PROPERTY MANAGEMENT SYSTEM

Name: B Nethra
Registration number:

BONAFIDE CERTIFICATE

Certified to be the Bonafide Project work in COMPUTER SCIENCE done by B NETHRA___, Registration Number ______________________of Class XII Section B of D.A.V GIRLS SR.SEC. SCHOOL, GOPALAPURAM, CHENNAI – 600086 during the year 2024-2025.

Signature of Principal

Signature of Subject Teacher

School Seal

Submitted for the Practical Examination held on _____________________ at D.A.V.GIRLS SR. SEC. SCHOOL, GOPALAPURAM, CHENNAI – 86

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

I would like to offer my sincere thanks to our Principal Smt. S.Sindhu for giving me the opportunity to do this project, which was a wonderful knowledge gaining experience.

I also extend my gratitude to our Computer Science Teacher Mrs. Uma Thiagarajan for her immense support and guidance throughout the completion of this project.

INDEX

AIM OF THE PROJECT

The primary purpose of making the online property management system is to create an automatic online based system which will provide a friendly format for purchasing and selling properties. Users can search and browse for property based on their own preferences and can also book their desired properties online using this python code which includes the usage of binary files. This code mainly concentrates on maintaining and managing the small print of the property. The code deals with buying and selling residential building, commercial and agricultural land. This code provides the functionality for the sellers to log into the system to add new properties and also to delete the existing ones. This system would definitely go to reduce labor and make business more profitable and promising to clients in locating properties without the brokers.

FILES USED

- **Land_for_sale.DAT:** Stores the data of plots of land that are put for sale by property owners such as Id, location, price, type (agricultural/ commercial), status (for sale/sold) and the details of owner such as name, contact number and email address. Each record in the file is in the form of a dictionary with fixed keys. Reading and writing in the file is done using python's pickle module.

  Structure of each record:

  {"ID": 10001, "Location": "Bangalore", "Price":2500,
  "Type": "Agricultural", "Owner name": "Alex",
  "Contact number": 8745692031,
  "Email address": "xyz@gmail.com", "Status": "Up for sale"}

- **Residential_building.DAT:** This file stores details about residential buildings such as Id, location, price, type (rent/ lease/apartment), status (for sale/sold) and details of owner such as name, contact number and email address. Each record in the file is in the form of a dictionary with fixed keys. Reading and writing in the file is done using python's pickle module.

  Structure of each record:

  {"ID": 10012, "Location": "Bangalore", "Price":2500,
  "Type": "Rent", "Owner name": "Riya",
  "Contact number": 8745296031,
  "Email address": "xyz@gmail.com", "Status": "Up for sale"}

USER DEFINED FUNCTIONS USED

In Seller module:

- ADD_land():
  Takes the details of new land, which has to be included in the file as inputs from the user and dumps it in the "Land_for_sale.dat" binary file.

- ADD_rb():
  Takes the details of new residential building, which has to be included in the file as inputs from the user and dumps it in the "Residential_building.dat " binary file.

- Delete(file):
  Its parameter is a binary file. It loads all record and deletes those records whose status is "Booked". That is it deletes all the booked properties from the file passed as parameter.

- Update(file):
  Its parameter is a binary file. This UDF is a menu driven code offering 3 choices to update the price, contact number or email address in the file passed as parameter.

In Buyer module:

- disp_land ():
  This UDF displays all the available land’s details in the "Land_for_sale.dat " binary file.

- disp_rb():
  This UDF displays the details of all the available residential buildings in the "Residential_building.dat " binary file.

- search_loc(file,place):
  This UDF takes two parameters, file and interested location. This UDF searches for land or residential building in their given respective files and displays those properties' details whose location matches with the passed one.

- search_pr(file,ll,ul):
  This UDF takes three parameters, a binary file, minimum and maximum price. This UDF searches and displays those properties' details whose price lies in the range of price passed as parameters from their respective files.

- search_type(file,t):
  This UDF takes two parameters. One is the file and other one is the type of land/residential building which the user is looking for whether commercial or agricultural if land or rent/ lease/ apartment if residential building and it displays the details of properties of that type.

- buy(file):
  This UDF takes one parameter. This UDF inputs the ID of the property which the user wishes to buy. It loads all records from the given binary file and checks if the ID matches with the inputted one, and if yes, changes its status to Booked

PROGRAM LISTING

#FUNCTIONS IN THE SELLER MODULE

```
import pickle,os
def ADD_land():
    with open('Land_for_sale.dat','ab+') as F:
        while True:
            I=int(input("Enter ID of property:"));c=0
            while True:
                try:
                    X=pickle.load(F)
                    if X['ID']==I:
                        c+=1;print("ID already used! Enter new ID"); break
                except EOFError: break
            if c==0: break
        loc=input("Location:")
        pr=int(input("price/sq feet:"))
        on=input("Owner name:")
        ph=int(input("Contact number:"))
        em=input("Email address:")
        s=input("Suitable for agriculture/commercial:")
        pickle.dump({'ID':I,'Location':loc,'Price':pr,'Type':s,'Owner  name':on,'Contact number':ph,'Email address':em,'Status':"up for sale"},F)
```

```
def ADD_rb():
    with open('Residential_building.dat','ab+') as F:
        while True:
            I=int(input("Enter ID of property:"))
            c=0
            while True:
                try:
                    X=pickle.load(F)
                    if X['ID']==I:
                        c+=1;print("ID already used! Enter new ID"); break
                except EOFError: break
            if c==0: break
        loc=input("Location:")
        pr=int(input("price:"))
        on=input("Owner name:")
        ph=int(input("Contact number:"))
        em=input("Email address:")
        t=input("Type(Apartment/rent/lease):")
        pickle.dump({'ID':I,'Location':loc,'Price':pr,'Type':t,'Owner name':on,'Contact number':ph,'Email address':em,'Status':"up for sale"},F)
```

```
def Delete(file):
    with open(file,'rb') as F, open('temp.dat','wb') as G:
        c=0
        while True:
            try:
                X=pickle.load(F)
                if X['Status']=='Booked':
                    pass
                else: pickle.dump(X,G);c+=1
            except :
                if c==0: print("No properties booked")
                else: print("Details of booked properties are deleted")
                break
    os.remove(file)
    os.rename('temp.dat',file)
def Update(file):
    print('''
    Update details of
    1. Price
    2. Contact number
    3.Email address''')
 with open(file,'rb+') as F:
        a=int(input("Enter choice:"))
        Id=int(input("Enter Id of your property:"));c=0
```

```
while True:
    try:
        pos=F.tell()
        X=pickle.load(F)
        if X['ID']==Id:
            c+=1
            if a==1:
                npr=int(input("Enter new price:")); X['Price']=npr
            elif a==2:
                ncn=int(input("Enter new contact number:"));
                  X["Contact number"]=ncn
            elif a==3:
                E=input("Enter new Email Id:"); X['Email address']=E
            else: print("Invalid choice:")
        F.seek(pos)
        pickle.dump(X,F)
    except EOFError:
        if c==0: print("Invalid ID")
        else: print("Details updated")
        break
```

#FUNCTIONS IN THE BUYER MODULE

```
import pickle
def disp_land():
    with open("Land_for_sale.dat","rb") as F:
        while True:
            try:
                X=pickle.load(F)
                for k in X:
                    print(k,X[k],sep=':')
                print()
            except : break
def disp_rb():
    with open("Residential_building.dat",'rb') as F:
        while True:
            try:
                X=pickle.load(F)
                for k in X:
                    print(k,X[k],sep=':')
                print()
            except : break
def search_loc(file,place):
    with open(file,'rb') as F:
        c=0
        while True:
            try:
                X=pickle.load(F)
                if X['Location']==place:
                    c+=1
                    for k in X:
                        print(k,X[k],sep=':')
                    print()
            except :
                if c==0: print("No property found in",place)
                break
def search_pr(file,ll,ul):
    with open(file,'rb') as F:
        c=0
        while True:
            try:
                X=pickle.load(F)
                if ll<= X['Price']<=ul:
                    c+=1
                    for k in X:
                        print(k,X[k],sep=':')
                    print()
            except :
                if c==0: print("No properties found in given price range")
                break
```

```
def search_type(file,t):
    with open(file,'rb') as F:
        c=0
        while True:
            try:
                X=pickle.load(F)
                if X['Type']==t:
                    c+=1
                    for k in X:
                        print(k,X[k],sep=':')
                    print()
            except :
                if c==0: print("No property of given type is found")
                break
def buy(file):
    i=int(input("Enter ID of property which you wanna buy:"))
    with open(file,'rb+') as F:
        c=0
        while True:
            try:
                pos=F.tell()
                X=pickle.load(F)
                if X['ID']==i:
                    X['Status']='Booked';c+=1
                F.seek(pos)
                pickle.dump(X,F)
            except :
                if c==0: print("Invalid ID")
                break

import BUYER,SELLER
while True:
    x=int(input('''Login as a
    1.SELLER
    2.BUYER
    3.EXIT: '''))
    if x==1:
        print("""        MENU
        1.Add property
        2.Delete booked properties
        3.Update property details
        4.Return to login """)
        n=int(input("Enter your choice:"))
        while True:
            if n==1:
                c=input("Is it a land or residential building:")
                if c.lower()=="land":
                    SELLER.ADD_land()
                    print("Your land's details  are added")
                elif c.lower()=="residential building":
                    SELLER.ADD_rb()
                    print("Your residential building's details are added")
                else :
                    print("Invalid entry")
            elif n==2:
                c=input("Is it a land or residential building:")
                if c.lower()=="land":
                    f='Land_for_sale.dat'
                    SELLER.Delete(f)
                elif c.lower()=="residential building":
                    f='Residential_building.dat'
                    SELLER.Delete(f)
                else : print("Invalid entry")
            elif n==3:
                c=input("Is it a land or residential building:")
                if c.lower()=="land":
                    f='Land_for_sale.dat'
                    SELLER.Update(f)
                elif c.lower()=="residential building":
                    f='Residential_building.dat'
                    SELLER.Update(f)
                else : print("Invalid entry")
            elif n==4:
                print("Returning to login");break
```

```
        else:
            print("Invalid input")
        n=int(input("Enter your choice:"))

elif x==2:
    c=input("do you want to buy land/residential building:")
    if c.lower()=="land":
        print("""           MENU
        1.Display the available land's details
        2.Display all lands in interested location
        3.Display all lands within the  expected price
        4.Diplay only agriculture/commercial lands
        5.To book a property
        6.Return to login""")
        while True:
            n=int(input("Enter your choice:"))
            if n==1:
                print("The available land's details are:")
                BUYER.disp_land()
            elif n==2:
                l=input("Enter the desired location:")
                print("The lands available in",l,"are:")
                BUYER.search_loc("Land_for_sale.dat",l)
            elif n==3:
                min=int(input("Enter the  minimum  affordable price:"))
            max=int(input("Enter the  maximum affordable price:"))
            print("The lands available in the given price range are:")
            BUYER.search_pr("Land_for_sale.dat",min,max)
        elif n==4:
            ty=input("Enter the type of land agricultural/commercial:")
            BUYER.search_type("Land_for_sale.dat",ty)
        elif n==5:
            BUYER.buy("Land_for_sale.dat")
            print("Land booked")
        elif n==6:
            print("Returning to login")
            break
        else:
            print("Invalid input")

if c.lower()=="residential building":
    print("""            MENU
    1.Display the available residential building's details
    2.Display all residential buildings in interested location
    3.Display all residential buildings within the price expected
    4.Display residential buildings of desired type
    5.To book a property
    6.Return to login""")
    n=int(input("Enter your choice:"))
```

```
while True:
  if n==1:
      print("The available residential building's details are:")
      BUYER.disp_rb()
   elif n==2:
      l=input("Enter the desired location:")
      print("The residential building's details available in",l,"are:")
      BUYER.search_loc("Residential_building.dat",l)
   elif n==3:
      min=int(input("Enter the  minimum  affordable price:"))
      max=int(input("Enter the  maximum affordable price:"))
      print("The residential buildings available in the given price range are:")
      BUYER.search_pr("Residential_building.dat",min,max)
   elif n==4:
      t=input("Enter the type of residential building apartment/rent/lease :")
      print("The",t,"type available residential buildings are:")
      BUYER.search_type("Residential_building.dat",t)
   elif n==5:
      BUYER.buy("Residential_building.dat")
      print("Residential building booked")
   elif n==6:
      print("Returning to login")
      break
```

```
            else:
                print("Invalid input")
            n=int(input("Enter your choice:"))
elif x==3:
    print("*THANKS FOR VISITING*")
    break
else:
    print("Invalid input")
```

SAMPLE OUTPUT

```
Login as a
    1.SELLER
    2.BUYER
    3.EXIT: 1
         MENU
        1.Add property
        2.Delete booked properties
        3.Update property details
        4.Return to login
Enter your choice:1
Is it a land or residential building:land
Enter ID of property:14001
Location:Chennai
price/sq feet:2000
Owner name:Priya
Contact number:8745693201
Email address:priya72@gmail.com
Suitable for agriculture/commercial:agriculture
Your land's details  are added
Enter your choice:1
Is it a land or residential building:residential building
Enter ID of property:17002
```

```
Location:Bangalore
price:2500
Owner name:Rahul
Contact number:9658740231
Email address:rahul644@gmail.com
Type(Apartment/rent/lease):rent
Your residential building's details are added
Enter your choice:1
Is it a land or residential building:residential building
Enter ID of property:12003
Location:Mumbai
price:3000
Owner name:Rohini
Contact number:8547693021
Email address:rohini836@gmail.com
Type(Apartment/rent/lease):apartment
Your residential building's details are added
Enter your choice:1
Is it a land or residential building:land
Enter ID of property:85004
Location:Delhi
price/sq feet:2000
Owner name:Subbarao
```

```
Contact number:6381957420
Email address:subbarao42@gmail.com
Suitable for agriculture/commercial:commercial
Your land's details  are added
Enter your choice:3
Is it a land or residential building:land

Update details of
1. Price
2. Contact number
3.Email address
Enter choice:2
Enter Id of your property:85004
Enter new contact number:7238401956
Details updated
Enter your choice:4
Returning to login
Login as a
   1.SELLER
   2.BUYER
   3.EXIT:2
do you want to buy land/residential building: residential building
```

```
          MENU
           1.Display the available residential building's details
           2.Display all residential buildings in interested location
           3.Display all residential buildings within the price expected
           4.Display residential buildings of desired type
           5.To book a property
           6.Return to login
Enter your choice:1
The available residential building's details are:
ID:17002
Location:Bangalore
Price:2500
Type:rent
Owner name:Rahul
Contact number:9658740231
Email address:rahul644@gmail.com
Status:up for sale

ID:12003
Location:Mumbai
Price:3000
Type:apartment
Owner name:Rohini
```

```
Contact number:8547693021
Email address:rohini836@gmail.com
Status:up for sale

Enter your choice:2
Enter the desired location:Mumbai
The residential building's details available in Mumbai are:
ID:12003
Location:Mumbai
Price:3000
Type:apartment
Owner name:Rohini
Contact number:8547693021
Email address:rohini836@gmail.com
Status:up for sale

Enter your choice:5
Enter ID of property which you wanna buy:12003
Residential building booked
Enter your choice:6
Returning to login
Login as a
   1.SELLER
   2.BUYER
   3.EXIT:2
do you want to buy land/residential building:land
         MENU
          1.Display the available land's details
          2.Display all lands in interested location
          3.Display all lands within the  expected price
          4.Diplay only agriculture/commercial lands
          5.To book a property
          6.Return to login
Enter your choice:3
Enter the  minimum  affordable price:1750
Enter the  maximum affordable price:2500
The lands available in the given price range are:
ID:14001
Location:Chennai
Price:2000
Type:agriculture
Owner name:Priya
Contact number:8745693201
Email address:priya72@gmail.com
Status:up for sale
```

```
ID:85004
Location:Delhi
Price:2000
Type:commercial
Owner name:Subbarao
Contact number:7238401956
Email address:subbarao42@gmail.com
Status:up for sale

Enter your choice:4
Enter the type of land agricultural/commercial:agriculture
ID:14001
Location:Chennai
Price:2000
Type:agriculture
Owner name:Priya
Contact number:8745693201
Email address:priya72@gmail.com
Status:up for sale

Enter your choice:5
Enter ID of property which you wanna buy:14001
Land booked
```

```
Enter your choice:6
Returning to login
Login as a
    1.SELLER
    2.BUYER
    3.EXIT:1
         MENU
        1.Add property
        2.Delete booked properties
        3.Update property details
        4.Return to login
Enter your choice:2
Is it a land or residential building: residential building
Details of booked properties are deleted
Enter your choice:4
Returning to login
Login as a
    1.SELLER
    2.BUYER
    3.EXIT:3
*THANKS FOR VISITING*
```

BIBLIOGRAPHY

- Computer Science with Python by Sumita Arora
- Compuet Science textbook for class XII –NCERT
- https://copyassignment.com/real-estate-management-system-project-in-python/