**LAB # 01**

**“ OVERVIEW OF TENSORFLOW AND ITS OPERATIONS ”**

* **OBJECTIVE:**

• Understand the basics of deep learning and its applications.

• Set up the Python environment for deep learning.

• Perform basic operations using TensorFlow and Scikit-learn.

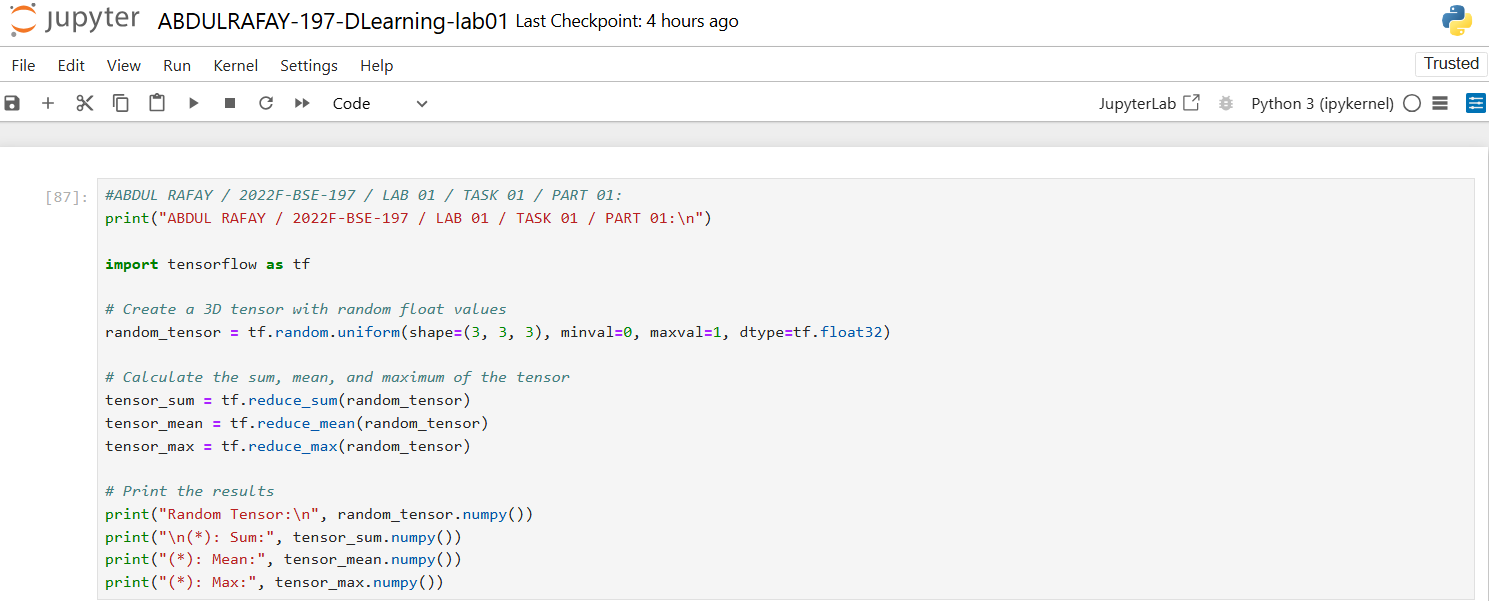
* **LAB TASKS:**

**TASK 1.**

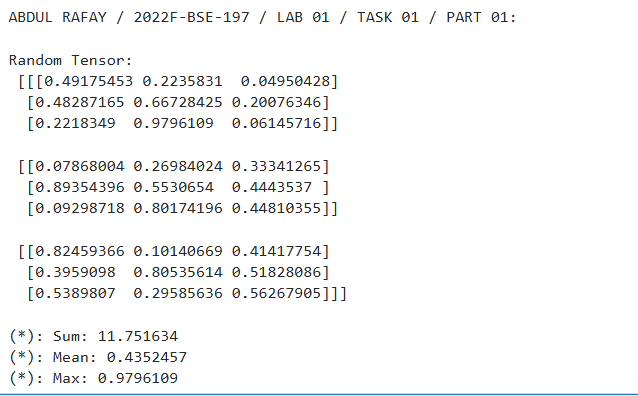
**(1):** Create a TensorFlow tensor with random float values. Calculate the sum, mean, and maximum of the tensor. **Hint:** Use tf.reduce\_sum(), tf.reduce\_mean(), tf.reduce\_max().

**(2):** Create a 3D TensorFlow tensor. Access and print a specific element, a 2D slice, and a 1D slice. **Hint:** Use standard Python indexing and slicing syntax)

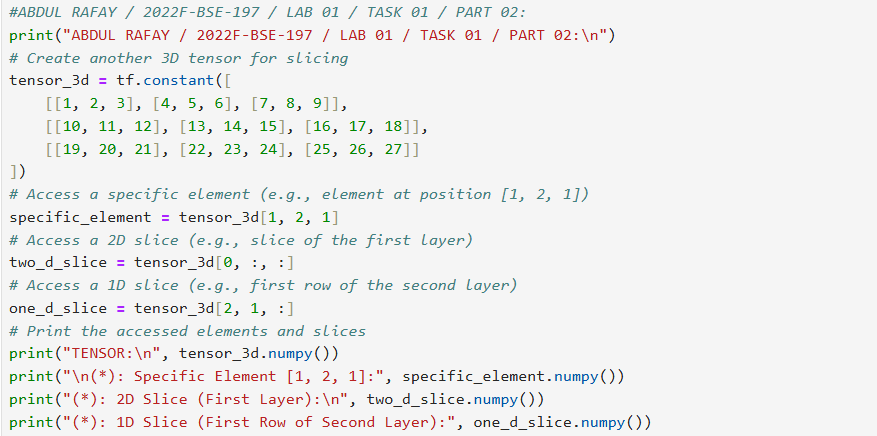
* **CODE (PART 1):**



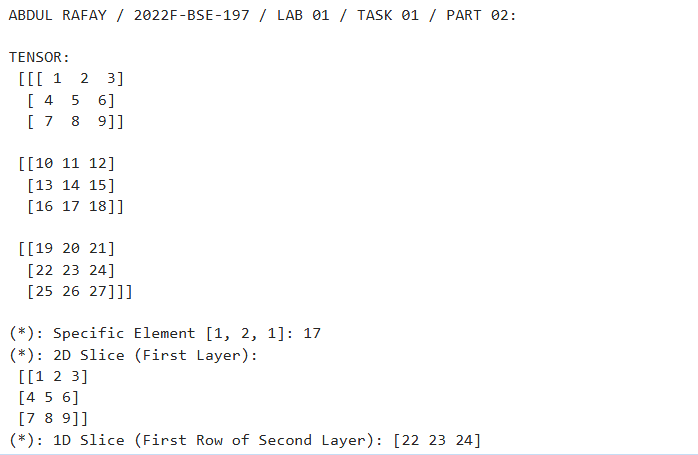
* **OUTPUT (PART 1):**



* **CODE (PART 2):**



* **OUTPUT (PART 2):**



**TASK 2.** Create a TensorFlow function called tensor operations that takes two tensors as input and performs the following arithmetic operations:

• Addition (tensor1 + tensor2)

• Subtraction (tensor1 - tensor2)

• Multiplication (tensor1 \* tensor2)

• Division (tensor1 / tensor2)

The function should return the results of these operations as a dictionary with the following keys:

• "addition"

• "subtraction"

• "multiplication"

• "division" etc

Test your function with the following tensors:

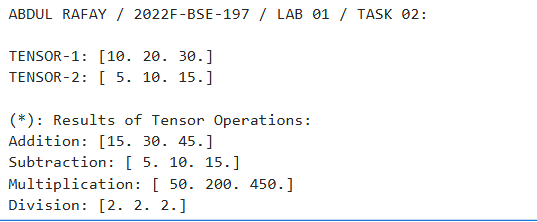
tensor1 = [10, 20, 30]

tensor2 = [5, 10, 15].

* **CODE:**



* **OUTPUT:**



**TASK 3.**

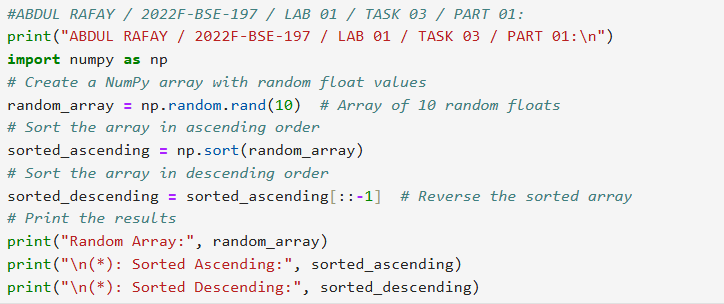
**(1):** Create a NumPy array with random float values. Sort the array in ascending and descending order. **Hint:** Use np.sort() and array slicing.

**(2):** Create a 2x2 NumPy matrix. Calculate its transpose and inverse.

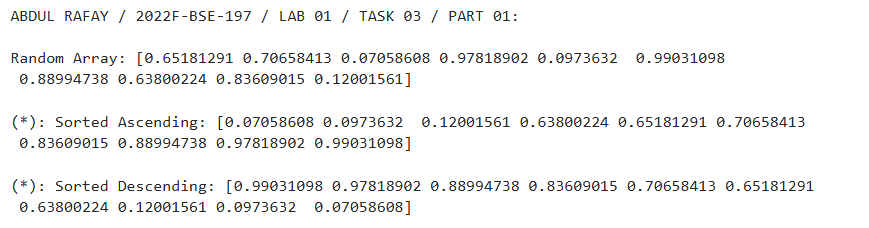
**Hint:** Use np.transpose() and np.linalg.inv().

**(3):** Create a 2D NumPy array. Calculate the sum of each row, the mean of each column, and the standard deviation of the entire array.

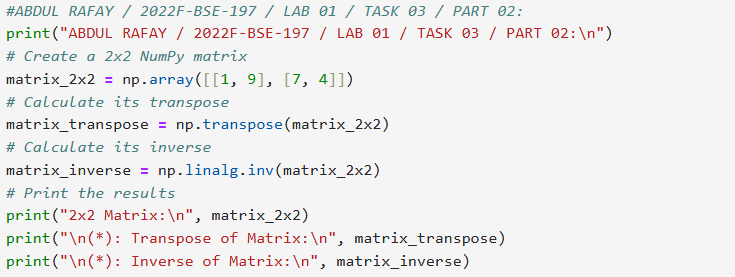
* **CODE (PART 1):**



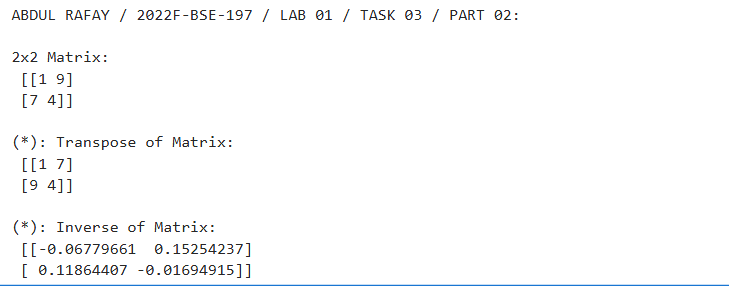
* **OUTPUT (PART 1):**



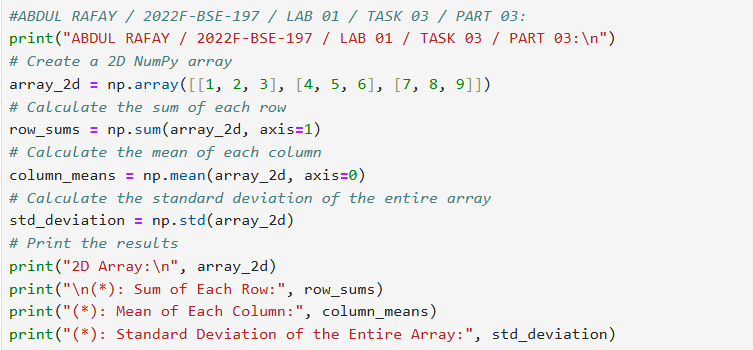
* **CODE (PART 2):**



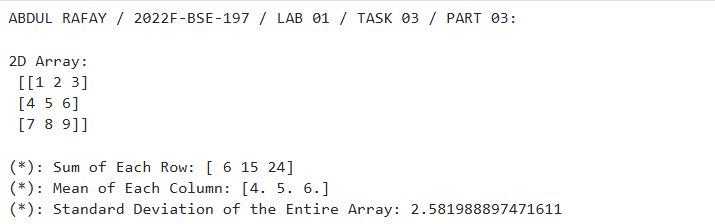
* **OUTPUT (PART 2):**



* **CODE (PART 3):**



* **OUTPUT (PART 3):**



* **GITHUB UPLOAD:**
* **KAGGLE UPLOAD:**