OOPS PRESENTATION

Hotel booking system

Submitted to:
Asst Prof Arpit Deo

Submitted by:
EN20CS301311 Prateeti Mehta Jain
EN20CS301327 PurviChelawat

EN20CS301328 RadhikaPatidar

WORKING OF THE PROJECT

INTRODUCTION TO PYTHON

- Python is a popular programming language that is often applied in scripting roles.
- Python is also called as interpreted language.
- Python’s biggest strength is that bulk of it’s library is portable. It also support GUI programming and can be used to create application portable on mac, windows and unix x-windows system.

Why python ?

Reasons Why You Must Consider Writing Software Applications in Python

1) Readable and Maintainable Code

2) Multiple Programming Paradigms

3) Compatible with Major Platforms and Systems

4) Robust Standard Library

5) Many Open Source Frameworks and Tools

6) Simplify Complex Software Development

7) Adopt Test Driven Development

Tkinter

Tkinter is the Python interface to the Tk GUI toolkit shipped with Python.

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

DATE AND TIME

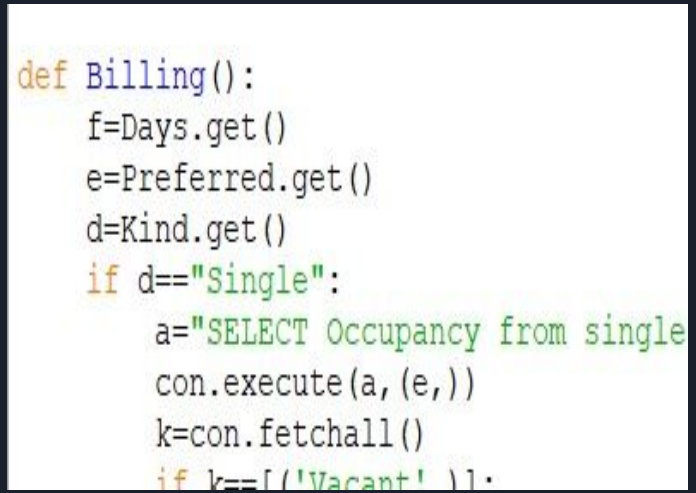
In Python, date and time are not a data type of their own, but a module named datetime can be imported to work with the date as well as time. Python Datetime module comes built into Python, so there is no need to install it externally.

Python Datetime module supplies classes to work with date and time. These classes provide a number of functions to deal with dates, times and time intervals. Date and datetime are an object in Python, so when you manipulate them, you are actually manipulating objects and not string or timestamps.

USER DEFINED FUNCTION

BILLING()

- This function is used for checking various condition and also generating the bill



```
def Billing():
    f=Days.get()
    e=Preferred.get()
    d=Kind.get()
    if d=="Single":
        a="SELECT Occupancy from single
        con.execute(a,(e,))
        k=con.fetchall()
        if k==[('Vacant',)]:
```

PAY()

- This function gives various gateway of payment.

```
                b1.grid(row=4,col
def Pay():
    win=Tk()
    win.geometry("300x200")
    b1=Button(win,text=("Pay
    b2=Button(win,text=("Pay
    b3=Button(win,text=("Pay
    l1=Label(win,text="Sorry!
    b1.pack()
    b2.pack()
    b3.pack()
    l1.pack()
```

BOOK()

- This function tells that the room has been booked and also does the necessary updates in the database.

```
def Book():
    a=Name.get()
    b=People.get()
    c=Kind.get()
    d=Preferred.get()
    e=Days.get()
    if c=="Single":
        i=("UPDATE single SET Occupancy='O
        con.execute(i,(a,b,e,d))
        mydb.commit()
        window=Tk()
```

DESCRIPTION()

- It tells about all the room that are available in a single kind along with the perks provided.

```
def Description():
    window=Tk()
    window.geometry("460x900")
    window.configure(bg="#add8e6")
    window.title("Waterfall Room Descr
    l1=Label(window, text="1. Single:-
    l2=Label(window, text="The room co
    l3=Label(window, text="Costing:-\t
    l4=Label(window, text="Rooms Numbe
    l1.grid(row=0, column=0, sticky=W)
    l2.grid(row=1, column=0, sticky=W)
    l3.grid(row=2, column=0, sticky=W)
    l4.grid(row=3, column=0, sticky=W)
```

RESTRICTION

- The database is not updated on checkout.
- It cannot be used for booking that are days away
- The screen does not close itself

FURTHER SCOPE

- This can be updated to make a proper hotel management system.
- Some feature can be added that will enable the user to book rooms for further days and even get discount.

THANK YOU

ANY DOUBTS ?