# **Python OOP Starter Guide:**

## **"Why OOP When Procedural Works?" + Basics of Class and Object**

## **💭 Why OOP When Procedural Works in Python?**

* ✅ Procedural works well for **small programs**
* ❌ But for **large, real-world apps**, procedural gets:  
  + Hard to **maintain**
  + Difficult to **reuse code**
  + Messy when handling **multiple data types**
* ✅ OOP helps you:  
  + **Model real-world objects** (like Student, Order, Vehicle)
  + Combine **data + logic** into one structure
  + **Reuse** code using inheritance and **organize** it better

## **🏗️ What is a Class?**

A **class** is a **blueprint or template** to create objects.

**Example:** A class called Car defines what all cars have (like model, brand) and do (like drive, brake).

## **🧱 What is an Object?**

An **object** is a **real instance** created using a class.

**Example:** "Toyota Innova" is an object created using the Car class.

## **❓ Why Object Creation is Required?**

* Objects allow you to **store real data** using a class structure.
* You can create **multiple versions** (objects) from the same class — each with its own values.
* Without creating an object, the class is just a design — you can’t use its functions or variables.

## **⚡ Function vs Method – One-liner**

**Function** is defined outside class.  
 **Method** is a function **inside a class** that works with an object.

## **🧪 Simple Example: Class & Object**

class Student:

def say\_hello(self):

print("Hi, I'm a student!")

s1 = Student() #

s1.say\_hello() #s1.say\_hello(s1)

Multiple Object for single class

class Student:

def \_\_init\_\_(self, name, grade):

self.name = name

self.grade = grade

def display(self):

print(f"{self.name} is in grade {self.grade}")

# 🎯 Multiple objects

s1 = Student("Gowtham", 10)

s2 = Student("nandini", 12)

s3 = Student("nila", 11)

s1.display() # Gowtham is in grade 10

s2.display() # Priya is in grade 12

s3.display() # Kiran is in grade 11

## **✅ When is Multiple Object Creation Needed?**

| **Use Case** | **Example** |
| --- | --- |
| Modeling multiple real things | Students, Cars, Products, Employees |
| Storing different user data | Login sessions, users, profiles |
| Game development | Each character, enemy, or item as an object |
| Data analysis tools | Each dataset or row becomes an object |
| Chatbots / messages | Each message/user interaction as an object |

## **❌ What if you don’t use multiple objects?**

* You'd overwrite the data every time
* Code becomes rigid, non-scalable
* Can’t maintain individual records

## **✅ Q2: *“Does object creation mean memory is allocated?”***

**Yes, absolutely!** When you write something like:

u1 = User("Gowtham", 33)

* Python creates a **new memory space** for this object
* Stores variables like name, age inside that memory
* You can check this with id(u1)

u1 = User("Gowtham", 33)

u2 = User("Priya", 28)

print(id(u1)) # 👉 Unique memory address

print(id(u2)) # 👉 Different address

Each object = separate space in memory

So you can store, change, and track them **independently**

### **About the Author**

**Gowtham SB** is a **Data Engineering expert, educator,** **and content creator** with a passion for **big data technologies, as well as cloud and Gen AI** . With years of experience in the field, he has worked extensively with **cloud platforms, distributed systems, and data pipelines**, helping professionals and aspiring engineers master the art of data engineering.

Beyond his technical expertise, Gowtham is a **renowned mentor and speaker**, sharing his insights through engaging content on **YouTube and LinkedIn**. He has built one of the **largest Tamil Data Engineering communities**, guiding thousands of learners to excel in their careers.

Through his deep industry knowledge and hands-on approach, Gowtham continues to **bridge the gap between learning and real-world implementation**, empowering individuals to build **scalable, high-performance data solutions**.

𝐒𝐨𝐜𝐢𝐚𝐥𝐬

🎥𝐘𝐨𝐮𝐓𝐮𝐛𝐞 - https://www.youtube.com/@dataengineeringvideos

📸𝐈𝐧𝐬𝐭𝐚𝐠𝐫𝐚𝐦 - <https://instagram.com/dataengineeringtamil>

📸𝐈𝐧𝐬𝐭𝐚𝐠𝐫𝐚𝐦 - [https://instagram.com/](https://instagram.com/dataengineeringtamil)thedatatech.in

🤝𝐂𝐨𝐧𝐧𝐞𝐜𝐭 𝐟𝐨𝐫 𝟏:𝟏 - https://topmate.io/dataengineering/

💼𝐋𝐢𝐧𝐤𝐞𝐝𝐈𝐧 - https://www.linkedin.com/in/sbgowtham/

🌐𝐖𝐞𝐛𝐬𝐢𝐭𝐞 - https://codewithgowtham.blogspot.com

💻𝐆𝐢𝐭𝐇𝐮𝐛 - http://github.com/Gowthamdataengineer

💬𝐖𝐡𝐚𝐭𝐬 𝐀𝐩𝐩 - https://lnkd.in/g5JrHw8q

📧𝐄𝐦𝐚𝐢𝐥 - atozknowledge.com@gmail.com

📱𝐀𝐥𝐥 𝐌𝐲 𝐒𝐨𝐜𝐢𝐚𝐥𝐬 - <https://lnkd.in/gf8k3aCH>