

Programming Project #1

Assignment Overview

This assignment involves coding and testing of a program based on the “Hello World” program from the first lab.

The basic design of the first programs that you construct in this class consists of a prompt for information, receiving information, processing that information then producing a display of the results.

This assignment is worth 10 points (1% of course grade), and must be completed before 11:59 PM on Tuesday, September 2nd. Projects are typically due on Mondays but it is one day later than usual because of Labor Day.

Background

This programming project will use the `raw_input` and `print` functions along with some simple mathematics for rate conversion. The important part of the project is to learn the skills needed to access the class web site to download a project description, create a new program in Python and finally to hand it in.

Program Specifications

Your program will prompt the user for a floating point number which stands for gallons of gasoline. You will reprint that value along with other information about gasoline and gasoline usage:

- Number of liters
- Number of barrels of oil required
- Number of pounds of CO₂ produced
- Equivalent energy amount of ethanol gallons
- Price in US dollars

You can find these measures on the web (try wikipedia), but just in case:

- 1 gallon is equivalent to 3.7854 liters
- 1 barrel of oil produces 19.5 gallons of gas (approximately, can vary depending on the oil source. This number will work for us). FYI, a barrel is 42 gallons.
- 1 gallon of gas produces approximately 20 pounds of CO₂. Again, good enough
- 1 gallon of gas produces 115,000 BTU (British Thermal Units). 1 gallon of ethanol produces 75,700 BTU.
- God knows what the cost should be, but let's peg it at \$4.00/gallon

Deliverables

proj01.py -- your source code solution (remember to include your section, the date, project number and comments).

1. Please be sure to use the specified file name, i.e. “proj01.py”
2. Save a copy of your file in your CS account disk space (H drive on CSE computers).

3. Electronically submit a copy of the file.

Assignment Notes:

To input the numbers it is necessary to use the `raw_input` function. The `raw_input` function takes a string, a sequence of characters between quotes, as a prompt to print to the user. It then waits until the user types a response, terminated by the user typing the Enter key. A string, again as a sequence of characters, is returned.

The returned string must be converted to a number. In this assignment we are strictly working with floating point numbers, a string is converted to an float using the `float` function. The `float` function takes as an argument a single string and returns the floating point number the string represents. A typical interaction would be something like:

```
numStr = raw_input('Please enter a number: ')
floatVar = float(numStr)
```

`print` is a command that will print on the output window any combination of variables, values and strings. Each item to be printed must be separated from other items by a comma. All the items will be printed together, followed by a new line. For example:

```
billsFloat = 3.14159
print 'The number ',billsFloat,' times two is ', billsFloat*2
```

This command has 4 items to print: a string ('The number '), the value in the variable `billsFloat` (3.14159), another string (' times two is ') and the result of an expression (6). What it will print is:

```
The number 3.14159 times two is 6.28318
```

Look at the program `numberInput.py` in the `proj01` directory as an example of using `raw_input`, `print` and `float`

Once converted to numbers, the operations on these numbers are, respectively: `+` (sum), `-` (difference), `*` (product), `/` (division) and `%` (remainder). The last two deserve special comment.

In Python, if an integer is divided by another integer, the result is an integer. Thus the result of `6/4` is `1`. `6.0/4` is `1.5`. That is, the `"/"` operation results in the **integer quotient if using integers, floats if using floats**. The result of `6%4` is the integer remainder of the division, thus `2` (6 divided by 4 is 1 with a remainder of 2). *Play around with the quotient and remainder operators in the Python shell window until you get comfortable with how it works.*

To clarify the problem specifications, we provide at the end of this document a snapshot of interaction with the already written program.

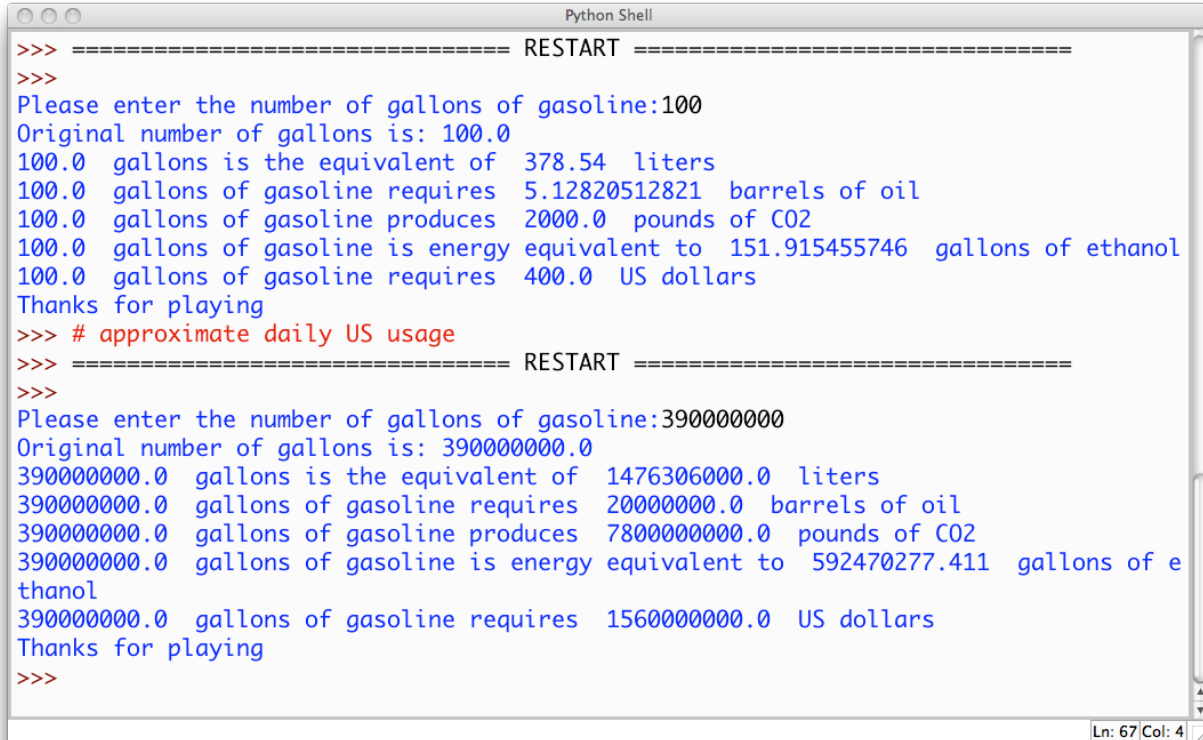
Getting Started

1. Using IDLE create a new program.
2. If you are in a CSE lab, select the H: drive as the location to store your file
3. Save the name of the project: proj01.py
4. Using the example from numberInput.py, write the code. Track down any errors (shouldn't be any at first).
5. Run the program
6. Use the web site to hand in the program (to make sure you can do it)
7. Edit the program
8. Now you enter a cycle of edit-run to incrementally develop your program.
9. Hand in your final version.

Questions for you to consider (not hand in)

1. What happens when you try to divide by zero when you run your program?
2. What happens when you enter a letter instead of a number at the prompt?

Sample Interaction



```
>>> ===== RESTART =====
>>>
Please enter the number of gallons of gasoline:100
Original number of gallons is: 100.0
100.0 gallons is the equivalent of 378.54 liters
100.0 gallons of gasoline requires 5.12820512821 barrels of oil
100.0 gallons of gasoline produces 2000.0 pounds of CO2
100.0 gallons of gasoline is energy equivalent to 151.915455746 gallons of ethanol
100.0 gallons of gasoline requires 400.0 US dollars
Thanks for playing
>>> # approximate daily US usage
>>> ===== RESTART =====
>>>
Please enter the number of gallons of gasoline:390000000
Original number of gallons is: 390000000.0
390000000.0 gallons is the equivalent of 1476306000.0 liters
390000000.0 gallons of gasoline requires 20000000.0 barrels of oil
390000000.0 gallons of gasoline produces 7800000000.0 pounds of CO2
390000000.0 gallons of gasoline is energy equivalent to 592470277.411 gallons of ethanol
390000000.0 gallons of gasoline requires 1560000000.0 US dollars
Thanks for playing
>>>
```