**NAME: Muhammad Ali Khatti**

**CMS ID: 387854**

**BS MATHS 21 SNS**

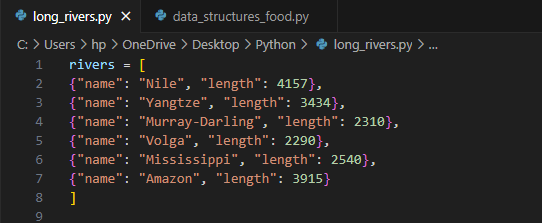
**LAB-01**

**Task 01 (To be submitted during lab timings):**

1. **Create a new file called long\_rivers.py**



1. **Copy and paste the given list of dicts to long\_rivers.py**



**3. Solve the tasks seen below**

**4. Remember to take a screenshot/screenshots of the result!**

**5. If you can't fit all solutions on a single screenshot, you can**

**zoom out in VS Code using CMD + - or CTRL + -**

**Task:**

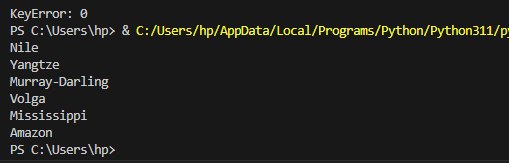
1. **In a for loop, print out each river's name!**

Code:

for i in range(6):

    print(rivers[i]['name'])

Screenshot of result:



**2. In another for loop, add up and print out the total length of**

**all the rivers!**

Code:

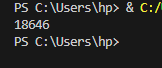
total\_length = 0

for i in range(6):

    total\_length += rivers[i]['length']

print(total\_length)

Screenshot of result:



**3.Print out every river's name that begins with the letter M !**

Code:

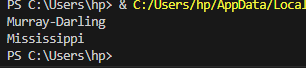
for i in range(6):

    river = rivers[i]['name']

    if river[0] == 'M':

        print(river)

Screenshot of result:



**4. The length of the rivers is in miles. Print out every river's**

**length in kilometres! (1 mile is roughly 1.6 km)**

Code:

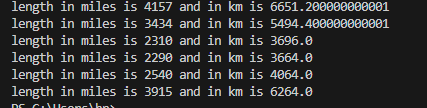
for i in range(6):

    length = rivers[i]['length']

    length\_km = length \* 1.6

    print(f'length in miles is {length} and in km is {length\_km}')

Screenshot of result:



**Task 02 (To be submitted during lab).**

Write the following functions:

**overlap() : Given two lists, find a list of the elements common to both**

**lists and return it.**

Code:

list1 = [1,2,3,4]

list2 = [3,4,5,6]

# Function for overlapp()\_\_\_

def overlapp(list1, list2):

    overlapp = []

    for i in range(len(list1)):

        for j in range(len(list2)):

            if list1[i] == list2[j]:

                overlapp.append(list1[i])

    print(overlapp)

overlapp(list1,list2)

Screenshot of result:



**join(): Given two lists, join them together to be one list without duplicate elements and return that list.**

Code:

list1 = [1,2,3,4]

list2 = [3,4,5,6]

# Function for join()\_\_\_

def join(list1, list2):

    joined = list1

    for i in range(len(list2)):

        if list2[i] in joined:

            continue

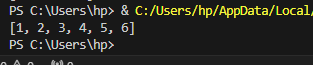
        else:

            joined.append(list2[i])

    print(joined)

join(list1,list2)

Screenshot of result:



**Task 03.**

**Download the data\_structures\_food.py file from LMS and run it in**

**VS code. Your goal is to practice manipulating sequences with the**

**Python tools.**

**In data\_structures\_food.py , there is a list of dictionaries**

**representing different spicy foods.**

1. get\_names:

Code:

def get\_names(spicy\_foods):

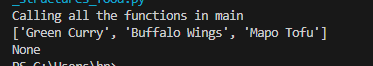
    food\_name = []

    for i in range(3):

        food\_name.append(spicy\_foods[i]['name'])

    print(food\_name)

Screenshot of output:



1. get\_spiciest\_foods:

Code:

def get\_spiciest\_foods(spicy\_foods):

    spiceiest\_food = []

    for i in range(3):

        if spicy\_foods[i]['heat\_level'] > 5:

            spiceiest\_food.append(spicy\_foods[i])

    print(spiceiest\_food)

ScreenShot of Output:



1. Get\_spiciest\_food(spicy\_foods)

Code:

def print\_spicy\_foods(spicy\_foods):

    for i in range(3):

        name = spicy\_foods[i]['name']

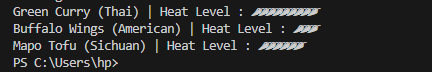
        cuisine = spicy\_foods[i]['cuisine']

        level = spicy\_foods[i]['heat\_level']

        emoji = '

        print(f'{name} ({cuisine}) | Heat Level : {level\*emoji}')

Screenshot of output:



1. **get\_spicy\_food\_by\_cuisine():**

**Code:**

def get\_spicy\_food\_by\_cuisine(spicy\_foods, cuisine):

    for i in range(3):

        if spicy\_foods[i]['cuisine'] == 'American' :

            print(spicy\_foods[i])

**Screenshot of output:**



1. Print\_spiciest\_food:

Code:

def print\_spiciest\_foods(spicy\_foods):

    for i in range(3):

        if spicy\_foods[i]['heat\_level'] > 5 :

            name = spicy\_foods[i]['name']

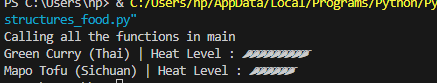
            cuisine = spicy\_foods[i]['cuisine']

            level = spicy\_foods[i]['heat\_level']

            emoji = '

            print(f'{name} ({cuisine}) | Heat Level : {level\*emoji}')

Screenshot:



1. get\_average\_heat\_level :

Code:

def get\_average\_heat\_level(spicy\_foods):

    sum = 0

    for i in range(3):

        sum += spicy\_foods[i]['heat\_level']

    average = sum/3

    print(f'The average is {average}')

Screenshot:



1. Create\_spicy\_food:

Code:

def create\_spicy\_food(spicy\_foods, spicy\_food):

    spicy\_foods.append(spicy\_food)

    return spicy\_foods

Screenshot:

