**Tkinter – Hotel Management**

**What is Project?**

This application manages the records of visitors in hotel .You can add, edit/update, delete, search or view all the records of the people staying at hotel.

This project is made using Tkinter and sqlite3.

**Getting Started**

Setting up project is simple.

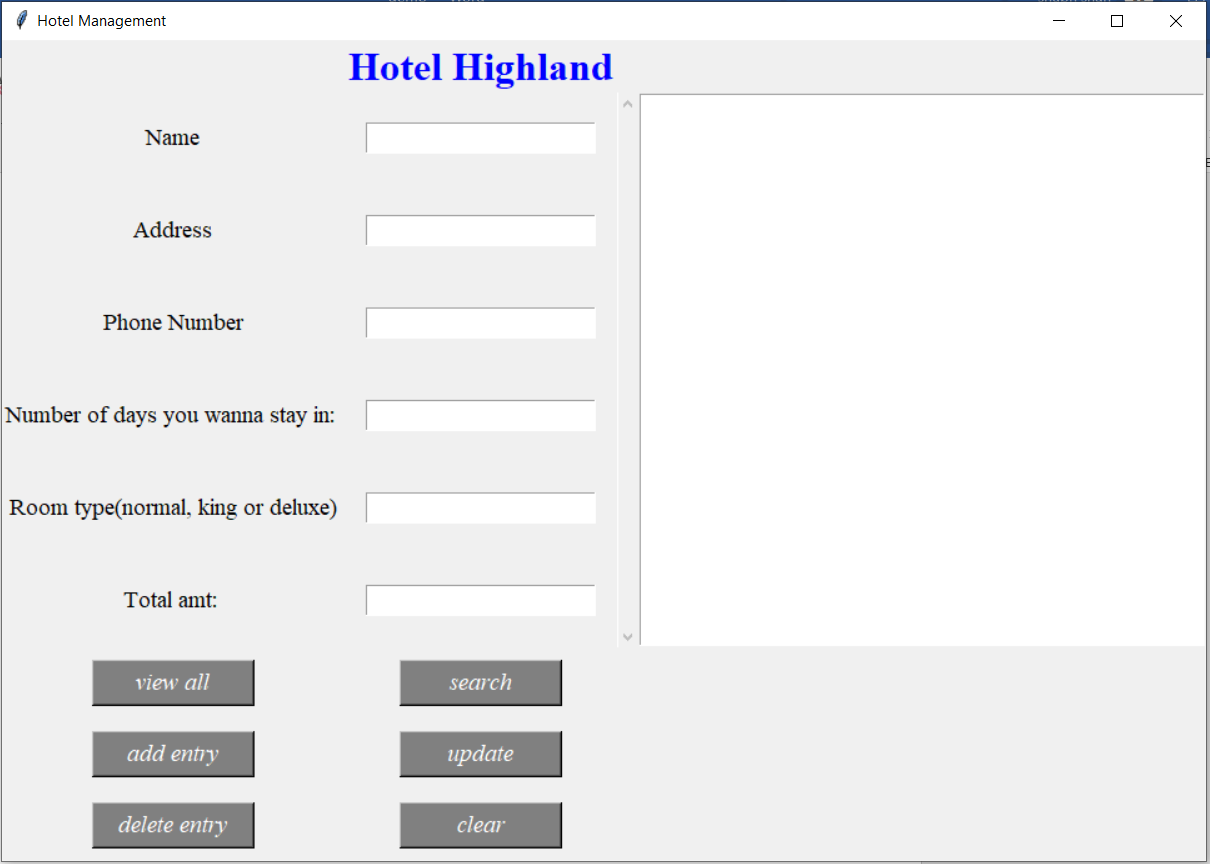
1. Clone this repo
2. cd into the root folder
3. run command ‘python front.py’ to start the project.

**Demo**

Here's a demo of what Project looks like once it's set up.

***Demo video file is uploaded as Demo.mp4 to this repo.***

***Please check it.***



**Notes (Used Tkinter for GUI):**

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.

Creating a GUI application using Tkinter is an easy task. All you need to do is perform the following steps −

* Import the *Tkinter* module.
* Create the GUI application main window.
* Add one or more of the above-mentioned widgets to the GUI application.
* Enter the main event loop to take action against each event triggered by the user.

**Now will discuss about various widgets we have used in our project:**

1. **Entry widget**: The Entry Widget is a Tkinter Widget used to Enter or display a single line of text.

Syntax: tk.Entry(parent, options)

**Parameters:**  
**1) parent:** The Parent window or frame in which the widget to display.

**2) Options:**

 **bg:**The normal background color displayed behind the label and indicator.

 **bd:**The size of the border around the indicator. Default is 2 pixels.

 **font:**The font used for the text.

 **fg:**The color used to render the text.

 **textvariable:**In order to be able to retrieve the current text from your entry widget, you must set this option to an instance of the StringVar class.

And many more exists…...

**Methods:**The various methods provided by the entry widget are:

* **get():**Returns the entry’s current text as a string.
* **delete():** Deletes characters from the widget
* **insert (index, ‘name’):** Inserts string ‘name’ before the character at the given index.

1. **Button Widget**: The Button widget is used to add buttons in a Python application. These buttons can display text or images that convey the purpose of the buttons. You can attach a function or a method to a button which is called automatically when you click the button.

Syntax: tk.Button(master, option=value ….)

Parameters

* **master** − This represents the parent window.
* **options** − Here is the list of most commonly used options for this widget. These options can be used as key-value pairs separated by commas.

**Example** **of various options:**

**1. activebackground -**

Background color when the button is under the cursor.

**2**. **activeforeground -**

Foreground color when the button is under the cursor.

**3.bd -**

Border width in pixels. Default is 2**.  
4.bg -**

Normal background color.

**5.command -**

Function or method to be called when the button is clicked.

And there are many more…...

1. **Label Widget** - This widget implements a display box where you can place text or images. The text displayed by this widget can be updated at any time you want. It is also possible to underline part of the text (like to identify a keyboard shortcut) and span the text across multiple lines.

## Parameters

* **master** − This represents the parent window.
* **options** − Here is the list of most commonly used options for this widget. These options can be used as key-value pairs separated by commas.

1. **Listbox Widget**: The Listbox widget is used to display a list of items from which a user can select a number of items.

Syntax: tk.Listbox(master, option,…)

## Parameters

* **master** − This represents the parent window.
* **options** − Here is the list of most commonly used options for this widget. These options can be used as key-value pairs separated by commas.

**It has many options like bg, bd, cursor, font, etc…**

1. **Scrollbar Widget:**

This widget provides a slide controller that is used to implement vertical scrolled widgets, such as Listbox, Text and Canvas. Note that you can also create horizontal scrollbars on Entry widgets.

Syntax: tk.Scrollbar(master, options..)

## Parameters

* **master** − This represents the parent window.
* **options** − Here is the list of most commonly used options for this widget. These options can be used as key-value pairs separated by commas.

**It has many options like activebackground, bg, bd, command, cursor, etc**..