

# CIT244 Python II

## Program 1: Classes

We're going to write a program that manages the email address of our contacts. Your program must contain a class named *contact* that has 3 attributes (instance variables): last name, first name, and email. Your class should also have a method to return the full name and email address for printing when requested. You may add other methods to the class if you like.

Please start early in the week that this program is due; this gives you plenty of time to ponder how it should work. Moreover, there will be more to ponder as the course progresses. I am glad to help troubleshoot programs, but I go to bed reasonably early and am unlikely to stay up until midnight of the due date.

Each contact should use a single instance of your contact class. You should use either a list or a dictionary to act as a container of the contact instances.

Also, the program should have a menu that allows the user to interact with your collection of contacts: displaying contacts or adding new contacts as long as the user wishes, then saying goodbye when finished. The menu should offer these options. Note we are not saving anything to disk here.

Program Options.

- 1.) Display all contacts
- 2.) Create new contact
- 3.) Exit

```
option = input("Enter 1, 2, or 3: ")
```

There are several examples of this kind of menu program scattered throughout the lecture notes for the first several weeks of the semester. The example most relevant to this assignment comes at the end of the notes on Classes.

For full credit.

- use a python class to represent the contact info and at least one helper method to print all of the contact info. When creating a contact you need to create at least 3 properties: firstname, lastname, and email.
- collect each new contact instance in a list or dictionary. When you print the contact info, loop through this list and use the helper function to print the data.
- please format printed output so all column data lines up.
- You do not necessarily need to reverse first and last names when printing.

Here is a possible print-out, where we display some pre-existing contacts, add a new contact, then re-display the list to see that our contact has been added. Again, you are not required to save to disk before exiting.

```
Program Options.
  1.) Display all contacts
  2.) Create new contact
  3.) Save and exit

Enter 1, 2, or 3: 1

Name          Email
smith, sue     ssmith@wer.com
Dean, hugh     hrdean@abc.com
green, sam     sgreen@rty.com

Program Options.
  1.) Display all contacts
  2.) Create new contact
  3.) Save and exit

Enter 1, 2, or 3: 2

Enter contact's first name: mac
Enter contact's last name: davis
Enter contact email: mdavis@asd.com

Program Options.
  1.) Display all contacts
  2.) Create new contact
  3.) Save and exit

Enter 1, 2, or 3: 1

Name          Email
smith, sue     ssmith@wer.com
Dean, hugh     hrdean@abc.com
green, sam     sgreen@rty.com
davis, mac     mdavis@asd.com

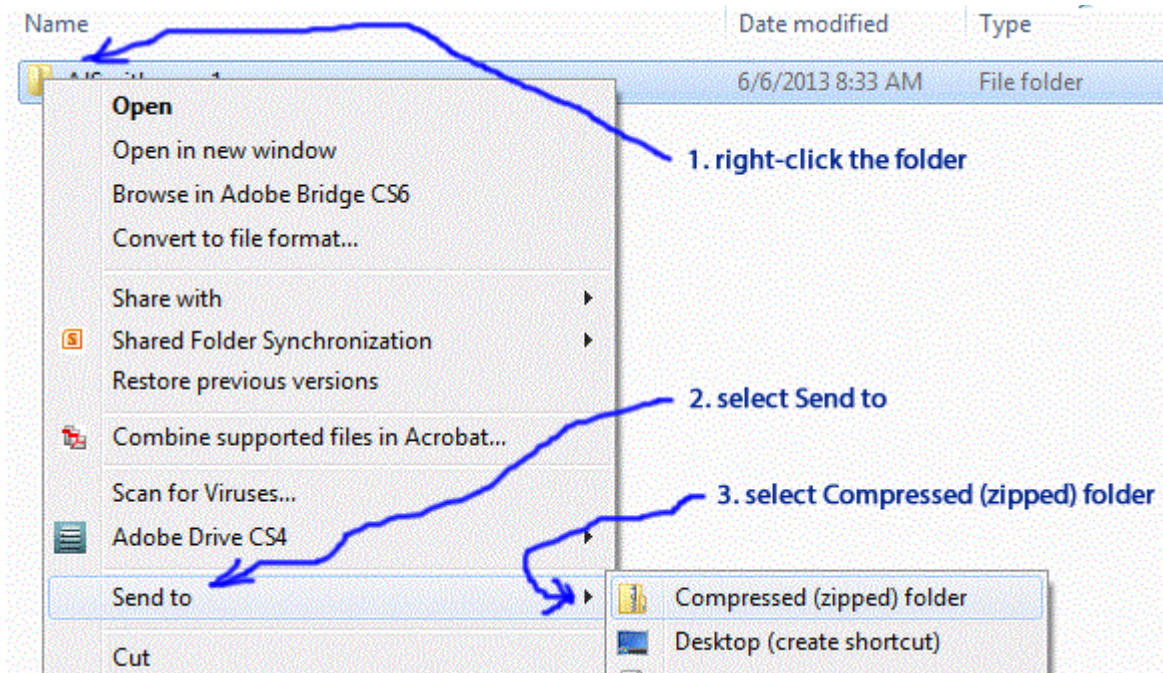
Program Options.
  1.) Display all contacts
  2.) Create new contact
  3.) Save and exit

Enter 1, 2, or 3:
```

If you get good and stuck, attach a compressed version of your code and mail it to me; I'll be glad to help you troubleshoot it.

# How To Submit Your Program

You need to compress the folder that holds your program file or files before you email it to me. Uncompressed code is likely to be filtered out by a suspicious email server. You probably know this, but to compress an existing folder, *right click it*, choose *Send to*, then choose *Compressed (zipped) folder*, as shown below. The zipped folder has a little zipper on it.



If you email me your finished program, unless it comes in late at night, I will usually grade it that same day. Also, if you get stuck on something you can also send me your code and I'll be happy to help you get it working. The earlier in the week the program is due that you start, the more time we have to sort out any problems there may be. So, please start on these program many days before the due date.

Please clearly name the folder with your code, with *your name, the class, and the program number*. For example, jsmith-pythonprogram3.

I get lots and lots of programs from several different classes, so it helps me a great deal in keeping things straight if I can tell what's in the folder by the folder label.

Once everything is working (or even if it isn't) attach the zipped folder to an email and send it to me..

If you have questions let me know: [mark.prather@kctcs.edu](mailto:mark.prather@kctcs.edu).