CIS 122 Summer 2015
Project 3: Booleans and Python Conditionals

Due Thursday 23rd July 2015 5:00 PM

Goals:

1. By the end of this homework you should be able to use the if,else,elif keyword
2. be able to build branching functions
3. be able to understand bool type

Getting started:

This homework uses the editor window (open Idle, then go to the menu and choose File->New File (may say "New Window" in some versions), a shortcut is ctrl-N on Windows, probably command-N on Macs. Add comments to indicate the start of a problem using the # sign.

Example: #Problem 1
The # sign tells Python to ignore everything after it.

Problem 1: project3a.py

OK or Not OK 10 points

Write the following functions:

- Write a function that takes one input parameter, word of type string and returns True if word is at least 5 characters long, and False otherwise.

- Write a function that takes one input parameter, word, of type string and returns True if word does NOT contain the characters 'E' or 'e',and False otherwise.
  Hint: Look at the different ways of writing the if-statement.

XC +5 points: Extend the second part to check for any other two vowels as well.
Note: Write doc-strings for the above functions. As given on p.47 of the textbook.

Problem 2: project 3b.py

Rock, Paper XOR Scissors 5+15=20 Points

- exclusive_or: Write a function, exclusive_or() which takes two Boolean arguments and returns true if exactly one of them is true, otherwise returns false. In other words
  >>> exclusive_or(True, False)
  True
  >>> exclusive_or(False, True)
  True
  >>> exclusive_or(False, False)
  False
  >>> exclusive_or(True, True)

False
Remember to write a docstring for exclusive_or(). You may use the test cases above if you like.

- Rock, paper, scissors is a two-player game in which each player chooses one of three items (rock, paper, or scissors). If both players choose the same item, the game is tied. Otherwise, Rock crushes/wins over Scissors, Scissors cuts/wins over Paper, and Paper covers/wins over Rock.

  Write a function, rps(), that asks two players in turn for their input choices ('r', 'p', or 's') and prints out which player wins and the reason for the win.
  For example,
  rps() Welcome to Rock, Paper, Scissors!
  First player, enter 'r', 'p', or 's': r
  Second player, enter 'r', 'p', or 's': s
  r beats s - player 1 wins
  Don't worry about a docstring since this function takes user input.

  Hint- there are nine total cases to consider but there may be a way cover 3 of the cases with one test, thus reducing the work needed substantially.

Problem 3: project3c.py

20 points

The City of Plutonium, Oregon has decided to raise funds using an income tax. The general idea: as your income goes up, so does the rate you pay. Following is the rubric for it.

At least	Up to	Rate	on amount over	Max Due for this rate
0	5000	0.0%	-	$0.00
> 5,000	15,000	1.0%	5,000	$100.00
> 15,000	50,000	1.5%	15,000 + $100	$625.00 100 + 525
> 50,000	250,000	2.0%	50,000 + $625	$4,625 625 + 4,000
> 250,000	any amount	2.5%	250,000 + 4,625	No max

Your program should ask each person for
name
income
Then print the amount of income tax due
Example
Joe earned 1650 last year. He owes $0 taxes.

Julie earned 12,000 last year. She owes 1.0% on the amount over 5000, or 1% of 7,000 + 0, for a total of $70.

Jane earned 35000 last year, so the 1.5% rate applies.
She owes 1.5% of the amount she earned over 15000, plus the max due on the next lower rate, or $100.

```
 35,000
-15,000
 ----------
 20,000 * 0.015 = $300
```

Her tax due $300 + $100 = $400

Howard earned 75,000 last year so he owes 2.0% of the amount over 50,000 or 0.015 * 25,000 = $500 plus $625 for a total due of $1125.

Wanda earned 300,000 last year so she owes 2.5% of the amount over 250,000 or 0.025 * 50,000 = $1250 + $4625, for a total due of $5875.

Grading Rubric:
4 points for each of the 4 non-zero rates, 16 total.
3 points for total tax due
1 point for calculating average tax due