**Assignment 6**

**Information Structures with Python**

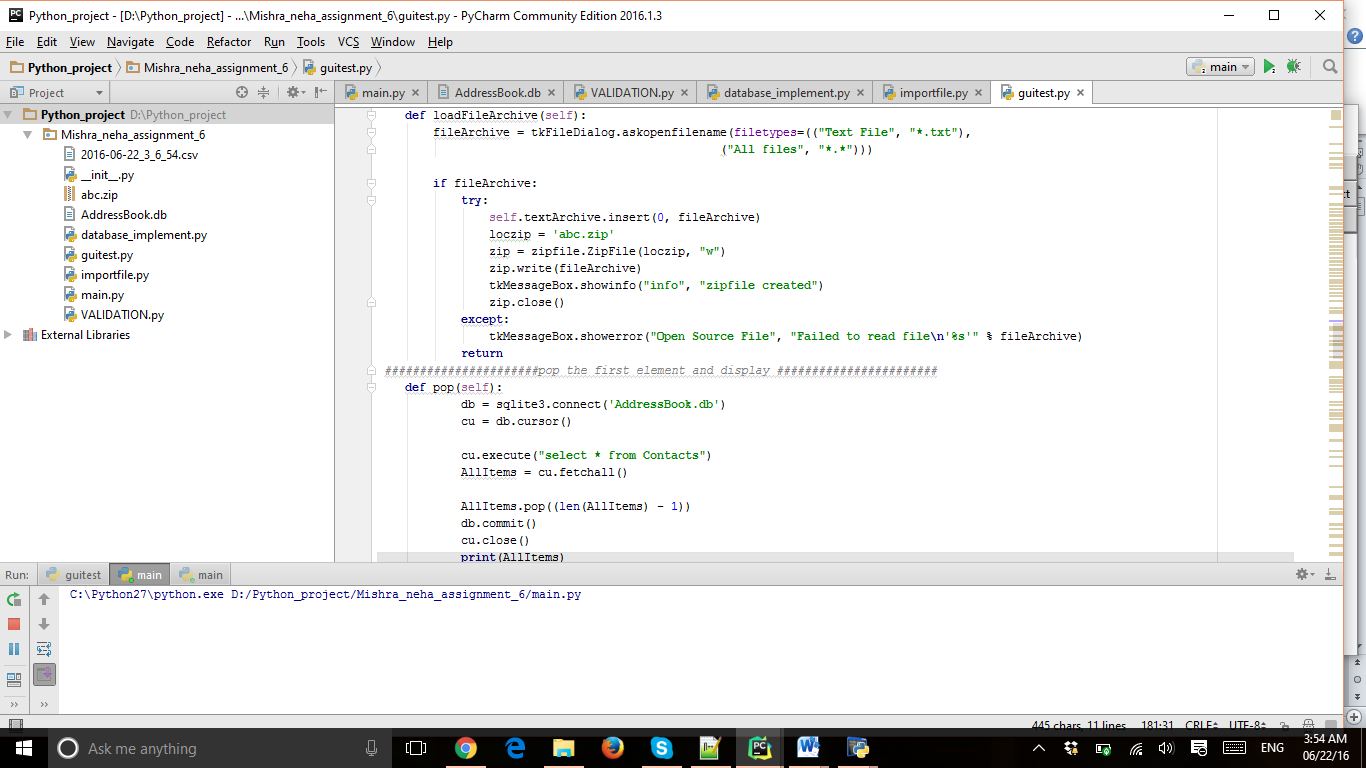
This project is about basic contactbook in which I have added features of creating zipfile , search google or open multiple sites by click of a button .

**Project Description :**

In this project , first I have created menu bar in which there are four options new ,open ,export ,exit

**Part 2 :**

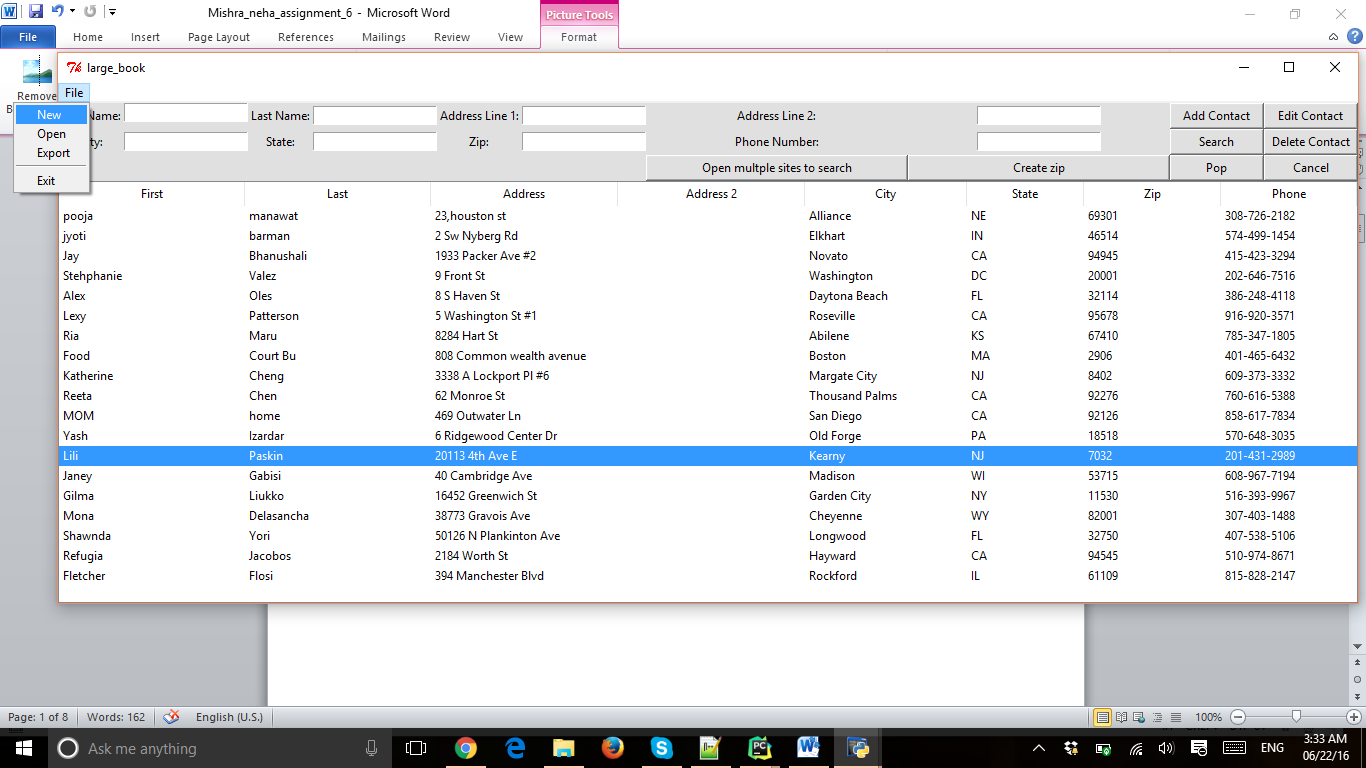
**Directory structure**



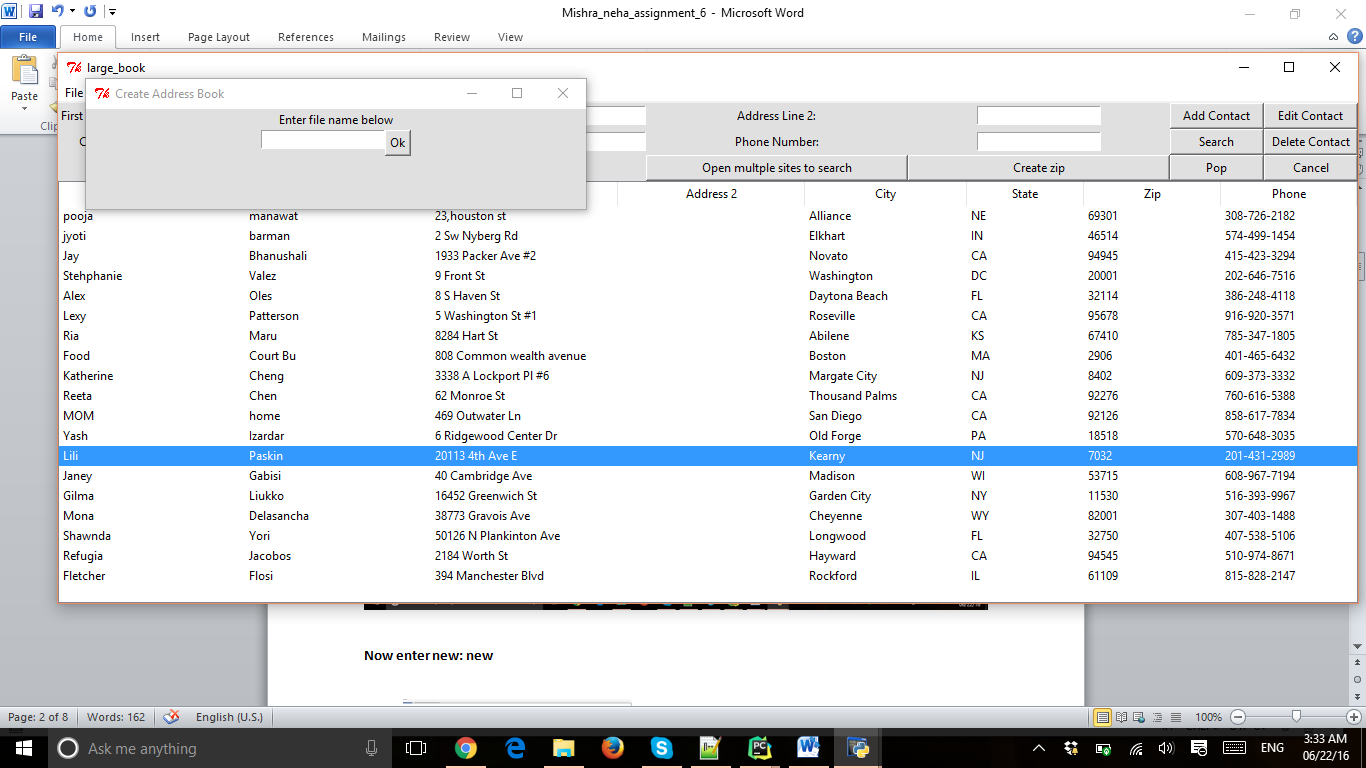
**Part 3:**

Screen shots of the functions

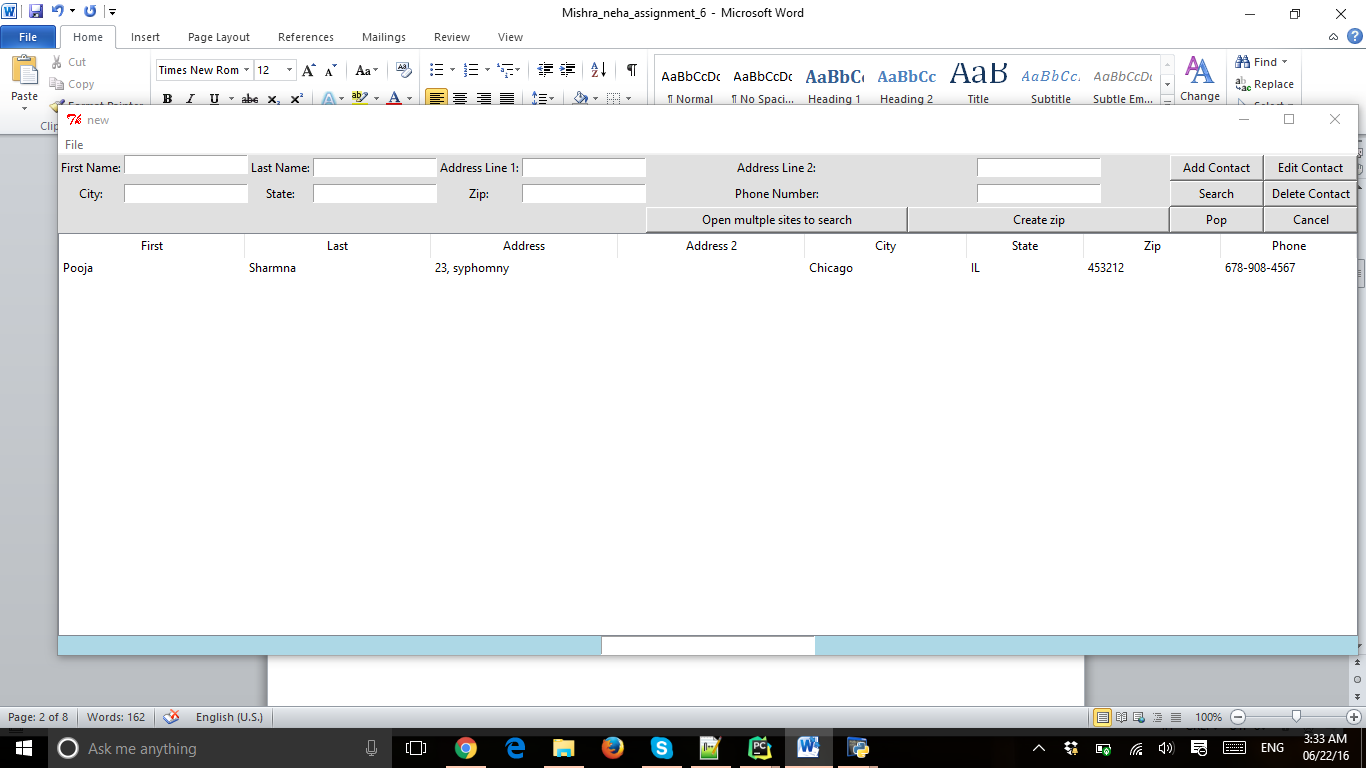
**First start with menu bar**



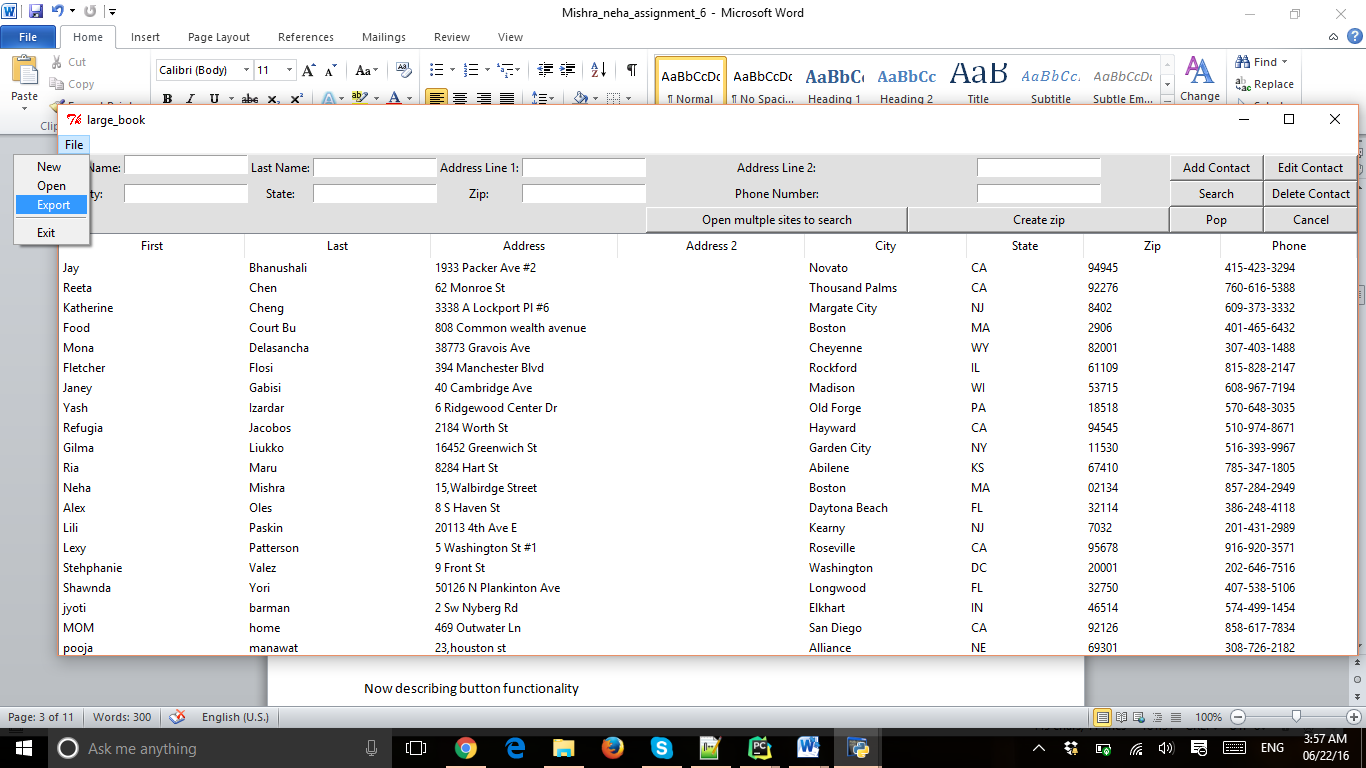
**Now new click :Enter addressbook name you want to create**



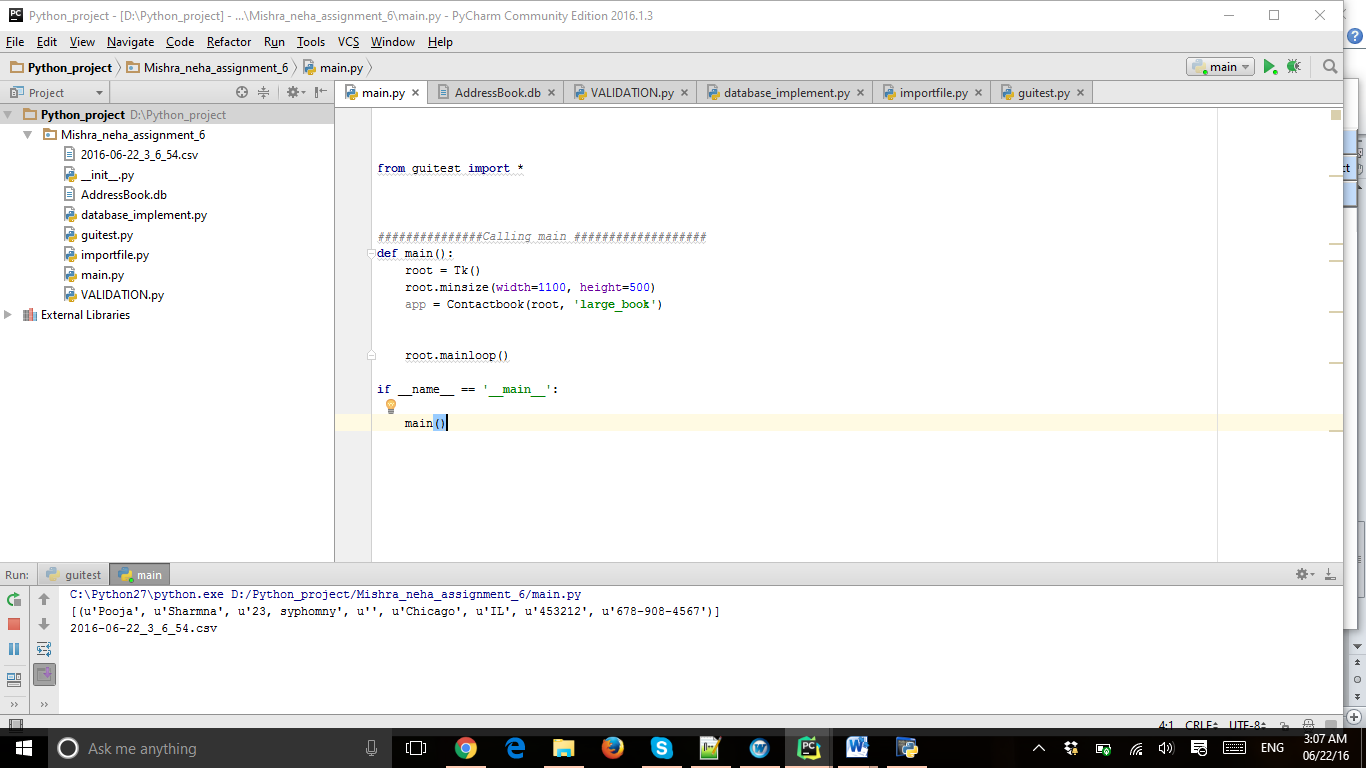
**Open file : new**



**Export Function:**

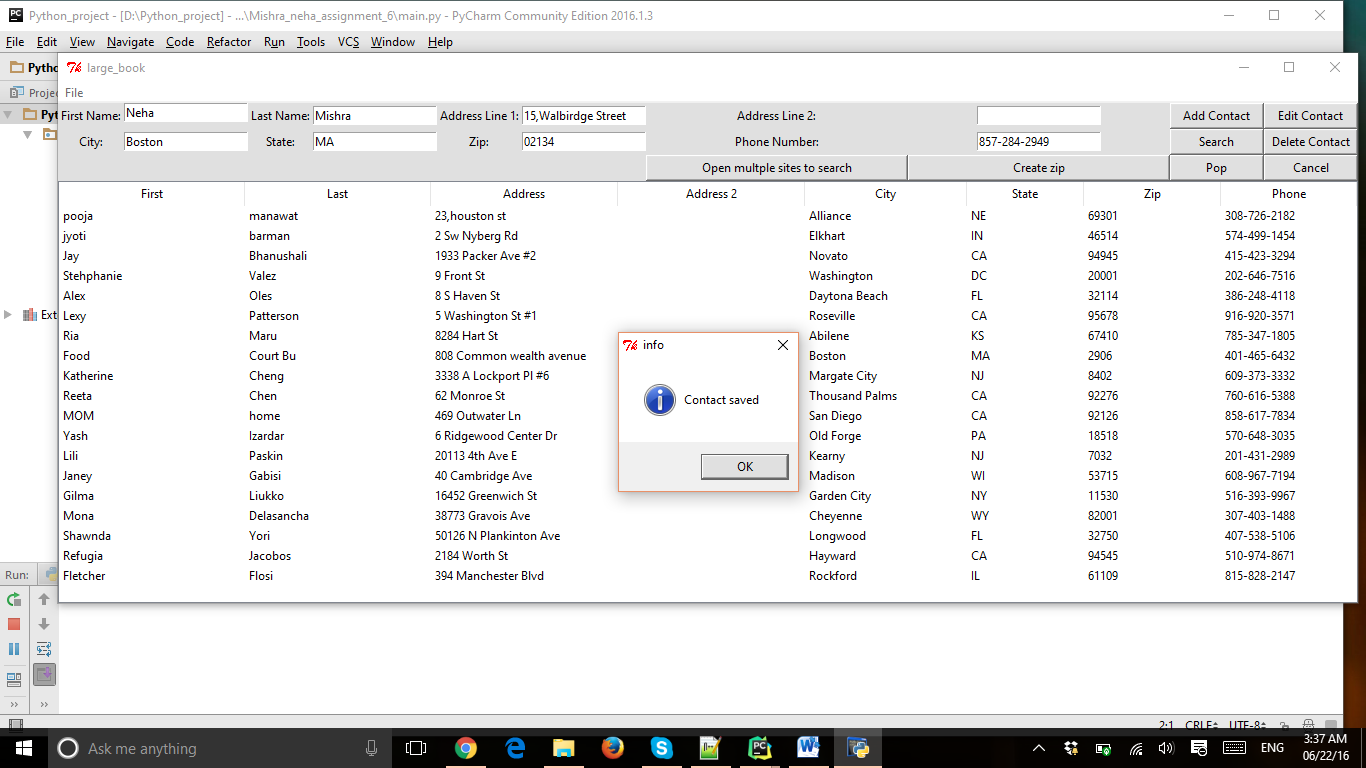


**Now the export function will create file name (current time).csv**



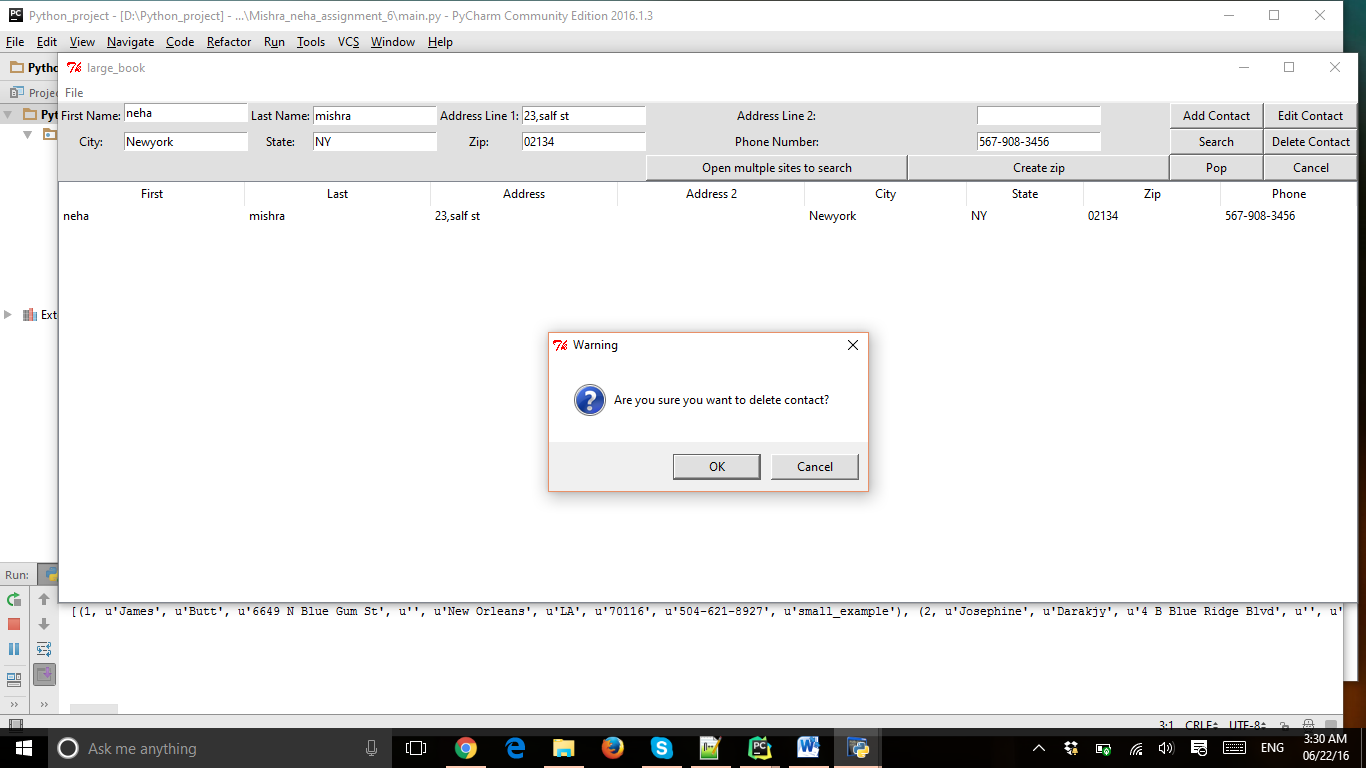
**Now describing button functionality**

**Add contact :**

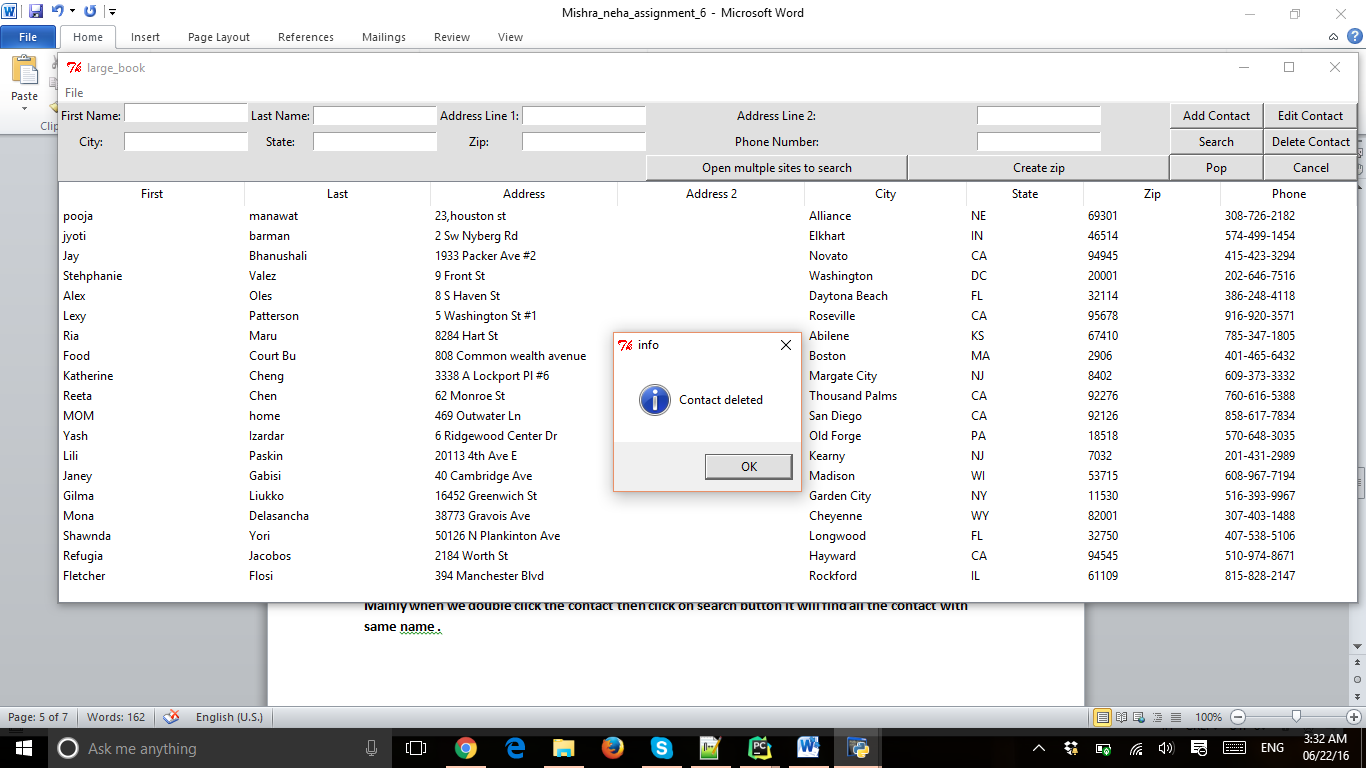


**Delete contact:**

For deleting delete contact **double -click** on the contact then it will come frame then you can delete it



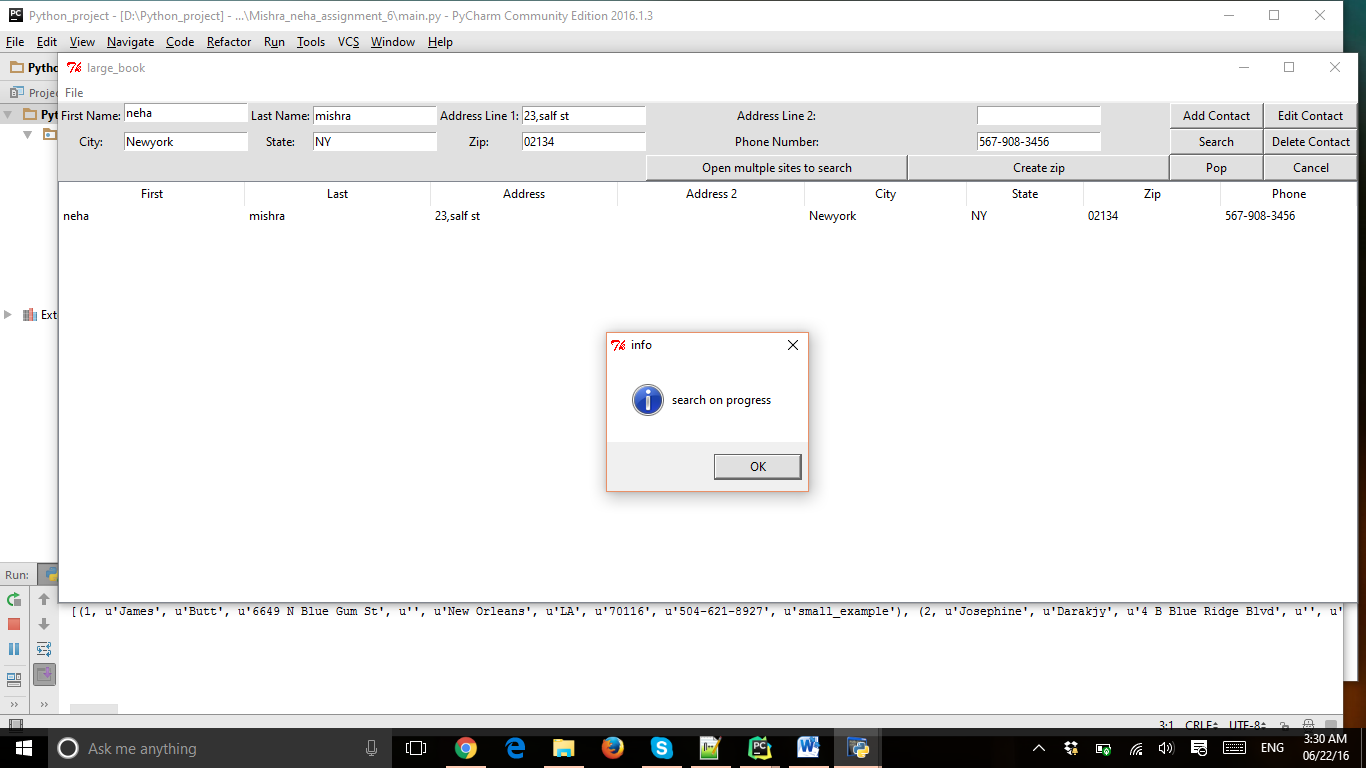
**After deletion :**



**For edit also Same : double click the contact and then click edit button**

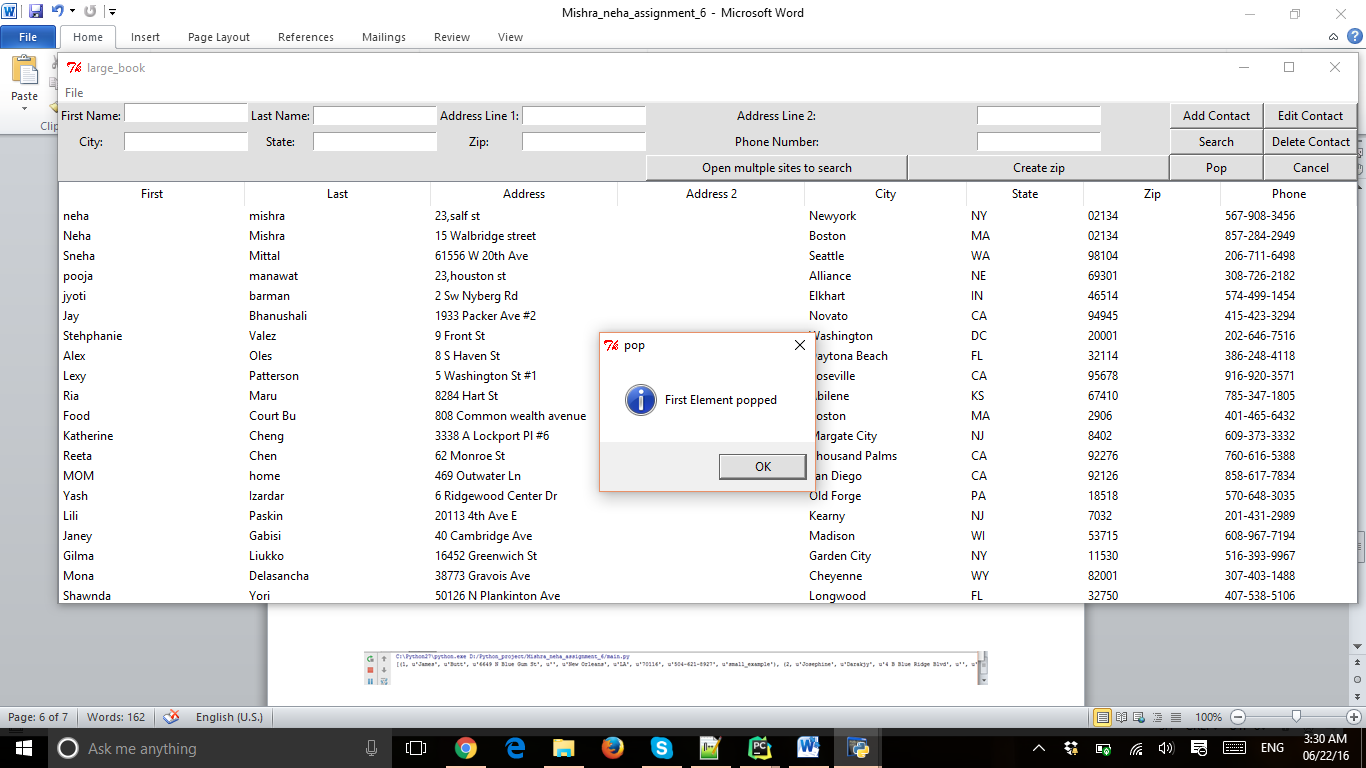
**Search button :**

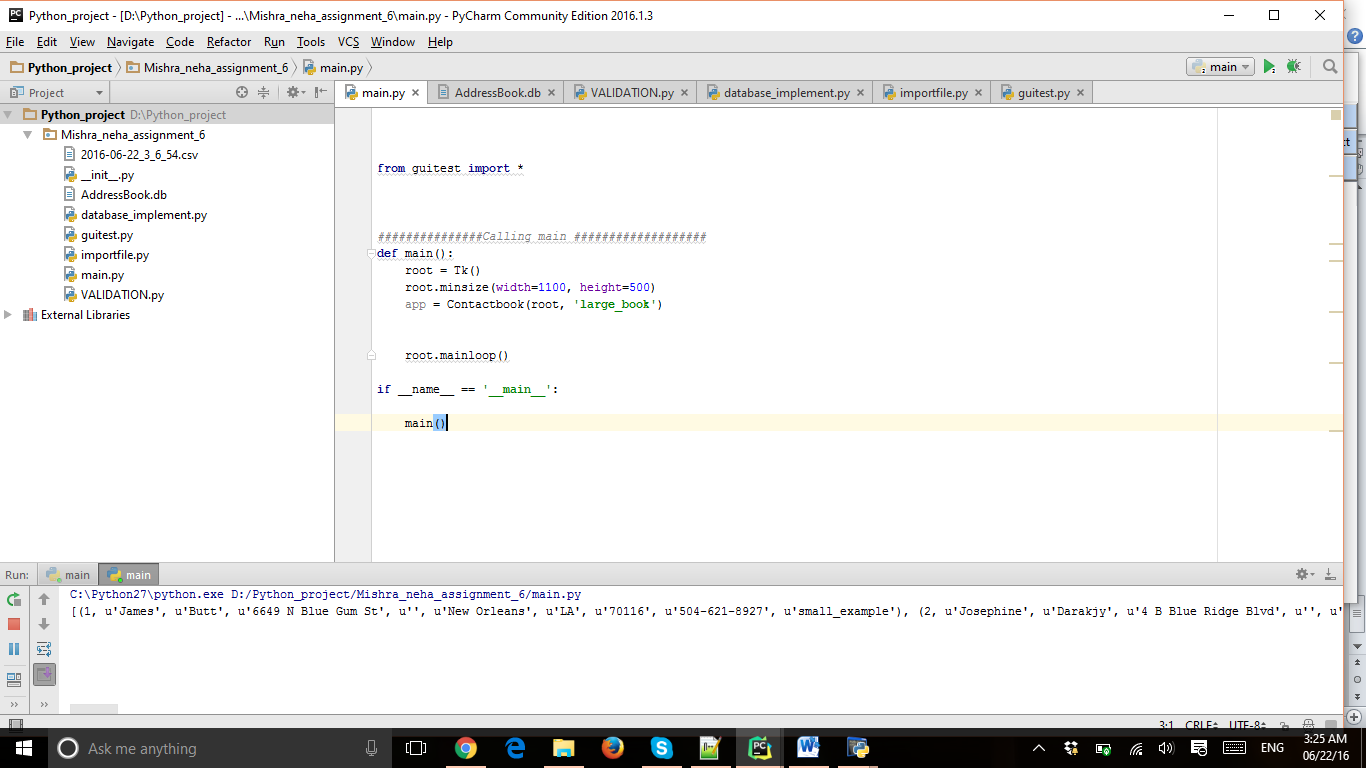
**Mainly when we double click the contact then click on search button it will find all the contact with same name .**



**POP button :**

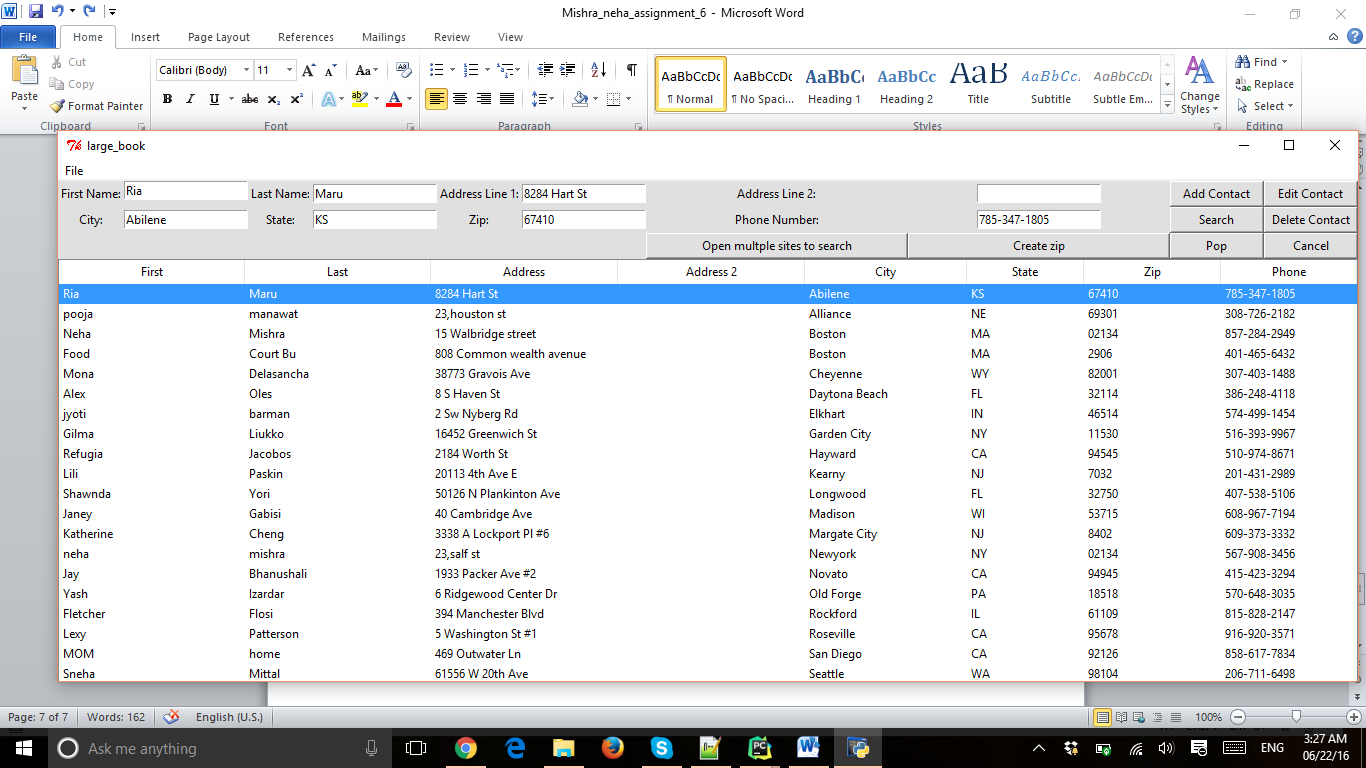
**POP button will pop the first element from address and display in the console**

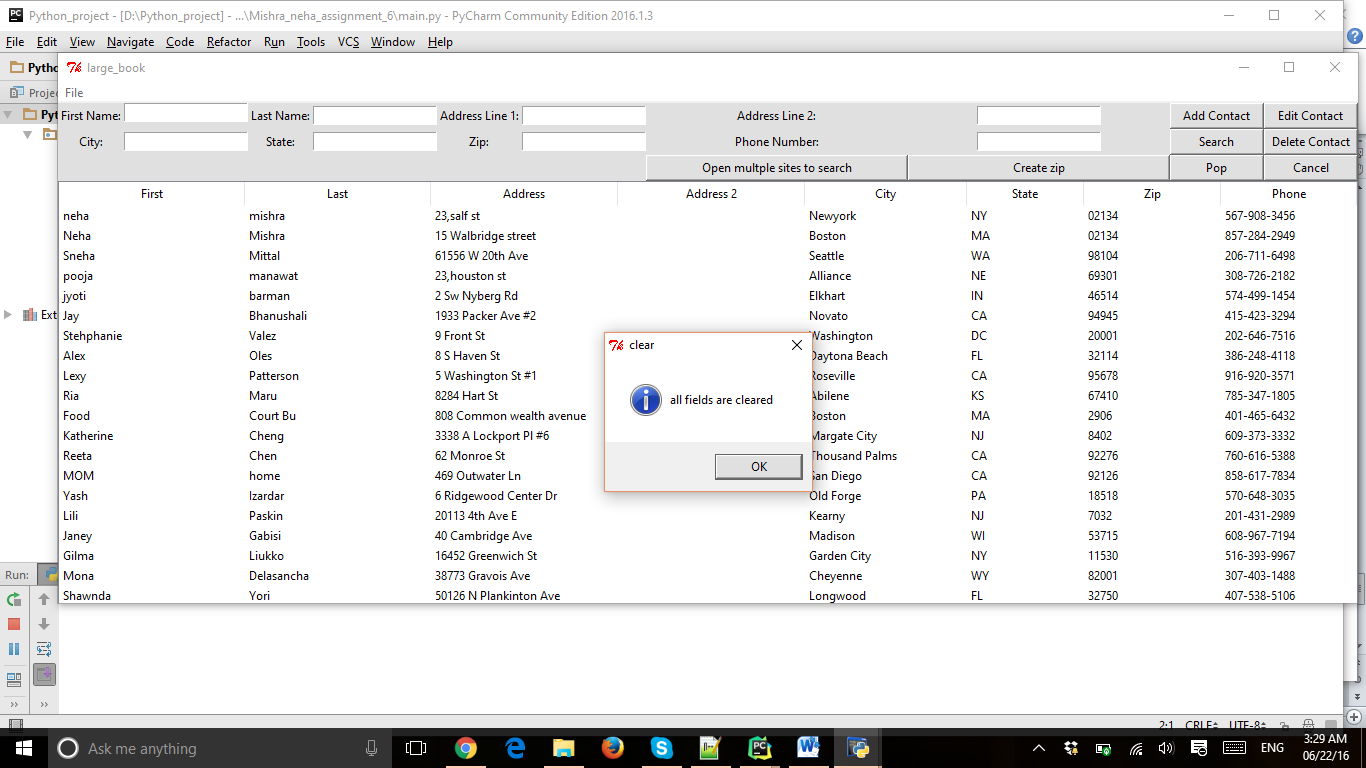




**Cancel :**

It clears the fields



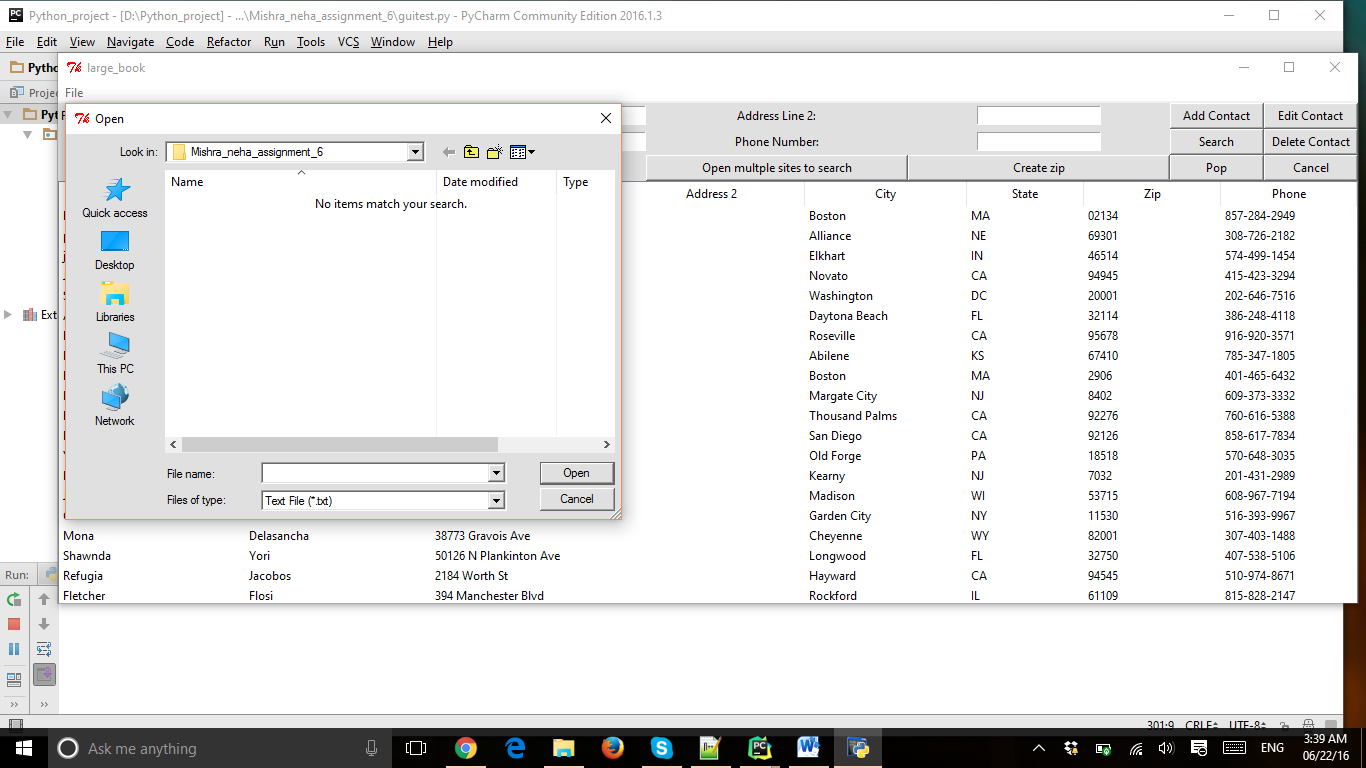


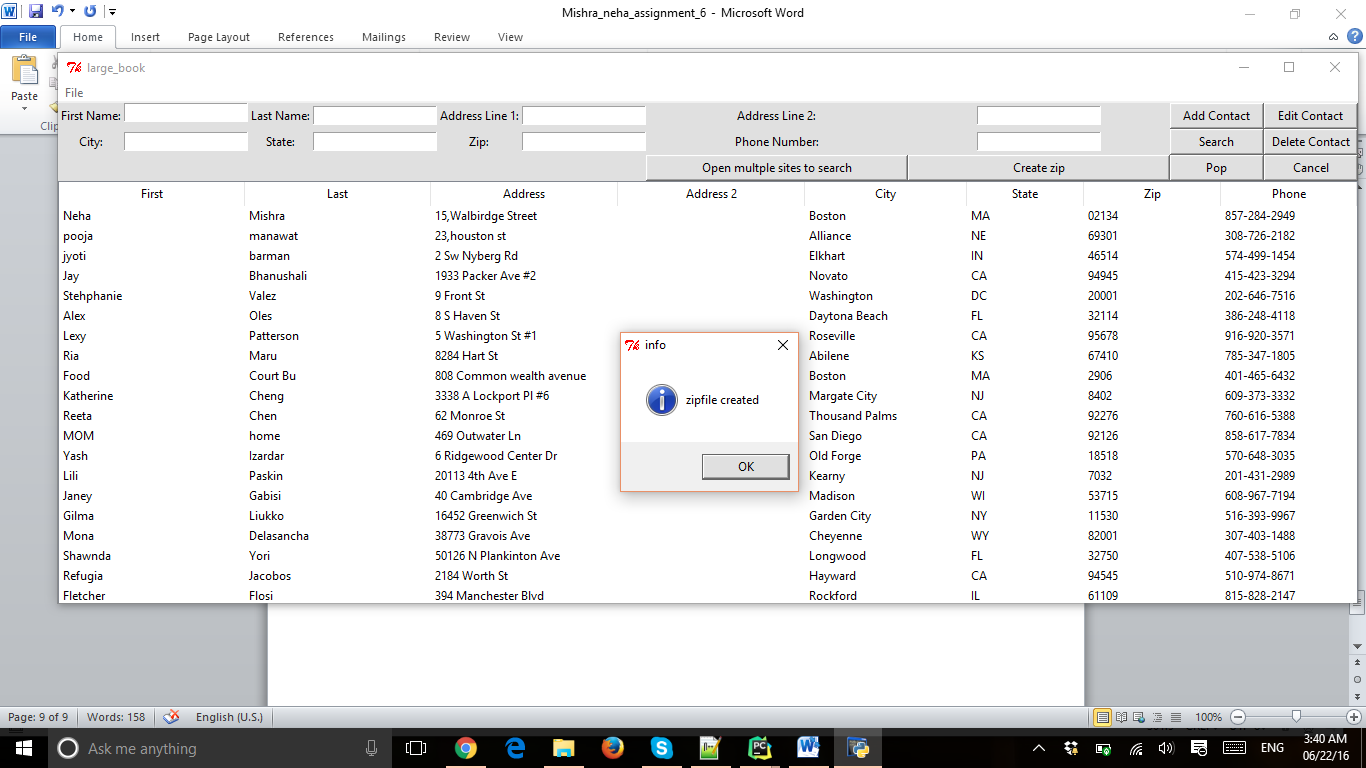
**Part 4 :**

**Python features included**

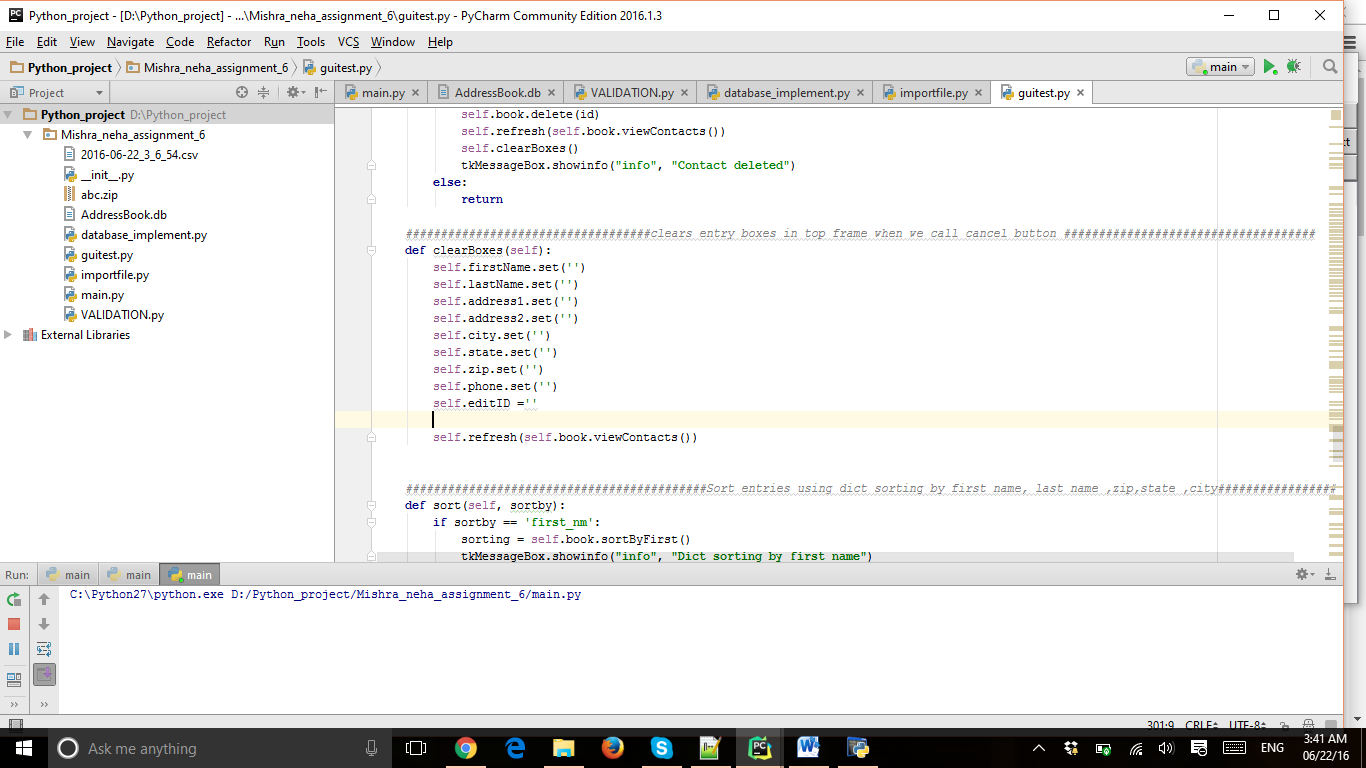
**Create Zip file:**

The function askopenfilename will ask to open a file which you want to create then



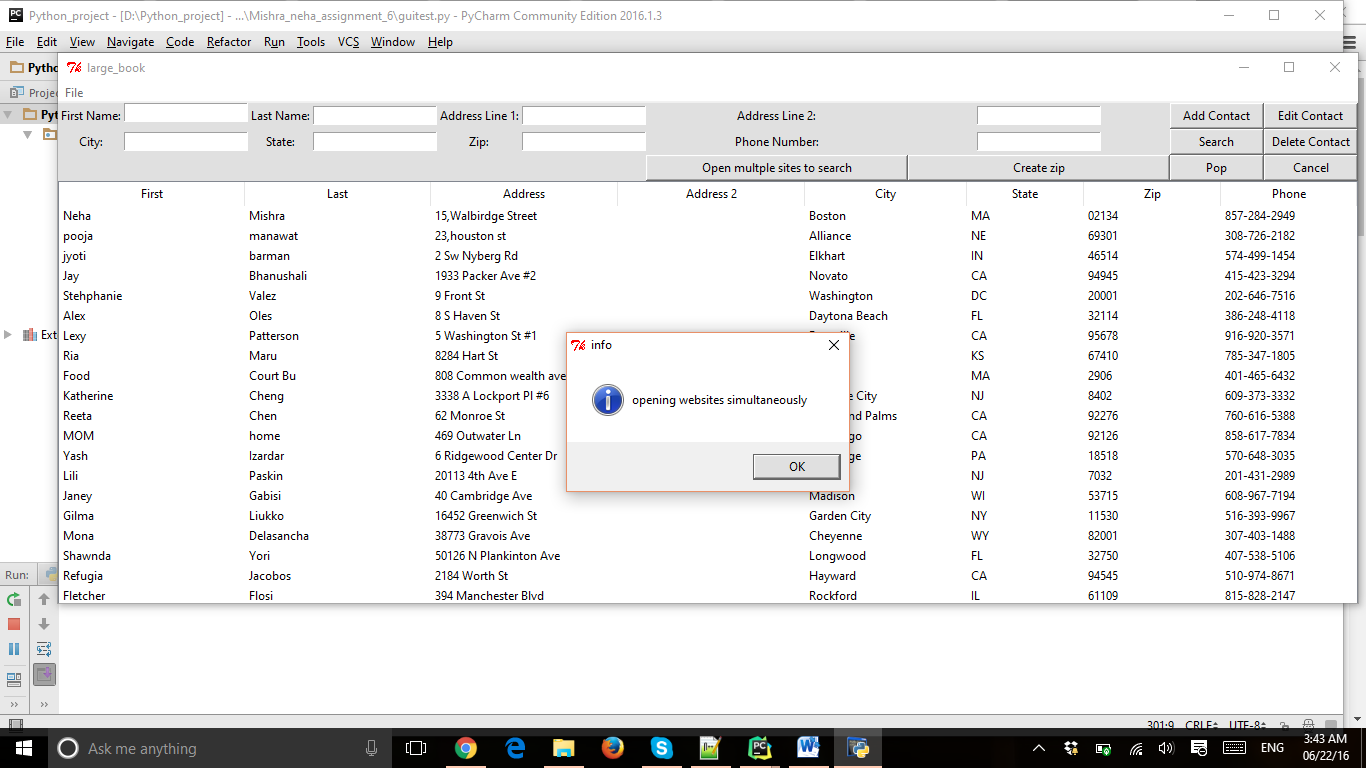
 The zip file function will create the zip file

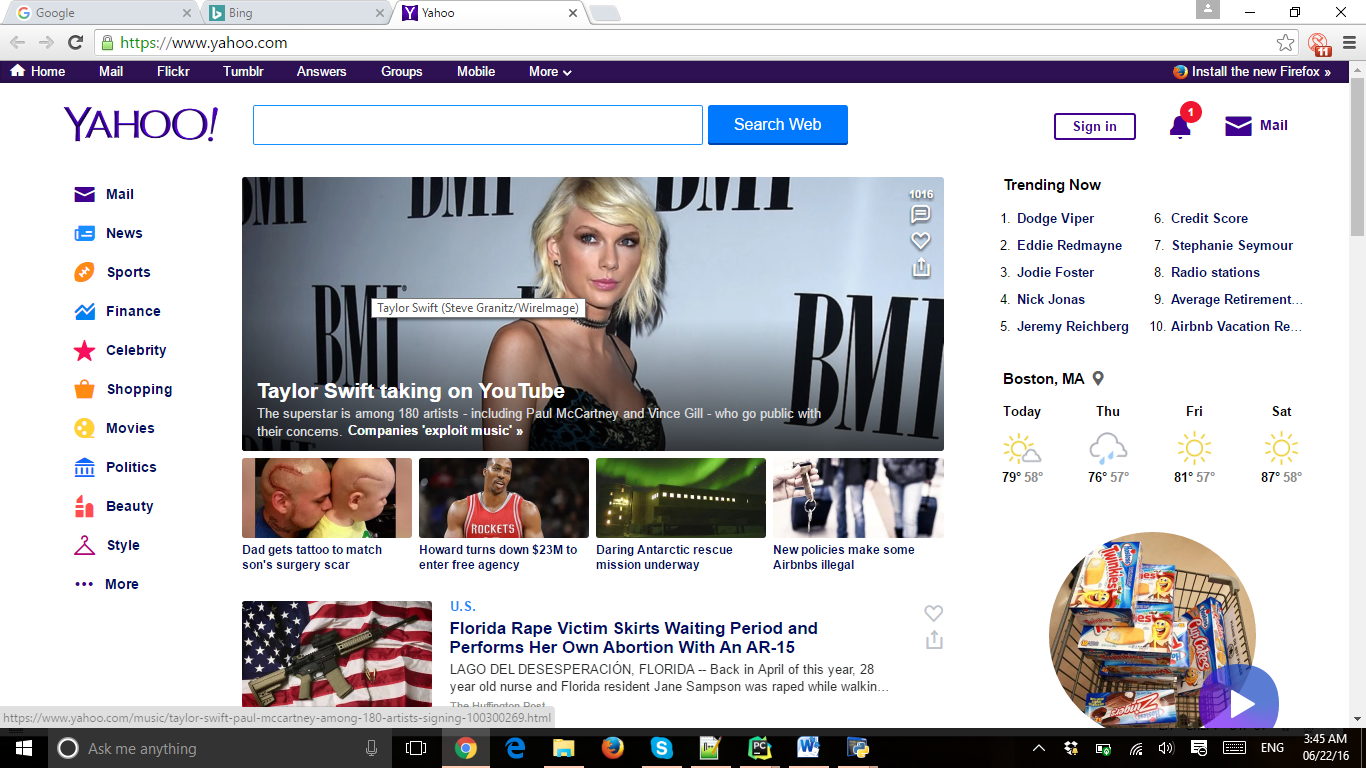
**The file is converted into abc.zip**



**Open Multiple sites :**

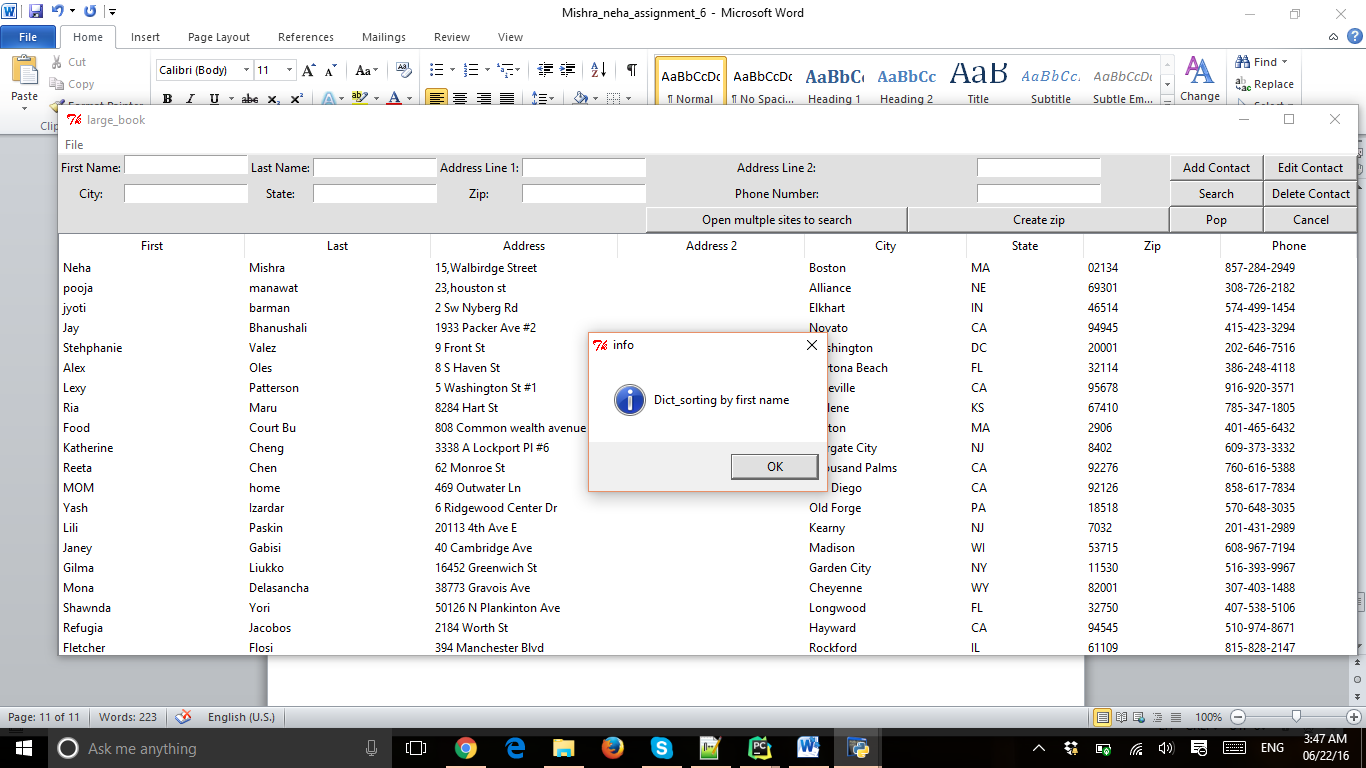
This button will open the websites given in the code which is google ,yahoo,bing



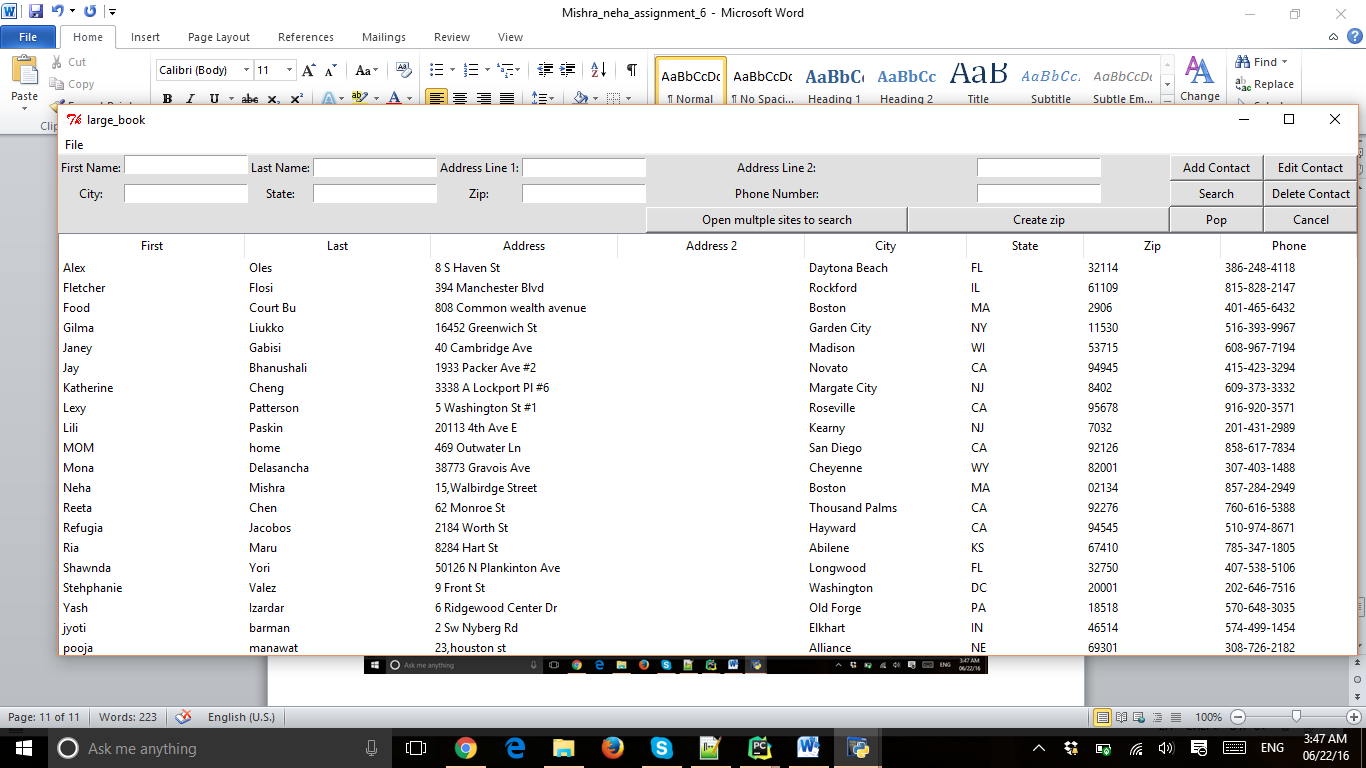


**sorting :**

When we click first , the addressbook will arrange according to first



**After sorting :**



**Similarly by last name, address,zipcode,state,city,phone**

**Part 5 :**

**Example code**

**I am explain the code of open multiple websites**

**def** startfunction(self):  
 **def** get\_url(a\_queue, a\_url):  
 a\_queue.put(urllib2.urlopen(a\_url).read())  
 webbrowser.open\_new(a\_url)  
  
 *# The url's we are interested (prepared for many of them)* the\_urls = [**"http://google.com"**, **"http://yahoo.com"**,**"http://www.bing.com"**]  
*# The content of the URL's in the\_urls are on the monitor*

the\_queue = Queue.Queue()  
 tkMessageBox.showinfo(**"info"**, **"opening websites simultaneously "**)  
 **for** url **in** the\_urls:  
 Thread(target=get\_url, args=(the\_queue, url)).start()

**Exporting the currenttime file :**

**import** csv  
**import** datetime  
  
  
  
*################Creating csv file of addressbook at current time ################################***def** currentTimeString():  
 time = datetime.datetime.now()  
 currentTimeStr = str(time.date()) + **"\_"** + str(time.hour) + **"\_"** + str(time.minute) + **"\_"** + str(time.second)  
 **return** currentTimeStr  
  
**def** createBasicCSV(dataList = (), header=(**'first\_nm'**, **'last\_nm'**, **'address1'**, **'address2'**, **'city'**, **'state'**, **'zip'**, **'phone'**)):  
 fileName = currentTimeString() + **'.csv'  
 print** fileName  
  
 **try**:  
 **with** open(fileName, **'w'**) **as** myfile:  
 wr = csv.writer(myfile, quoting=csv.QUOTE\_ALL)  
 wr.writerow(header)  
 **for** row **in** dataList:  
 wr.writerow(row)  
 myfile.close()  
 **except**:  
 **print "error in making csv"  
 return** None  
 **return** fileName

**Sorting code :**

**def** sort(self, sortby):  
 **if** sortby == **'first\_nm'**:  
 sorting = self.book.sortByFirst()  
 tkMessageBox.showinfo(**"info"**, **"Dict\_sorting by first name"**)  
 **elif** sortby == **'last\_nm'**:  
 sorting = self.book.sortByLast()  
 tkMessageBox.showinfo(**"info"**, **"Dict\_sorting by last name"**)  
  
 **elif** sortby == **'address1'** :  
 sorting = self.book.sortByAddress()  
 tkMessageBox.showinfo(**"info"**, **"Dict\_sorting by Address"**)  
  
 **elif** sortby == **'zip'**:  
 sorting = self.book.sortByZip()  
 tkMessageBox.showinfo(**"info"**, **"Dict\_sorting by zip"**)  
 **elif** sortby == **'state'**:  
 sorting = self.book.sortByState()  
 tkMessageBox.showinfo(**"info"**, **"Dict\_sorting by state"**)  
 **elif** sortby == **'city'**:  
 sorting = self.book.sortByCity()  
 tkMessageBox.showinfo(**"info"**, **"Dict\_sorting by city"**)  
 **else**:  
 sorting = self.book.sortByPhone()  
 tkMessageBox.showinfo(**"info"**, **"Dict\_sorting by phone"**)  
  
 self.refresh(sorting)

*#########################sorting by first name ###########################################* **def** sortByFirst(self):  
  
 cur = self.connection.cursor()  
  
 cur.execute(**'SELECT \* FROM contacts WHERE bookName = ? ORDER BY first\_nm DESC'**, (self.bookName,))  
  
 **return** cur.fetchall()  
  
 *#########################sorting by last name ###########################################* **def** sortByLast(self):  
  
 cur = self.connection.cursor()  
  
 cur.execute(**'SELECT \* FROM contacts WHERE bookName = ? ORDER BY last\_nm DESC'**, (self.bookName,))  
  
 **return** cur.fetchall()  
  
  
  
 *#########################sorting by Address1 ###########################################* **def** sortByAddress(self):  
  
 cur = self.connection.cursor()  
  
 cur.execute(**'SELECT \* FROM contacts WHERE bookName = ? ORDER BY address1 DESC'**, (self.bookName,))  
  
 **return** cur.fetchall()  
  
 *#########################sorting by first name ###########################################* **def** sortByZip(self):  
  
 cur = self.connection.cursor()  
  
 cur.execute(**'SELECT \* FROM contacts WHERE bookName = ? ORDER BY zip DESC'**,(self.bookName,))  
  
 **return** cur.fetchall()  
  
 *#########################sorting by State ###########################################* **def** sortByState(self):  
  
 cur = self.connection.cursor()  
  
 cur.execute(**'SELECT \* FROM contacts WHERE bookName = ? ORDER BY state DESC'**, (self.bookName,))  
  
 **return** cur.fetchall()  
  
 *#########################sorting by Phone ####################################################* **def** sortByPhone(self):  
  
 cur = self.connection.cursor()  
  
 cur.execute(**'SELECT \* FROM contacts WHERE bookName = ? ORDER BY phone DESC'**, (self.bookName,))  
  
 **return** cur.fetchall()  
  
 *#########################sorting by City ###################################################* **def** sortByCity(self):  
  
 cur = self.connection.cursor()  
  
 cur.execute(**'SELECT \* FROM contacts WHERE bookName = ? ORDER BY city DESC'**,(self.bookName,))  
  
 **return** cur.fetchall()

**All the code and database is attached in the zipfile**