PYTHON CODING

* **print**"\*\*\*\*\*\*\*\*\*\*\*\*\***LAB: SPECIES::MANAGEMENT**\*\*\*\*\*\*\*\*\*\*\*"

"""\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**MODULES USED IN PROJECT**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

Import pickle

Import os

"""\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**CLASS USED IN PROJECT**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

**class account**(object):

**def \_\_init\_\_(s):**

s.acno=0

s.name=""

s.deposit=0

s.type=""

**def create\_account(s):** #function to get data from user

name=raw\_input("\n\nEnter the name of the SPECIES: ")

s.name=name.capitalize()

type=raw\_input("\nEnter TYPE OF SPECIES : ")

s.type=type.upper()

s.deposit=input("\nEnter NUMBER OF SPECIMENS : ")

**def show\_account(s):** #function to show data on screen

print "\nLAB SPECIMEN NUMBER. :", s.acno

print "\n NAME OF SPECIES: ", s.name

print "\nTYPE OF SPECIES", s.type

print "\nNUMBER OF SPECIMENS EXIST IN LAB: ", s.deposit

**def modify(s):** #function to get new data from user

print "\n SPECIMEN LAB NO.. : ", s.acno

s.name=raw\_input("\n\nEnter the name of Species: ")

type=raw\_input("\n\nEnter type of Species: ")

s.type=type.upper()

s.deposit=input("\nEnter the NUMBER OF SPECIMENS : ")

**def dep(s,x):** #function to accept amount and add to balance

s.deposit+=x

**def draw(s,x):** #function to accept amount and subtract from balance

s.deposit-=x

**def report(s):** #function to show data in tabular format

print "%-15s"%s.acno,"%-20s"%s.name,"%-15s"%s.type,"%-15s"%s.deposit

**def retacno(s):** #function to return account number

return s.acno

**def retdeposit(s):** #function to return balance amount

return s.deposit

**def rettype(s):** #function to return type of account

return s.type

"""\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**FUNCTION TO GENERATE ACCOUNT NUMBER**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

**def gen\_acno():**

try:

inFile=open("account2.dat","rb")

outFile=open("text2.dat","wb")

n=inFile.read()

n=int(n)

while True:

n+=1

outFile.write(str(n))

inFile.close()

outFile.close()

os.remove("account2.dat")

os.rename("text2.dat","account2.dat")

yield n

except IOError:

print "I/O error occured"

"""\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**FUNCTION TO WRITE RECORD IN BINARY FILE**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

**def SPECIES\_ADDING():**

try:

ac=account()

outFile=open("account.dat","ab")

ch=gen\_acno()

ac.acno=ch.next()

ac.create\_account()

pickle.dump(ac,outFile)

outFile.close()

print "\n\n SPECIES ADDED Successfully"

print "\n\n YOUR SPECIES LAB NUMBER IS: ",ac.retacno()

except:

pass

"""\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**FUNCTION TO DISPLAY ACCOUNT DETAILS GIVEN BY USER**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

**def display\_sp(n):**

flag=0

try:

inFile=open("account.dat","rb")

print "\nDISPLAYING RECORDS OF SPECIEMENS\n"

while True:

ac=pickle.load(inFile)

if ac.retacno()==n:

ac.show\_account()

flag=1

except EOFError:

inFile.close

if flag==0:

print "\n\nSpecimen Lab number not exist"

except IOError:

print "File could not be open !! Press any Key..."

"""\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**FUNCTION TO MODIFY RECORD OF FILE**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

**def modify\_account(n):**

found=0

try:

inFile=open("account.dat","rb")

outFile=open("temp.dat","wb")

while True:

ac=pickle.load(inFile)

if ac.retacno()==n:

print 30\*"-"

ac.show\_account()

print 30\*"-"

print "\n\nEnter The New Details of Species"

ac.modify()

pickle.dump(ac,outFile)

print "\n\n\tSpecimen Record Updated"

found=1

else:

pickle.dump(ac,outFile)

except EOFError:

inFile.close()

outFile.close()

if found==0:

print "\n\nRecord Not Found "

except IOError:

print "File could not be open !! Press any Key..."

os.remove("account.dat")

os.rename("temp.dat","account.dat")

"""\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**FUNCTION TO DELETE RECORD OF FILE**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

**def delete\_account(n):**

found=0

try:

inFile=open("account.dat","rb")

outFile=open("temp.dat","wb")

while True:

ac=pickle.load(inFile)

if ac.retacno()==n:

found=1

print "\n\n\tSpecimen Record Deleted .."

else:

pickle.dump(ac,outFile)

except EOFError:

inFile.close()

outFile.close()

if found==0:

print "\n\n Specimen Record Not Found"

except IOError:

print "File could not be open !! Press any Key..."

os.remove("account.dat")

os.rename("temp.dat","account.dat")

"""\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**FUNCTION TO DISPLAY ALL ACCOUNT DETAILS**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

**def display\_all():**

print "\n\n\t LAB SPECIMENS LIST\n\n"

print 73\*"="

print "%-15s"%"SPECIMEN LAB NO..","%-15s"%"NAME OF SPECIES","%-10s"%"Type OF SPECIES ","%-10s"%"NUMBER OF SPECIMENS "

print 73\*"=","\n"

try:

inFile=open("account.dat","rb")

while True:

ac=pickle.load(inFile)

ac.report()

except EOFError:

inFile.close()

except IOError:

print "File could not be open !! Press any Key..."

"""\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* **FUNCTION TO DEPOSIT/WITHDRAW AMOUNT FOR GIVEN ACCOUNT**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*”

**def deposit\_withdraw(n,option):**

found=0

try:

inFile=open("account.dat","rb")

outFile=open("temp.dat","wb")

while True:

ac=pickle.load(inFile)

if ac.retacno()==n:

ac.show\_account()

if option==1:

print "\n\n\tTO ADD NUMBER OF SPECIMENS"

amt=input("Enter the number of specimens: ")

ac.dep(amt)

elif option==2:

print "\n\n\tTO TAKEOUT THE SPECIMENS"

amt=input("ENTER NUMBER OF REMOVE SPECIMEN : ")

bal=ac.retdeposit()-amt

if((bal<5 and ac.rettype()=="S")or(bal<10 and ac.rettype()=="C")):

print "Insufficient number of species"

else:

ac.draw(amt)

pickle.dump(ac,outFile)

found=1

print "\n\n\tSPECIES Record Updated"

else:

pickle.dump(ac,outFile)

except EOFError:

inFile.close()

outFile.close()

if found==0:

print "\n\n Species Record Not Found"

except IOError:

print "File could not be open !! Press any Key..."

os.remove("account.dat")

os.rename("temp.dat","account.dat")

"""\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**INTRODUCTORY FUNCTION**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

**def intro():**

print "\n\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*LAB SPECIES MANAGEMENT\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

print "\n\n\n MADE BY : JAGMOHAN SINGH"

print "\n SCHOOL : ARMY PUBLIC SCHOOL , JODHPUR"

"""\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**THE MAIN FUNCTION OF PROGRAM**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

**intro()**

while True:

print 2\*"\n",73\*"="

print """MAIN MENU

1. SPECIES ADDING

2. ADD NUMBER OF SPECIMENS

3. REMOVE NUMBER OF SPECIMENS

4. NUMBER OF SPECIMENS ENQUIRY

5. ALL SPECIES LIST

6. DELETING SPECIES

7. UPDATING SPECIES

8. Exit

"""

try:

ch=input("Enter Your Choice(1~8): ")

if ch==1:

SPECIES\_ADDING()

elif ch==2:

num=input("\n\nEnter lab Specimen Number: ")

deposit\_withdraw(num,1)

elif ch==3:

num=input("\n\nEnter lab Specimen Number: ")

deposit\_withdraw(num,2)

elif ch==4:

num=input("\n\nEnter lab Specimen Number: ")

display\_sp(num)

elif ch==5:

display\_all()

elif ch==6:

num=input("\n\nEnter lab Specimen Number: ")

delete\_account(num)

elif ch==7:

num=input("\n\n Enter lab Specimen Number: ")

modify\_account(num)

elif ch==8:

raw\_input("\n\n\n\n\n THANK YOU\n\n Press any key to exit...")

exit()

else:

print "Input correct choice...(1-8)"

**except** NameError:

print "Input correct choice...(1-8)"

"""\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**END OF PROJECT**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""