Developers Institute

Python Course

Week 1

Day 3

Exercises

Group Exercise – Favourite Fruits

1. Ask the user to type in his/her favourite fruits.
2. What's the best way to deal with multiple favourites?
3. We'll do it differently here – get input in one string, and ask the user to separate between fruits with a single space. We will also accept a single fruit as the input here.
4. Store the favourite fruit(s) in a list.
5. Now ask the user to type in the name of a fruit.
6. If the user inputs the name of a fruit that is **not** one of his favourites, print a message like this: "Are you sure? You really like: apples, pears, oranges".
7. (Bonus: add the word "and" before the last fruit in the list – but only if there are more than 1 favourites!)
8. But if the user inputs the name of a fruit that **is** one of his favourites – print a message like this: "You like those! Good choice!"

Exercise 1 – Retirement

1. Retirement age in Israel is 67 for men, and 62 for women born after April, 1947.
2. Ask the user's sex and date of birth in the form "1993/09/21".
3. Print a message telling the user his/her age.
4. Determine whether the user is above or below Israeli retirement age. Enclose this logic into a function, which will return True if the user is above retirement age, or False if not. This function should be named '**can_retire**', and it should take 2 arguments: sex and date of birth.
5. Display a message to the user informing them whether they can retire or not.
6. (Bonus: extend your 'can_retire' function using the short form of the 'if' statement:
    1. Create a variable called **can_retire**
    2. Assign it a value in a single statement.
    3. You may need to do some initial steps first, to get the user's date of birth into a useful form.
    4. As always, test your code to ensure it works.)

Exercise 2 – Paragraph Analysis

1. Find an interesting paragraph of text online. (Please keep it appropriate to the social context of our class.)
2. Paste it into your code, and store it as a variable.
3. Python code lines should be no longer than 79 characters. Format the paragraph to fit with this. Use string concatenation.
4. Let's analyze the paragraph. Print out a nicely formatted message saying:

1. How many sentences the paragraph contains
2. How many words it contains
3. How many characters it contains (this one is easy...)
4. (Bonus: How many **non-whitespace** characters it contains)
5. How many unique words it contains
6. (Bonus: The average amount of words per sentence in the paragraph)
7. (Bonus: the amount of non-unique words in the paragraph)

Exercise 3 – Car Manufacturers

1. Here is a list of popular car manufacturers: https://pastebin.com/bkBRuvAZ
2. Paste it into your code, and store it in a variable.
3. Convert it into a list using Python (don't do it by hand!)
4. Print out a message saying how many manufacturers/companies are in the list
5. There are a few duplicates in the list!
    1. Remove these programmatically. (Hint: use sets to help you)
    2. Print out the companies without duplicates, in a comma-separated list with no line-breaks (eg. "Acura, Alfa Romeo, Aston Martin, ..."), and also print out a message saying how many companies are now in the list.
6. Print the list of manufacturers in reverse/descending order (Z-A)
7. Using list comprehension:
    1. Find out how many manufacturers' names have the letter 'o' in them.
    2. Find out how many manufacturers' names **do not** have the letter 'i' in them
    3. Print the above information out with meaningful output messages.
8. (Bonus: print out the list of manufacturers in ascending order (A-Z), but reverse the letters of each manufacturer's name)