
    - **What happens to the "sum" variable each time the program goes through the loop?** What happens to the "counter" variable?
    - **How do "sum" and "counter" get used to find the average weight?** Is this how you would think about an average?
- **Modify the chunk of code you just added so that the program calculates average acceleration instead of average weight.**
    - Think: what parts of the code need to change?
    - Are there any parts of the code that can stay the same?

Mario Kart Drivers Program 2 - Use, Modify, Create (25 min)

(CSTA standards in this activity: 3A-AP-17, 3A-AP-18)

This work was made possible through generous support from the National Science Foundation (Award # 2141655). API Can Code is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0) License

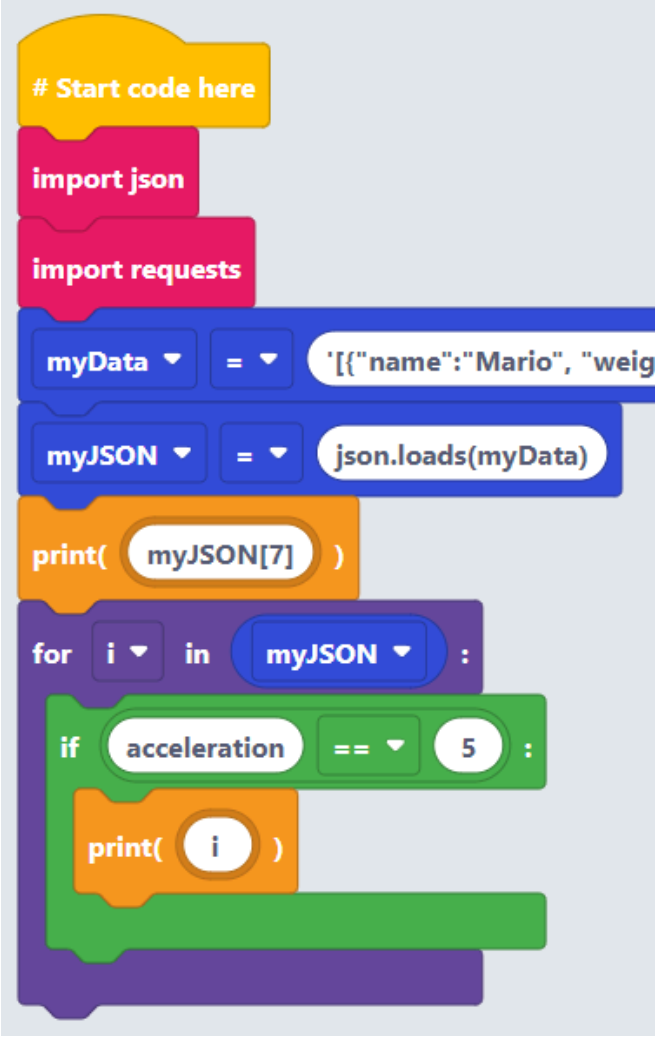
- Students should open this program. As they did before, be sure to clone, save, and rename the project in your EduBlocks account. Some further questions for discussion below:
    - What prints out when you run this program? Is that what you would expect? (Something to think about: Why doesn't Mario's information print out, since he's the first one in the data?)
- Attach the chunk of code that starts with "for i in myJSON".
    - Run the program again. What does this add?
    - What is the purpose of the purple for loop in this section of code?
    - What is the purpose of the green if statement in this section of code?
    - Modify the code to only print Daisy's speed value.
- **Task for students:** Create a new section on this program. The code should find the maximum speed value in the list of drivers, and then print out the name of the driver.
- Some things to consider:
    - How could you loop through the JSON data to find the maximum speed value?
    - How could you use that speed value to find the driver who has it in the JSON?
    - What should happen if there is a tie?
- TEACHER KEY: here is an example of a project that prints the maximum speed value and all drivers that have it.

Exit Ticket - Conditional Statements in EduBlocks (10 min)

(CSTA standards in this activity: 3A-CS-03)

- Students are given a program with several issues in it. They need to find the errors, explain what is wrong, and suggest solutions.

This work was made possible through generous support from the National Science Foundation (Award # 2141655). API Can Code is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0) License



- Issues:
    - The JSON has 7 values, but that means the max index is 6. There is no “myJSON[7]”. To get Toadette’s value, students could reference “myJSON[6]”.
    - In the green if statement, “acceleration” is a bad reference. Instead, students need to reference “i[‘acceleration’]”.

Assessment:

Assess student understanding through participation in class discussions and class activities.

This work was made possible through generous support from the National Science Foundation (Award # 2141655). API Can Code is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0) License