Python Lists:

Mastering Sequence Data Types

by Xintong

What is a List?

- A list is a collection of items in a specific order.

List Basics

```
# Creating lists
empty_list = []
numbers = [1, 2, 3, 4, 5]
fruits = ["apple", "banana", "cherry"]
mixed = [1, "apple", True, 3.14]
```

- **Ordered** collection of items
- **Mutable** - can change after creation
- **Heterogeneous** - can contain different types
- **Zero-indexed** - first element at index 0

Accessing Elements

```
fruits = ["apple", "banana", "cherry"]

print(fruits[0])   # "apple"
print(fruits[-1])  # "cherry" (negative indexing)
print(fruits[1:3]) # ["banana", "cherry"] (slicing)

# Modifying elements
fruits[1] = "blueberry"
```

The result is:

```
['apple', 'blueberry', 'cherry']
```

Watch Out!

❗ Python will raise an `IndexError` if the index is out of range.

Common List Operations

```
colors = ["red", "green"]

# Add elements
colors.append("blue")        # Add to end
colors.insert(1, "yellow")  # Insert at index

# Remove elements
colors.remove("green")       # By value
popped = colors.pop(1)       # By index
```

Some Essential List Methods

```
# Create a list
lst = [3, 1, 2]
```

Method	Description	Example
`append()`	Add item to end	`lst.append(4)`
`insert()`	Insert at index	`lst.insert(0, 'a')`
`remove()`	Remove first match	`lst.remove(3)`
`index()`	Find first occurrence	`lst.index('a')`
`sort()`	Sort in-place	`lst.sort(reverse=True)`

Practice Exercises

```
lst = [1, 2, 3, 4, 5]
```

Do the following exercises one by one.

1. Basic: Reverse a list in-place
2. Intermediate: Insert an element "a" at index 2
3. Intermediate: Find the index of element "3"