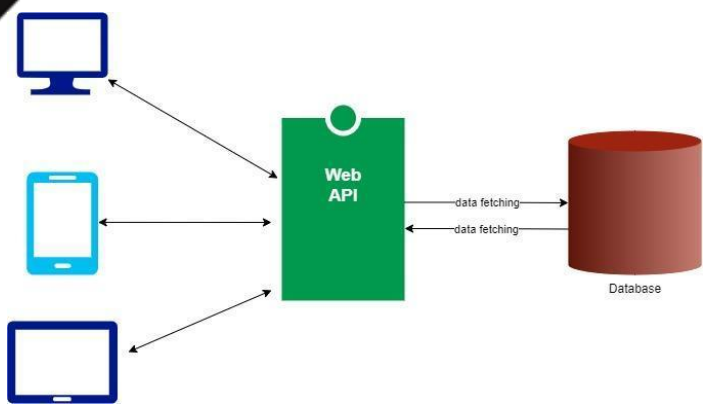
2025

nguyen.a.tu@gmail.com
www.facebook.com/nguyenanhtucom
www.youtube.com/@nguyenanhtucom

EXERCISE
Internet of Things
PROGRAMMING



MỤC LỤC

Setup Development Environment

Example 0.01

Mục tiêu: Thiết lập môi trường lập trình Java Spring Boot và tạo project

Yêu cầu:

- ✓ Cài đặt JDK và Visual Studio Code
- ✓ Tạo project có tên project là **example01**

Hướng dẫn:

Bước 1: Cài đặt JDK và Visual Studio Code

JDK Development Kit 17.0.7 downloads

JDK 17 binaries are free to use in production and free to redistribute, at no cost, under the Oracle No-Fee Terms and Conditions.

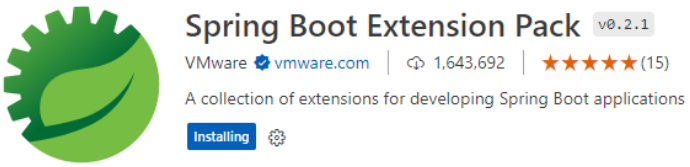
JDK 17 will receive updates under these terms, until September 2024, a year after the release of the next LTS.

Linux **macOS** **Windows**

Product/file description	File size	Download
x64 Compressed Archive	172.19 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.zip (sha256)
x64 Installer	153.28 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.exe (sha256)
x64 MSI Installer	152.07 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.msi (sha256)

Bước 2: Để cài đặt Extension Java cho Visual Studio Code bạn hãy thực hiện Mở Extensions (Ctrl+Shift+X), tìm kiếm **java**

Bước 3: Để cài đặt Extension Spring Boot cho Visual Studio Code bạn hãy thực hiện Mở Extensions (Ctrl+Shift+X), tìm kiếm **Spring Boot Extension Pack**

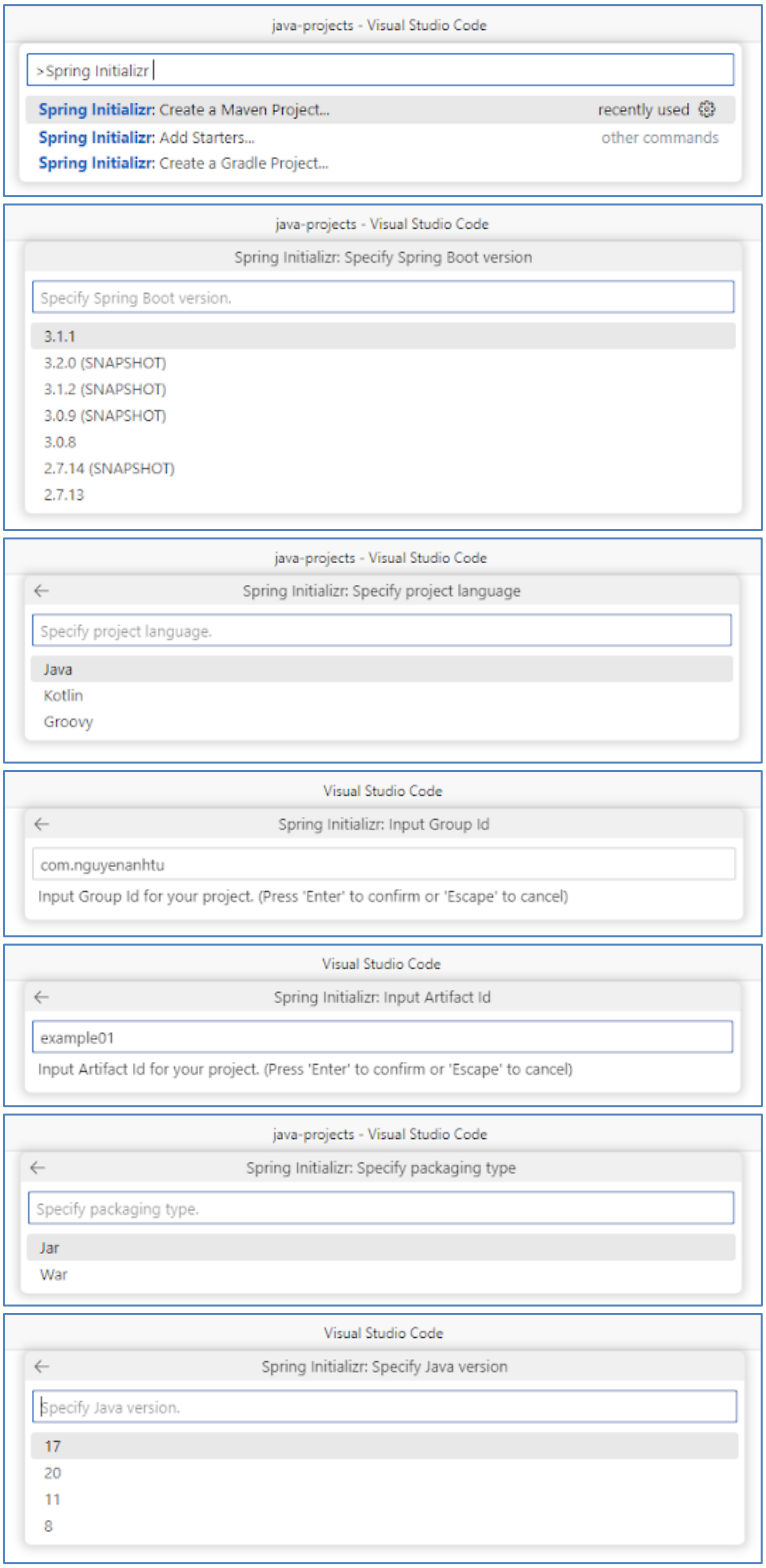


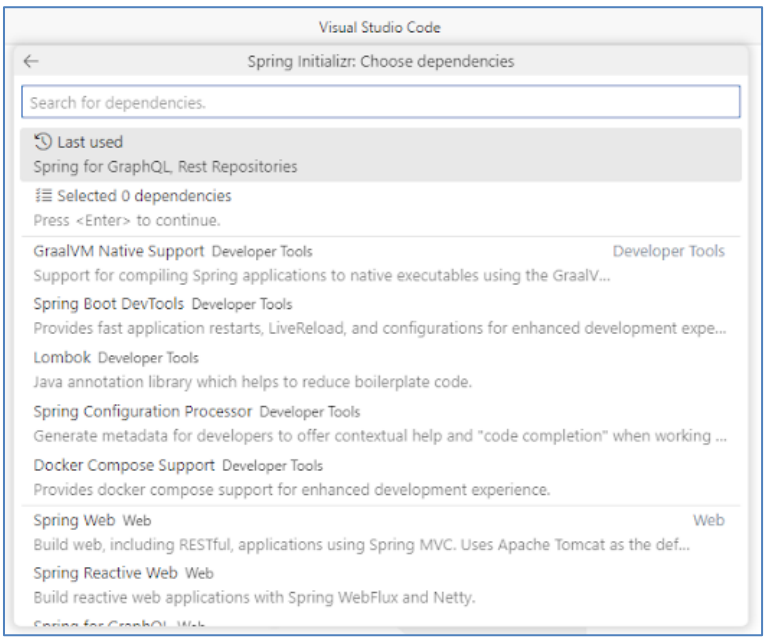
Bước 4: Sử dụng Visual Studio Code tạo project **Maven** với thông tin như sau:

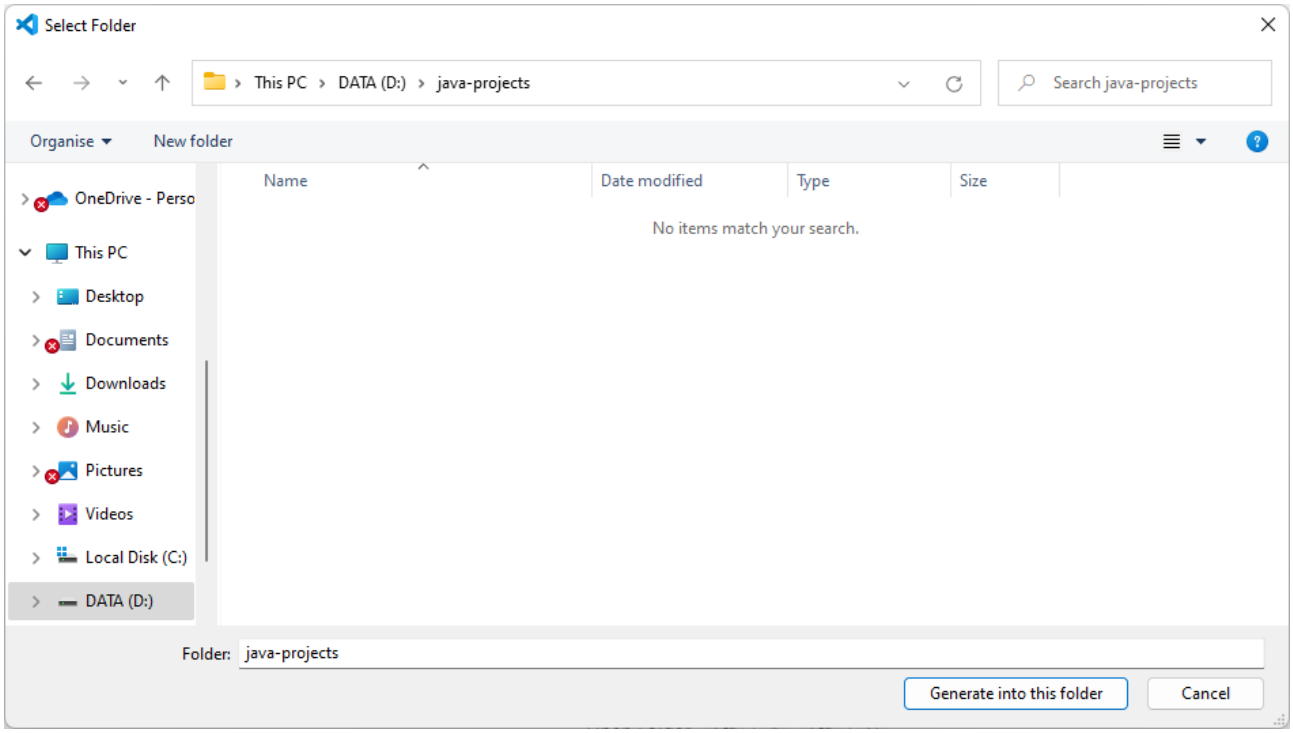
- ✓ Spring Boot version: **3.x.x**
- ✓ Language: **Java**
- ✓ Group Id: **com.nguyenanhtu**
- ✓ Artifact Id: **example01**
- ✓ Packaging type: **Jar**
- ✓ Java Version: **17**
- ✓ Dependencies: **Selected 0 dependencies**

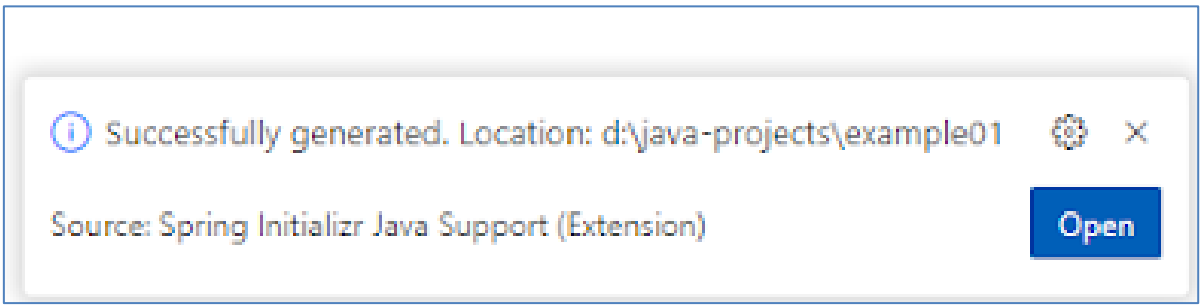
✓ Vị trí project: **D:\java-projects**

Hãy mở Bảng lệnh (Ctrl+Shift+P) và nhập **Spring Initializr** để bắt đầu tạo dự án **Maven**

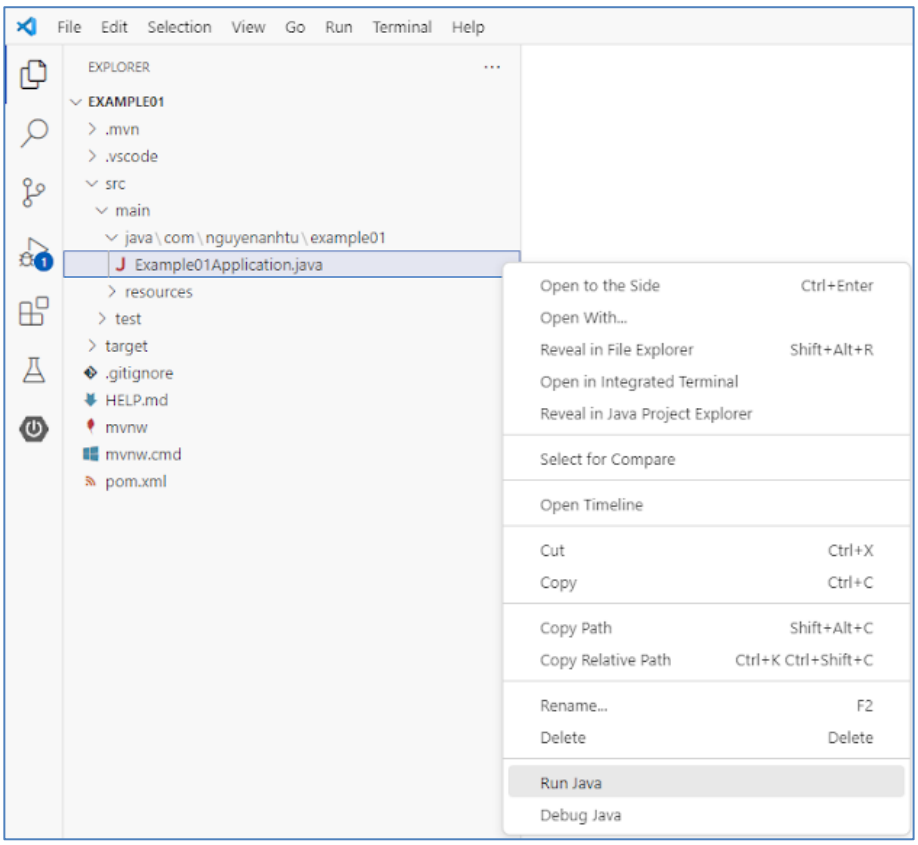


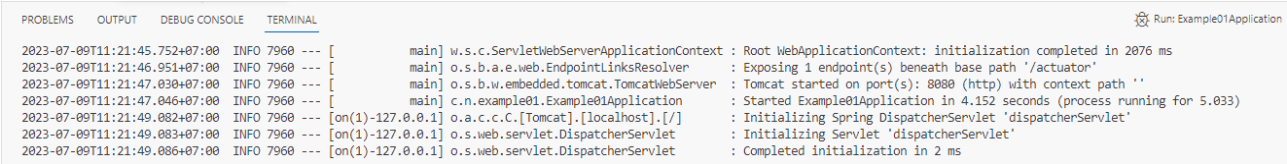






Bước 5: Click chuột phải vào lớp **Example01Application** sau đó Click vào **Run Java**.


File Edit Selection View Go Run Terminal Help
EXPLORER
EXAMPLE01
.mvn
.vscode
src
main
java \ com \ nguyenanhtu \ example01
Example01Application.java
resources
test
target
.gitignore
HELP.md
mvnw
mvnw.cmd
pom.xml
Open to the Side Ctrl+Enter
Open With...
Reveal in File Explorer Shift+Alt+R
Open in Integrated Terminal
Reveal in Java Project Explorer
Select for Compare
Open Timeline
Cut Ctrl+X
Copy Ctrl+C
Copy Path Shift+Alt+C
Copy Relative Path Ctrl+K Ctrl+Shift+C
Rename... F2
Delete Delete
Run Java
Debug Java


PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Run: Example01Application
2023-07-09T11:21:45.752+07:00 INFO 7960 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 2076 ms
2023-07-09T11:21:46.951+07:00 INFO 7960 --- [main] o.s.b.a.e.web.EndpointLinksResolver : Exposing 1 endpoint(s) beneath base path '/actuator'
2023-07-09T11:21:47.030+07:00 INFO 7960 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
2023-07-09T11:21:47.046+07:00 INFO 7960 --- [main] c.n.example01.Example01Application : Started Example01Application in 4.152 seconds (process running for 5.033)
2023-07-09T11:21:49.082+07:00 INFO 7960 --- [on(1)-127.0.0.1] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring DispatcherServlet 'dispatcherServlet'
2023-07-09T11:21:49.083+07:00 INFO 7960 --- [on(1)-127.0.0.1] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'
2023-07-09T11:21:49.086+07:00 INFO 7960 --- [on(1)-127.0.0.1] o.s.web.servlet.DispatcherServlet : Completed initialization in 2 ms

IoT Broker

Example 1.01

Mục tiêu: Tạo và quản lý ứng dụng **Java** sử dụng **MQTT Paho** kết nối **MQTT Broker** có tên kiến trúc như sau:

Yêu cầu: Ứng dụng **Java** Thực hiện các chức năng sau:

- ✓ Subscribe với **Broker** với topic có tên */test/topic*
- ✓ Publish nội dung "Hi from the IoT application" đến **Broker** với topic có tên */test/topic*

Hướng dẫn:

Bước 1: Sử dụng Visual Studio Code, mở Command Palette (Ctrl+Shift+P) và nhập **Java: Create Java Project…)** để bắt đầu tạo project với các thông tin như sau:

- ✓ Select the project type: **Maven**
- ✓ Vị trí project: **D:\java-projects**
- ✓ Artifact Id: **example101**
- ✓ Group Id: **com.nguyenanhtu**
- ✓ Select an archetype: **No Archetype**

Bước 2: Sửa file pom.xml như sau:

pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>

    <groupId>com.nguyenanhtu</groupId>
    <artifactId>example102</artifactId>
    <version>1.0-SNAPSHOT</version>

    <dependencies>
      <dependency>
            <groupId>org.eclipse.paho</groupId>
            <artifactId>org.eclipse.paho.client.mqttv3</artifactId>
            <version>1.2.5</version>
      </dependency>
    </dependencies>

    <properties>
        <maven.compiler.source>17</maven.compiler.source>
        <maven.compiler.target>17</maven.compiler.target>
    </properties>
</project>
```

Bước 4: Tao Entity **Employee** như sau:

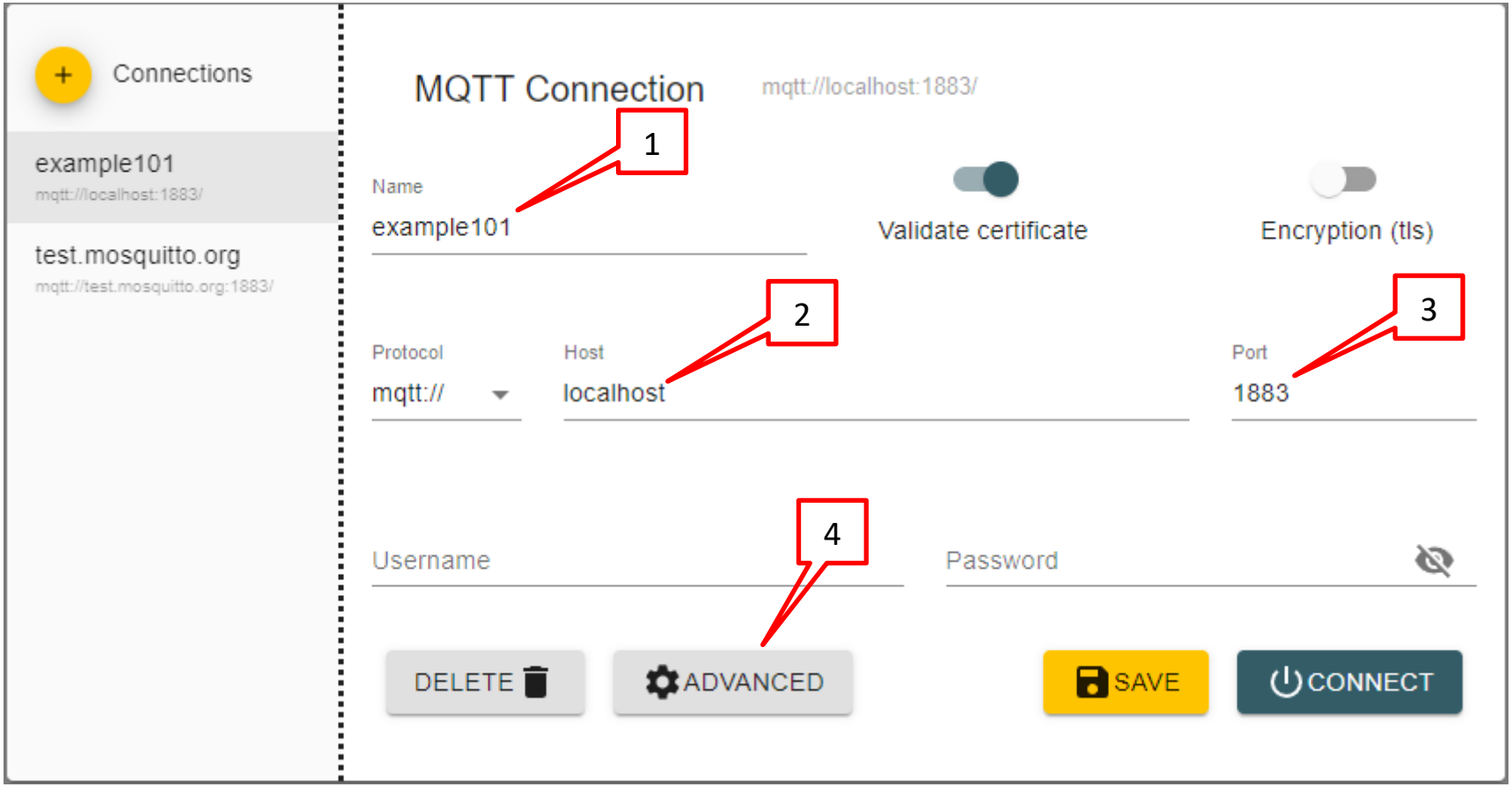
Main.java

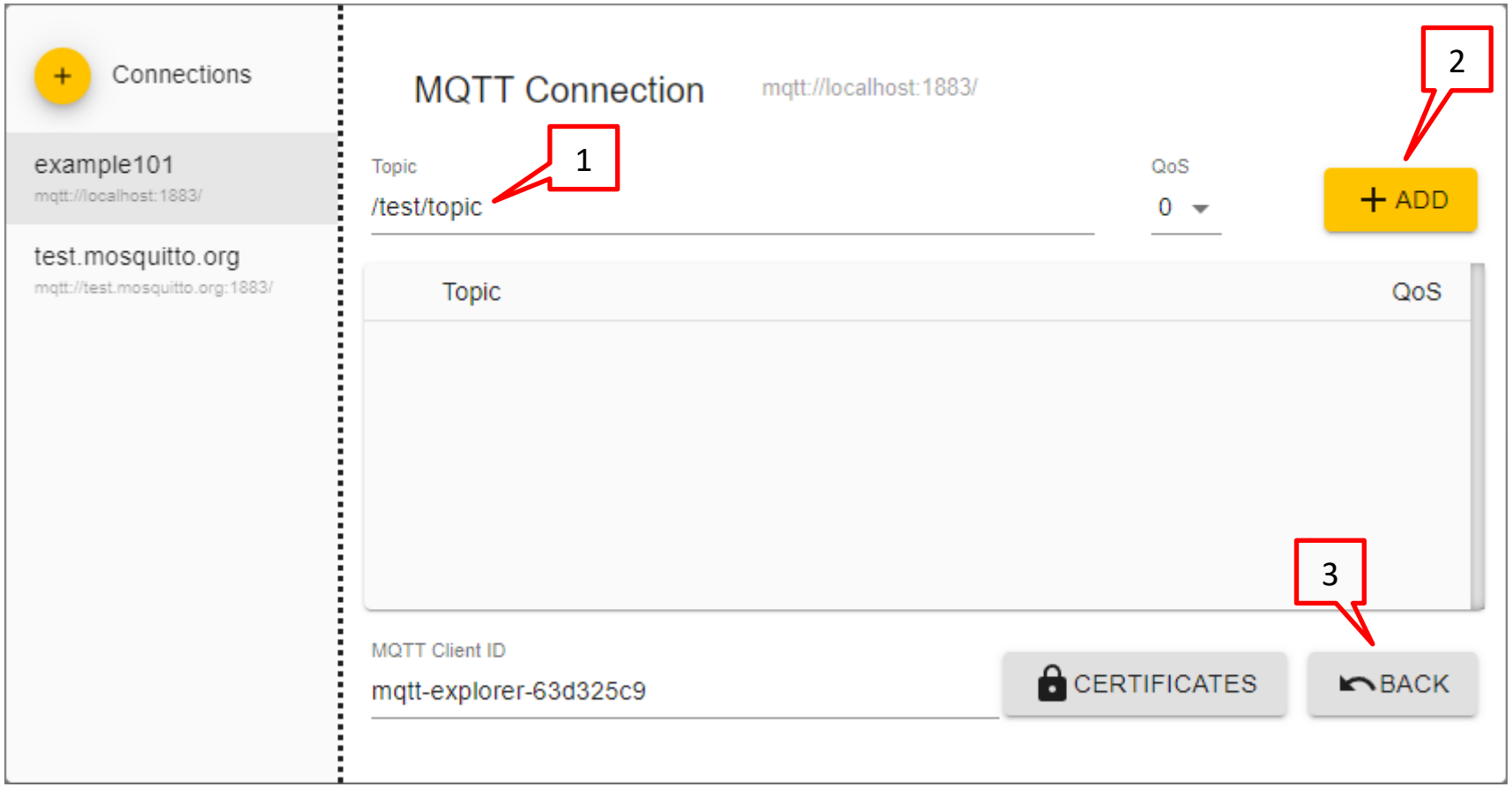
```
package com.nguyenanhtu.example101;

import java.io.IOException;

import org.eclipse.paho.client.mqttv3.IMqttDeliveryToken;
import org.eclipse.paho.client.mqttv3.MqttCallback;
import org.eclipse.paho.client.mqttv3.MqttException;
import org.eclipse.paho.client.mqttv3.MqttMessage;
import org.eclipse.paho.client.mqttv3.MqttConnectOptions;
import org.eclipse.paho.client.mqttv3.MqttClient;

public class Main {
    public static void main(String[] args) {
        String mqttBroker = "tcp://localhost:1883";
        String mqttTopic = "/test/topic";
        String username = "IoTClient";
        String password = "IoTPass";
        String testMsg = "Hi from the IoT application";
        int qos = 1;

        try {
            MqttClient mqttClient = new MqttClient(mqttBroker, "mqttClient");
            MqttConnectOptions mqttOptions = new MqttConnectOptions();
            mqttOptions.setUserName(username);
            mqttOptions.setPassword(password.toCharArray());
            mqttClient.connect(mqttOptions);

            if (mqttClient.isConnected()) {
                mqttClient.setCallback(new MqttCallback() {
                    @Override
```

```
                public void messageArrived(String topic, MqttMessage message) throws Exception {
                    System.out.println("Received message: " + new String(message.getPayload()));
                }

                @Override
                public void connectionLost(Throwable cause) {
                    System.out.println("Connection is lost: " + cause.getMessage());
                }

                @Override
                public void deliveryComplete(IMqttDeliveryToken token) {
                    System.out.println("Message publish is complete: " + token.isComplete());
                }
            });

            /* The client subscribe to a topic */
            mqttClient.subscribe(mqttTopic, qos);
            /* Preparing a message to be published */
            MqttMessage mqttMsg = new MqttMessage(testMsg.getBytes());
            mqttMsg.setQos(qos);
            /* A message is published on the same subscribed topic */
            mqttClient.publish(mqttTopic, mqttMsg);
        }

        /* Keep the application open, so that the subscribe operation can tested */
        System.out.println("Press Enter to disconnect");
        System.in.read();
        /* Proceed with disconnecting */
        mqttClient.disconnect();
        mqttClient.close();

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

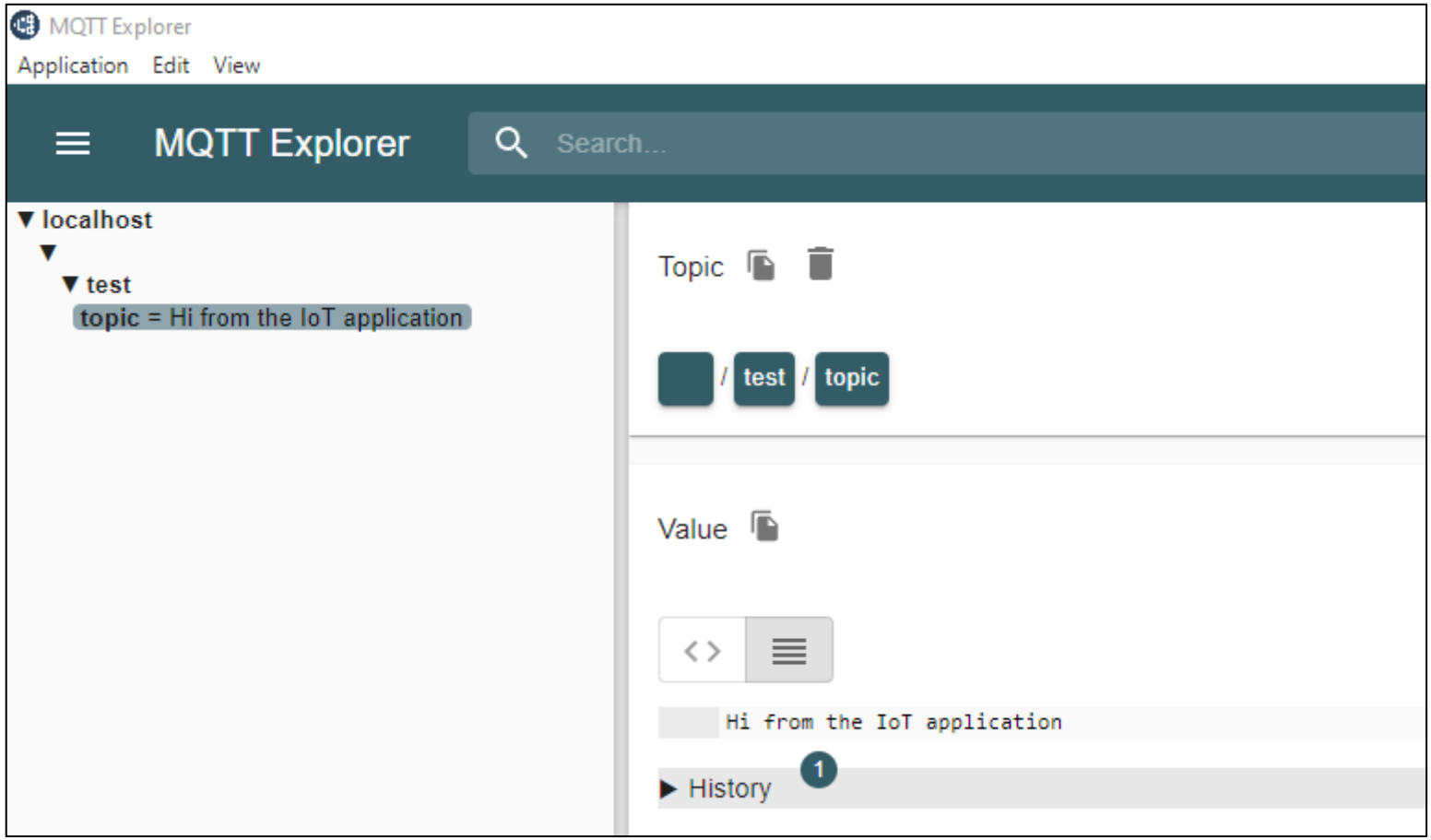
```
        } catch (IOException e) {
            throw new RuntimeException(e);
        }
    }
}
```

Bước 3: Sử dụng **MQTT-Explorer** đăng ký topic có tên **/test/topic** với Broker

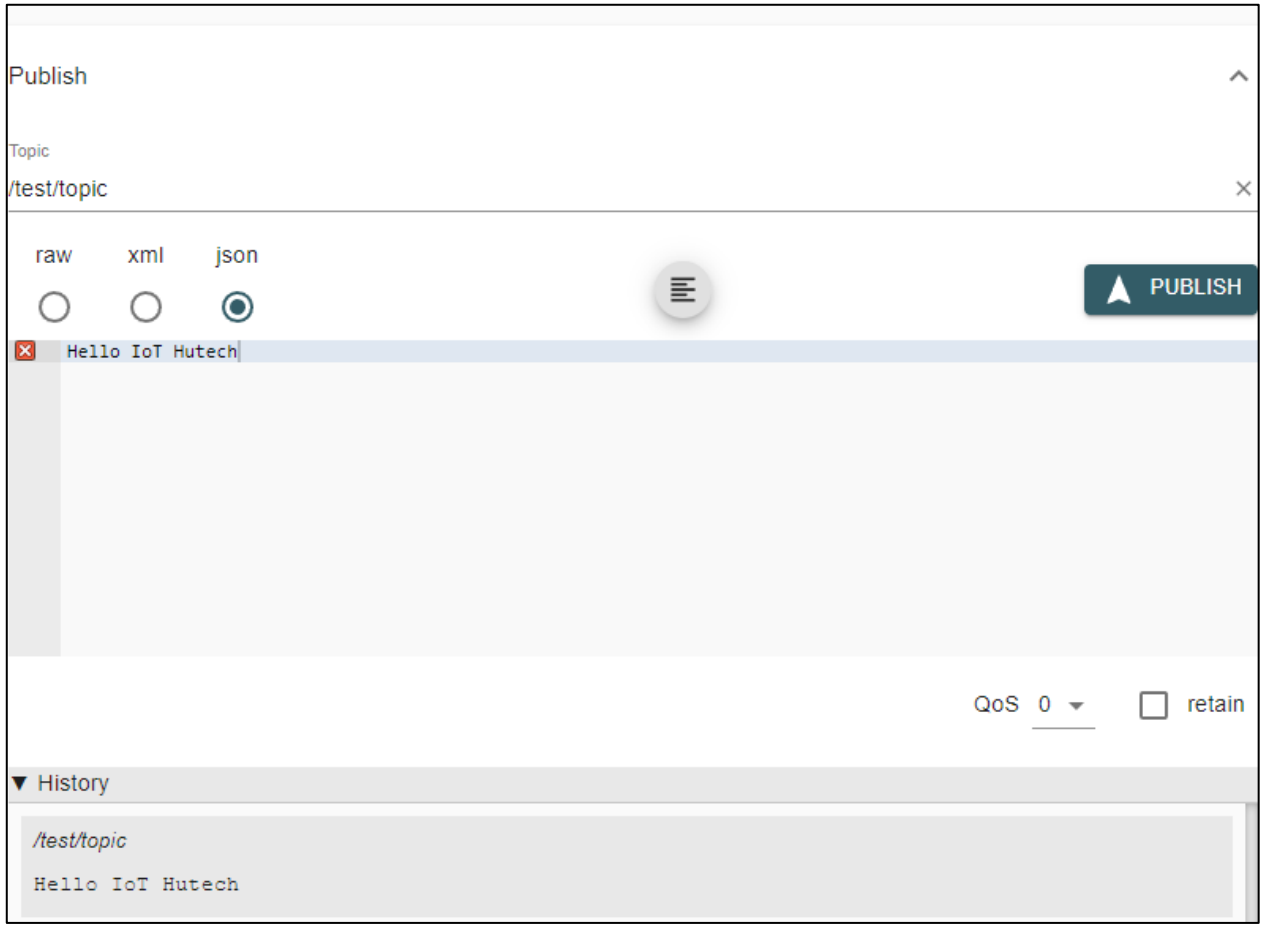




Bước 4: Click chuột phải vào lớp **Main →Run Java,** sau đó quan sát ứng dụng **MQTT-Explorer** như sau**:**



Bước 4: Sử dụng **MQTT-Explorer** publish tới Broker nội dung **Hello IoT Hutech** với topic có tên **/test/topic**



Bước 5: Kiểm tra dữ liệu nhận được trong cửa sổ Terminal của Visual Code

```
Press Enter to disconnect
Received message: Hi from the IoT application
Message publish is complete: true
Received message: Hello IoT Hutech
```