

## COMP1002 – Advanced Python

### Lab3

#### Assignment 1

Imagine you are developing an inventory management system for a retail store. You have a dictionary where the keys represent the product names and the values represent the quantity of each product in stock.

For example:

```
inventory = {  
    'laptop': 50,  
    'headphones': 25,  
    'blender': 30,  
    'microwave': 40,  
    'desk lamps': 20  
}
```

Implement functionality to display the inventory in different ways based on user requirements:

##### a. Display by Product Name:

Sort the inventory dictionary by product names in alphabetical order and display it.

Sample Output:

Sorted Inventory: blender: 30 desk lamps: 20 headphones: 25 laptop: 50 microwave: 40
-----------------------------------------------------------------------------------------------------

##### b. Display by Quantity:

Sort the inventory dictionary by the quantity of products in ascending order and display it.

Sample Output:

Sorted Inventory (by Quantity in Ascending Order): desk lamps: 20 headphones: 25 blender: 30 microwave: 40 laptop: 50
--------------------------------------------------------------------------------------------------------------------------------------

##### c. Display Low Stock Items:

Filter the inventory to display only items with a quantity less than a certain threshold, say 30, and sort them by their quantities.

For threshold value 30, the sample output:

Low Stock Items: desk lamps: 20 headphones: 25
------------------------------------------------------

## Assignment 2

Write a Python function that takes two lists of numbers and returns a new list containing the result of pairwise comparisons of the elements from the given lists (e.g., greater than, less than or equal).

For example, the sample given lists are shown below.

```
list1 = [1, 3, 5, 6, 9]
list2 = [2, 3, 4, 10, 1]
```

The output is ['<', '=', '>', '<', '>']

## Assignment 3

Create three lists: **names**, **student\_ids** and **grades**, containing names, student ids and grades of students, respectively.

Use the `zip()` function to iterate over these lists simultaneously and print out each student's information in the format: "Name: <name>, Student Id: <student\_id>, Grade: <grade>".

Sample Output:

```
Name: Ali, Student Id: 12345, Grade: 75
Name: Ahmet, Student Id: 98765, Grade: 80
Name: Zeynep, Student Id: 54321, Grade: 78
```

## Assignment 4

Write a function **filter\_numbers(\*args, \*\*kwargs)** that takes any number of positional arguments and keyword arguments. The function should return a list containing only the positional arguments that are divisible by the value associated with the keyword argument "divisor". If "divisor" is not provided, default to 1.

For example,

#Function Call	#Output
<code>filter_numbers(1, 2, 3, 4, 5, 6, divisor=3)</code>	<code>[3, 6]</code>
<code>filter_numbers(10, 20, 30, 40, 50)</code>	<code>[10, 20, 30, 40, 50]</code>
<code>filter_numbers(3, 6, 9, 12, 15, divisor=5)</code>	<code>[15]</code>