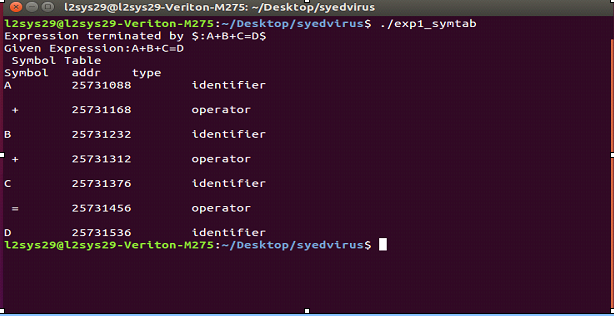
**Experiment No 1**

**AIM:**  
To write a program for implementing Symbol Table   
 **ALGORITHM**

**Step1:** Start the program for performing insert, display, delete, search and modify option in symbol table  
**Step2:**Define the structure of the Symbol Table  
**Step3:** Enter the choice for performing the operations in the symbol Table  
**Step4:** If the entered choice is 1, search the symbol table for the symbol to be inserted. If the symbol is  
already present, it displays “Duplicate Symbol”. Else, insert the symbol and the corresponding address in  
the symbol table.  
**Step5:** If the entered choice is 2, the symbols present in the symbol table are displayed.  
**Step6:**If the entered choice is 3, the symbol to be deleted is searched in the symbol table.  
**Step7:**If it is not found in the symbol table it displays “Label Not found”. Else, the symbol is deleted.  
**Step8:** If the entered choice is 5, the symbol to be modified is searched in the symbol table.

**Sample Input and Output:**[](https://4.bp.blogspot.com/-7ZMLcvjzN_w/WiY4jj8Og3I/AAAAAAAAC3I/HE_HY3MBrl8uRjaF_xitsa7-w4IWoT0HACLcBGAs/s1600/sym.PNG)

**Code :**

import string

import pandas as pd

import numpy as np

import random

import os.path

from csv import writer

from tabulate import tabulate

def take\_input():

  print("1 . Create Table  2 . Search Table  3. Enter Symbol 4. Remove Symbol 5. View Table 6 . Exit")

  n = int(input("Enter Your Choice "))

  return n

IDENTIFIERS = list(string.ascii\_letters) + ['1','2','3','4','5','6','7','8','9','0']

OPERATORS = ['+' ,'-' , '\*','/','=']

INPUT\_LIST = []

FILE\_NAME = "Table\_Data.csv"

COLUMN = ["SYMBOL",'ADDRESS','TYPE']

def create\_table():

  print("Creating Table in Progress")

  global INPUT\_LIST

  if(len(INPUT\_LIST) > 0):

    for expression in INPUT\_LIST:

      generate\_table(expression)

    INPUT\_LIST = []

  else:

    print("ENTER EXPRESSION FIRST")

def generate\_table(expression):

  letters = [x for x in expression]

  data = {

      "SYMBOL":[],

      "ADDRESS":[],

      "TYPE":[]

  }

  try:

    file\_read = pd.read\_csv(FILE\_NAME,index\_col=[0])

    symbols = file\_read['SYMBOL'].to\_list()

  except FileNotFoundError:

    symbols = []

    pass

  for sym in letters:

    if not (sym in symbols) or len(symbols) == 0 :

      address = id(sym)

      data["ADDRESS"].append(address)

      data["SYMBOL"].append(sym)

      if sym in OPERATORS :

        data["TYPE"].append("operators")

      elif sym in IDENTIFIERS:

        data["TYPE"].append("identifiers")

    else:

      print(f"Sorry Unable To Update Table As {sym} already exits in File ")

      return

  # print(data)

  new\_file = pd.DataFrame(data)

  try:

    old\_file = pd.read\_csv(FILE\_NAME,index\_col=[0])

    df = pd.concat([old\_file , new\_file], ignore\_index= True)

    df.to\_csv(FILE\_NAME )

    print("SuccessFully Created Table")

  except FileNotFoundError:

    new\_file.to\_csv(FILE\_NAME)

    print("File Created")

# generate\_table("a=b+c-d\*5")

def enter\_input():

  expression = input("Enter Your Expression")

  # INPUT\_LIST.append(expression)

  generate\_table(expression)

def search\_in\_table(alphabet):

  try:

    file\_read = pd.read\_csv(FILE\_NAME,index\_col=[0])

    result = file\_read[file\_read['SYMBOL'] == alphabet]

    print(result)

  except FileNotFoundError:

    print("Sorry Couldnt Read File As It Does Not Exist")

    return

# search\_in\_table("C")

def remove\_from\_table(alpha):

  try:

    file\_read = pd.read\_csv(FILE\_NAME ,index\_col=[0])

    print(file\_read['SYMBOL'])

    if alpha in file\_read['SYMBOL'].to\_list():

      new\_file = file\_read[file\_read['SYMBOL'] != alpha]

      new\_file.to\_csv(FILE\_NAME)

      print(f"{alpha} Removed From Table Data")

    else:

      print(f"{alpha} Does Not Exist in Table Data")

  except FileNotFoundError:

    print("Sorry Couldnt Read File As It Does Not Exist")

    return

# remove\_from\_table('M')

def view\_table():

  try:

    file\_read = pd.read\_csv(FILE\_NAME,index\_col=[0])

    print(tabulate(file\_read, headers='keys', tablefmt='psql'))

  except FileNotFoundError:

    print("Sorry Couldnt Read File As It Does Not Exist")

    return

# view\_table()

while True:

  t = take\_input()

  if t == 1 :

    # create table

    create\_table()

  elif t == 2:

    #  search in table

    sym = input("Enter Alphabet To Be Searched....")

    search\_in\_table(sym)

  elif t == 3:

    # Enter Symbol

    enter\_input()

  elif t == 4 :

    # Remove Symbo

    sym = input("Enter Symbol To Be Removed From Table")

    remove\_from\_table(sym)

  elif t == 5:

    # View Table

    view\_table()

  elif t == 6:

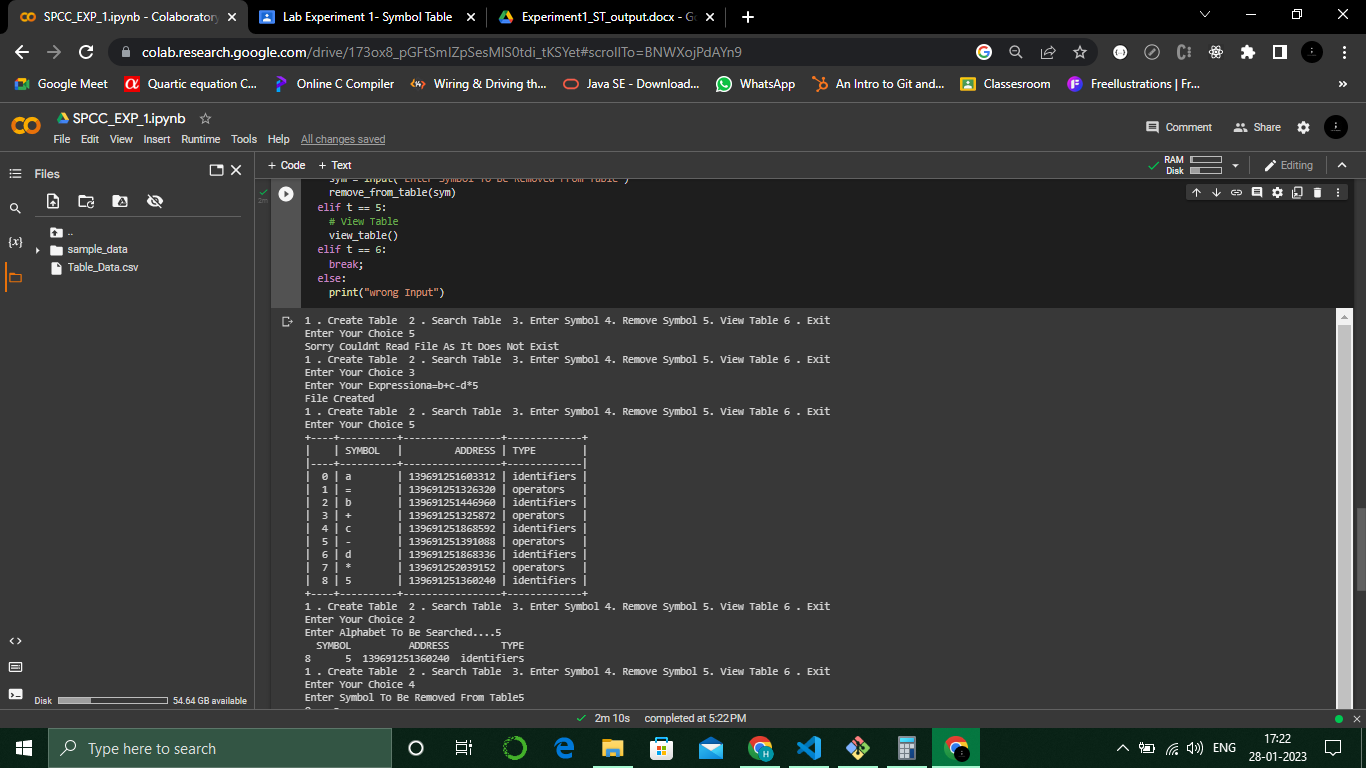
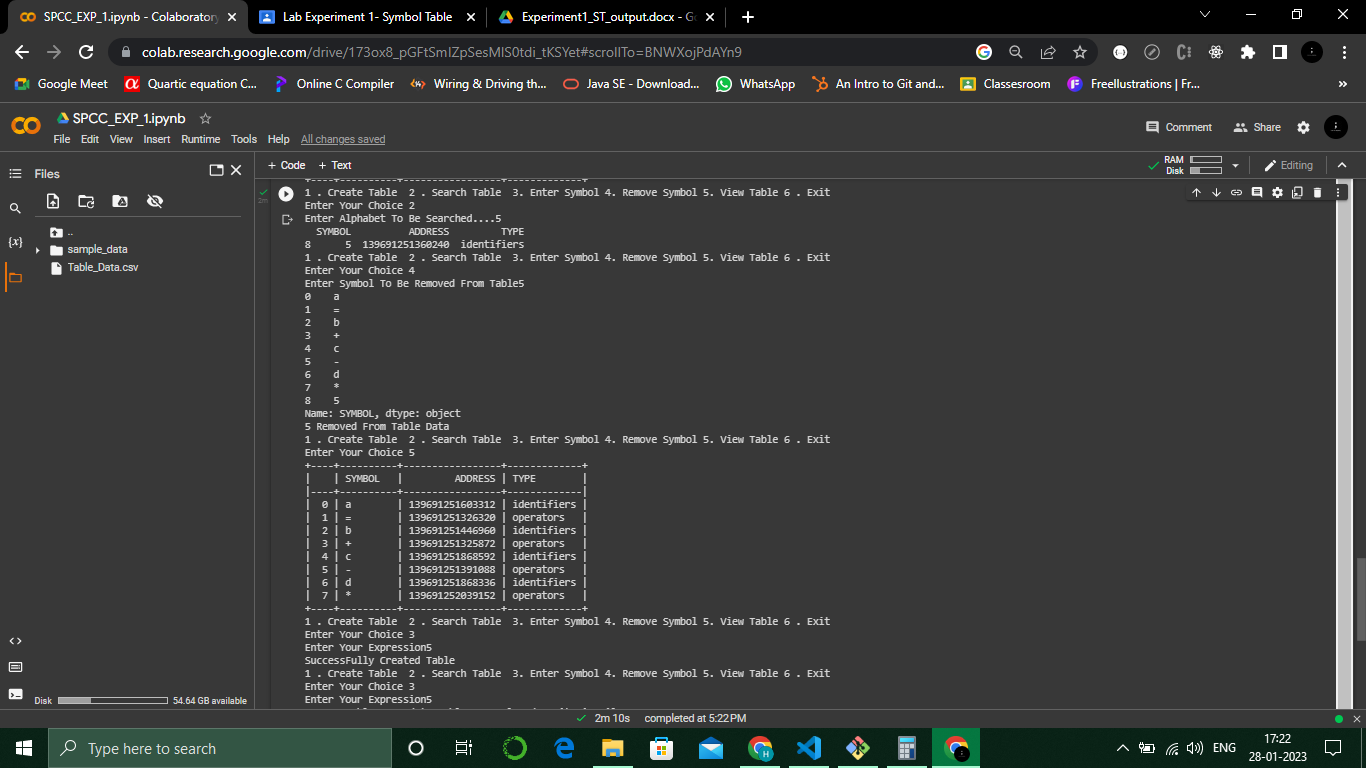
    break;

  else:

    print("wrong Input")

**Postlab Questions:**

1. Explain different phases of compiler. Illustrate all the output after each phase for the following statement

a= b +c -d \*5

