1- $array = [2,4,3,1,6,7,5,8,0,9] .

Write a PHP program to sort this array to  ASC . (without using  built in functions)

2- use a previouse  array and sort it as  DESC . (without using  built in functions)

3- write a program to calculate the avg of numbers array.

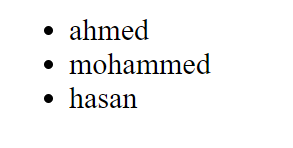
4- write a program to filter even numbers from an array, delete all even numbers.

5- write a program to find the max number of numbers group [10,30,20], using if.

6-  We have a numeric array for names:

$names = ['ahmed', 'mohammed', 'hasan'];

Write a script to display names in a dynamic HTML list:



7 -  We have a 2D array for instructors (outer is numeric, inner is associative):

$instructors = [

    'kareem fouad' => [

        'job' => 'backend developer',

        'track' => 'php',

    ],

    'ahmed bahnasy' => [

        'job' => 'frontend developer',

        'track' => 'angular',

    ],

    'ahmed nasr' => [

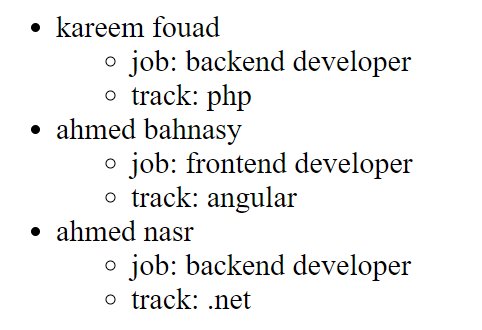
        'job' => 'backend developer',

        'track' => '.net',

    ],

];

Write a script to display instructors data in a dynamic HTML nested list:



8- **write a script add-product.php containing a form to add new product**

**form inputs:**

name: text

description: textarea

price: number

form is submitting a POST request to script handle-add-product.php

the handle file contains the following:

* validate the inputs by rules

name → required | string | min:5 | max:255

description → optional | string

price → required | number | min:0

* if not valid then display errors

else then print data (name - description if exists - price - price after discount using the function *getPriceWithDiscount* from question 2)

**Images Drive**

A simple drive that allows the user to upload/download images.

The exercise has two pages:

* Upload page: contains a form with two inputs (image file input - text input to set the name of the image if he wants to rename it to a specific name)
* Index page: displays all stored images, each image must have a download link.

**Hint:**

You may use the *scandir()* function to get all file names in your uploads folder.

<https://www.w3schools.com/php/func_directory_scandir.asp>

**Product Cart**

A simple cart that allows the user to add products to it and display his cart items.

The exercise has two pages:

* Products page: displays all products from an array of *$products.* Each product has (image - name - price) with an “add to cart” button.
* Cart page: display the names of the products that the user added to the cart.

Hint:

* We have no database yet, So put any dummy content in your *$products* array. Also, put images in the project folder to be displayed with product data.
* Store cart items in session under a key called ‘cart’. The value of the key ‘cart’ is an array of the selected products’ names.
* Use bootstrap cards on the products page.