Develop a Python script on the IoT devices to send real-time environmental data to the monitoring platform.

Certainly, here's a basic Python script that you can use on an IoT device to send real-time environmental data to a monitoring platform. This example assumes you are using a Raspberry Pi with a sensor to collect environmental data (e.g., temperature and humidity) and send it to a remote server. You'll need to adapt it to your specific IoT device and sensor.

```
import time
import requests

# Configure your IoT device to collect environmental data
def collect_environmental_data():
    # Replace this with actual data collection code, e.g., using sensors
    temperature = 25.5
    humidity = 50.0
    return temperature, humidity

# URL of the monitoring platform to send data
monitoring_platform_url= "https://your-monitoring-platform.com/api/data"

# Main loop to continuously collect and send data
while True:

    try:
 temperature, humidity = collect_environmental_data()

        # Create a JSON payload with the collected data
        data = {
            "temperature": temperature,
            "humidity": humidity
        }

        # Send the data to the monitoring platform
        response = requests.post(monitoring_platform_url, json=data)

        if response.status_code == 200:

            print("Data sent successfully")

        else:
```

```
        print(f"Failed to send data. Status code: {response.status_code}")

        # Wait for a specific interval (e.g., 5 minutes) before sending the next data
        time.sleep(300)

    except Exception as e:
        print(f"An error occurred: {str(e)}")
```

Remember to replace **"https://your-monitoring-platform.com/api/data"** with the actual URL of your monitoring platform's API. Additionally, you'll need to implement the

collect_environmental_data() function to collect data from your IoT sensors.

This script collects data and sends it to the monitoring platform in a continuous loop. Make sure to handle any exceptions that may occur during data collection or transmission and add any additional security measures required for your specific IoT device and platform.