Tab 1

Gowtham SB
www.linkedin.com/in/sbgowtham/ Instagram - @dataengineeringtamil

🧠 Python Guide: Higher-Order Functions (HOF)

✅ What is a Higher-Order Function?

A function is called a **Higher-Order Function (HOF)** if it:

1. **Takes one or more functions as arguments**, or
2. **Returns a function as its result**

✅ In Python, functions are **first-class citizens**, so they can be:

- Passed to other functions
- Returned from functions
- Assigned to variables

🔄 Types of HOF

Type	Definition	Keyword Feature
Type 1	Takes function(s) as argument(s)	`func(arg_func)`
Type 2	Returns a function (nested function)	`return inner_func`

🔧 Type 1: Function as Argument

📌 Example:

```
def shout(text):
    return text.upper()

def whisper(text):
    return text.lower()

def speak(func, message):
    return func(message)

print(speak(shout, "Gowtham"))   # GOWTHAM
print(speak(whisper, "Gowtham")) # gowtham
```

☑️ When to Use:

- You want to pass different **behaviors** (functions) into another function
- Common in `map()`, `filter()`, `sorted(key=...)`, etc.

📐 Type 2: Function that Returns Another Function

📌 Example:

```
def greeting_builder(prefix):
    def greet(name):
        return f"{prefix}, {name}!"
    return greet

tamil_greet = greeting_builder("Vanakkam")
print(tamil_greet("Gowtham"))  # Vanakkam, Gowtham!
```

☑️ When to Use:

- You want to create a **customized function generator**

- Used in **decorators**, **closures**, and **dynamic behavior creation**

🔥 Real-World Use Cases

Use Case	HOF Type	Description
SQL Query Builder	Type 2	Return a query builder function for specific table/condition
Caption Stylizer	Type 1	Pass different stylizing functions (`uppercase`, `emoji`)
Email Generator	Type 2	Create `gmail()`, `ymail()` style functions from domain input
Sorting	Type 1	Pass function as `key` in `sorted()`
Decorators	Type 2	Python decorators are built using returning functions

---

🧪 Closure Concept (Important for Type 2)

A **closure** is when an inner function remembers the variables from its outer function
Even after the outer function is done executing

```
def outer(msg):
    def inner():
        return f"Message is: {msg}"
    return inner

say_hi = outer("Vanakkam da mapla")
print(say_hi())  # Message is: Vanakkam da mapla
```

🔁 Comparison of All 3 Styles

Feature	Normal Function	HOF Type 1	HOF Type 2
Pass logic dynamically	❌ No	✅ Yes	✅ Yes
Return logic dynamically	❌ No	❌ No	✅ Yes
Useful for plugins/patterns	❌ Limited	✅ Good	✅✅ Best
Reuse with preset config	❌ Manual	⚠️ Needs args	✅ Clean
Supports Closures	❌ No	❌ No	✅ Yes

🛠️ Built-in Python HOFs

- `map(function, iterable)`
- `filter(function, iterable)`
- `sorted(iterable, key=function)`
- `functools.reduce(function, iterable)`
- `functools.partial` (for partial application)

🏁 Final Tips

✅ Use **HOF Type 1** when:

- You want to plug in **existing logic**
- You're dealing with **strategies**, **filters**, or **formatters**

✅ Use **HOF Type 2** when:

- You want to **generate logic on-the-fly**
- You want to **bind values early and reuse the logic later**
- You're writing **decorators**, **factory functions**, or **plugins**

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/thedatatech.in

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

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

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

Gowtham SB
www.linkedin.com/in/sbgowtham/ Instagram - @dataengineeringtamil

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

💬 **Whats App** - https://lnkd.in/g5JrHw8q

📧 **Email** - atozknowledge.com@gmail.com

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