Topic 55 - Classes: A Bit of Housekeeping

**What**

* When creating a class in Python, some basic setup (or "housekeeping") is needed to properly assign attributes to instances of the class.
* This example shows how to set up an attribute within the Patient class using self.last\_name = last\_name to link an instance’s attribute to the input provided during creation.

**Why**

* **Attribute Assignment**: Using self.attribute\_name = attribute\_value links the instance's attribute (self.last\_name) to the provided input (last\_name), ensuring each instance has its own data.
* **Clarity and Consistency**: Duplicating attribute names (e.g., using self.last\_name = last\_name) clarifies the attribute’s purpose, making the code more readable and maintainable.
* **Foundation for OOP**: This process is crucial for Object-Oriented Programming, where each instance of a class needs individualized attributes.

**How**

1. **Creating the Class with Attribute Assignment**  
   Here’s how to set up the Patient class with an assigned attribute for each instance.

python

Copy code

class Patient:

# Initialization method to define attributes

def \_\_init\_\_(self, last\_name):

self.last\_name = last\_name # Assigns the last\_name attribute to each instance

* + **Class Definition**: The class starts with the class keyword, followed by the class name (Patient) and a colon.
  + **Constructor (\_\_init\_\_)**: The \_\_init\_\_ method takes last\_name as a parameter.
  + **Attribute Assignment with self**: self.last\_name = last\_name assigns the instance's last\_name attribute to the provided input, allowing each Patient object to hold its own unique last\_name value.

1. **Creating an Instance and Accessing Attributes**  
   After defining the class, you can create an instance of Patient and access its last\_name.

python

Copy code

# Creating an instance of Patient

patient1 = Patient("Smith")

print(patient1.last\_name) # Output: Smith

* + **Instantiation**: patient1 is an instance of Patient, with last\_name set to "Smith".
  + **Accessing Attributes**: patient1.last\_name retrieves the last\_name attribute for patient1.

**Things to Remember**

* **Attribute Naming**: Using the same name (e.g., last\_name) for both the parameter and the instance attribute is common and enhances readability.
* **Indentation**: Each line within a class method (like \_\_init\_\_) is indented one level.
* **Using self**: self differentiates instance-specific data from other variables, ensuring each instance has its own attributes.