

## Environmental monitoring

### Development part:

Here's a simplified Python code example for air and water quality monitoring in an IoT project. This example uses a hypothetical "sensor" module for demonstration purposes.

```
```python
import time
import random # Simulate sensor readings
from my_air_quality_sensor import AirQualitySensor
from my_water_quality_sensor import WaterQualitySensor
from my_iot_platform import IoTPlatform

# Initialize your sensors (replace with actual sensor initialization)
air_sensor = AirQualitySensor()
water_sensor = WaterQualitySensor()

# Initialize your IoT platform (replace with your IoT platform setup)
iot = IoTPlatform()

while True:
    # Read air and water quality data (replace with actual sensor readings)
    air_quality = air_sensor.read_quality()
    water_quality = water_sensor.read_quality()

    # Send data to the IoT platform
    iot.send_data(air_quality, water_quality)

    # Add some delay between readings
    time.sleep(60) # Sample data every minute
...
```
```

In this code:

1. Import necessary libraries and modules. You should replace `my\_air\_quality\_sensor`, `my\_water\_quality\_sensor`, and `my\_iot\_platform` with the actual libraries and platform you are using.
2. Initialize your air quality and water quality sensors. Replace `AirQualitySensor` and `WaterQualitySensor` with your actual sensor initialization code.
3. Initialize your IoT platform. Replace `IoTPlatform` with the code required to set up your IoT platform, like AWS IoT, Azure IoT, or any other platform you're using.

4. In the main loop, read air and water quality data from your sensors. Replace ``read_quality()`` with the actual method used to read data from your sensors.

5. Send the collected data to your IoT platform using the ``send_data()`` method. You need to replace this with the actual code to send data to your chosen platform.

6. Add a delay between readings to control the frequency of data collection. In this example, data is collected every 60 seconds, but you can adjust this to your project's requirements.

Remember to adapt this code to the specific sensors, hardware, and IoT platform you're using in your project.