Weekly Reflection Week 4-5 DDP1-D

What I've learned:

1. We learned about string methods and slicing throughout week 4, and we learned how to access data from text files.
2. A key insight into string manipulation is that strings are a list/array of characters, and so can be accessed in a similar way. For example, the list[n] method fetches the item at the n-th index of the list, while the str[n] method fetches the character at the n-th index of the string.
3. With this knowledge in mind, we can do what is called string slicing, where we can take out substrings from strings. It is implemented in Python using the [] operator.

    For example,

    ```
    str1 = "This is a string"
    print(str[0:5:1])
    ```

    Will output the characters from indices 0 to 5, with a "step" of 1, which in this case is "This ".
4. There are many other string methods as defined within its python documentation, most of which pivot off of list/array methods, just packaged in a way that is more friendly to implementation in strings.
5. Accessing a file means establishing two way communication between the program and the file in question. This can come in the form of reading the contents of the file, writing to it, overwriting a file, appending to it, etc.
6. We can access files in Python using the open() method which opens a fie located in the same directory as the program.
7. To read the contents of the file, we could use methods such as file.read() and file.readlines(), which would return the contents of said files in a readable form (strings and lists).
8. Modifying the access privileges a program has to a file is done using the access parameters of the open() function. For example, open("file.txt", "r") grants the program read only access from the file, while open("file.txt", "w") grants the program write access to the file.
9. We learned to handle exceptions, which are runtime errors in the program.
10. There are many kinds of exceptions in python, such as IndexError, ValueError, TypeError, etc. These errors correspond to different causes and patterns of error in a program, such as TypeError indicating that some methods used in a program are not supported by a certain data type, or ValueError meaning a mismatch in data types, and so on and so forth.

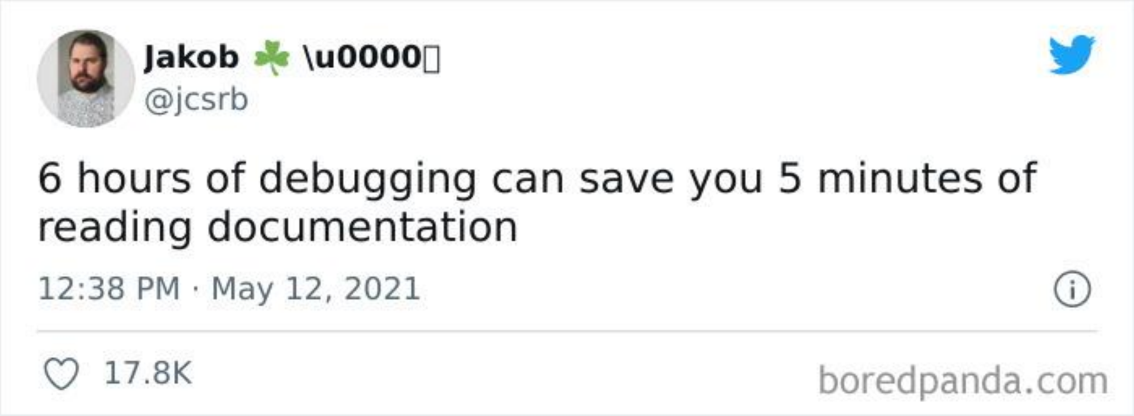
Improvements for Learning Experience

So far I feel like it's mostly been fine, but maybe it could be nice to have the people coming up to the board to do practice questions be in a fixed list(?) in the case where nobody wants to volunteer themselves. I feel like it would be able to involve everyone instead of the "random sampling" method that has been applied these past few meetings wkwk.

Questions:

How would the concepts of string slicing and such be used in the real world? So last labwork involving string slicing and string methods have seemed to be more of a practice in manipulating lists in a rather janky way. In that scenario, it’s clear that using lists would’ve made the process much easier. Because of that, I’d like to ask, when would using string slicing ever come in handy other that in input and output processing?

Have experienced this on more than one occasion no doubt



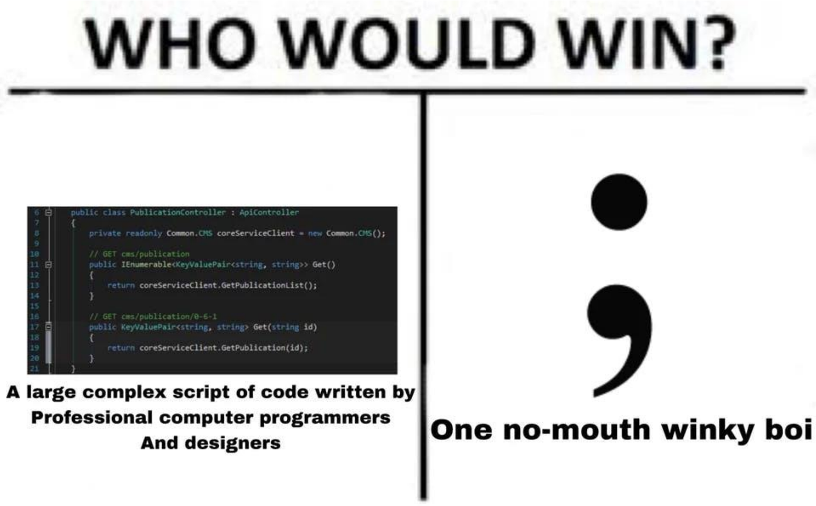
Jakob ☘ \u0000▯
@jcsrb

6 hours of debugging can save you 5 minutes of reading documentation

12:38 PM · May 12, 2021

♡ 17.8K

boredpanda.com



```
public class PublicationController : ApiController
{
    private readonly Common.CMS coreServiceClient = new Common.CMS();

    // GET cms/publication
    public IEnumerable<KeyValuePair<string, string>> Get()
    {
        return coreServiceClient.GetPublicationList();
    }

    // GET cms/publication/0-6-1
    public KeyValuePair<string, string> Get(string id)
    {
        return coreServiceClient.GetPublication(id);
    }
}
```

WHO WOULD WIN?
A large complex script of code written by Professional computer programmers And designers
;
One no-mouth winky boi

When I am writing the code in IDE..