

## LEC 03 - Object Oriented Programming

### Fundamental Transition:

- The study of computer science allows us to create technology and use technology
- Computer scientists design programs according to client specifications

Skills to learn:

- Design, implement, writing documentation, thorough testing
- 

### Classes and Objects:

What is a class?

- Abstract data structure that models a real world concept
- Describes the attributes and methods of the concept (called objects)
- Ex. `int`, `str`, `list`, or user defined classes such as `Tweet`, `User`, etc.

What is an object?

- Instance of a class
  - Everything in Python is an object
- 

### Example (Analyze the specification):

Attribute   Object   Method   Class

“The `Twitter` application allows users to broadcast short messages called `tweets`. A `tweet` includes the `message content` (of up to 280 characters), the `user` who wrote the `tweet`, when the `tweet` was `created`, and how many “`likes`” the `tweet` has. Once a `tweet` is `created`, it may be `liked` by other users. Furthermore, the tweet may be `edited` by its owner.”

---

### Design Roadmap:

**Define the class API according to the class design recipe:**

1. Class name and brief description (docstring)
2. Examples of client code (as doctests in the class docstring)
3. Public methods
  - Apply the function design recipe

**Implement the class:**

4. Public attributes
5. Internal (private) attributes
6. Representation invariants
7. Implement public methods

Jan 13, 2025

### **LEC 03 - Object Oriented Programming**

- Privateness and representation invariants will be revisited later, but keep them in mind

Reminders:

- Read carefully what the question asks
  - Formulate answers on your own
  - Discuss with neighbors
  - Ask questions along the way
- 

#### **Rebinding self:**

- Rebinding `self` in a class does not mutate anything