Class 4 Summary — Data Structures I: Lists & Tuples

Computational Thinking with Python
Conceptual lens: State → Transition → Invariant (SSTI)

1. Big Idea

Programs manage **state**. Data structures group related values into a single variable so we can reason clearly about changes (transitions) and guarantees (invariants).

2. Strings as a Starting Point

Strings are ordered collections of characters and already behave like data structures.

- **Indexing**: returns a single character (no new object)
- **Slicing**: creates a new string object
- Strings are immutable: methods never change the original string

3. Indexing vs Slicing (Core Rule)

This rule applies to strings, lists, and tuples without exception.

- Indexing **points** to a value
- Slicing **copies** and can be used to clone

4. Lists

Lists are ordered and **mutable**. List methods frequently cause state transitions.

- Support indexing and slicing
- Can grow and shrink
- Often used for changing collections (queues, stacks, carts)

Common List Methods

Method	Effect
append(x)	Add one item
extend(it)	Add many items
pop()	Remove & return last item
pop(i)	Remove & return item at index i
remove(x)	Remove first matching value
sort()	Reorder list in place
reverse()	Reverse list in place
len(list)	Observe length (no mutation)

5. Tuples

Tuples are ordered but **immutable**. Their contents cannot be changed after creation.

- Support indexing and slicing
- No mutating methods
- Strong invariants: fixed structure

6. Methods vs Functions

Methods belong to objects and use dot notation. Functions stand alone.

- Method: nums.append(4)
- Function: len(nums), sorted(nums)

7. SSTI Applied

- **State**: the entire list or tuple
- **Transition**: list methods that change contents
- **Invariant**: ordering, size expectations, immutability (tuples)

8. Key Takeaways

- Indexing points; slicing copies
- Lists change state; tuples protect state
- Methods often cause transitions; functions usually observe or create new objects

Next: Dictionaries & Sets → then Loops (repeated transitions with invariants).