TASK 1

Review the following codes,find and fix errors and explain the error

https://docs.google.com/document/d/1em2gA_FPlGVb6ul0AvBznSUafZvBzqWpCApyfI51sgo/edit?usp=sharing

Objective: To identify and fix errors in a Python program that manipulates strings.

Errror:

The f-string formatting around the print statement since the reversed_string variable is not being used. Instead, I directly concatenate the "Reversed string:" text with the reversed_string variable inside the print statement.

Corrected Code:

```
def reverse_string(s):
    reversed = ""
    for i in range(len(s) - 1, -1, -1):
        reversed += s[i]
    return reversed
def main():
    input_string = "Hello, world!"
    reversed_string = reverse_string(input_string)
    print("Reversed string: {reversed_string}")
if __name__ == "__main__":
    main()
```

Explanation of Error:

However, it seems that the variable reversed_string is not defined or used anywhere else in the code. This is why you received a NameError stating that the name 'reversed_string' is not defined.

TASK 1

To correct this error, I modified the code by removing the f-string formatting and directly concatenating the "Reversed string:" text with the reversed_string variable inside the print statement:

print("Reversed string:", reversed_string)

Code2: Objective: To identify and fix errors in a Python program that validates user input.

Error:

TypeError: '>=' not supported between instances of 'str' and 'int'

Corrected Code:

```
def get_age():
    age = input("Please enter your age: ")
    if age.isnumeric() and int(age) >= 18:
        return int(age)
    else:
        return None
def main():
    age = get_age()
    if age is not None:
        print(f"You are {age} years old and eligible.")
    else:
        print("Invalid input. You must be at least 18 years old.")
```

```
if __name__ == "__main__":
    main()
```

Explanation of Error:

In this code, I made a small adjustment in the main function. Instead of checking if age, I changed it to if age is not None. This ensures that the program only prints the age if it is a valid input.

Code3:

Objective: To identify and fix errors in a Python program that reads and writes to a file

Issues of code:

Certainly! I reviewed the code, and it seems to be mostly correct. However, there's one potential issue related to the use of the same variable name (file) for both the reading and writing contexts within the same with block. This might lead to unexpected behavior, and it's better to use different variable names.

Corrected code:

```
def read_and_write_file(filename):
    try:
        with open(filename, 'r') as read_file:
            content = read_file.read()
        with open(filename, 'w') as write_file:
            write_file.write(content.upper())
        print(f"File '{filename}' processed successfully.")
```

```
    except Exception as e:
        print(f"An error occurred: {str(e)}")
def main():
    filename = "sample.txt"
    read_and_write_file(filename)
if __name__ == "__main__":
```

This correction involves using different variable names (read_file and write_file) to avoid potential conflicts in the same with block.

Code4:

Corrected Code:

```
def merge_sort(arr):
    if len(arr) <= 1:
        return arr
     mid = len(arr) // 2
    left = arr[:mid]
    right = arr[mid:]
     left = merge_sort(left)
    right = merge_sort(right)
        i = j = k = 0
     while i < len(left) and j < len(right):
        if left[i] < right[j]:
            arr[k] = left[i]
            i += 1
```

```
        else:
            arr[k] = right[j]
            j += 1
        k += 1
    while i < len(left):
        arr[k] = left[i]
        i += 1
        k += 1
     while j < len(right):
        arr[k] = right[j]
        j += 1
        k += 1
    arr = [38, 27, 43, 3, 9, 82, 10]
   merge_sort(arr)
  print(f"The sorted array is: {arr}")
```

Explanation of Bugs:

The merge_sort function is missing a return statement after sorting the subarrays. You need to return the sorted arr at the end of the function.

In the code block where you call merge_sort(left) and merge_sort(right), the returned sorted arrays are not captured. You need to assign the returned arrays to left and right respectively.

Done by - K.Chandrika

TASK 1