

# WEEK1- EXERCISES-SUBJECT

## I) Working on variables

### STRING

#### **Exercise 1**

- 1) Write variables and give them values:
  - *addressNumber*
  - *addressStreet*
  - *country*
- 2) Write a variable *global\_address*, and concatenate inside, the variables:
  - *addressNumber*, *addressStreet*, *country*, to create a sentence

Example : *global\_address* should display « I live in BenYehuda 5, in Israel »
- 3) Print *global\_address*

#### **Exercise 2**

- 1) Create a numerically indexed table: *pets*, like this  
['cat','dog','fish','rabbit','cow'].
- 2) Print dog
- 3) Add to the array *pets*, the pet horse. Remove the pet rabbit
- 4) Print with the for loop the array *pets*, and its length

#### **Exercise 3**

- 1) Copy this article in a variable *article*

This news is about a Chihuahua. It is very small. It is only 4.5 kilograms heavy

- 2) Check and Print how many letters are in Chihuahua
- 3) Count and Print how many -h- are in "Chihuahua"
- 4) Print "huahua" from the variable *chihuahua* and put it in a variable called *newDog*
- 5) Put the variable *newDog* in uppercase and then in lowercase and Print it
- 6) Check if the word Chihuahua is in the article
  - if yes, Print 'I LOVE Chihuahua, it's my favorite dog'. Where dog is the item in the array *pets* created in **Exercise2**
  - if not, Print -I dont care, I prefer CATS

## NUMBER

### Exercise 4

- 1) Ask a number to the user, and save it to a variable called *num*
- 2) Check if num is an even number -

If yes, Print, 'num is an even number'. Where *num* is the actual number of the user

If no, Print 'num is not an even number'. Where *num* is the actual number of the user

### Exercise 5

- 1) Write a Python program that prints all the numbers from 0 to 6 except 3 and 6.  
If the number is 3 or 6, don't print it,  
If not, print the number

## NUMBER/LOOPS - HARDER EXERCISES

### Exercise 6

- 1) Write a Python program to print alphabet pattern 'A'.

Hint : Use Row and Column



## Exercise 7

- 1) Write a Python function that takes a parameter: `myAge`, and return the age of `myMum` and `myDad`.

### Explanation

In the function, create two variables `myMum` and `myDad`.

`myMum` age is: twice my age, plus 5 years

`myDad` age is: three times my age

`myAge` is 20

## Exercise 8

Let's create functions that calculate your vacation's costs:

- Define a function called `hotel_cost`, that takes one argument `nights`. This argument is an input.

The hotel costs \$140 per night. So, the function `hotel_cost` should return `140 * nights`.

- Define a function called `plane_ride_cost` that takes one argument `city`. This argument is an input.

The function should return a different price depending on the location.

```
"London": 183$  
"Paris": 220$  
All other destination : 300$
```

- Define a function called `rental_car_cost` that takes one argument `days`. This argument is an input.

Calculate the cost of renting the car:

Every day you rent the car costs \$40.(cost=40\*days)

- If you rent the car for 7 or more days, you get \$50 discount on your total (Example: 100\$ (is the cost) - 50\$ ).
- If you rent the car for 3 or more days, you get \$20 off your total.
- You cannot get both of the above discounts.

Return that cost.

- Define a function called `trip_cost` that takes three arguments, `city`, `night` and `days`.

The function has to return the total cost of the vacation by calling `the rental_car_cost(days)`, `hotel_cost(days)`, and `plane_ride_cost(city)` functions.

Example : The car cost: \$x, the hotel cost: \$y, the plane tickets cost: \$z.

- Call the function `trip_cost`

## Exercise 9

- 1) Write a Python function that takes a list and returns a new list with unique elements of the first list.

Example list=[1,2,3,3,3,3,4,5] newList = [1,2,3,4,5]

Example list=[1,2,3,3,3,3,4,5] newList = [1,2,3,4,5]

## DICTIONARIES

## Exercise 10

Follow the steps:

- First, make a list called *myGroceries* with the values "banana", "orange", and "apple".
- Create this two dictionaries:

```
stock = {  
    "banana": 6,  
    "apple": 0,  
    "orange": 32  
}  
  
prices = {  
    "banana": 4,  
    "apple": 2,  
    "orange": 1.5  
}
```

- Define a function *myBill* that takes one argument *food*. (this argument is *myGroceries* list created before)

In the function, create a variable *total* with an initial value of zero.

For each item in the food list, add the price of that item to *total*.

Example: Banana cost 4 (look at the *prices* dictionary)

Return the *total*.

- Call the function *myBill*

## Bonus

- In the function *myBill*

While you loop through each item of food, only add the price of the item to the *total* if the item's stock is greater than zero.

If the item is in stock, decrease the item's stock by 1

Example: They are 6 bananas in the stock(look at the *stock* dictionary). Banana is in stock so you can add the price of one banana in the variable *total*. Then decrease the stock of bananas. Now the stock is 5.

