### **Lab 1: Decoding Sensor Data Using Python Abstract Base Classes (ABCs)**

**Exercise: Decoding Sensor Data using Python Abstract Class**

Creating an advanced lab exercise on abstract base classes (ABCs) in Python with a focus on sensor data is a great way to teach students about modeling real-world scenarios and using abstraction in their code. In this exercise, we will design a sensor data processing system using abstract base classes to handle various types of sensors and data.

**Exercise Description:**

Imagine you are developing a sensor data processing system for a smart home. The system should be able to handle different types of sensors, such as temperature sensors, humidity sensors, and motion sensors. To ensure a consistent interface for all sensor types, we decide to use abstract base classes.

**Instructions:**

1. **Create an Abstract Base Class (ABC):** Start by creating an abstract base class called Sensor with the following abstract methods:
   1. **read\_data(self):** An abstract method that simulates reading data from the sensor. This method should return a dictionary with sensor-specific data (e.g., temperature, humidity, motion status).
2. **Create Concrete Classes:** Implement three concrete classes that inherit from the Sensor class:

* **TemperatureSensor:** Represents a temperature sensor. It should generate random temperature readings in Celsius.
* **HumiditySensor:** Represents a humidity sensor. It should generate random humidity readings as a percentage.
* MotionSensor: Represents a motion sensor. It should randomly detect motion (True or False).

1. **Implement Abstract Method:** In each of the concrete classes, implement the read\_data method to generate sensor-specific data according to the rules for each type of sensor.
2. **Test the Classes:** Create instances of each of the concrete classes and demonstrate that the read\_data methods work as expected. Display the generated sensor data.
3. **Polymorphism:** Demonstrate polymorphism by creating a list of various sensors (instances of the concrete classes) and reading data from them in a loop.

This exercise helps participants practice abstract base classes, inheritance, method overriding, and polymorphism while simulating real-world sensor data processing.