Lab: Python Lists and Their Operations

Lab Objectives

By the end of this lab, students will be able to:
- Create and manipulate Python lists.
- Apply indexing, slicing, and updating techniques.
- Use built-in list methods and functions.
- Implement list comprehensions and loops for list processing.

Part 1: List Creation and Access

Tasks:
1. Create a list named `subjects` with at least 5 course names.
2. Print the first and last elements using indexing.
3. Slice and print the middle three elements.

Part 2: Updating and Deleting Elements

Tasks:
1. Replace the second element in `subjects` with "AI".
2. Append "Data Science" to the list.
3. Remove "AI" using `remove()` and delete the last element using `del`.

Part 3: Built-in Functions

Tasks:
1. Create a list of numbers: `scores = [85, 90, 78, 92, 88]`.
2. Use `len()` to find the number of scores.
3. Use `sum()` to calculate the total score.
4. Find the average score.

Part 4: List Operations

Tasks:
1. Concatenate `scores` with another list `[75, 80]`.
2. Repeat the list `["Hi!"]` 3 times.
3. Check if `90` is in `scores`.

Part 5: List Methods

Tasks:

1. Use `pop()` to remove the third score and store it in a variable.
2. Use `count()` to find how many times `88` appears.
3. Use `insert()` to add `95` at index 2.
4. Use `sort()` to sort the scores in descending order.

Part 6: Iteration and List Comprehensions

Tasks:

1. Use a `for` loop to print each score with a message: "Score: X".
2. Create a new list using list comprehension that contains only scores above 85.
3. Create a list of squares from `[1, 2, 3, 4, 5]` using list comprehension.

Bonus Task: Nested List Comprehension

Create a list of tuples representing the cross product of `[1, 2]` and `['a', 'b']` using list comprehension.

Submission Requirements

- Submit a `.py` or `.ipynb` file, or screenshots showing the output for all completed tasks.
- Include comments explaining each step.
- Ensure code is clean and readable.