**ASSIGNMENT:1**

**Task:1**

**Code1:**

***ERROR:***

**Indentation error is occurred in the given program.**

***CORRECTED CODE:***

***def reverse\_string(s):***

***reversed\_str = ""***

***for i in range(len(s) - 1, -1, -1):***

***reversed\_str += s[i]***

***return reversed\_str***

***def main():***

***input\_string = "Hello, world!"***

***reversed\_string = reverse\_string(input\_string)***

***print(f"Reversed string: {reversed\_string}")***

***if \_\_name\_\_ == "\_\_main\_\_":***

***main()***

**EXPLANATION:**

The provided code seems correct, but there is a minor issue with the indexing in the loop. The range function should include the starting index (0) and exclude the ending index (len(s)), so we need to adjust it.

**CODE:2**

**ERROR:**

Provided age is in string form. To compare it we have to convert it in integer value.

**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:

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:**

The issue in the program is that the input function returns a string, and when comparing age with 18, it's comparing a string to an integer. To fix this, you should convert age to an integer before the comparison in the get\_age function.

***CODE:3***

***ERROR:***

Error in the program is that the file should be open in append mode ('a') instead of write mode ('w').

***CORRECTED CODE:***

def read\_and\_write\_file(filename):

try:

with open(filename, 'r') as file:

content = file.read()

with open(filename, 'a') as file:

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\_\_":

main()

***EXPLANATION:***

The code provided has a potential issue that can lead to data loss. When opening the file in write mode ('w'), it truncates the file before writing the new content. To fix this, you can open the file in append mode ('a') instead of write mode.

***CODE:4***

***ERROR:***

The error in the given code is that recursive calls to merge\_sort for left and right .

***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:***

The issue the program is that the recursive calls to merge\_sort for left and right are not capturing the sorted subarrays. You need to assign the sorted subarrays back to left and right.