=======================================================================================

SECTION-III

Answer any 5 from the following List of Programs.

Each Program carries 5 Marks 5x5=25 Marks

=======================================================================================

1) Write a Python Program to Count Uppercase and Lowercase and special symbols (except spaces) in a given Line text.

Sample Input: NaResh I TeCh

Number of Upper Case Letters: 5

Number of lower Case Letters: 6

2) Write a Python class Employee with attributes like emp\_id, emp\_name, emp\_salary, and emp\_department and

methods like calculate\_emp\_salary, emp\_assign\_department, and print\_employee\_details.

Sample Employee Data:

"RAJESH", "E7876", 50000, "ACCOUNTING"

"RAKESH", "E7499", 45000, "RESEARCH"

"RAM", "E7900", 50000, "SALES"

"RAJIT", "E7698", 55000, "OPERATIONS"

=>Use 'assign\_department' method to change the department of an employee.

=>Use 'print\_employee\_details' method to print the details of an employee.

=>Use 'calculate\_emp\_salary' method takes two arguments: salary and hours\_worked, which is the number of

hours worked by the employee. If the number of hours worked is more than 50, the method computes overtime and adds it to the salary. Overtime is calculated as following formula:

=>overtime = hours\_worked - 50

=>Overtime amount = (overtime \* (salary / 50))

3. Write a Python Program for Validating Name of a Person OR Product Name or Place Name:

Sample Input: Guido Van Rossum----Valid Name bcoz all alphabets are present

Sample Input: Gui$do Van Rossum----InValid Name bcoz $ symbols not allowed

Sample Input: Gui2do Van R-ossum----InValid Name bcoz 2 (digit) and special symbol(-) are not allowed

4. Write a Python Program for Obtaining Names and Marks of Students by using Regular Expressions.

Ensure that student data present in student.data file

--------

HINT: student.data<------It is File Name

-------

Rossum got 55 marks , Ritche got 67 marks and Gosling got 69 marks

and Travis got 99 marks and Jhon got 33 marks

--------------------------

Expected Output

--------------------------

Names Marks

---------------------------

Rossum 55

Ritche 67

Gosling 69

Travis 99

Jhon 33

-----------------------------

5. Write a Python Program by Accepting Employee number,Employee Name,Salary and Designation and save those

details as record in "EMP.DATA" file by using Pickling Process (Object Serialization Process)

6. Write a Python Program for Creating a CSV File by using csv module

( HINT: use csv.writer() OR use csv.DictWriter() )

(HINT: Use Ur own sample data for csv file)

7. Write a Pandas program for creating a Data Frame and to get the first 3 rows of a given DataFrame.

-----------------------------

Take Sample Input

-----------------------------

Sample Python dictionary data and list labels:

exam\_data = {'name': ['Raj', 'Kv', 'Rossum', 'James', 'Emily', 'Hunter', 'Matthew', 'Laura', 'Kinney', 'Jonas'],

'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],

'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],

'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}

labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

---------------------------

Expected Output:

---------------------------

First three rows of the data frame:

attempts name qualify score

a 1 Raj yes 12.5

b 3 Kv no 9.0

c 2 Rossum yes 16.5

==========================================x==================================================

Q1) ANS ----CODE

line=input("Enter a Line of Text:")

noupper=0

nolower=0

for ch in line:

if(ch.islower()):

nolower=nolower+1

if (ch.isupper()):

noupper = noupper + 1

print("Number of Upper Case Letters=",noupper)

print("Number of Lower Case Letters=",nolower)

==========================================================================================

Q2) ANS----CODE

class Employee:

def \_\_init\_\_(self, name, emp\_id, salary, department):

self.name = name

self.id = emp\_id

self.salary = salary

self.department = department

def calculate\_salary(self, salary, hours\_worked):

overtime = 0

if hours\_worked > 50:

overtime = hours\_worked - 50

self.salary = self.salary + (overtime \* (self.salary / 50))

def assign\_department(self, emp\_department):

self.department = emp\_department

def print\_employee\_details(self):

print("\nName: ", self.name)

print("ID: ", self.id)

print("Salary: ", self.salary)

print("Department: ", self.department)

print("----------------------")

employee1 = Employee("RAJESH", "E7876", 50000, "ACCOUNTING")

employee2 = Employee("RAKESH", "E7499", 45000, "RESEARCH")

employee3 = Employee("RAM", "E7900", 50000, "SALES")

employee4 = Employee("RAJIT", "E7698", 55000, "OPERATIONS")

print("Original Employee Details:")

employee1.print\_employee\_details()

employee2.print\_employee\_details()

employee3.print\_employee\_details()

employee4.print\_employee\_details()

# Change the departments of employee1 and employee4

employee1.assign\_department("OPERATIONS")

employee4.assign\_department("SALES")

# Now calculate the overtime of the employees who are eligible:

employee2.calculate\_salary(45000, 52)

employee4.calculate\_salary(45000, 60)

print("Updated Employee Details:")

employee1.print\_employee\_details()

employee2.print\_employee\_details()

employee3.print\_employee\_details()

employee4.print\_employee\_details()

==========================================================================================

Q3) ANS----CODE

while True:

name=input("Enter Ur Name:")# name="Guido Van Ro3ssum"

words=name.split() # words=[Gu2ido, Van , Rossum]

res=True

for word in words:

if(not word.isalpha()):

res=False

break

if(res):

print("{} is Valid Name:".format(name))

break

else:

print("\t\t{} is In Valid Name:".format(name))

==========================================================================================

Q4: ---ANS CODE

import re

try:

with open("student.data", "r") as fp:

filedata=fp.read()

nameslist=re.findall("[A-Z][a-z]+",filedata)

markslist=re.findall(r"\d{2}",filedata)

print("="\*50)

print("Names\t\tMarks")

print("=" \* 50)

for names,marks in zip(nameslist,markslist):

print("{}\t\t{}".format(names,marks))

print("=" \* 50)

except FileNotFoundError:

print("File does not Exist")

==========================================================================================

Q5---ANS---CODE

import pickle

with open("emp.data","ab") as fp:

while(True):

#accept the employee values from KBD

print("-------------------------------------------------------")

empno=int(input("Enter Employee Number:"))

ename=input("Enter Employee Name:")

sal=float(input("Enter Employee Salary:"))

dsg=input("Enter Employee Designation:")

print("-------------------------------------------------------")

#create an empty list and place emp values

lst=list()

lst.append(empno)

lst.append(ename)

lst.append(sal)

lst.append(dsg)

#Save OR Transfer lst data into the file

pickle.dump(lst,fp)

print("Employee Record Saved in File sucessfully:")

print("-------------------------------------------------------")

ch=input("Do u want to Insert Another Record(yes/no):")

if(ch.lower()=="no"):

print("Thx for using this program")

break

==========================================================================================

Q6) ANS ---CODE

#Program for creating CSV File though Python Lang with csv.reader()

#CSVWriteEx1.py

import csv # Step-1

hname=["EMPNO","NAME","SAL"] # Step-2

records=[ [100,"Rossum",4.5],

[200,"Travis",5.6],

[300,"Dennis",3.4],

[400,"Strup",4.5],

[500,"Ritche",6.7] ] # Step-3---List in List--called Nested List

#Choose the CSV File and Open it into write Mode---Step-4

with open("emp.csv","w") as fp:

csvwr=csv.writer(fp) # Step-5---here csvwr is an object of <class,csv.writer>

#write the Header Names---

csvwr.writerow(hname) # Step-6

#Write the Records

csvwr.writerows(records) # Step-7

print("CSV File Created Dynamically through Code--verify")

(OR)

#Program for Creating CSV File bY using Dict Data

#CSVDictWriteEx.py

import csv # Step-1

hnames=["ENO","NAME","SAL"] # Step-2

records=[{"ENO":100,"NAME":"RS","SAL":4.5},

{"ENO":110,"NAME":"TR","SAL":6.5},

{"ENO":120,"NAME":"DR","SAL":2.5},

{"ENO":140,"NAME":"ST","SAL":2.6},

{"ENO":160,"NAME":"KV","SAL":0.0}] # Step-3

#Choose the CSV File and Open in Write Mode

with open("emp.csv","w") as fp: # Step-4

csvdwr=csv.DictWriter(fp,fieldnames=hnames) # Step-5

csvdwr.writeheader() # Step-6

csvdwr.writerows(records) # Step-7

print("CSV File Created--Verify")

----------------------------------------------------------------------------------------------------------------------------------------------------------

Q7 ANS: CODE

---------------------------------------------------------------------------------------------------------------------------------------------------------

import pandas as pd

import numpy as np

exam\_data = {'name': ['Raj', 'Kv', 'Rossum', 'James', 'Emily', 'Hunter', 'Matthew', 'Laura', 'Kinney', 'Jonas'],

'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],

'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],

'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}

labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

df = pd.DataFrame(exam\_data , index=labels)

print("First three rows of the data frame:")

print(df.iloc[:3])

==========================================================================================