7. Program to calculate students Grades using Dictionary

7.1 Aim

Creating a module in Python read data of the students into a dictionary and find the average grades.

7.2 Software Used

1. Anaconda Navigator
2. Jupyter Notebook

7.3 Pre Lab Questions

1. Create a dictionary for five states and capital. Print the same.
2. Modify the value/Key value in the dictionary by adding two more states and capitals.
3. How to delete a specific value in a dictionary?
4. Is a python dictionary modifiable? Write one example

7.4 (a) Algorithm

1. Create a dictionary with the key of Student name, assignments, test and lab mark.
2. Create database for five students using step1.
3. Create a function "get_average" to find avg. marks of assignment, test and lab Marks
4. Create a function "calculate_total_mark" to find total marks based on weightage of assignment, test and lab marks
5. Create a function assign_letter_grade to allocate grade based on the specification
6. Create a Function to calculate the total average marks of the whole class
7. Execute the program enter input and check the final result

Formula 1: calculate Grade based on total mark

score >= 90 : **"A"**
score >= 80 : **"B"**
score >= 70 : **"C"**
score >= 60 : **"D"**
for other : "E"

Formula 2: calculate total marks based on weightage of each component

Return the total marks based on weightage supplied

10 % from assignments

70 % from test

20 % from lab-works

1 & 2 .Creating a dictionary which consists of the student name, assignment, Lab, test marks

#s1 to s5 are dictionary of five students

3.Function to calculates average marks

4.Function to calculates total mark for each student based on weightage of assignment,test and lab mark

Return the total marks based on weightage supplied
10 % from assignments
70 % from test
20 % from lab-works

Grade will be calculated according to
1. score >= 90 : **"A"**
2. score >= 80 : **"B"**
3. score >= 70 : **"C"**
4. score >= 60 : **"D"**

5 Calculate letter grade of each student

Function to calculate the total average marks of the whole class

Student list consisting the
dictionary of all students
Iterate through the students list and calculate their respective total marks and letter grade

6.Calculate the average of whole class

Program:

```
student = {
    'name': 'Kunal',
    'assignments': 45,
    'test': 50,
    'lab mark': 89,
}
print(student)

students = [
    {
    'name': 'Kunal',
    'assignments': 45,
    'test': 50,
    'lab mark': 89,
},
    {
    'name': 'Louhith',
    'assignments': 100,
    'test': 100,
    'lab mark': 100,
},
    {
```

```
    'name': 'Sangam',
    'assignments': 69,
    'test': 69,
    'lab mark': 69,
},
    {
    'name': 'Sandeepan',
    'assignments': 95,
    'test': 100,
    'lab mark': 99,
},
    {
    'name': 'Harish',
    'assignments': 96,
    'test': 100,
    'lab mark': 80,
},
    {
    'name': 'Likhit',
    'assignments': 80,
    'test': 100,
    'lab mark': 85,
}
]
print(students)

def calculate_total_marks(list):
    for student in list:
        total = student['lab mark'] + student['test'] + student['assignments']
        student['total'] = total
    return list;

print(calculate_total_marks(students))

def get_average(list):
    for student in list:
        student['average'] = (student['total']/300)*100
    return list;

print(get_average(students))

def total_average_marks_of_class(list):
    class_len =len(list)
    class_average = []
    for student in list:
        class_average.append(student['average'])

    return (sum(class_average)/class_len)

print(total_average_marks_of_class(students))
```

```
def grade_based_on_mark(list):
    for student in list:
        if(student['average'] >= 90):
            student['grade'] = 'A'
        elif(student['average'] >= 80):
            student['grade'] = 'B'
        elif(student['average'] >= 70):
            student['grade'] = 'C'
        elif(student['average'] >= 60):
            student['grade'] = 'D'
        else:
            student['grade'] = 'E'
    return list;

grade_based_on_mark(students)
for student in students:
    print(student['name'], 'scored', student['grade'], 'grade')
```

Output response:

```
student = {
    'name': 'Kunal',
    'assignments': 45,
    'test': 50,
    'lab mark': 89,
}
print(student)
```

```
{'name': 'Kunal', 'assignments': 45, 'test': 50, 'lab mark': 89}
```

```
students = [
    {
    'name': 'Kunal',
    'assignments': 45,
    'test': 50,
    'lab mark': 89,
},
    {
    'name': 'Louhith',
    'assignments': 100,
    'test': 100,
    'lab mark': 100,
},
    {
    'name': 'Sangam',
    'assignments': 69,
    'test': 69,
    'lab mark': 69,
},
    {
    'name': 'Sandeepan',
    'assignments': 95,
    'test': 100,
    'lab mark': 99,
},
    {
    'name': 'Harish',
    'assignments': 96,
    'test': 100,
    'lab mark': 80,
},
    {
    'name': 'Likhit',
    'assignments': 80,
    'test': 100,
    'lab mark': 85,
}
]
print(students)
```

```
[{'name': 'Kunal', 'assignments': 45, 'test': 50, 'lab mark': 89}, {'name': 'Louhith', 'assignments': 100, 'test': 100, 'lab mark': 100}, {'name': 'Sangam', 'assignments': 69, 'test': 69, 'lab mark': 69},
{'name': 'Sandeepan', 'assignments': 95, 'test': 100, 'lab mark': 99}, {'name': 'Harish', 'assignments': 96, 'test': 100, 'lab mark': 80}, {'name': 'Likhit', 'assignments': 80, 'test': 100, 'lab mark': 8
5}]
```

```
def calculate_total_marks(list):
    for student in list:
        total = student['lab mark'] + student['test'] + student['assignments']
        student['total'] = total
    return list;

print(calculate_total_marks(students))
```

```
[{'name': 'Kunal', 'assignments': 45, 'test': 50, 'lab mark': 89, 'total': 184, 'average': 100}, {'name': 'Louhith', 'assignments': 100, 'test': 100, 'lab mark': 100, 'total': 300, 'average': 100}, {'nam
e': 'Sangam', 'assignments': 69, 'test': 69, 'lab mark': 69, 'total': 207, 'average': 100}, {'name': 'Sandeepan', 'assignments': 95, 'test': 100, 'lab mark': 99, 'total': 294, 'average': 100}, {'name': 'Ha
rish', 'assignments': 96, 'test': 100, 'lab mark': 80, 'total': 276, 'average': 100}, {'name': 'Likhit', 'assignments': 80, 'test': 100, 'lab mark': 85, 'total': 265, 'average': 100}]
```

```
def get_average(list):
    for student in list:
        student['average'] = (student['total']/300)*100
    return list;

print(get_average(students))
```

```
[{'name': 'Kunal', 'assignments': 45, 'test': 50, 'lab mark': 89, 'total': 184, 'average': 61.33333333333333}, {'name': 'Louhith', 'assignments': 100, 'test': 100, 'lab mark': 100, 'total': 300, 'average':
100.0}, {'name': 'Sangam', 'assignments': 69, 'test': 69, 'lab mark': 69, 'total': 207, 'average': 69.0}, {'name': 'Sandeepan', 'assignments': 95, 'test': 100, 'lab mark': 99, 'total': 294, 'average': 98.
0}, {'name': 'Harish', 'assignments': 96, 'test': 100, 'lab mark': 80, 'total': 276, 'average': 92.0}, {'name': 'Likhit', 'assignments': 80, 'test': 100, 'lab mark': 85, 'total': 265, 'average': 88.3333333
3333333}]
```

```
def total_average_marks_of_class(list):
    class_len =len(list)
    class_average = []
    for student in list:
        class_average.append(student['average'])

    return (sum(class_average)/class_len)

print(total_average_marks_of_class(students))
```

```
84.77777777777777
```

```
def grade_based_on_mark(list):
    for student in list:
        if(student['average'] >= 90):
            student['grade'] = 'A'
        elif(student['average'] >= 80):
            student['grade'] = 'B'
        elif(student['average'] >= 70):
            student['grade'] = 'C'
        elif(student['average'] >= 60):
            student['grade'] = 'D'
        else:
            student['grade'] = 'E'
    return list;

grade_based_on_mark(students)
for student in students:
    print(student['name'], 'scored', student['grade'], 'grade')
```

```
Kunal scored D grade
Louhith scored A grade
Sangam scored D grade
Sandeepan scored A grade
Harish scored A grade
Likhit scored B grade
```

7.4 (b) Follow the steps bellow: -Create a new dictionary called prices using {} .

Put these values in your prices dictionary:

```
"banana": 4,
"apple": 2,
"orange": 1.5,
"pear": 3
```

Loop through each key in prices. For each key, print out the key along with its price and stock information. Print the answer in the following format:

```
apple
price: 2
```

stock: 0

- Let's determine how much money you would make if you sold all of your food.
- Create a variable called total and set it to zero.
- Loop through the prices dictionaries. For each key in prices, multiply the number in prices by the number in stock. Print that value into the console and then add it to total.
- Finally, outside your loop, print total.

Program:

```
#Create the prices dictionary:
#Add values
#Create the stock dictionary
#Add values
#Show all prices and stock
for food in prices:

prices = {
   'banana': 4,
   'apple': 2,
   'orange': 1.5,
   'pear': 3
}

stock = {
   'banana': 8,
   'apple': 0,
   'orange': 2,
   'pear': 6
}

for item in prices:
   print(item)
   print('price:', prices[item])
   print('stock:', stock[item], '\n')

total = 0

for item in prices:
   total = total + (prices[item]*stock[item])

print(total)
```

Output response:

[2]:
```
prices = {
    'banana': 4,
    'apple': 2,
    'orange': 1.5,
    'pear': 3
}

stock = {
    'banana': 8,
    'apple': 0,
    'orange': 2,
    'pear': 6
}

for item in prices:
    print(item)
    print('price:', prices[item])
    print('stock:', stock[item], '\n')
```

```
banana
price: 4
stock: 8

apple
price: 2
stock: 0

orange
price: 1.5
stock: 2

pear
price: 3
stock: 6
```

[3]:
```
total = 0

for item in prices:
    total = total + (prices[item]*stock[item])

print(total)
```

```
53.0
```

7.5 Post Lab Question

1. Write a program that accepts a sentence and calculate the number of letters and digits in the given sentence

Suppose the following input is supplied to the program:
Scientific python! 2022
Then, the output should be:
LETTERS 16
DIGITS 4

Hint : use isdigit(), isalpha() function

2. Write a python program using dictionary to create a small dictionary (min of five words) of synonyms. the program should accept a word and generate synonyms for the same

Hint: same key with different value

7.6 Result

Creating a module in Python read data of the students into dictionary, find the average grades and price verified the result.