

## Bài thí nghiệm 3

### GIAO TIẾP HTTP API TRONG PHẦN MỀM NHÚNG

#### Mục đích

- Nắm được kỹ thuật giao tiếp HTTP REST API sử dụng cho phần mềm nhúng để giao tiếp trao đổi dữ liệu với server. Viết chương trình minh họa trên Raspberry Pi 4, giao tiếp API từ server Thingspeak

#### Chuẩn bị

- Máy tính PC dùng Windows hoặc Ubuntu có các công cụ cần thiết: Visual Studio Code, SSH (xem hướng dẫn trong bài mở đầu)
- KIT Raspberry Pi 4

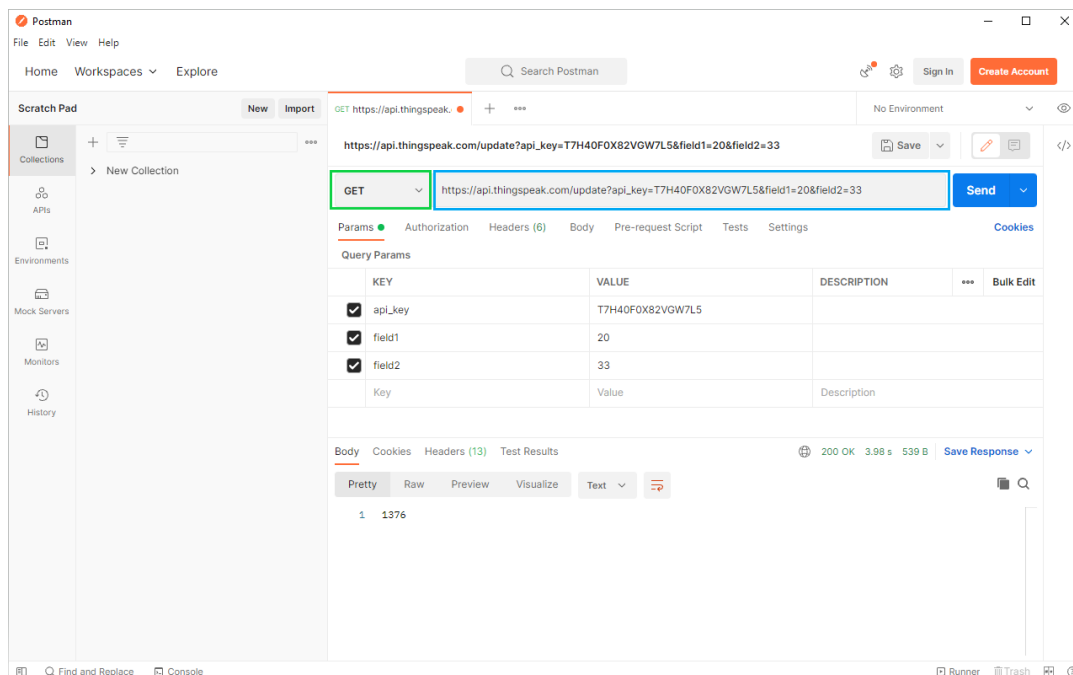
#### Tiến hành thí nghiệm

##### Bài 1. Gửi dữ liệu lên server Thingspeak

a) **Gửi dữ liệu lên Thingspeak qua API:** Dữ liệu gồm 2 trường field1 (Temperature) và field2 (Humidity) được đóng gói trong URL (urlencoded) như sau:

```
GET https://api.thingspeak.com/update?api_key=T7H40FOX82VGW7L5&field1=20&field2=33
```

- Kiểm thử API bằng Postman
  - Nhập url ở trên vào thanh địa chỉ và chọn kiểu phương thức HTTP request. Sau đó, chọn Send để gửi yêu cầu và chờ đợi kết quả.



- Như trong ảnh, kết quả nhận được gồm mã trả về (200 OK) và body của bản tin phản hồi (1376 – tức số thứ tự của gói tin gửi lên sau khi gửi yêu cầu kể trên)
- Viết chương trình chạy bằng Java để thực hiện lại công việc trên. Code chương trình mẫu:

File: **bth1a.java**

```
import java.io.*;
import java.net.URL;
import java.net.HttpURLConnection;

public class Main {
    /* Define Key API and URL of Thingspeak cloud */
    private static final String keyApi = "T7H40F0X82VGW7L5";
    private static final String thinkSpeakUrl = "https://api.thingspeak.com";

    /* Send field1 and field2 to cloud using URL */
    public static void updateDataUsingUrl(String urlStr) {
        URL url;
        String line;
        int responseCode;
        BufferedReader reader;
        HttpURLConnection conn = null;
        StringBuilder responseBody = new StringBuilder();
        try {
            url = new URL(urlStr);
            conn = (HttpURLConnection) url.openConnection();
            conn.setRequestMethod("GET");

            /* Get response code from connection */
            responseCode = conn.getResponseCode();
            System.out.println("GET Response Code : " + responseCode);
            if (responseCode == HttpURLConnection.HTTP_OK) {
                reader = new BufferedReader(new
InputStreamReader(conn.getInputStream()));
                while ((line = reader.readLine()) != null) {
                    responseBody.append(line);
                }
                System.out.println(responseBody.toString());
                reader.close();
            }
            conn.disconnect();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
public static void main(String args[]) {  
    int field1 = 15;  
    int field2 = 01;  
    //TODO: Hoàn thiện url của request  
    String url = thinkSpeakUrl + "/update?api_key=" + ....  
  
    updateDataUsingUrl(url);  
}  
}
```

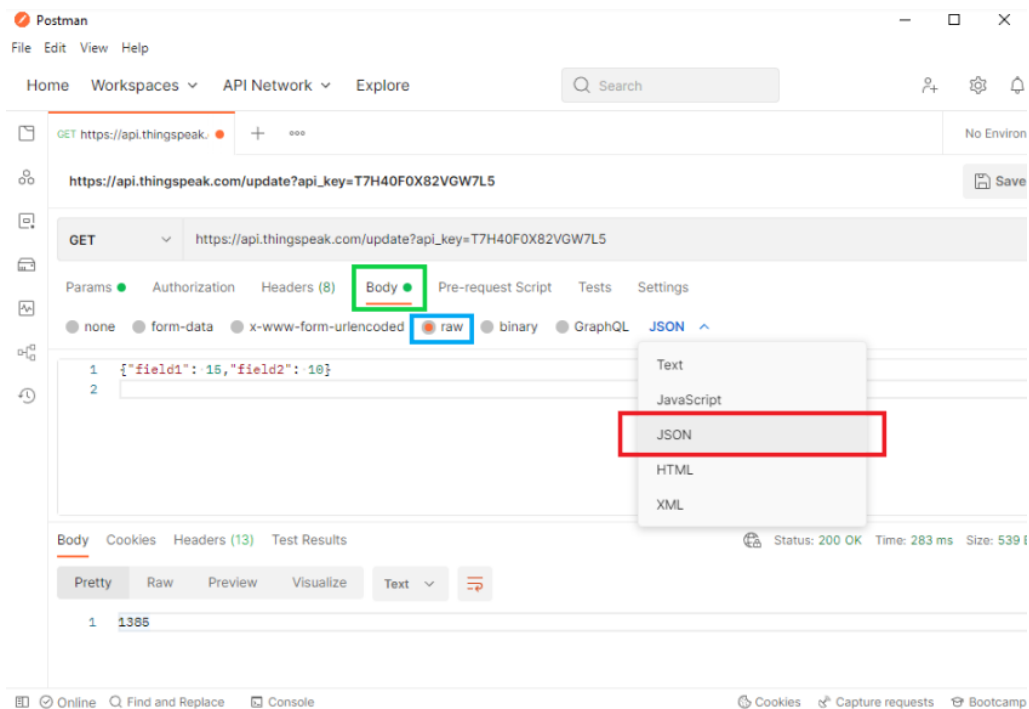
**b) Gửi dữ liệu lên Thinkspeak qua API:** Dữ liệu gồm 2 trường field1, field2 được đóng gói trong request body bằng Json.

```
GET https://api.thingspeak.com/update?api_key=T7H40F0X82VGW7L5
```

Ví dụ: body request

```
{  
    "field1": 20,  
    "field2": 33  
}
```

- Kiểm thử API bằng Postman:
  - Nhập URL và chọn phương thức HTTP request tương tự như trên
  - Chọn mục Body → raw → Json để nhập gói json vào body bản tin request
  - Sau đó nhấn Send và nhận kết quả trả về là thứ tự của bản tin cuối cùng đẩy lên server



- Viết chương trình thực hiện công việc trên bằng Java. Ngoài ra, phần bài tập này yêu cầu sử dụng thư viện Json để thao tác với các đối tượng json. Do đó, ta cần phải bổ sung thêm thư viện Json trong quá trình biên dịch:

**pi@raspberrypi: ~/code/bth1 \$** javac -cp <path to json.jar>:. bth1b.java

**pi@raspberrypi: ~/code/bth1 \$** java -cp <path to json.jar>:. bth1b.java

- Ví dụ, json.jar được đặt ở đường dẫn ~/json.jar thì câu lệnh biên dịch sẽ như sau

**pi@raspberrypi: ~/code/bth1 \$** javac -cp ~/json.jar:. bth1b.java

**pi@raspberrypi: ~/code/bth1 \$** java -cp ~/json.jar:. bth1b.java

- Code chương trình mẫu

File: **bth1b.java**

```
import org.json.JSONObject;
import java.io.*;
import java.net.URL;
import java.net.HttpURLConnection;

public class bth1b {
    /* Define Key API and URL of Thingspeak cloud */
    private static final String keyApi = "T7H40F0X82VGW7L5";
    private static final String thinkSpeakUrl = "https://api.thingspeak.com";
```

```

/* Send field1 and field2 to cloud using JSON */
public static void updateDataUsingJSON(String urlStr, JSONObject data) {
    URL url;
    String line;
    int responseCode;
    BufferedReader reader;
    HttpURLConnection conn = null;
    StringBuilder responseBody = new StringBuilder();
    try {
        url = new URL(urlStr);
        conn = (HttpURLConnection) url.openConnection();
        conn.setRequestMethod("POST");
        conn.setRequestProperty("Content-Type", "application/json;
charset=UTF-8");
        conn.setDoOutput(true);

        /* Put JSON into request body */
        OutputStream os = conn.getOutputStream();
        BufferedWriter writer = new BufferedWriter(new
OutputStreamWriter(os, "UTF-8"));
        writer.write(data.toString());
        writer.flush();
        writer.close();
        os.close();
        conn.connect();

        /* Get response code from connection */
        responseCode = conn.getResponseCode();
        System.out.println("GET Response Code : " + responseCode);
        if (responseCode == HttpURLConnection.HTTP_OK) {
            reader = new BufferedReader(new
InputStreamReader(conn.getInputStream()));
            while ((line = reader.readLine()) != null) {
                responseBody.append(line);
            }
            System.out.println("Last entry ID: " +
responseBody.toString());
            reader.close();
        }
        conn.disconnect();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

public static void main(String args[]) {
    //TODO: Tạo đối tượng json chứa dữ liệu (field1, field2)
    muốn gửi lên server
    JSONObject jsonObject = .....

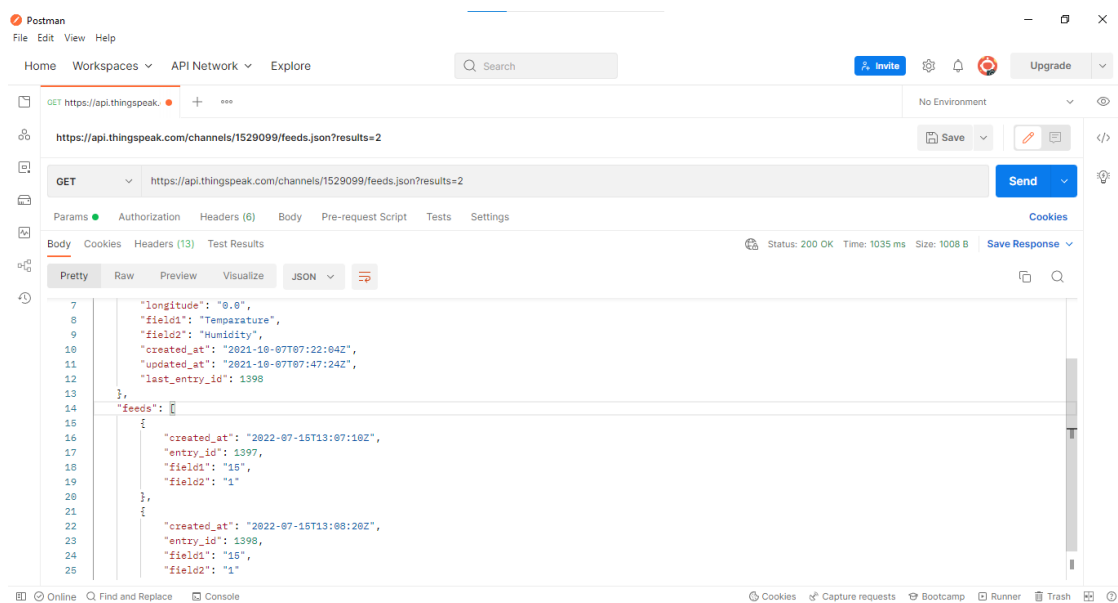
```

```
String url = thinkSpeakUrl + "/update?api_key=" + keyApi;
updateDataUsingJSON(url, jsonObject);
}
}
```

## Bài 2. Lấy dữ liệu từ server: Lấy dữ liệu về từ Thingspeak API

GET <https://api.thingspeak.com/channels/1529099/feeds.json?results=2>

- Kiểm thử API bằng Postman
  - Tương tự như trên, nhập url để gửi request lấy dữ liệu từ Channel 1529099 với results là số lượng bản tin mới nhất được đẩy lên kênh. Ví dụ, với đường dẫn trên, results=2 nghĩa bản tin trả về sẽ chứa dữ liệu của 2 bản tin mới nhất



- a) Viết chương trình bằng Java để lấy dữ liệu và hiển thị dữ liệu nhận được. Code chương trình mẫu.

File: **bth2.java**

```
import java.io.*;
import java.net.URL;
import java.net.HttpURLConnection;

public class bth1b {
    /* Define Key API and URL of Thingspeak cloud */
    private static final String keyApi = "T7H40F0X82VGW7L5";
    private static final String thinkSpeakUrl = "https://api.thingspeak.com";
}
```

```

/* Get data from Thingspeak */
public static void getDataFromChannel(String urlStr) {
    URL url;
    String line;
    int responseCode;
    BufferedReader reader;
    HttpURLConnection conn = null;
    StringBuilder responseBody = new StringBuilder();
    try {
        url = new URL(urlStr);
        conn = (HttpURLConnection) url.openConnection();
        conn.setRequestProperty("Content-Type", "application/json; charset=UTF-8");
        conn.setRequestProperty("Accept", "application/json");
        conn.setRequestMethod("GET");

        responseCode = conn.getResponseCode();
        System.out.println("GET Response Code : " + responseCode);
        if (responseCode == HttpURLConnection.HTTP_OK) {
            reader = new BufferedReader(new
InputStreamReader(conn.getInputStream()));
            while ((line = reader.readLine()) != null) {
                responseBody.append(line);
            }
            System.out.println(responseBody.toString());
            reader.close();
        }
        conn.disconnect();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

public static void main(String args[]) {
    int numChannel = 1529099; /* Channel ID */
    int numRecord = 2; /* Number of record want to get */
    String urlStr = "https://api.thingspeak.com/channels/" + numChannel +
"/feeds.json?results=" + numRecord;
    getDataFromChannel(urlStr);
}
}

```

- b) Parsing dữ liệu nhận được để lấy ra 2 trường field1 (temperature) và field2 (humidity) và hiển thị ra màn hình từ response body

```

public static void parseData(String data) {

```

```
try {
    JSONObject jsonObject = new JSONObject(data);
    JSONObject channel = jsonObject.getJSONObject("channel");
    //TODO: Parsing dữ liệu json nhận được để lấy Temperature
    và Humidity của mỗi bản ghi
    .....

} catch (Exception e) {
    e.printStackTrace();
}
}
```

### **Yêu cầu nộp bài:**

Viết báo cáo mô tả ngắn gọn gồm: code các bài, *chụp ảnh màn hình thực hiện chương trình và kết quả ở các yêu cầu.*

Submit: file báo cáo, file code