Python Basic - Final Work

Ex 1:
Write a for loop that prints the numbers from 12 to 24.

Ex 2:
Write a for loop that prints the ODD numbers from 7 to 31

Ex 3:
Write a for loop that prints the EVEN numbers from 10 to -20.

Ex 4:
Write a for loop that iterates through all numbers from 1 to 45. Print the following:

- For each number that multiples of 3 print “Fizz”
- For each number that multiples of 5 print “Buzz”
- For each number that multiples of 3 and 5 print “FizzBuzz”

Ex 5:
Write a function that receives an array as a parameter and calculates the sum of all the numbers in the given array (don’t use sum() function).

For example if the given array is: [1,13,22,123,49,34,5,24,57,45]
The result should be 373

Ex 6:
Write a function that receives an array of objects.
Each object should represent a student with the properties:

- id
- first name
- last name
- age
- country
- city

In addition, the function should receive a property to change.

1 - The function should check for each property in each object in the array if the given property exists and if it does, the function should delete it from the object.
2 - Write a function that prints each property of each object in the given array.
3 - Write a function that sorts the array by the students age from the oldest to the youngest and return the sorted array.

Ex 7:

```
our_pets = [
    {
        "animal_type": "cat",
        "names": [
            "Meowzer",
            "Fluffy",
            "Kit-Cat"
        ]
    },
    {
        "animal_type": "dog",
        "names": [
            "Spot",
            "Bowser",
            "Frankie"
        ]
    }
]
```

1 - Write a function that receives the array shown above and prints only animalType: cat.
2 - Write a function that receives the array shown above and the animal type. The function should print all names of that animal type if this type exists in the object.
3 - Write a function that that receives the array shown above and animal name The function should add the specified animal name to each ‘names’ array in each animal_type if that name does not exist in the ‘names’ array.

Ex 8:

```
student = {
    'name': 'John',
    'age': 20,
    'hobbies': ['reading', 'games', 'coding'],
}
```

1 - Write a function that prints all the student data (each student property should be printed in a new line).
2 - Write a function that receives the student object and a hobby, the function should add the hobby to the student's hobbies array if it’s not exist already.
3 - Use the function that you wrote in ex 1 to print the data of the student and check that the new hobby has been added.
4 - Write a function that receives an object of a student and hobby, the function should delete the hobby from the student's hobbies.
5 - Use the function that you wrote in ex 1 to print the data student and check that the hobby has been deleted from the object student.
6 - Add to the object student new property: family_name and add a value.

Ex 9:

Write a function that prints all the elements of a 2D array using nested for loops.

```
matrix =
[
  [1, 2],
  [3, 4],
  [5, 6]
]
```

print_matrix(matrix) → Should print: 1 2 3 4 5 6

Ex 10:

Write a function to count how many numbers of zeros appear in a 2D matrix using nested for loops and increment operation.

```
matrix =
[
  [0,1,1],
  [0,1,0],
  [1,0,0]
]
```

print(zero_count(matrix)) → Should print: 5

Ex 11:
Write a function to return an array of all the elements that are repeated more than once in a given array.
arr = [4,2,34,4,1,12,1,4]
print(find_dup(arr)) Should print: [4, 1]

Ex 12:
Write a function using a for loop that gets an array and returns a new array with the elements from the given array appearing in reverse order. (Don't use array reverse() method)

For example:
arr = [43, "what", 9, true, "cannot", false, "be", 3, true];

Function output should be:
[ture, 3, "be", false, "cannot", true, 9, "what", 43]

Ex 13:
Given two arrays of integers. Add up each element in the same position and create a new array containing the sum of each pair.
Assume both arrays are of the same length.

For example:
first_array = [4, 6, 7];
second_array = [8, 1, 9];

Function output should be:
[12, 7, 16]

Ex 14:
Write a program that will check if two strings are palindromes.
A palindrome is a word that spells the same forward and backward.
Palindrome: a word, phrase, or sequence that reads the same backward as forward, examples for valid palindromes: madam, nurses run.

For example:
first_str = "racecar"
second_str = "Java"

Function output should be:

True (for first_str)
False (for second_str)

Ex 15:
Write a while loop that iterates as long as the counter is less than 100, on every iteration the counter is multiplied by 2 starting from 1.

Ex 16:
Write a while loop that iterates as long as the counter is greater than 50 , on every iteration the counter is divided by 2.
The counter should start with the value 900000 before the first iteration.

Ex 17:
Write a function that gets an array of strings as parameter and returns a new array containing all the values that appear more than once. In your solution use only while loops.

Ex 18:
Write a function that gets an array of strings as parameter and returns a new array containing all the values from the provided array in the same order but without any duplicated values. In your solution use only while loops.

For example:
names = ['Chris', 'Kevin', 'Naveed', 'Pete', 'Victor', ‘Chris’, ‘Kevin’]

Function output should be:
['Chris', 'Kevin', 'Naveed', 'Pete', 'Victor']

Ex 19:
Write a function that gets an array of strings as parameter and returns a new array containing all the values from the provided array in the same order but without any duplicated values.
If the string ‘pete’ is a value inside the array your function should skip it and not copy it to the new array. In your solution use only while loops.

For example:
names = ['Chris', 'Kevin', 'Naveed', 'Pete', 'Victor', ‘Chris’, ‘Kevin’]

Function output should be:
['Chris', 'Kevin', 'Naveed', 'Victor']

Ex 20:
Use a while loop to iterate on a boolean array.
As long as the next index is different from the previous index the iteration continues, otherwise, return the index of the element with the same value.
If there are not two successive values, the function will return -1.

For example:
array= [true, false, false, true, true, false] → return 2
array= [true, false, true, false, true, true]; → returns 5
array= [true, false, true, false, true, false]; → returns -1

Ex 21:
Write a python program that gets user input (use input() function for this).
The first input will be the user full name
Second input will be the user age
Third input will be the user email
Write validation for each input provided by the user and allow the user to try again in case the user provided invalid input.

Validation for full name input → string type with 2 words for first name and last name.
Validation for age input → int type between 1 - 130.
Validation for email input → string type with '@' inside.