

The Code To Grow Educational Curriculum

By Jack Segil - January 2024

I have developed a 12-step curriculum for the Code To Grow students to follow through the school year between their summers attending coding camp. This is a self-guided program with weekly mentoring calls that lets them work at their own pace while learning the basics of how to code in Python. They come away from this program with great skills, more confidence, and a strong foundation to continue down the computer science path in middle school and high school and then to college.

Goal

The goal is of course to make learning how to code easy and interesting. I've always found myself thriving in project-based learning, especially when it is self-paced, so the curriculum is fashioned as such. When I was starting out making games, I had to scavenge the internet for tutorials and resources in order to successfully gain the knowledge to code what I wanted to code. The curriculum could be described as a more refined version of this.

Execution

Purpose

The purpose of each assignment is to teach a new concept. It does this by instructing the student to create a program that, unbeknownst to the student, can only practically be created using the new concept the assignment is aiming to teach.

Given Information

- The objectives of what the program is supposed to do are clearly listed, and a video demonstrating the full possible output of the program the student should aim to create is provided. A piece of example code is also provided. This code utilizes the new concept in the way the student needs to in order to successfully satisfy the assignment but gives it a different coat of paint. However, the example code is not similar enough to where the student could copy and paste code.

Objectives

Now, let's get started. The objectives for this assignment are as follows:

- You will give the user the option to search for specific names and ages in the passport and see if there are any users who have inputted those names and ages. You will use the name and age lists you worked on previously to accomplish this, and you will use for loops to search through them.

How Your Program Should Work:

- Before anything else happens in the while loop, ask the user "Would you like to add a new user or search for names and ages? Type '0' for adding a new user, and type '1' for searching:"
- If they typed 1:
 - Ask them "Do you want to search for Names or Ages? Type '0' for names, and type '1' for ages"
 - If they typed 1:
 - Ask them to enter the age they wish to search for.
 - Search for that age in your age list.
 - As soon as you find a match, stop the search and print out "There exists a user with this age"
 - If the search manages to complete, this means no matches have been found. Print out "No users have that age".
 - Restart the while loop.
 - If they typed 0:
 - Ask them to enter the name they wish to search for.
 - Search for that age in your name list.
 - As soon as you find a match, stop the search and print out "There exists a user with this name"
 - If the search manages to complete, this means no matches have been found. Print out "No users have that name".
 - Restart the while loop.
- If they typed 0:
 - Go ahead and run the code that we've been working on for the last few projects(the code that asks them their name and stuff and prints it out)

Look at this demonstration video for more context. Your code should behave IDENTICALLY to this:

[Video](#)

Example of objectives for an assignment. Objectives will vary drastically depending on the module.

- Both video and written resources are given to the student that pertain to new concepts, and contain all the information needed to utilize the new concept to the extent necessary.

Resources

Now here are some resources to help you create this!

Remember that we're working in Python 3, not 2, so all methods should be for 3.

There's a file called `examplecode.py` here, separate from the `passport.py` file(which is where you should put your work). This has some code that is intended to help you learn (most) of the new concepts needed to complete this project in a hands-on way. Before you even touch this file, you absolutely have to look through the W3 and/or videos.

- If you use the resources below and feel confident that you know what to do, that's great! You don't even need to touch this file.
- However, if you're still confused, you can look at the example code for additional help. This code uses all of the new concepts needed to complete this project in a different way than how the project uses them. Think of it as a halfway step to just reading the solution.
 - First, you should run this code, and then you should look at the code itself to see how it works. You probably shouldn't copy-paste this code into your project, but no one will stop you if you will. You'll learn it much better if you write your code by hand, trust me.
 - Reach out to your mentor through the pull request if you're still stuck after this.

W3:

[For Loops](#)

Videos:

[For Loops \(stop at 3:14\)](#)

An example of a resources section.

- Each assignment also contains the solution to the previous assignment. The code will be commented on extensively, and all comments persist across solutions, so if the student is rusty with earlier concepts they still have a recent reference. The student's code is not required to be identical

to the solution, instead, it simply provides a fallback if things get too messy, and illuminates to the student another way of solving the problem, which is beneficial in the long run.

```
# You always have to define the variable(s) used in the condition (in this case, the boolean "done")
done = False
# for the while loop before you create the while loop. This is important.
done = False
# The list of names
name_list = []
# The list of ages
age_list = []
while(not done) :
    # The input() function prints a question and returns the user's answer as a string.
    first_name = input("Enter first name: ")
    last_name = input("Enter last name: ")
    age = input("Enter age: ")
    ice_cream = input("Do you like ice cream? Type 'Yes' if yes, type 'No' if no: ")
    video_games = input("Do you like video games? Type 'Yes' if yes, type 'No' if no: ")
    is_done = input("Are you done using the passport? Type 'Yes' if yes, type 'No' if no: ")
    full_name = first_name + " " + last_name
    # "Append" is a fancy word that means to add something to the list. When you use the append function, it adds it to the end of the list.
    name_list.append(full_name)
    age_list.append(age)

# Creates username by slicing the "first_name" string to get the first letter, and doing the same to "last_name".
# It takes those, and adds them together with age (remember, the input function always returns a string!)
# When you "add" strings, it's like you're combining two words to make one word. Example: "One" + "word" = "Oneword".
# This process is also called concatenation.
username = first_name[0:1] + last_name[0:1] + age
```

A snippet of code from a solution

Outside Help

The student meets with a mentor for an hour or two once a week. Here is where the roadblocks the student will inevitably come across will be overcome by the mentor. Additionally, if the student considers the assignment to be done, the mentor will look at the code and give it a seal of approval if it gets the job done, and the student will move on to the next project.

Infrastructure

We use a combination of GitHub Classroom and visual studio code to provide an excellent and command-line-free experience for the students, regardless of their operating system or computer power. I'll spare the technical details, but all the student has to do is download visual studio code with some extensions, download Python, and log in with the GitHub account we'll give them. Then, they can clone assignments to their local machine with a quick button press, and edit their code in real-time with their mentor. Plus, the mentor can open up an online instance of visual studio code with the student's code at any time, allowing for quick and easy access.

Conclusion

Overall, the curriculum is designed to be something I would've loved to have done when I was still learning to code. I believe that kids who have a passion for coding will really enjoy this method of learning, and I can't wait to put it out in the field in the coming months. Hopefully, this curriculum will aid in the overall mission of Code To Grow, and really give a conduit to the passion for coding these kids have.