3064.(092) – Python for Programmers

Final Project

**Visitors Log – Desktop Application**

**Introduction:**

This project is about creating a desktop application for maintaining a visitors log. This program can be used as a register of visitor where the visitor’s details like name, phone number, email id, person they are visiting and the date of visit can be stored. It also provides with many search options:

* View all the visitors name
* Filter the visitors based on dates
* Filter the visitors based on the names

It also does input data validation and displays the appropriate error messages. It also has an option to download this information into an excel sheet.

**Requirements:**

The following list of modules needs to be installed to run this application:

|  |  |  |
| --- | --- | --- |
|  | **Module** | **Description** |
| 1 | wxPython | This module is used for the UI development |
| 2 | Sqlite3 | This module is used to connect to the database |
| 3 | xlwt | This module is used to write into excel files |
| 4 | re | This module is used for regular expression |
| 5 | datetime | This module id used for the date conversion |
| 6 | wx.lib.scrolledpanel | This module adds a scrollable window |
| 7 | sys | This module contains system-specific functionality |

**Description:**

This program has the following main features:

* Designing the UI - windows
* Input validation
* Adding the data to the database
* Retrieving the data from a database based on certain conditions
* Writing the data to an excel file

Database : visitor.db (sqlite3)

Output : visitorlist.xls

The frames/screens have been developed using wxPython. Three **classes** have been used for the three frames.

**[Code line: 23-205]**

The first screen is called by creating the instance of the class mainwindow. This class is inherited from wx.Frame class. All the necessary text boxes, buttons and panel are set and the frame are customized. All the buttons are bound to a function i.e. whenever a button is pressed the corresponding function is called.

The following are the validations done:

* The text filed for the name and the person they are visiting accepts only alphabets and ‘.’
* The phone number only accepts numbers and should be in the format XXX-XXX-XXXX
* The email id accepts only alphabets ,letters, ‘.’, ‘\_’ in the address name followed by ‘@’ and then only alphabets for the account followed by ‘.’ And only alphabets after that.

i.e [(alphabets/letter/./\_)@(alphabets).(alphabets)]

* The date field accepts only numbers in the format MM/DD/YYYY.

Apart from this if any of the fields is left blank, am ‘—empty—‘ is displayed in red in the empty text boxes.

The following are the three buttons present in the window:

* ***Submit*** **:** Once the user enters the input data in the fields, and clicks on the submit button the data is first validated using regular expressions. If any of the data entered in the fields are invalid, the invalid data is highlighted in red and an error message appears at the bottom of the screen. If the data entered is valid and if none of the fields are left blank, the data is stored into database using sqlite3.A new entry is added into the visitors table present in the visitor database.
* ***Cancel***: When the user presses this button all the data entered is erased from the text boxes and data is not stored into the database.
* ***Search Options* :** Once the user presses this button, a new window named “SEARCH OPTION” is opened.

**[Code line: 212 - 337]**

The second screen is called by creating the instance of the class newwindow when the ‘Search Options’ button is pressed in the first screen. This class is also inherited from wx.Frame class. All the necessary text boxes, buttons and panel are set and the frames are customized. All the buttons are bound to a function. The following are the search options present in the window:

* ***View:*** the entire visitor list is displayed in a new window.
* ***Filter using date value:*** The visitor list can also be filtered using the date options. The name of the visitor registered between the from and to date is displayed
* ***Filter using visitor name:*** The visitor list can also be filtered based on the name either by giving the starting letter or the entire name.

If any of the button present in the window is pressed, a new window named ‘Visitor list’ is opened. The data entered for searching [from date, to date, visitor name] is also validated. If invalid data is given, the code is highlighted in red and appropriate error message is displayed at the bottom of the screen. Even if the field is left blank an ‘empty’ highlighted in red is displayed in the text boxes.

**[Code line: 344 – 386]**

The third window is called by creating an instance of the class listwindow when any button is pressed from the second window. This class is also inherited from wx.Frame class. The data is displayed according to the search query .

A button for download is also present. When the user clicks on this button, the information displayed is written to an excel sheet named visitorlist.xls

**Python elements used:**

List Comprehension:

* Line number: 333

The data retrieved from sqlite3 is in Unicode format, which when displayed directly has a ‘u’ in from of the string I.e. u’string’. So to remove the u present in the front the strings, each data fetched from the query is converted to a list using list comprehension.

* Line number : 365 , 383

Used list comprehension to capitalize the first and the last name of the visitor while displaying.

Function:

Each class used has many functions performing various tasks.

Classes:

Three classes inheriting wx.Frame have been used for the three windows.

Importing external modules:

The modules listed in the Requirements section have been imported.

File input and output:

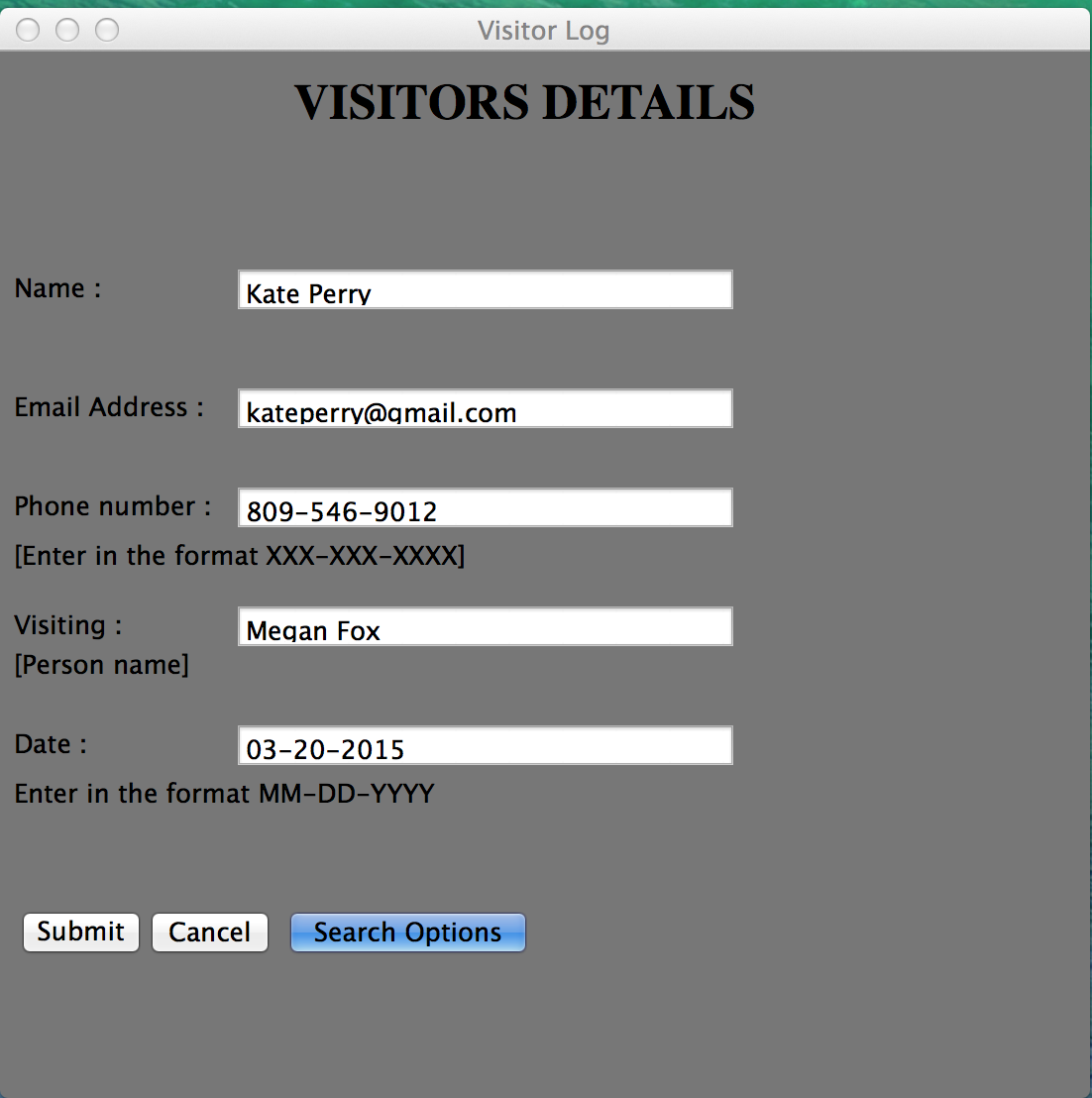
Data has been written into excel files using xlwt module.

Regular expression:

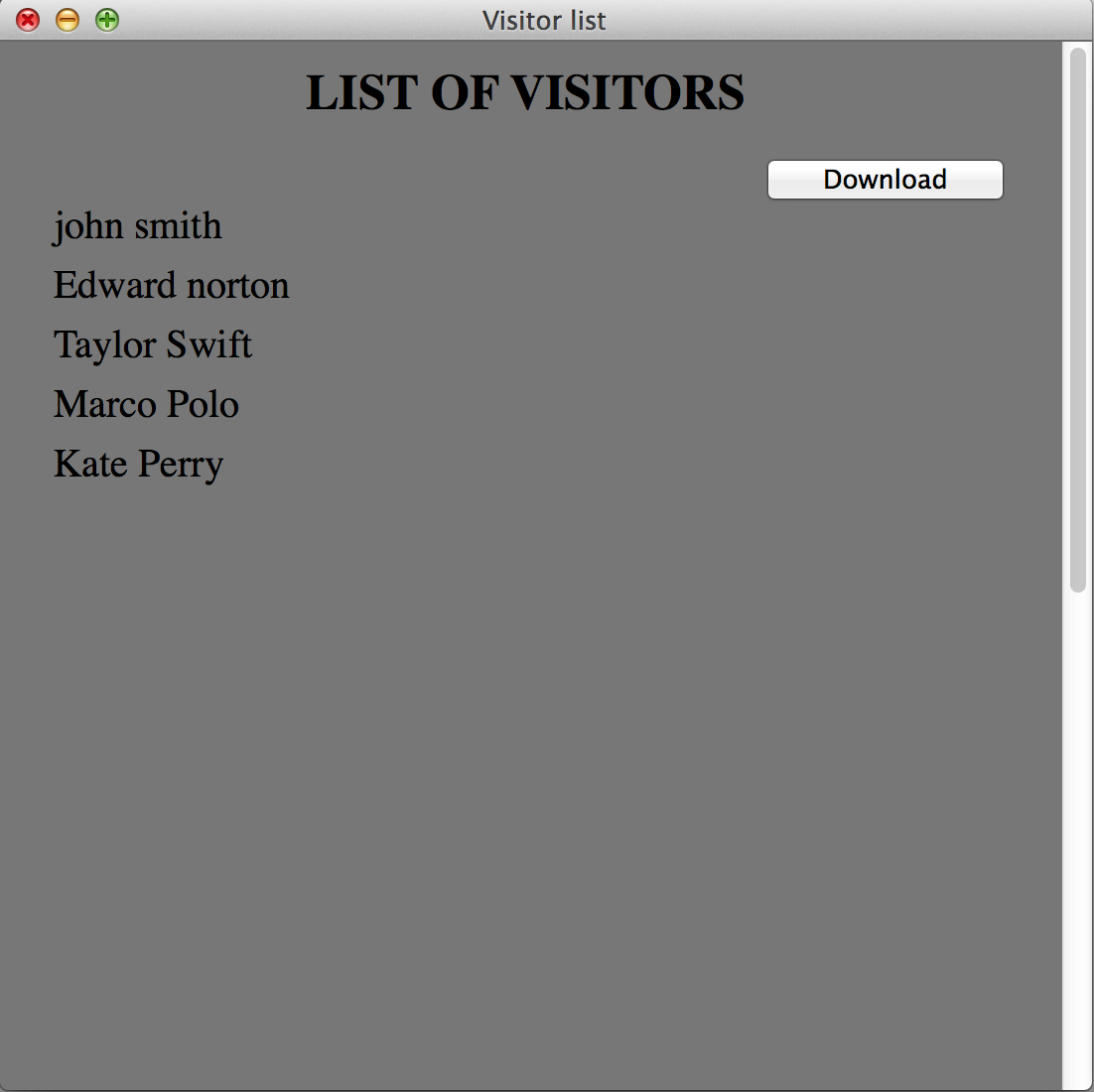
Input data validation has been done through regular expression

**Screenshots of the output:**

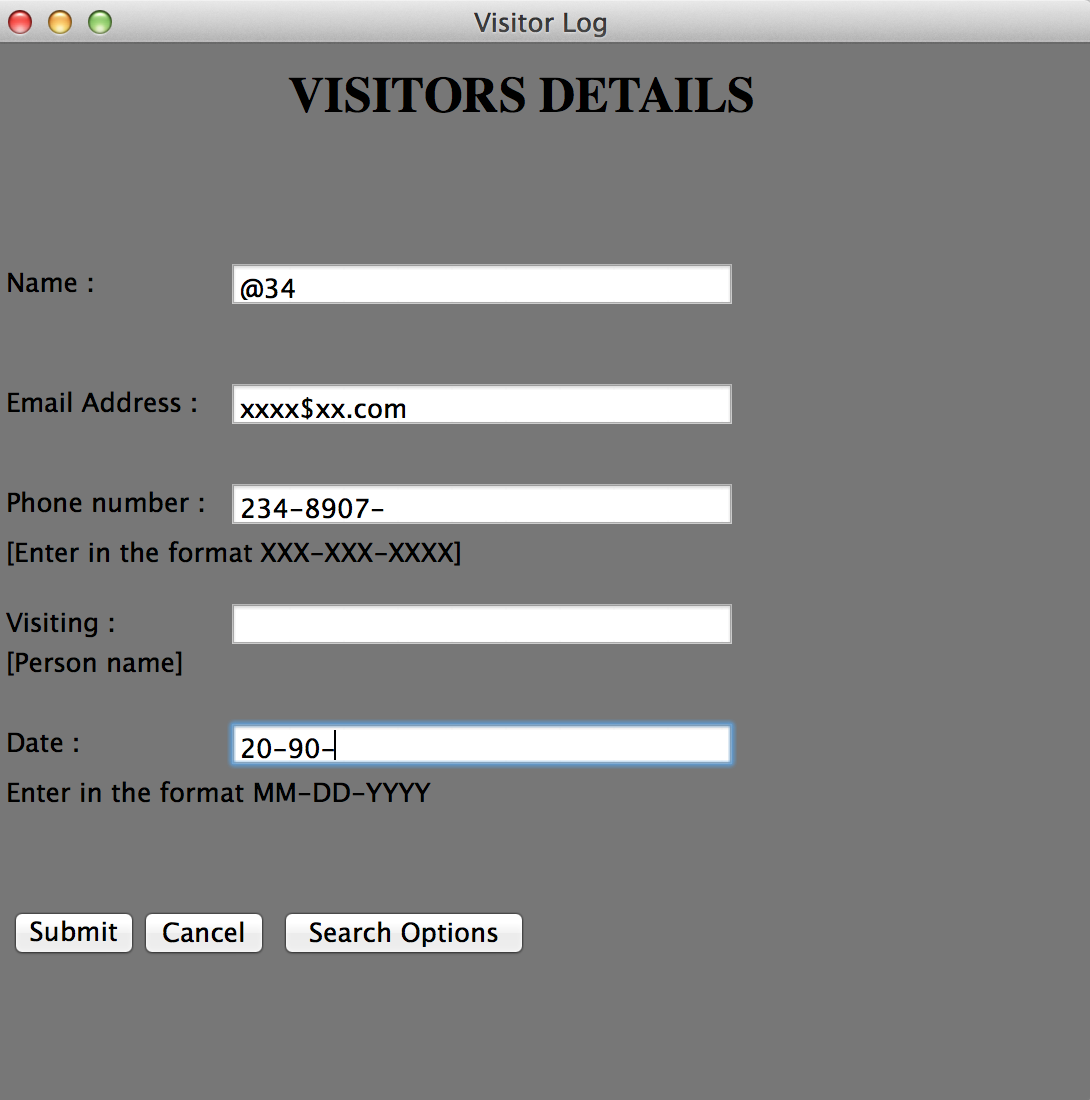
Users enters valid input and presses ‘Submit’ button

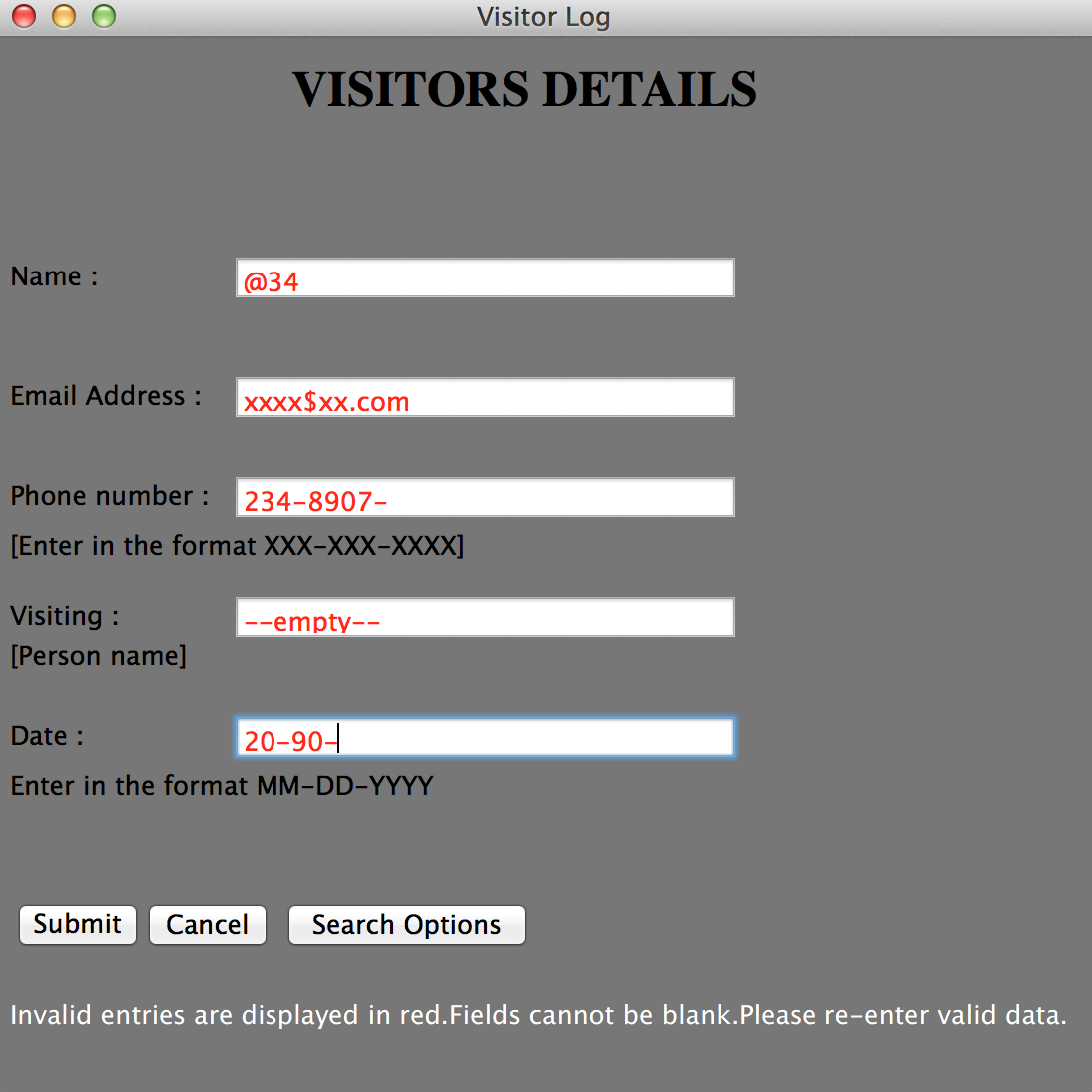


The entered data is stored in the database

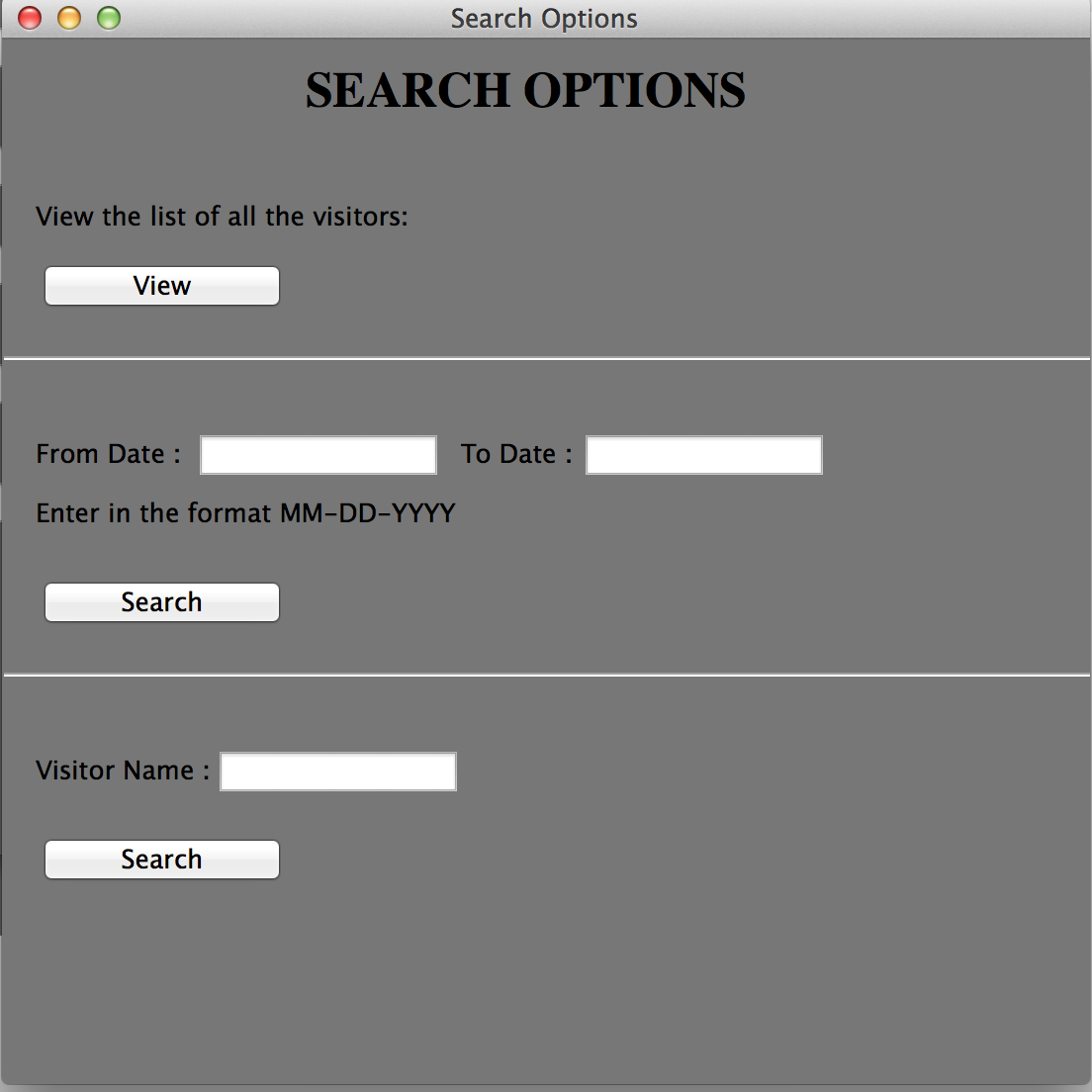
.

User enters invalid data and presses submit:

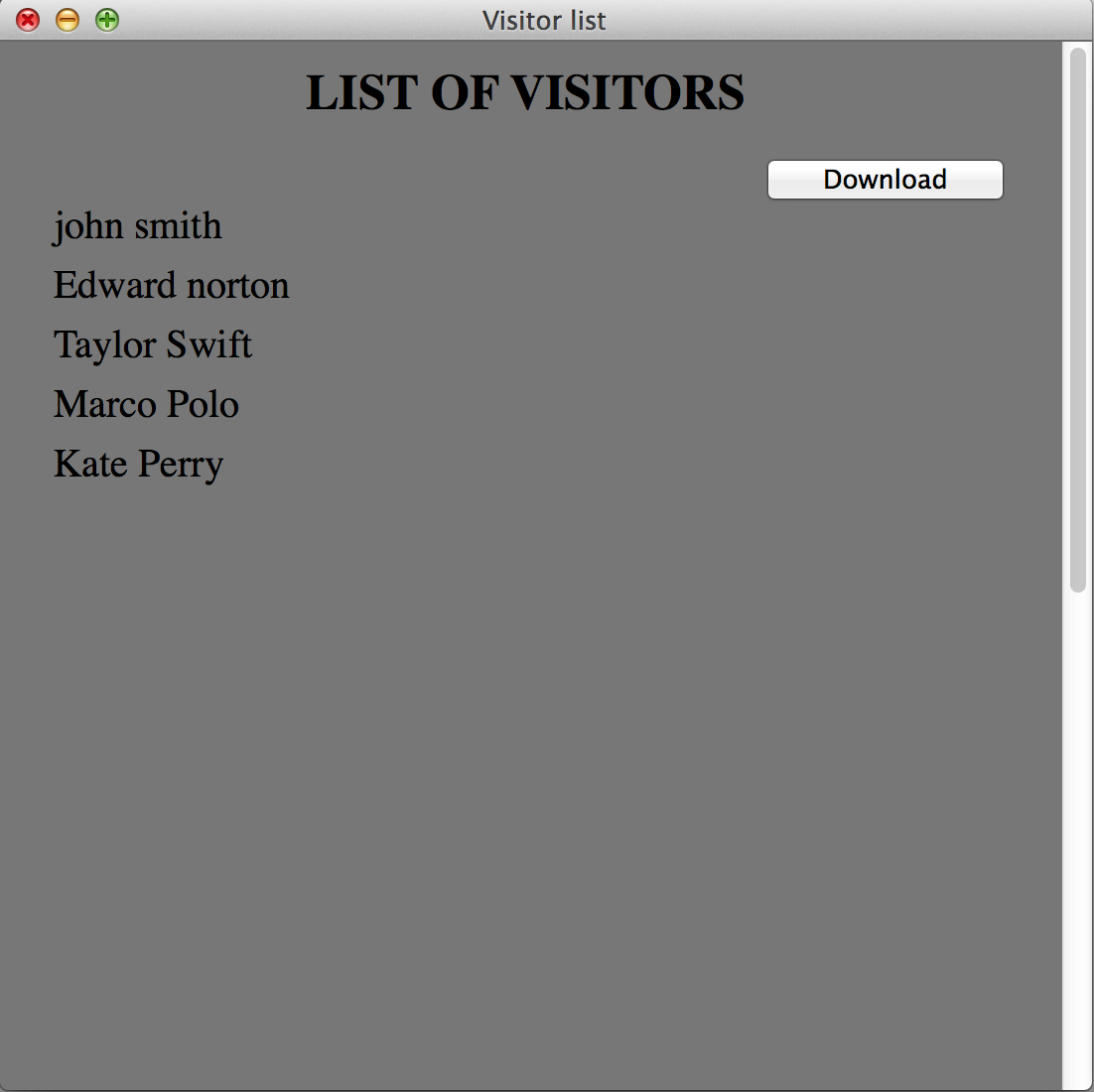


****

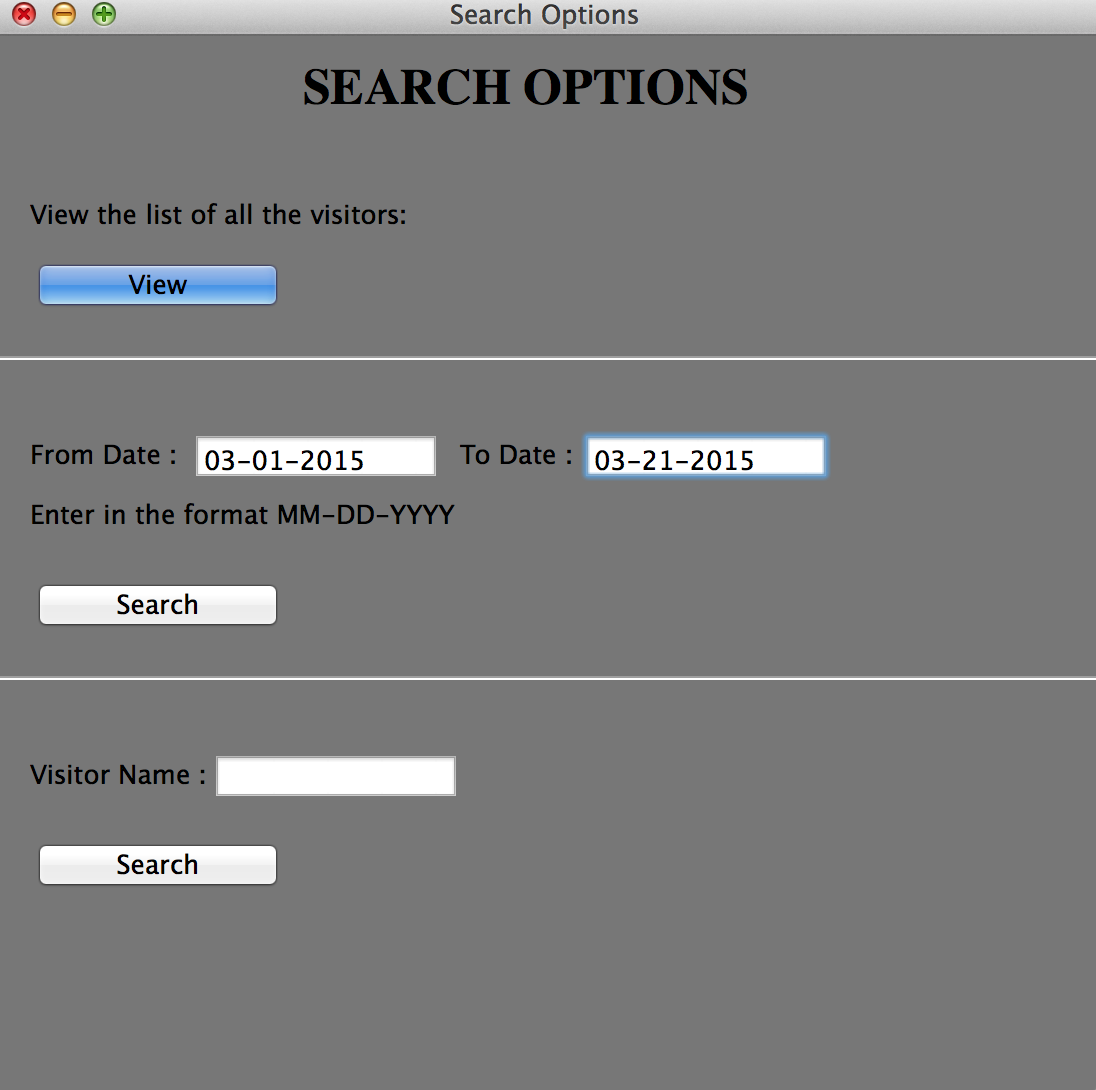
**Search options:**

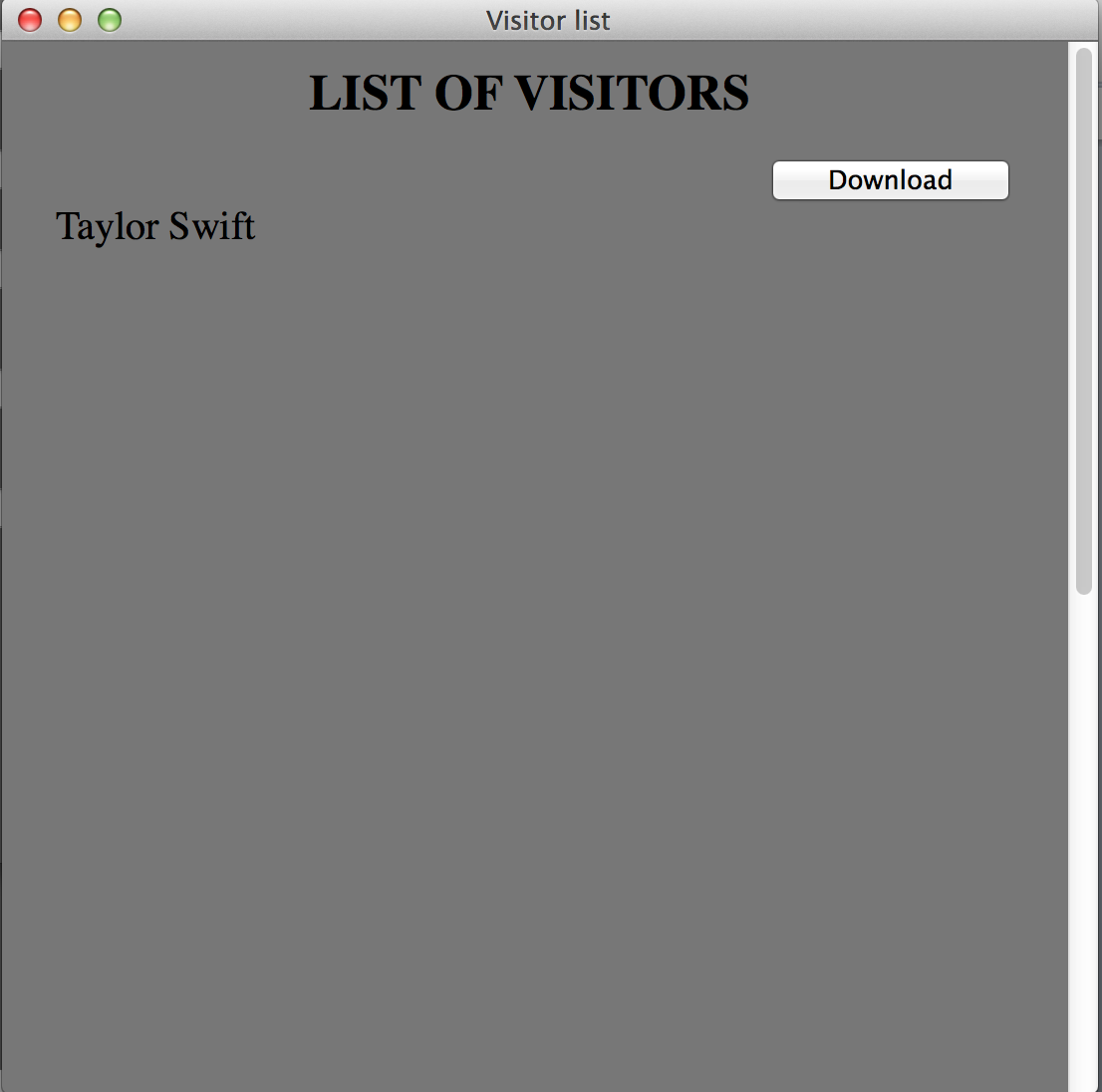
****

User presses View to view the entire visitor list

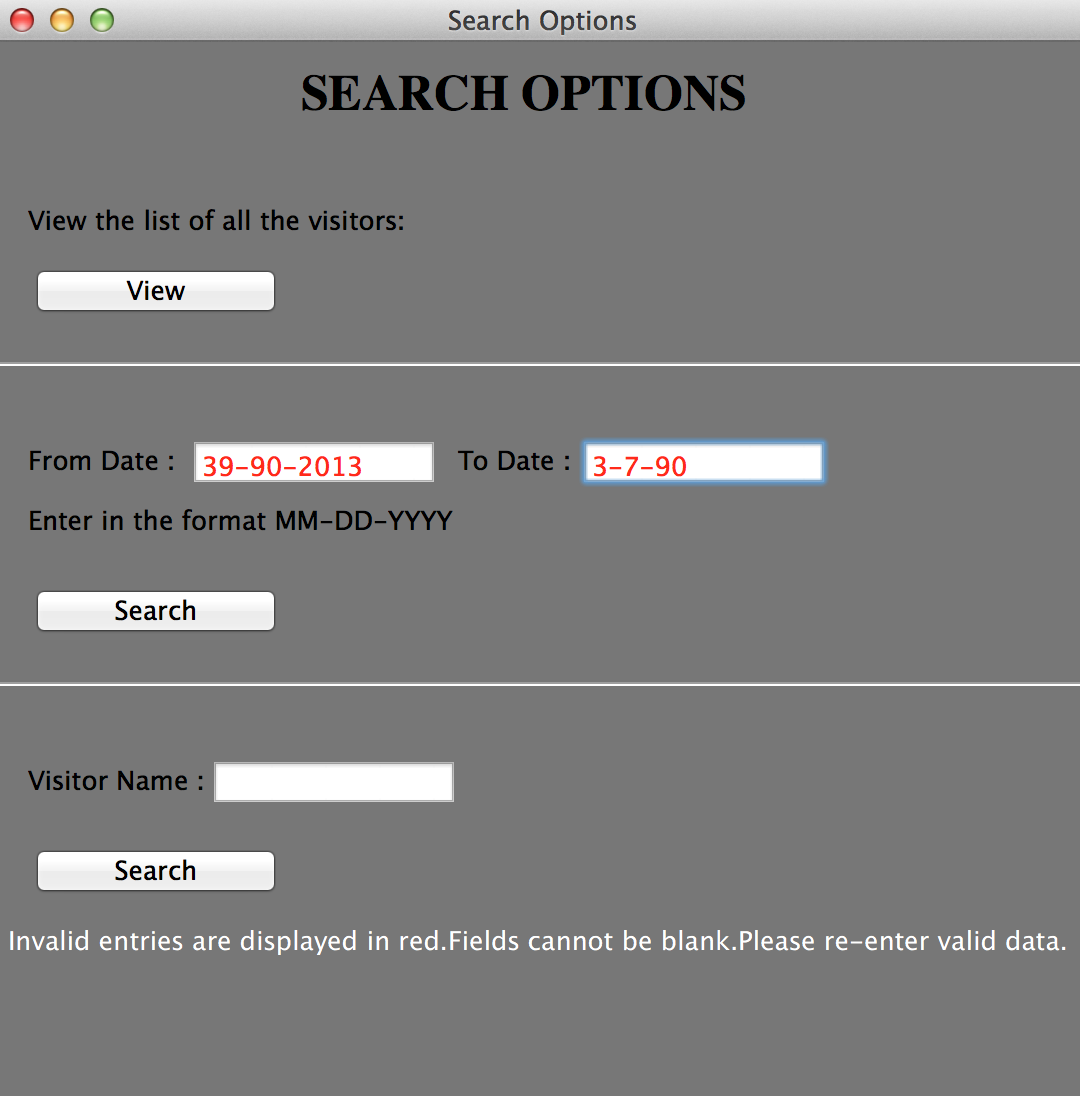
****

User filters the visitor list based on dates

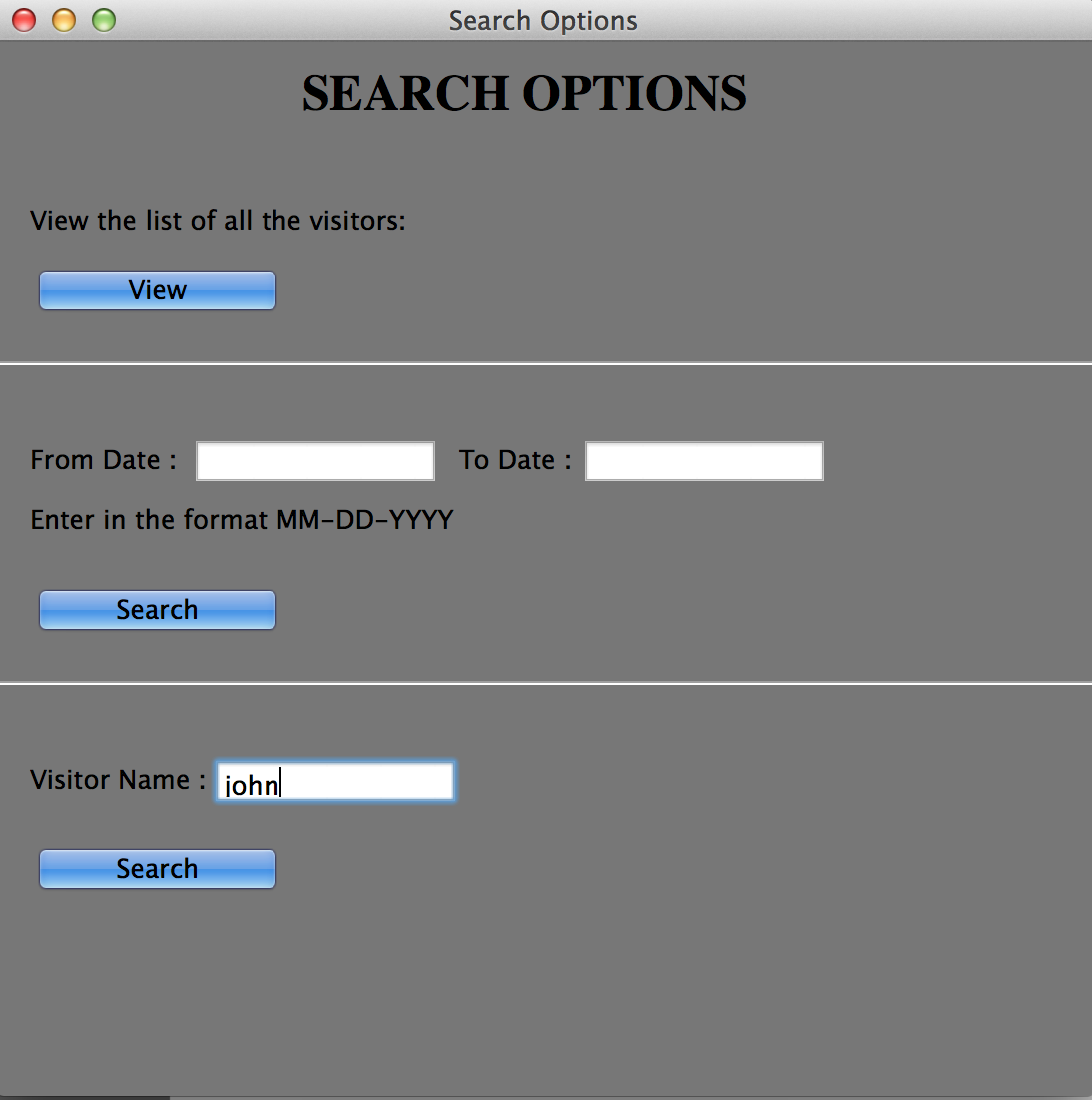
****

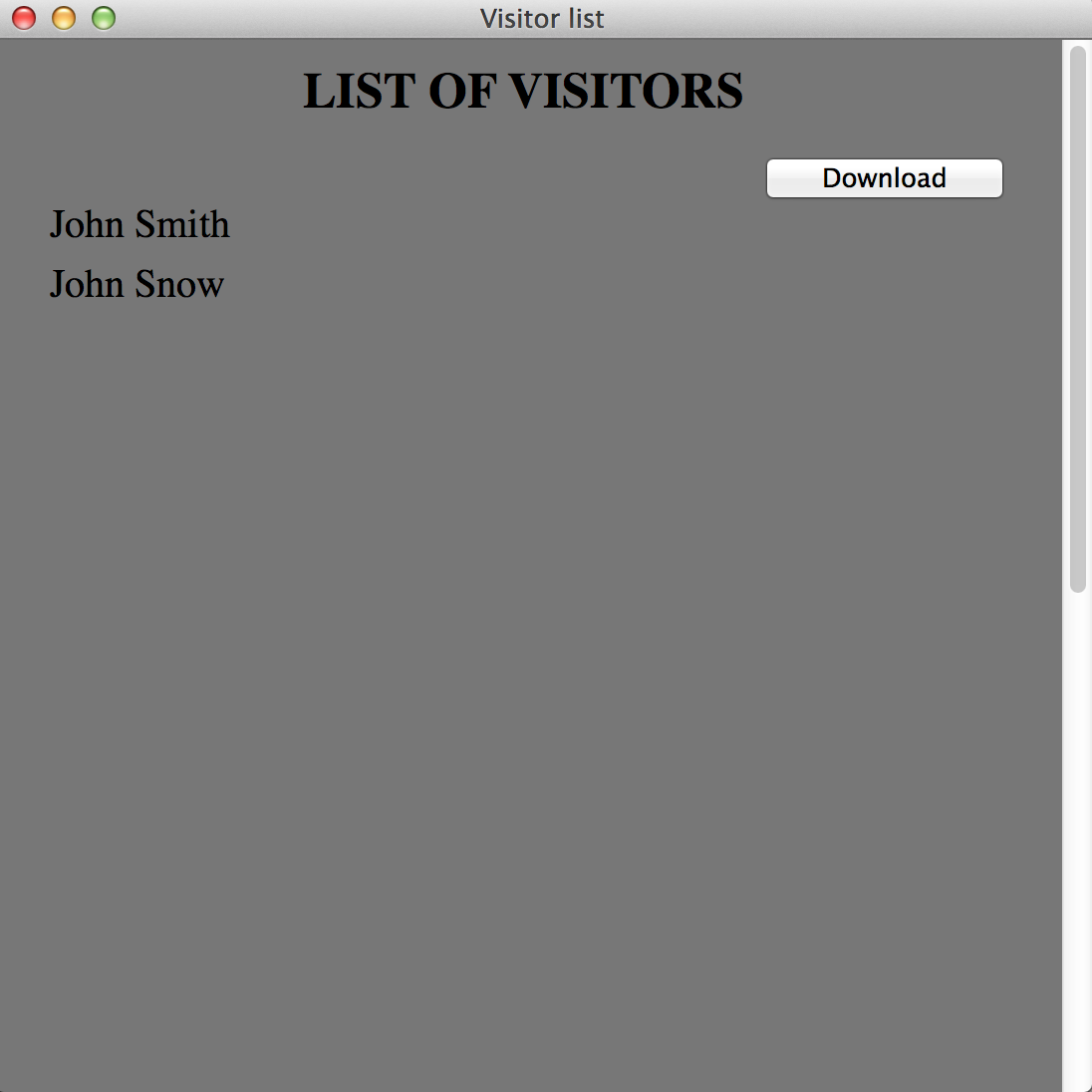
****

User enters invalid data in the from and to date fields – Highlights the error in red and displays error messages.



User filters the visitor’s list based on name

****

****

**Conclusion:**

A visitor log has been created which allows the user to enter data into the database with proper input validations. Provided the user with few search options - filtering based on name and dates. Users can also use the feature to download the data into an excel sheet.