**TABLE OF CONTENTS:-**

Introduction and Assumptions……………………………………………………..……...1

Design of the program………………………………………………………………..……3

Source code and explanation…………………………………………………………..….13

Screenshots of sample Input/Output and explanation……………………………….……20

Conclusion………………………………………………………………………………...24

**LIST OF FIGURES:-**

Figure 1:- Main function flow chart…………………………………………….3

Figure 2:- Registration function flow chart…………………………………….4

Figure 3:- Vaccine administration function flow chart…………………………5

Figure 4:- Menu output………………………………………………………….20

Figure 5:- Wrong input………………………………………………………….20

Figure 6:- 1st option selection. ………………………………………………….20

Figure 7:- Patient registration. …………………………………………………21

Figure 8:- Patient.txt ……………………………………………………………21

Figure 9:- Vaccine administration ………………………………………………21

Figure 10:- D1 option……………………………………………………………22

Figure 11:- D2 option……………………………………………………………22

Figure 12:- 3rd menu selection. ………………………………………………….22

Figure 13:- Output of 4th option…………………………………………………23

Figure 14:- Statistics text file……………………………………………………..24

**INTRODUCTION AND ASSUMPTIONS**

Coronavirus, also known as SARS-CoV-2 (severe acute respiratory syndrome coronavirus 2), is a lethal virus that has caused a global epidemic. The virus causes infections in the respiratory tract that can range from mild to fatal. Coronavirus disease is the designation given by the World Health Organization (WHO) to the sickness caused by coronavirus (COVID-19). There is currently no specific medication or therapy that has been discovered to treat COVID-19 patients. As a result, hospitals frequently offer COVID-19 patients with supportive care as part of their treatment. This involves symptom relief, hydration therapy, oxygen support, and prone positioning as needed, as well as drugs or devices to support other essential organs that are damaged. The COVID-19 vaccine has been discovered to be a safer strategy to help build virus resistance and terminate the pandemic. This has prompted many countries to promptly vaccinate their citizens, and to do so, the COVID-19 vaccine process is being carried out concurrently in several vaccination centers across the country.

The vaccine center needs to have a system which contains all the information and allow the vaccine process to work as needed. Python language is used as the programming language as mentioned in the assignment requirement. Python is an interpreted, object-oriented, high-level programming language with dynamic semantics that may be executed on a computer's desktop. High-level data structures, along with dynamic type and dynamic binding, making it a very attractive choice for Rapid Application Development as well as a scripting or glue language for connecting existing components. In addition, Python's straightforward, simple-to-learn syntax prioritizes readability, which lowers the overall cost of program maintenance. As a result of Python's support for modules and packages, program modularity and code reuse are encouraged. Python's interpreter and huge standard library are available for free in source or binary form for all major platforms, and they can be freely distributed if they are not altered in any way.

**ASSUMPTIONS MADE:-**

* Unique id provided to the patient will starts from 100 and keep incrementing by 1.
* The vaccination center information will be stored as per the abbreviation allocated to the vaccination center.
* Patients will be required to go through the first dosage before the second unless if they have opted for EC vaccine.
* There is no requirement of sequence vaccination.

**DESIGN OF THE PROGRAM**

**FLOW CHART:-**

Diagram

Description automatically generated

Figure 1:- Main function flow chart

Diagram

Description automatically generated

Figure 2:- Registration function flow chart.

Diagram

Description automatically generated

Figure 3:- Vaccine administration function flow chart.

**PSEUDO CODE:-**

Considering the total lines of code is more than 350 lines of code, only few part of the codes were mentioned in the pseudo code.

READINGFILE function:-

FUNCTION reading\_file(input\_patient\_id, filename):

WITH open(file\_path) as get\_code:

lines <- [line.rstrip().split(",") for line in get\_code

get\_code.close()

id\_found <- False

break\_out\_flag <- False

FOR rows, columns in enumerate(lines):

FOR j in columns:

IF lines[rows][0] = input\_patient\_id:

searched\_line <- columns

break\_out\_flag <- True

id\_found <- True

break

ENDIF

ENDFOR

IF break\_out\_flag:

break

ENDIF

ENDFOR

IF id\_found:

RETURN True, searched\_line

ELSE:

RETURN False, 0

Registeration function

FUNCTION registration():

validvalue <- False

while not validvalue:

vc <- input("""Select your vaccination center:-

VC1

VC2

:-""")

IF vc = "VC1" OR vc = "VC2":

validvalue <- True

ELSE:

OUTPUT "Wrong input"

ENDIF

ENDWHILE

valid\_age <- False

while not valid\_age:

try:

age <- int(input("Enter your age:-"))

except ValueError:

OUTPUT "Your age should be a number! Thats evident"

IF 0 < age < 110:

valid\_age <- True

IF age >11:

validineer <- False

while not validineer:

IF age < 18:

code <- input(""" Select the Vaccination Code :-

ENDIF

Vaccine Code

IF (code = "AF" OR code = "BV" OR code = "CZ" OR code = "DM" OR code = "EC"):

validineer <- True

ELSE:

OUTPUT "Wong code selected"

ENDIF

ELSEIF 11 < age < 45:

code <- input(""" Select the Vaccination Code :-

Vaccine Code

IF (code = "AF" OR code = "BV" OR code = "CZ" OR code = "DM" OR code = "EC"):

validineer <- True

ELSE:

OUTPUT "Wong code selected"

ENDIF

ELSE:

code <- input(""" Select the Vaccination Code :-

Vaccine Code

IF (code = "AF" OR code = "BV" OR code = "CZ" OR code = "DM" OR code = "EC"):

validineer <- True

ELSE:

OUTPUT "Wong code selected"

ENDIF

ENDIF

ENDWHILE

ELSE:

valid\_age <- False

OUTPUT "Sorry! Thats not a valid age. Maybe Thats a Typo"

ENDIF

ENDWHILE

validphonenumber <- False

while not validphonenumber:

try:

phone\_number <- int(input("Enter your phone number"))

str\_phone\_number <- str(phone\_number)

IF(len(str\_phone\_number) = 10 ):

validphonenumber <- True

ENDIF

except ValueError:

OUTPUT "Phone number itself means number! Please input a number"

ENDWHILE

RETURN vc, age, code, phone\_number

Statistics function:-

FUNCTION statistic\_file():

working\_directory <-

with open(working\_directory) as get\_code:

lines <- [line.rstrip().split(",")

ENDFOR

get\_code.close()

totalvc1d1, totalvc1d2, totalvc2d1, totalvc2d2 <- 0, 0, 0, 0

WD1, WD2 <- 0, 0

ENDIF

for rows, columns in enumerate(lines):

IF(columns[1] = "VC1"):

WD1,WD2 <- reading\_status(rows)

totalvc1d1 <- totalvc1d1 + WD1

totalvc1d2 <- totalvc1d2 + WD2

ENDIF

IF(columns[1] = "VC2"):

WD1,WD2 <- reading\_status(rows)

totalvc2d1 <- totalvc2d1 + WD1

totalvc2d2 <- totalvc2d2 + WD2

for j in columns:

ENDFOR

ENDIF

ENDFOR

RETURN totalvc1d1, totalvc1d2, totalvc2d1, totalvc2d2

ENDFUNCTION

FUNCTION reading\_status(linenumbers):

d1, d2 <- 0, 0

try:

working\_directory <-

with open(working\_directory) as get\_code:

innerlines <- [line.rstrip().split(",") for line in get\_code]

ENDFOR

get\_code.close()

IF(innerlines[linenumbers][2] = "D1"):

d1 <- 1

ENDIF

IF(innerlines[linenumbers][2] = "D2" OR innerlines[linenumbers][2] = "D"):

d2 <- 1

ENDIF

except(IndexError):

d <- 0

RETURN d1, d2

ENDFUNCTION

v1,v2,v3,v4 <- statistic\_file()

**PROGRAM SOURCE CODE AND EXPLANATION:-**

The complete program consists of 348 lines of code, only those code which uses different programming techniques will be explained.

import os

from Newpatientregisteration import registeration

from vaccineadministration import vaccineadministration

from readfile import reading\_file

from statistics import statistic\_file

The above lines of code are the header files , the first import is the import os (operating system) module in Python includes functions for creating and removing directories (folders), retrieving their contents, modifying and identifying the current directory, among other things. The use of this module was done to extract the current directory location and also for read and write operation using getcode() function. The next four import are the files created for the program and their function name. Newpatient registration function file is for the registration of the patient which is the first requirement in the assignment. Vaccine administration python file was imported for the second requirement in the assignment related to doses and similarly third and fourth import are for the third and fourth requirements.

def menu():

try:

menu\_selection = int(input("""

------------------MENU SELECTION------------------

1]= NEW PATIENT REGISTERATION ]

2]= VACCINE ADMINISTRATION ]

3]= SEARCH PATEINT RECORD ]

4]= SATISTICS AS PER VACCINE CENTER ]

5]= EXIT ]

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

:-"""))

return menu\_selection

except ValueError:

print("Input can only be number form 1 to 5")

The above line of codes is of the menu function of the main file of the program, the def keyword is used in python followed by the file name to for the creation of the function. The menu function includes try and except block, the try block is where the code is included while the except block is when the error occurs, and it handles the error which allows the program to run without throwing the error. For this program the try and catch block was used for error handling for input from the user, where the input is forced to be an integer if the user enters a string or any other special characters the program will ask the user to enter an integer value between 1 to 5.

while True:

selected\_menu = menu()

if selected\_menu == 1:

id\_file = open(r"D:\WORK\Python Programming\Covid mangement system\unique\_id.txt", "r+")

id = int(id\_file.read())

id\_file.seek(0)

id= int(id) +1

id\_file.write(str(id))

id\_file.truncate()

#id\_file.write("\n")

id\_file.close()

vc, age, code, phone\_number = registeration()

pateints\_file = open("D:\WORK\Python Programming\Covid mangement system\patients.txt", "a+")

pateints\_file.write(str(id)+","+str(vc) +"," + str(age) + "," + str(code) + "," + str(phone\_number)+"\n")

pateints\_file.close()

input("Press Enter to continue...")

The above line of code works after the user enters the input, a while loop with true condition was used to make the program works in loop until the user select for the exit option. If else conditional statements were used for the program menu, if the user enter a number outside the range of 1 to 5 the program will ask the user to enter in the provided range. If the entered input is 1 the first line in the condition is for the open() function which opens a file and return a file object, id.txt file was provided as the argument to the function, the file consist of a unique id. The id is then read to a variable, python reads the data form the file in string format hence type conversion function of python was used to convert the string to Int. The id is increased by one, to store the id write function is used which replace the old data with the new data. Truncate method is used to delete the contents of the file, which basically truncate the file to 0 bytes. Registration function is called where the function returns four variables required for the patient information as required in the assignment. Then similar write file algorithm is used to write the data but since there will be a greater number of patients the mode for the file is append mode(a+). Similar if else conditions were used for the selection of menu for four different tasks.

**Vaccine administration function.**

from readfile import reading\_file

import shutil

def vaccineadministration(inputid):

state,vaccination\_file = reading\_file(inputid,"patients.txt")

break\_out\_flag = False

try:

if (state == 0):

print("ID NOT FOUND! PLEASE REGISTER FIRST")

else:

id, vaccine\_type = [vaccination\_file[i] for i in (0, 3)]

valid\_input = False

if vaccine\_type == "EC":

dosage\_file = open("D:\WORK\Python Programming\Covid mangement system\\vaccination.txt", "a+")

dosage\_file.write(id+","+vaccine\_type+","+ "D"+"\n")

print("Congratulations you have been fully vaccinated")

else:

while not valid\_input:

The above lines of codes are of the vaccineadministration function where the first line is of the call of reading file function, try and except block was used for handling the wrong input by the user. If statement was used for displaying the id not found error, since the EC type vaccine doesn’t require second dose, the patient data is written to the vaccination.txt file and the mode is append since there can be multiple user which will be added.

dose\_input = input("Select the Dose (D1 or D2):-")

if(dose\_input == "D1"):

with open("D:\WORK\Python Programming\Covid mangement system\\vaccination.txt") as file:

file\_content = [line.rstrip().split(",") for line in file]

for rows, columns in enumerate(file\_content):

for j in columns:

if j == id:

if file\_content[rows][2] == "D1":

print("You have already opt for dose")

else:

new\_patient\_update = open("D:\WORK\Python Programming\Covid mangement system\\vaccination.txt","a+")

new\_patient\_update.write(id+","+vaccine\_type+","+ "D1"+"\n")

if(vaccine\_type == "BV" or vaccine\_type == "CZ"):

print("Please come after 3 weeks for your second dose")

elif(vaccine\_type == "AF"):

print("Please come after 2 weeks for your second dose")

else:

print("Please come after 4 weeks for your second dose")

valid\_input = True

The above lines of codes is for the condition when the users opts for DOSE D1, line.split() function is used to split the data from the vaccination.txt file by providing the “,” as delimiter. Since the fileobject contains 2d dimensional data in form of lists, nested for loop was implemented to go through the file line one by one. Based on the vaccine selected the information was printed.

elif(dose\_input == "D2"):

with open(r"D:\WORK\Python Programming\Covid mangement system\\vaccination.txt","r") as file:

newline = [line.rstrip().split(",") for line in file]

for row, columns in enumerate(newline):

for j in columns:

if str(id) in columns:

if newline[row][2] == "D1":

newline[row][2] = "D2"

else:

print("You are already Vaccinated")

for row, columns in enumerate(newline):

for j in columns:

a,b,c = columns

update\_file = open("D:\WORK\Python Programming\Covid management system\\vaccination\_temp.txt", "a+")

update\_file.seek(0)

update\_file.write(a+","+b+","+c+"\n")

update\_file.close()

with open("D:\WORK\Python Programming\Covid mangement system\\vaccination.txt", "w") as clear:

clear.truncate()

clear.close()

shutil.copyfile("D:\WORK\Python Programming\Covid mangement system\\vaccination\_temp.txt","D:\WORK\Python Programming\Covid mangement system\\vaccination.txt")

open("D:\WORK\Python Programming\Covid mangement system\\vaccination\_temp.txt", "w").close

valid\_input = True

else:

print("Wrong input")

except(TypeError):

print("Id not found")

The above line of codes for the condition when the user selects for second Dose, similar readstrip method was used to read the data in list form file. Enumerate method is used for reading the files object which is in list form to convert into enumerating object types, by that we will have rows and columns of the list and hence will be easier to implement. Since the 3 columns(2nd index) contains the information of the dose that element is replace by D2 if the index contains D1. The new value is stored in a temporary file, the values of the previous file is deleted by opening the file in write mode, truncate function is used to erase the file data. The temporary file’s data is then copied to the previous file, the temporary file data is deleted which is required for the new operation.

**RESULTS AND ANALYSIS**

Text

Description automatically generated

Figure 4:- Menu output

Figure 4 shows output of the menu which contains five options as required in the assignment.

Text

Description automatically generated

Figure 5:- Wrong input

Figure 5 shows the output when the user enters selection out of the range required. It’s shows the error and ask the user to enter between 1 to 5.

Text

Description automatically generated

Figure 6:- 1st option selection.

Figure 6 shows the output when the user selects the first option it shows the user to select between the vaccination center VC1 and VC2.

Schematic, box and whisker chart

Description automatically generated

Figure 7:- Patient registration.

Figure 7 shows the output of the patient registration where age, vaccination code and phone number is entered by the user, in figure 8 line 6 it can be seen that a unique id is assigned to the user and the information is stored in the file.

Text

Description automatically generated

Figure 8:- Patient.txt

Text

Description automatically generated

Figure 9:- Vaccine administration

Text

Description automatically generated

Figure 10:- D1 option

Text

Description automatically generated Text

Description automatically generated

Figure 11:- D2 option

Figure 9, 10 and 11 are the vaccine administration where figure 9 shows the output when the user enter a patient id which is not registered yet. Figure 10 is when the user selects for the Dose D1 but the user is already vaccinated hence the output informing the user to opt for D2. Figure 11 shows the output when the user enters the D2 option and the file changed in the pateinet.txt file.

Text

Description automatically generated

Figure 12:- 3rd menu selection.

Figure 12 shows the statistics of the desired patient id, in this case it was 101.

Text

Description automatically generated

Figure 13:- Output of 4th option.

Text

Description automatically generated Text

Description automatically generated

Figure 14:- Statistics text file

Figure 13 is for the output of 4th option which is showing the statistics of the number of patients as per vaccination center. From figure 14, there are total 6 registration done during the time of screenshot of the operation, while only 5 patient was registered for their doses.

**CONCLUSION:-**

The assignment was to develop a python program to manage the vaccination record for the two vaccination centers.There were 4 objective of the assignment, first objective was to generate a unique id for each individual and store their information. This objective was completed by creating a registration function which returns the stored information back to the mainfile. Second objective was to record the vaccine administered, this objective was achieved by creating a separate function vaccineadministration which incorporates all the error handling techniques and required parameters, the output of the file is stored in vaccination.txt. Third and fourth objective was related to the information display for the patient and as per the vaccination center, this objective was achieved, and results are shown in the section. The program incorporates several functionality of the python such as inclusion of file handling, try and catch block, while, for and if conditional statements, predefined functions such as truncate, enumerate, shutil, Input output operations, comments, indentation, user defined function etc. It can be said that by incorporating all the functionality and achieving all the objective the overall objectives of the assignment were achieved.