Python Day 3 Assignment

February 2, 2025

1 Python Day 3 Assignment

1.1 Question 1

Write a Python program to read a file line by line and store it into a list.

[1]:
```
f = open("../readfile", "r")
l = [line for line in f]
print(l)
```

```
['python\n', 'web dev\n', 'day 3']
```

1.2 Question 2

Create a list of strings, add to it your name, then write the list to a new file.

[2]:
```
name = "Mohamed"
string_list = ["Hello", "World", "Python", "Programming"]
string_list.append(name)
string_res = "\n".join(string_list)
# if the file is not there it needs to be run twice that is because we didn't␣
 ↪close the file
f = open("./output", "w")
f.write(string_res)
f.close()
```

1.3 Question 3

Create a package directory called `Calculator` and add a file called `my_functions` that has the following functions:

- Sum Function
- Subtract Function
- Divide Function
- Multiply Function

[3]:
```
from Calculator.my_functions import *

print(subtract(3, 4))
print(sum(3, 4))
```

```
print(multiply(3, 4))
print(divide(3, 4))
```

```
-1
7
12
0.75
```

1.4 Question 4

Use the created package in Python code that takes input operand from the user:

- 0 → Sum
- 1 → Subtract
- 2 → Divide
- 3 → Multiply

If one of the two numbers is zero and the operand is subtract, raise a `ValueError` with the message "subtracting zero from Number".

If one of the two numbers is zero and the operand is divide, raise a `ZeroDivisionError` with the message "can't divide with zero".

If one of the two numbers is zero and the operand is multiply, raise a `ValueError` with the message "Multiply with Zero".

[4]:
```
functions = {0: sum, 1: subtract, 2: divide, 3: multiply}

try:
    op = int(input("Enter operand (0:sum, 1:subtract, 2:divide, 3:multiply): "))
    if op not in functions:
        raise ValueError("Invalid operand")

    num1 = float(input("Enter first number: "))
    num2 = float(input("Enter second number: "))

    if op == 1 and (num1 == 0 or num2 == 0):
        raise ValueError("subtracting zero from Number")
    if op == 2 and num2 == 0:
        raise ZeroDivisionError("can't divide with zero")
    if op == 3 and (num1 == 0 or num2 == 0):
        raise ValueError("Multiply with Zero")

    result = functions[op](num1, num2)
    print(f"Result: {result}")
except ValueError as ve:
    print(f"ValueError: {ve}")
except ZeroDivisionError as ze:
    print(f"ZeroDivisionError: {ze}")
```

```
ZeroDivisionError: can't divide with zero
```

1.5 Extra Problem 1

Write a function in Python that accepts two string parameters.

- The first parameter will be a string of characters.
- The second parameter will be the same string of characters, but they'll be in a different order and have one extra character.

The function should return that extra character.

Example:

- If the first parameter is `eueiieo` and the second is `eueiiedo`, then the function should return `d`.

[5]:
```
def getExtraChar(s1: str, s2: str):
    s1 = sorted(s1)
    s2 = sorted(s2)
    while True:
        if s1[0] == s2[0]:
            s1.pop(0)
            s2.pop(0)
        else:
            if len(s1) > len(s2):
                return s1[0]
            else:
                return s2[0]


print(getExtraChar("eueiiedo", "eueiieo"))
```

```
d
```

1.6 Extra Problem 2

Write a function that accepts a number as a parameter.

- The function should return a number that's the difference between the largest and smallest numbers that the digits can form in the number.

Example:

- If the parameter is `"213"`, the function should return `"198"`, which is the result of `123` subtracted from `321`.

[6]:
```
def getDifference(number):
    num_str = str(number)

    smallest_num = int("".join(sorted(num_str)))
    largest_num = int("".join(sorted(num_str, reverse=True)))

    difference = largest_num - smallest_num
```

```
    return difference


number = 213
print(getDifference(number))
```

```
198
```