Chapter 1: Introduction to IQ Data and Signal Fundamentals

Objective

- Understand what IQ data is and why it matters for signal analysis.
- Learn the mathematical and visual representation of IQ signals.
- Visualize simple IQ data in Python to build intuition.

Scope

- Introduce I (In-phase) and Q (Quadrature) components.
- Explain complex representation of signals.
- Phasor visualization and concept of magnitude & phase.
- Simple plotting using Python.

Fundamentals Involved

1. **IQ Representation**:
   - Any bandpass signal can be represented as **two orthogonal components**:

$$s(t) = I(t) \cdot \cos(2\pi f_c t) - Q(t) \cdot \sin(2\pi f_c t)$$

   - In Python, `I` and `Q` are usually stored as float32 or int16 arrays.
2. **Complex Form**:

$$x(t) = I(t) + jQ(t)$$

   - Magnitude: $|x(t)| = \sqrt{I^2 + Q^2}$
   - Phase: $\phi(t) = \arctan 2(Q, I)$
3. **Phasor Concept**:
   - Visualizes IQ as points rotating in the complex plane.
   - Useful to see modulation, phase shifts, or signal irregularities.

Step 1: Simulate IQ Data

For learning, we can start with synthetic IQ data.

```python
import numpy as np
import matplotlib.pyplot as plt

# Parameters
fs = 1e3  # Sampling frequency, Hz
t = np.arange(0, 1, 1/fs)  # 1 second of samples
f_signal = 50  # Signal frequency in Hz

# Simulate IQ components
I = np.cos(2 * np.pi * f_signal * t)  # In-phase
Q = np.sin(2 * np.pi * f_signal * t)  # Quadrature

# Complex representation
IQ = I + 1j*Q

# Magnitude and Phase
magnitude = np.abs(IQ)
phase = np.angle(IQ)

# Plot I & Q
plt.figure(figsize=(12,4))
plt.plot(t, I, label='I')
plt.plot(t, Q, label='Q')
plt.title('I and Q Components')
plt.xlabel('Time (s)')
plt.ylabel('Amplitude')
plt.legend()
plt.grid(True)
plt.show()

# Plot Phasor (Complex plane)
plt.figure(figsize=(6,6))
plt.plot(I, Q)
plt.title('Phasor Plot (I vs Q)')
plt.xlabel('I')
plt.ylabel('Q')
plt.axis('equal')
plt.grid(True)
plt.show()

# Plot Magnitude and Phase
plt.figure(figsize=(12,4))
plt.subplot(2,1,1)
plt.plot(t, magnitude)
```

```
plt.title('Magnitude of IQ Signal')
plt.xlabel('Time (s)')
plt.ylabel('Magnitude')
plt.grid(True)

plt.subplot(2,1,2)
plt.plot(t, phase)
plt.title('Phase of IQ Signal')
plt.xlabel('Time (s)')
plt.ylabel('Phase (radians)')
plt.grid(True)
plt.tight_layout()
plt.show()
```

Exercise

1. Change `f_signal` to different frequencies (100 Hz, 200 Hz, etc.) and observe the phasor rotation.
2. Modify the amplitude of I and Q to be unequal; notice how the phasor shape changes.
3. Add a small random noise to I and Q and visualize how it affects magnitude and phase.

> Goal: Build **intuition for real IQ signals** before moving to captured drone or RF signals.