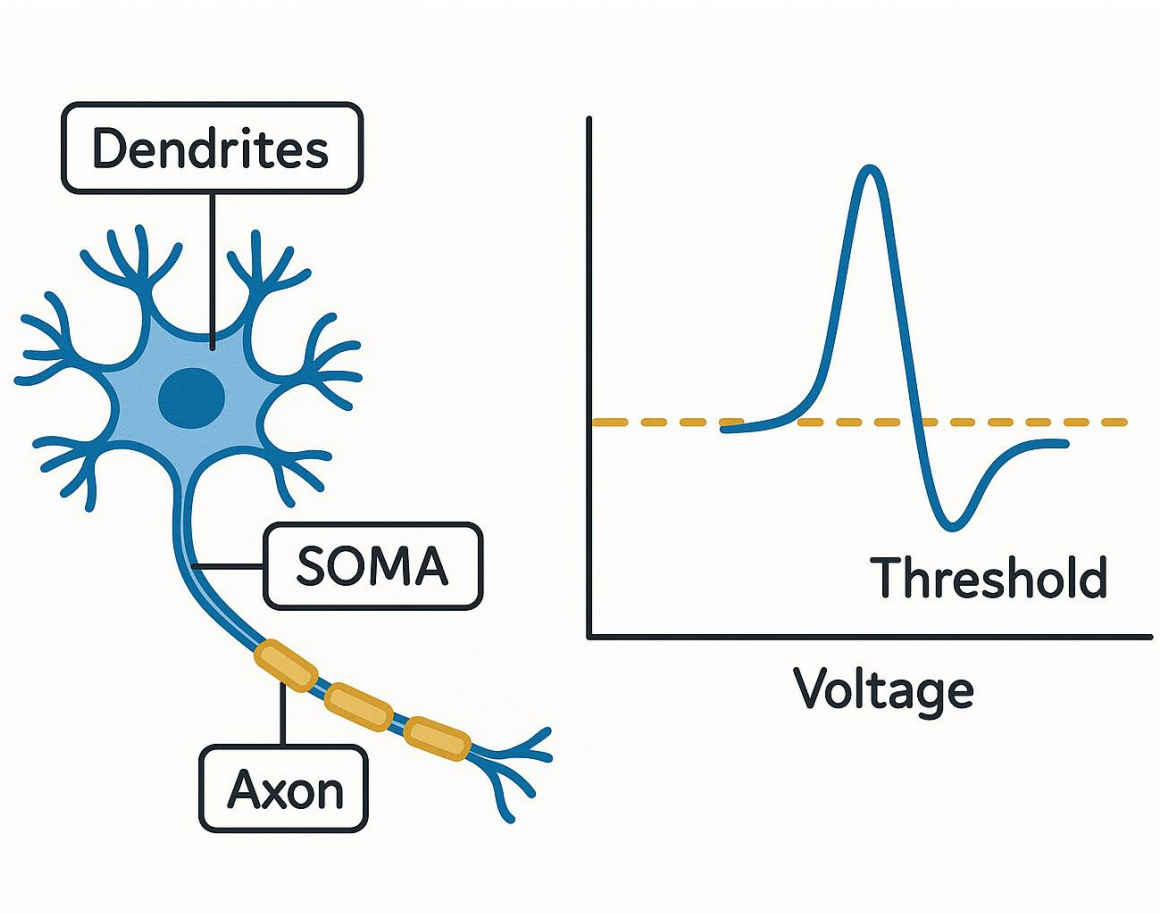
# From Python Programming Basics to Neuron Simulation

Dr. Sahil Sharma, PhD

Postdoctoral Researcher

ISRC, Ulster University

# Why this is cool?




**LIF: Leaky Integrate and Fire**

## 🔌 Biological View → Electrical Analogy

•Neuron **receives input** via dendrites.
•**Soma integrates** the incoming signals over time.
•If input is strong enough, the neuron **fires** (spike sent through axon).

### ⚡ LIF Model Abstraction

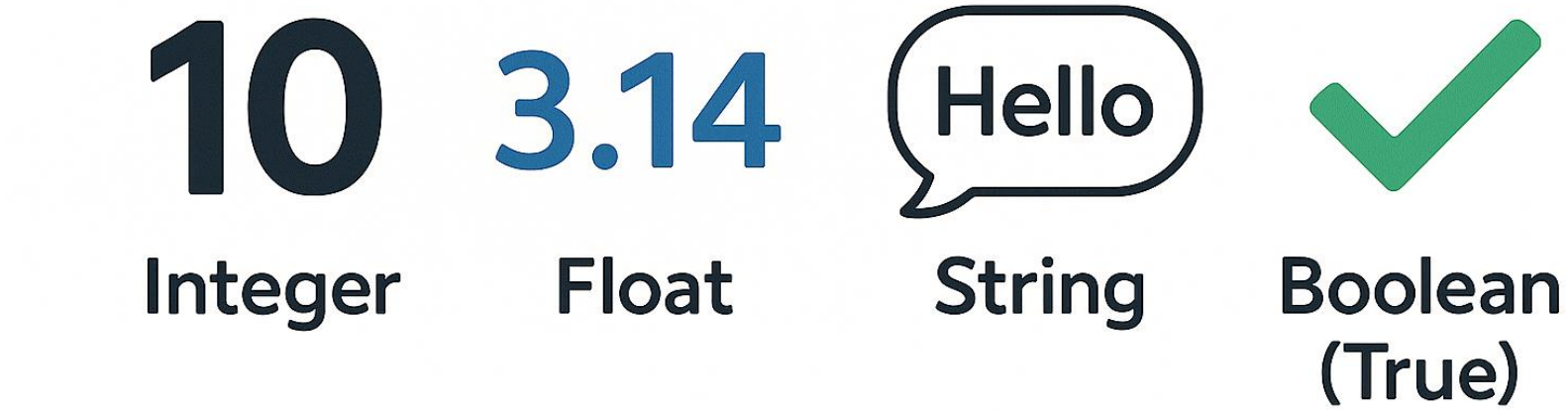
•The neuron acts like a **leaky capacitor**:
- **Capacitor (C)** stores voltage (membrane potential).
- **Resistor (gL)** leaks voltage over time.

### 📈 Voltage Behavior (Right Graph)

•Input causes **voltage to rise**.
•If voltage crosses the **Threshold**:
- A **spike is generated**.
- Voltage **resets** (back to baseline).

# Core Python Datatypes

| Type | Example | Description |
|---|---|---|
| **int** | 10 | Whole numbers, positive or negative |
| **float** | 3.14 | Decimal numbers |
| **bool** | True, False | Boolean values |
| **str** | "Hello" | Text (string of characters) |



# Python Collections (Container Types)

| Type | Example | Description |
| --- | --- | --- |
| **list** | [1, 2, 3] | Ordered, mutable sequence |
| **tuple** | (1, 2, 3) | Ordered, immutable sequence |
| **dict** | {"key": "value"} | Key-value pairs |
| **set** | {1, 2, 3} | Unordered, unique elements |

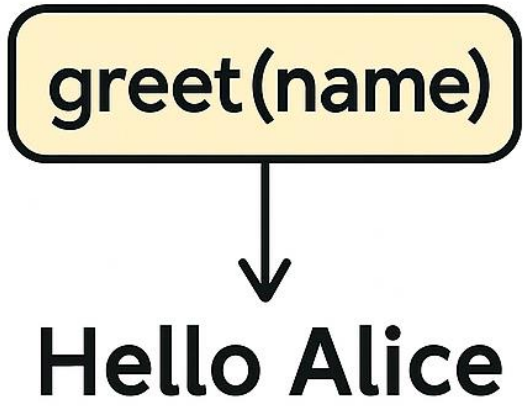
# Making Decisions with Python

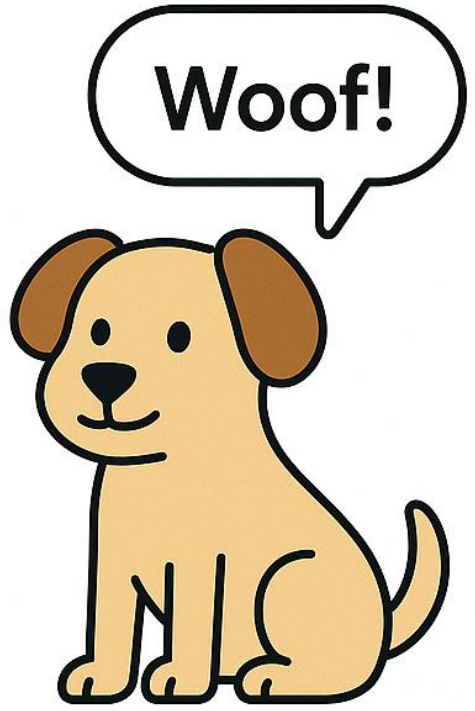
- **Conditionals:**
  ```
  if age < 18:
      print("Minor")
  elif age == 18:
      print("Exactly 18!")
  else:
      print("Adult")
  ```
- **Loops:** repeat actions easily
  ```
  for i in range(5):
      print(i)
  ```

# Functions and Classes

- **Function example:**
  ```
  def greet(name):
      return "Hello " + name
  ```

- **Class example:**
  ```
  class Dog:
      def bark(self):
          return "Woof!"
  ```




*Functions and classes help us model more complex systems → like a neuron.*

# NumPy and Matplotlib

- **NumPy** = fast math with arrays

import numpy as np
np.mean([1,2,3,4,5]) → 3.0

- **Matplotlib** = easy plots

    - Example: sine wave plot

# *Leaky Integrate-and-Fire Neuron*

The **Leaky Integrate-and-Fire neuron** is a simple model used to explain how neurons behave electrically. It helps us understand how neurons "decide" when to send a signal (a spike).

## The RC Circuit Analogy

Imagine a neuron like an **electrical circuit**:

🔋 **Battery/Input Current**
•This represents the electrical input (like a stimulus).

🧱 **Capacitor (C) – *Stores charge***
•Think of this as the **membrane of the neuron** that stores voltage (membrane potential).

🔩 **Resistor (gL) – *Leak conductance***
•This represents how the **membrane leaks charge** over time. It's why it's called a “leaky” neuron.

⟶ Together, this RC (Resistor-Capacitor) circuit builds up voltage as input current comes in, but also slowly leaks it.

# Voltage Over Time

Imagine watching the **voltage inside the neuron** over time:

- The curve rises as input charges the capacitor.
- Once the voltage crosses a **threshold** (dashed line), the neuron **"fires"** — a sharp spike occurs (the neuron sends a signal).
- Then it **resets** back to a low value (baseline) — ready for the next input.

**Summary: What Happens?**

1. **Input comes in** → capacitor charges → voltage rises.
2. **Voltage leaks** slowly due to resistor.
3. If voltage **crosses the threshold**, the neuron **fires a spike**.
4. Then, it **resets** and starts again.

# Python Code for
# *Leaky Integrate-and-Fire Neuron*