LAB DIRECTIONS for LAB 4/23/2015

Advanced Files & Structures

1. Create a folder named **Lesson Set 12**. Put all files you create for this lab in this folder.

2. Bring in the program from **Lesson Set 12 Source Code** folder named **`petProgram.cpp`** . Also bring in the text file named **`pets.txt`**. You are going to modify this program.

    a. Create a comment block at the top of petProgram.cpp

    b. **MAIN** FUNCTION MODIFICATIONS:

        i. After all the variables are defined, print a menu to the screen with the options below. Validate the user's choice.

```
Please choose from the following options:
1. Enter pet data.
2. Print pet data.
3. Quit the program.
Choose 1-4:
```

        ii. Use a switch statement to determine the user's choice. If the user selected 1 – to enter pet data – then call the enterPetData function – sending the myPets array and the current number of pets to this function. This function should return the new number of pets as an integer.

        iii. If the user selected 2 – to print pet data – then call the printPetData function – sending the myPets array and the current number of pets to this function. This function should not return anything.

        iv. If the user selected 3 – say "GoodBye!" and then end the program

        v. You will be cutting the other code that is currently inside of the main function (where you enter pet data & print it out) and moving it to the other functions below. HINT: the only variables that should be left in the main function after you are done is the pet array, curNumPets, and a variable to hold the user's choice. All other variables will be moved to one of the other functions described below.

    c. **enterPetData** FUNCTION:

        i. You will now be using the code that was provided for you in the main function – but you will move it to this function and modify it. Instead of the inFile being defined as ifstream, define the file as fstream and make sure to open it in input file mode. Then, make sure to close the file. Also, print out a confirmation that says how many pets were added to the array. (Look at screen capture below – yellow highlighted parts are where the user entered data.)

```
Please choose from the following options:
1. Enter pet data.
2. Print pet data.
3. Quit the program.
Choose 1-4:  1

Please enter the name of the file that contains your pet data:  pets.txt
3 pets have been added to the array.

Please choose from the following options:
```

        ii. This function should return the curNumPets variable.

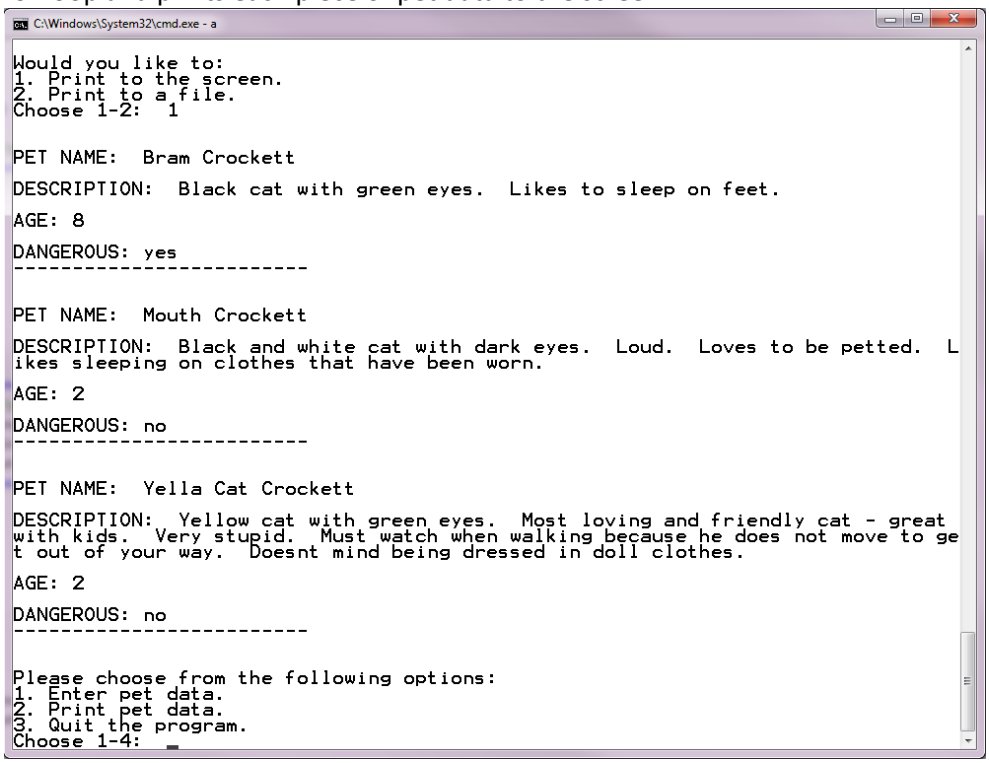
d. **printPetData** FUNCTION:

i. You need to ask the user if they wish to print the pet data to the screen or to a file. Make sure to validate the user's choice

```
Please choose from the following options:
1. Enter pet data.
2. Print pet data.
3. Quit the program.
Choose 1-4:  2


Would you like to:
1. Print to the screen.
2. Print to a file.
Choose 1-2:
```

ii. If they chose to print to the screen then you will now be using the code that was provided for you in the main function – but you will move it to this function. This code simply goes through a for loop and prints each piece of pet data to the screen.



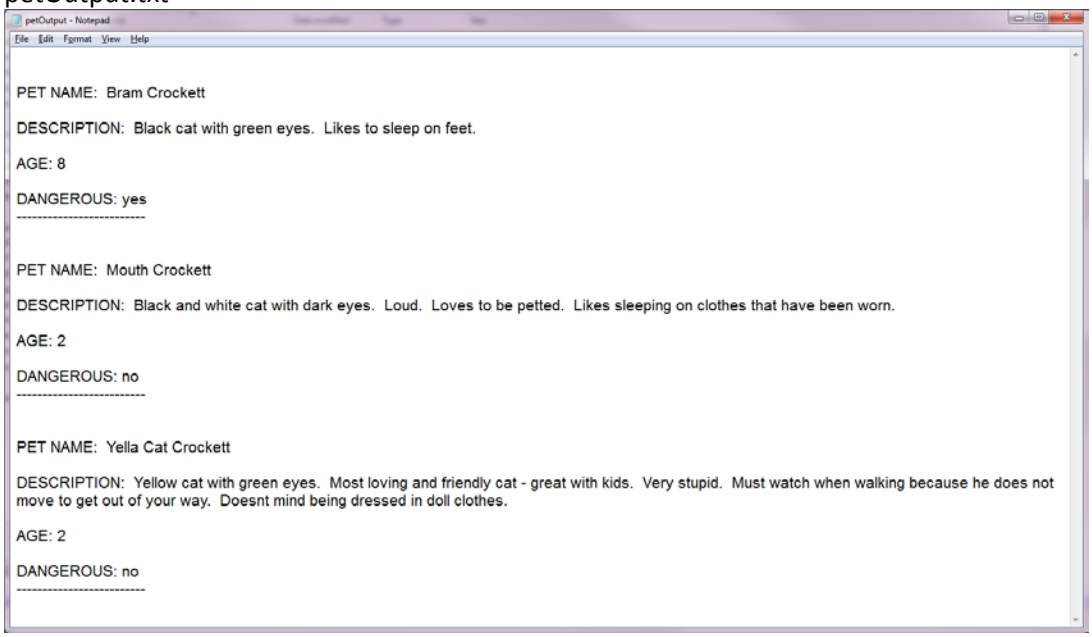
iii. If they chose to print to a file then you will have to ask them what they want the name of the file to be. CHOOSE to define the file as fstream instead of ofstream and open it in output mode. Print to the file EXACTLY in the format you printed to the screen. Make sure to close the file. Also, print out a confirmation to the screen that says that pet data has been printed to their file.

```
Would you like to:
1. Print to the screen.
2. Print to a file.
Choose 1-2:  2


What is the name of the file?
petOutput.txt
Pet data has been printed to petOutput.txt


Please choose from the following options:
1. Enter pet data.
```

petOutput.txt



3. Compile & run your program using the same data as in the samples above and then create a screen capture of your code running. Also take a screen capture of your output file like I did above. Put your screen capture in a document named **Lesson Set 12 Screen Captures**.

What to Turn In: (by Wednesday, April 29, 2015)

- petProgram.cpp
- pets.txt
- petOutput.txt
- Lesson Set 12 Screen Captures

How you will be graded

petProgram.cpp	70 points	FOLLOWS SPECIFICATIONS - Main function is modified to only have the menu of options and a switch statement which calls the functions if applicable. Menu should show until user chooses option 3 – to end the program. -Enter pets function defines the file as an fstream object and opens with ios::in. Pet data is appropriately read from the file and put in the array. The file is closed after used. A confirmation is printed showing how many pets were added to the array. curNumPets is returned from this function. -Print pet data function defines the file as an fstream object and opens with ios::out. Allows user to choose to either print to the file or the screen. Validates user's choice. Prints neatly to the screen. Asks the user for the name of the file and then opens and prints to the file just like you did when you printed to screen. File is closed. A message is printed to the screen indicating that the pet data was printed to the file.
petOutput.txt	15 points	Output to this text file looks exactly like the output to the screen when user chose to print to the screen.
pets.txt	5 points	You do not have to modify this file at all – but it is helpful if you include it so I can test your program quickly. I will give you 5 points just for including it in your zipped file.
Lesson Set 11 Screen Captures	10 points	Screen capture of petProgram.cpp running and also of petOutput.txt after running.