

PYTHON - ASSIGNMENT II

1. Create a variable, paragraph, that has the following content:
"Python is a great language!", said Fred. "I don't ever remember having this much fun before."
2. Write an if statement to determine whether a variable holding a year is a leap year.
3. Write code that will print out the anagrams (words that use the same letters) from a paragraph of text.
4. Create a list. Append the names of your colleagues and friends to it. Has the id of the list changed? Sort the list. What is the first item on the list? What is the second item on the list?
5. Create a tuple with your first name, last name, and age. Create a list, people, and append your tuple to it. Make more tuples with the corresponding information from your friends and append them to the list. Sort the list. When you learn about sort method, you can use the key parameter to sort by any field in the tuple, first name, last name, or age.
6. Create a list with the names of friends and colleagues. Search for the name 'John' using a for a loop. Print 'not found' if you didn't find it.
7. Create a list of tuples of first name, last name, and age for your friends and colleagues. If you don't know the age, put in None. Calculate the average age, skipping over any None values. Print out each name, followed by old or young if they are above or below the average age.

8. Write a function, `is_prime`, that takes an integer and returns `True` if the number is prime and `False` if the number is not prime.
9. Write a binary search function. It should take a sorted sequence and the item it is looking for. It should return the index of the item if found. It should return `-1` if the item is not found.
10. Write a function that takes camel-cased strings (i.e. `ThisIsCamelCased`), and converts them to snake case (i.e. `this_is_camel_cased`). Modify the function by adding an argument, `separator`, so it will also convert to the kebab case (i.e. `this-is-camel-case`) as well.
11. Create a variable, `filename`. Assuming that it has a three-letter extension, and using slice operations, find the extension. For `README.txt`, the extension should be `txt`. Write code using slice operations that will give the name without the extension. Does your code work on filenames of arbitrary length?
12. Create a function, `is_palindrome`, to determine if a supplied word is the same if the letters are reversed.
13. Write a function to write a comma-separated value (CSV) file. It should accept a filename and a list of tuples as parameters. The tuples should have a name, address, and age. The file should create a header row followed by a row for each tuple. If the following list of tuples was passed in:
`[('George', '4312 Abbey Road', 22), ('John', '54 Love Ave', 21)]`
it should write the following in the file:

```
name,address,age
George,4312 Abbey Road,22
```

John,54 Love Ave,21

14. Write a function that reads a CSV file. It should return a list of dictionaries, using the first row as key names, and each subsequent row as values for those keys.

For the data in the previous example it would return:

```
[{'name': 'George', 'address': '4312 Abbey Road', 'age': 22}, {'name': 'John', 'address': '54 Love Ave', 'age': 21}]
```

15. Imagine you are designing a banking application. What would a customer look like? What attributes would she have? What methods would she have?
16. Imagine you are creating a Super Mario game. You need to define a class to represent Mario. What would it look like? If you aren't familiar with SuperMario, use your own favorite video or board game to model a player.
17. Write a program that serves as a basic calculator. It asks for two numbers, then it asks for an operator. Gracefully deal with input that doesn't cleanly convert to numbers. Deal with division by zero errors.
18. Find a package in the Python standard library for dealing with JSON. Import the library module and inspect the attributes of the module. Use the help function to learn more about how to use the module. Serialize a dictionary mapping 'name' to your name and 'age' to your age, to a JSON string. Deserialize the JSON back into Python.
19. Write a Python class to find validity of a string of parentheses, '(', ')', '{', '}', '[' and ']'. These brackets must be close in the correct order, for example "()" and "()[]{}" are valid but "[)", "({[})", and "{{{" are invalid

20. Write a Python class to find the three elements that sum to zero from a list of n real numbers.

Input array : [-25, -10, -7, -3, 2, 4, 8, 10]

Output : [[-10, 2, 8], [-7, -3, 10]]