

Part I

Write Python programs that will solve problems listed below:

1. Get the user's first and last name with two `raw_input` calls and outputs them in reverse order with a comma between them.
2. Get the user's full name with one `raw_input` call and display the name as last, first
3. Gets the coordinates for two points from the user and computes the distance. Round your answer to two decimal places.
4. At Jenny's birthday party she orders a 32 piece pizza. Have the user (probably Jenny) enter the number of people at the party that will be eating pizza and output the number of slices each one gets. As you know the pizza might not divide evenly. There are two ways to handle this. The first is to ignore the extra pieces and give everyone the same amount. The second way is to cut up the extra pieces so that everyone gets the same amount. Your program must output both options. E.g.

```
Number of guests: 10
Option 1: 3 slices each, 2 left over
Option 2: 3.2 slices each
```

5. Get a sentence from the user and display it back with one word per line.
6. When building an enclosure for a python the amount of area at the base of the enclosure should be proportionate to the length of the snake. The minimum needed is $1/2$ square foot for each foot in length up to and including 6 and $3/4$ square foot for each foot after that.

e.g. 9' python needs 5.25 square feet ($6 * 1/2 + 3 * 3/4$)

Create a program that asks the user how long their python is and tell them the minimum area they need for the base of its enclosure.
7. Palindromes are words that are the same forwards and backwards. Create a program that gets a word from the user and tells them if it is palindrome or not. Your program should not care about capitalization.
8. As you might know, the persistence of a number is the number of steps it takes to get a one-digit number when separating the digits and multiplying them together. What you may not have noticed is that persistence is inherently recursive. The persistence of any one-digit number is zero. The persistence of any other number is one plus the persistence of the digits multiplied together. Create a recursive function called `persistence` that computes the persistence of any whole number.

Part II

1. Write a simple Django app that will allow users to run the above programs by entering appropriate urls into a web browser. The arguments that each program needs should form part of the respective url.

Prepared by Krzysztof Bandurski using exercises found on:
<http://compsci.ca/v3/viewtopic.php?t=16962>