Please note:

- Read carefully the questions.
- Feel free to add comments in your answers and read the documentations as you see fit.
- Take all the time you need
- Any Memory optimization or use of Python’s built-in functions is considered as a plus.
- Do test your code.

1/ Test 1(mandatory):

Create a new file called *web.py* in within a function that sends a post request to this url *https://httpbin.org/anything* with this parameters:

> *msg=welcomeuser*
> *isadmin=1*

and print the response body.

Now, send the same request this time acting as a mobile user and print the request’s headers.

2/ Test2 :

Create a file and name it *functional.py*

Write a function that takes a list of strings a returns the sum of the list items that represents an integer (skipping the other items).

3/ Test 3:

In the same file. Create a function “persistence” that takes an integer “n”, and return the number of times you must multiply the digits in “n” until you reach a single digit.

Examples

> persistence(67) ➞ 3   # Because 6*7 = 42, 4*2 = 8 and 8 has only one digit
>
> persistence(999) ➞ 4   # Because 9*9*9 = 729, 7*2*9 = 126, 1*2*6 = 12, and finally 1*2 = 2
> persistence(5) ➞ 0   # Because 5 is already a one-digit number

4/ test4 (mandatory):

In the same file. Create a function that takes a list of integers and return the sum of the numbers that repeat consecutively (return your result as a list).

Examples

> sum_consecutives([0, 7, 7, 7, 5, 4, 9, 9, 0]) ➞ [0, 21, 5, 4, 18, 0]
> sum_consecutives([4, 4, 5, 6, 8, 8, 8]) ➞ [8, 5, 6, 24]
> sum_consecutives([-5, -5, 7, 7, 12, 0]) ➞ [-10, 14, 12, 0]

5/ Test 5(mandatory):

In a separate folder *src,* create a python module with the following constraints:

a/it imports the *data.json* file which is located in the root folder. The file holds information about a category including the products.

b/ it prints each available product in this particular format:

"You can buy *Product_Name* at our store at *Product_Price*"

where :

> Product_Name is the product name truncated at 30 .
>
> Abbotts Village Bakery Ghrainy Wolemeal 850g ==> Abbotts Village Bakery Grainy
>
> Product_Price is the rounded product price in *dd.d* format
>
> Example: 13.34 ==> 13.3

c/ if the product is unavailable, it logs the product id and product name.
d/ if a clue of the product's availability can't be found, it logs an error.
e/ it saves the available products in a csv file.

P.S :

- Use any tool/language/method/website to find the products in the json file.
- To "truncate" means "to shorten by cutting off the top or end".

6/ Web Scraping (mandatory):

In the root folder, start a new scrapy project and build a spider that crawls the webpage below, extracting the products names, and the breadcrumb in a list format. Make sure to take advantage of *Scrapy.Items* to output the products.
Test your spider and save the results in a csv file.

Webpage:

> https://www.woolworths.com.au/shop/browse/drinks/cordials-juices-iced-teas/iced-teas

Breadcrumb :

> Home > Drinks > Cordials, Juices & Iced Teas > Iced Teas

The desired format:

> ["Home", "Drinks", "Cordials, Juices & Iced Teas", " Iced Teas"]