

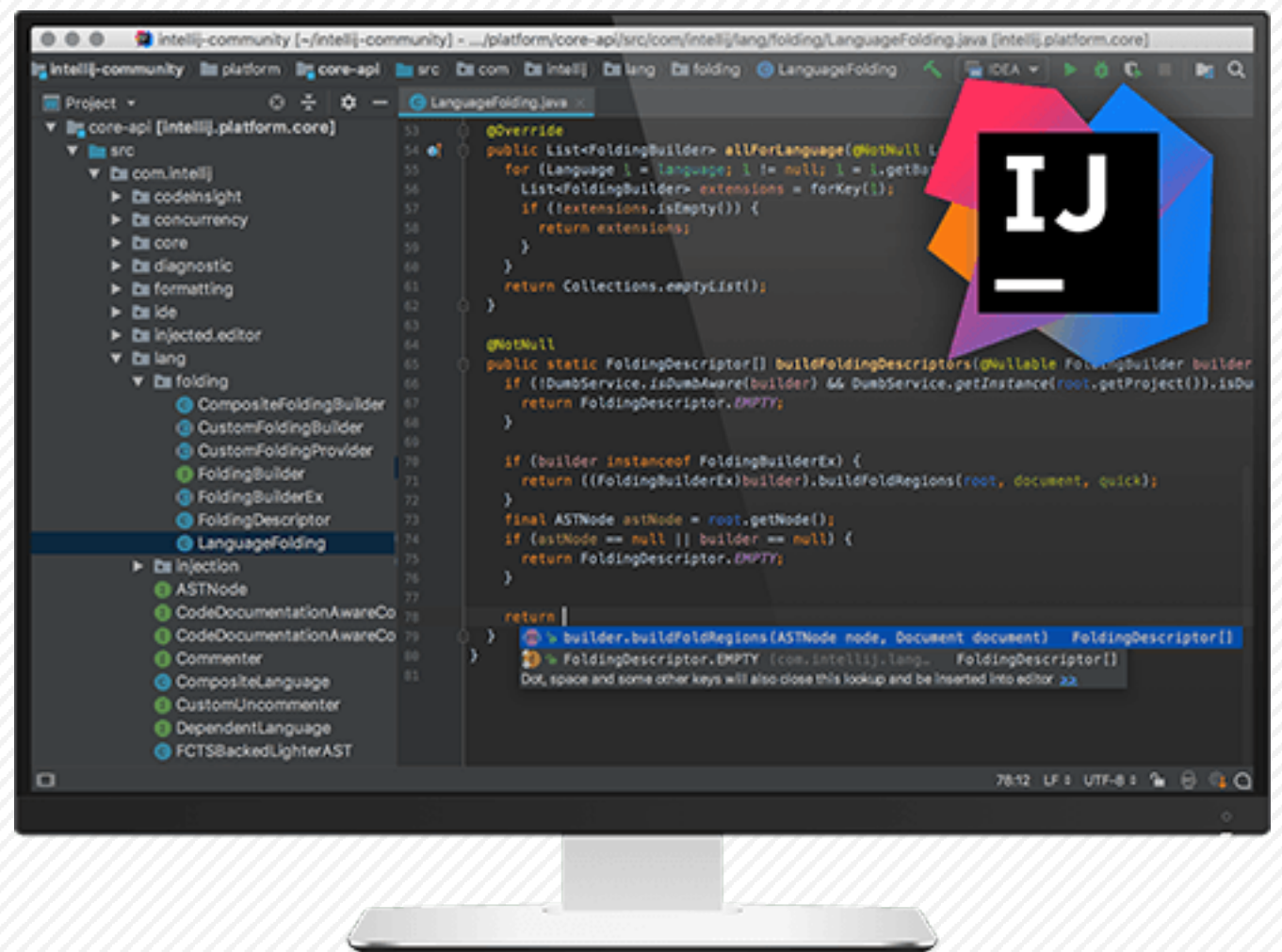
# Xây dựng Rest API Cơ Bản bằng **Kotlin SpringBoot**



# Môi trường phát triển



## IntelliJ IDEA Community Edition



# Tạo Init Project

Vào <https://start.spring.io> để tạo project.

**Project**  
☐ Maven Project  
☒ Gradle Project

**Language**  
☐ Java  
☒ Kotlin  
☐ Groovy

**Spring Boot**  
☐ 2.5.0 (SNAPSHOT) ☐ 2.4.2 (SNAPSHOT)  
☐ 2.4.1 ☐ 2.3.8 (SNAPSHOT) ☒ 2.3.7

**Project Metadata**  

Group

com.techbasevn

Artifact

demo

Name

demo

Description

Package name

com.techbasevn.demo

Packaging

☒ Jar ☐ War

Java


☐ 15 ☒ 11 ☐ 8

**Dependencies**

ADD ... ⌘ + B


**Spring Web** WEB

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.



Chọn dependency Spring-Web để sử dụng các annotation như @RestController, @RequestMapping

Nên sử dụng Gradle Build Tool vì tính đơn giản và tối ưu thời gian build hơn Maven



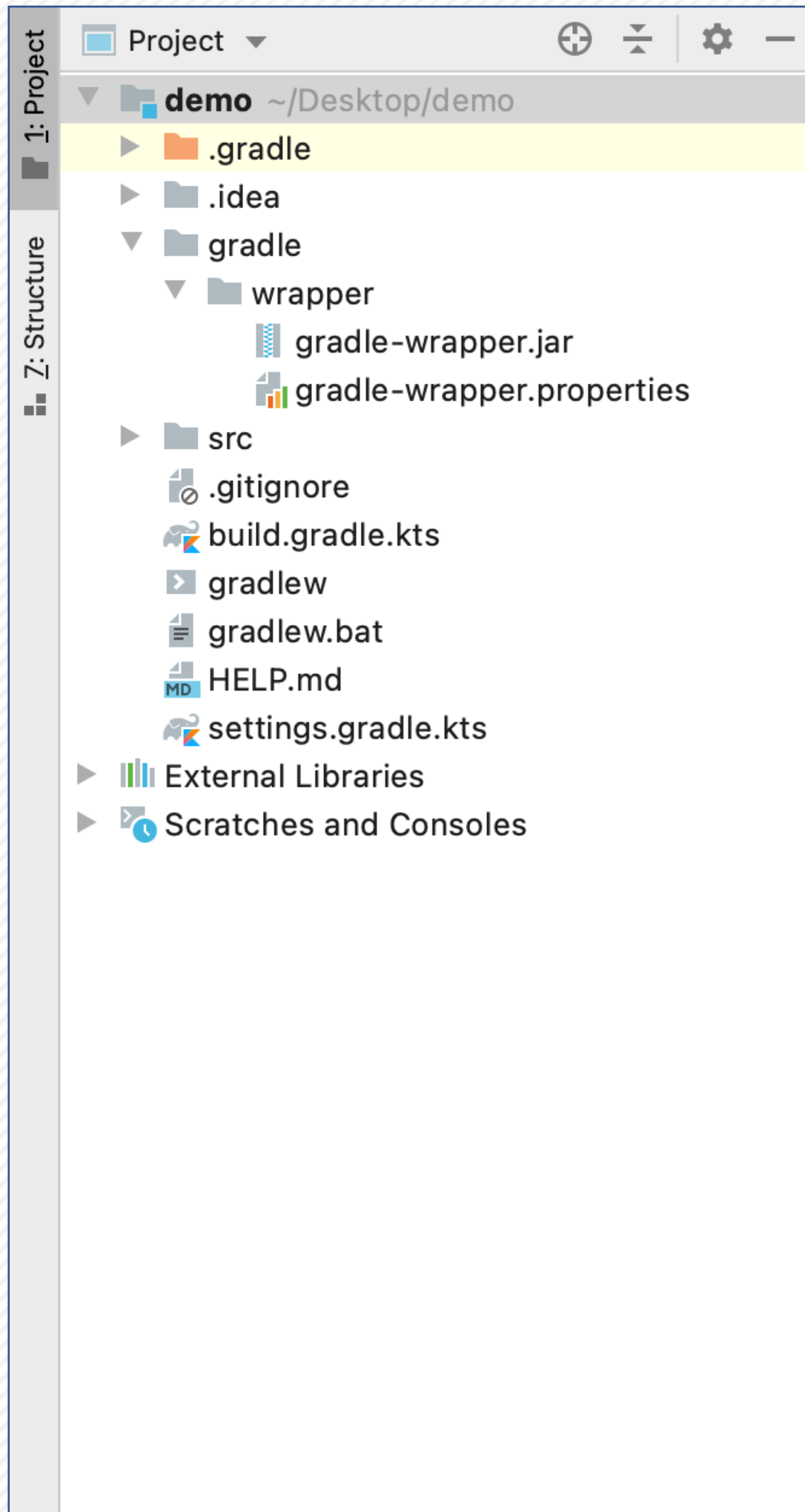
GENERATE ⌘ + ↵

EXPLORE CTRL + SPACE

SHARE...

# Cấu trúc project

# Gradle Build Tool



## ✓ gradle-wrapper.properties

Thông tin version gradle mà project sử dụng. Project sử dụng gradle-wrapper sẽ tự quản lý build tool theo version độc lập với các project khác.

## ✓ gradle-wrapper.jar

Dựa trên wrapper.properties, wrapper.jar sẽ download gradle tool theo version tương ứng.

## ✓ gradlew & gradlew.bat

Script thực thi build trên Linux & Windows.

## ✓ build.gradle.kts

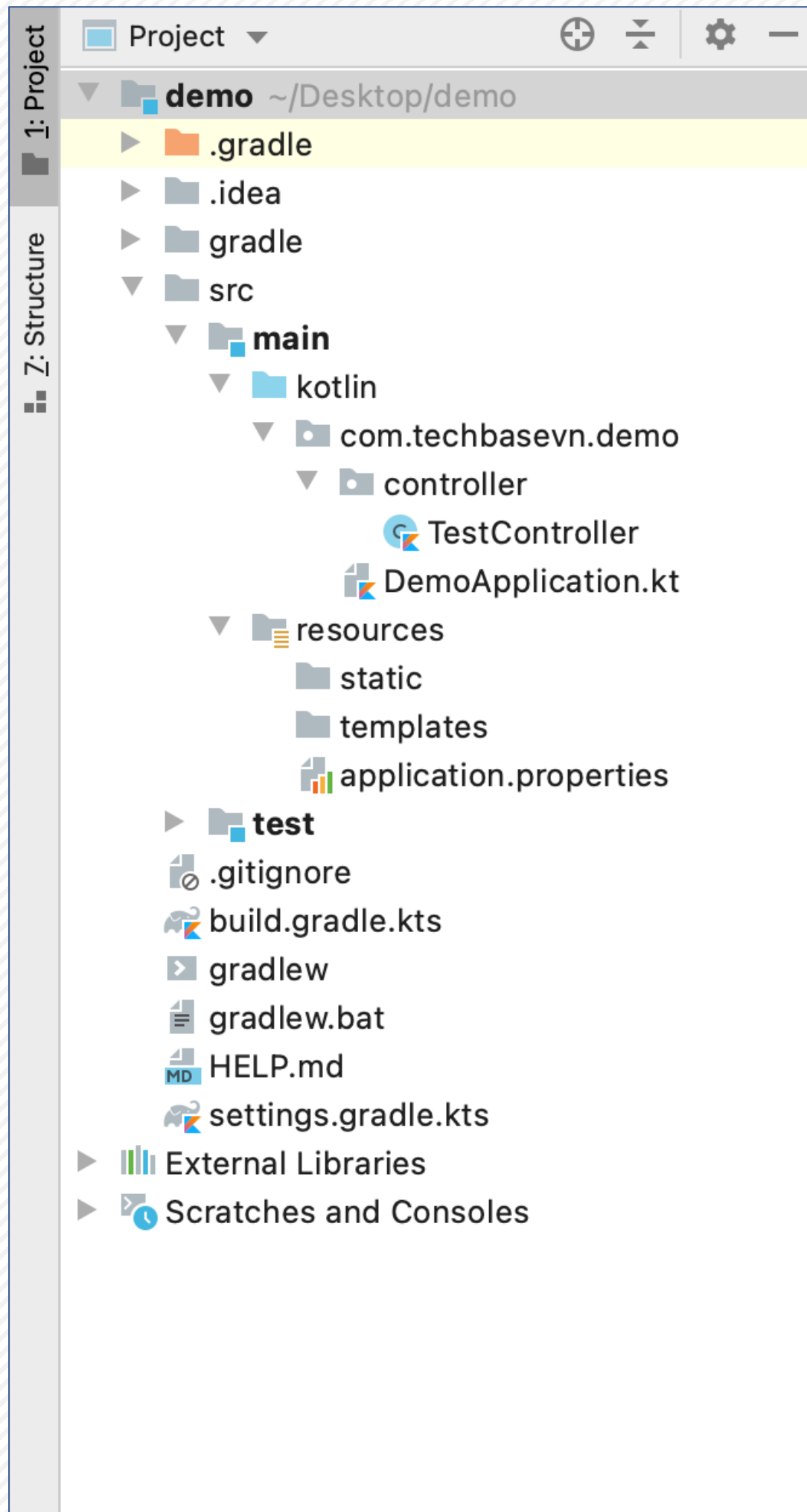
Khai báo các dependency, plugin, task của module.

Gradle cho phép viết các khai báo này theo ngôn ngữ Kotlin.

## ✓ settings.gradle.kts

Khai báo thông tin module và sub-module mà gradle cần quản lý.

# Cấu trúc project SpringBootApplication



## ✓ DemoApplication.kt

Chứa hàm main, start SpringApplication. Spring sẽ auto-configuration và scan tất cả component từ package cùng cấp hoặc sâu hơn từ class được đánh dấu @SpringBootApplication.

## ✓ package **[controller]**

Chứa các Controller class.

★ Controller là thành phần nhận request từ DispatcherServlet, yêu cầu các Service thực thi business Model, và quyết định Model nhận được theo View (html, xml, json,...) như thế nào.

## ✓ application.properties

(Hoặc application.yml)

Cho phép khai báo property dùng trong app, và cho các dependency.

★ Spring cho phép khai báo các bộ property profile theo từng environment (dev, prod, stg,...).



# Tạo RestController class

...trong package [controller] với annotation  
@RestController. Spring MVC sẽ



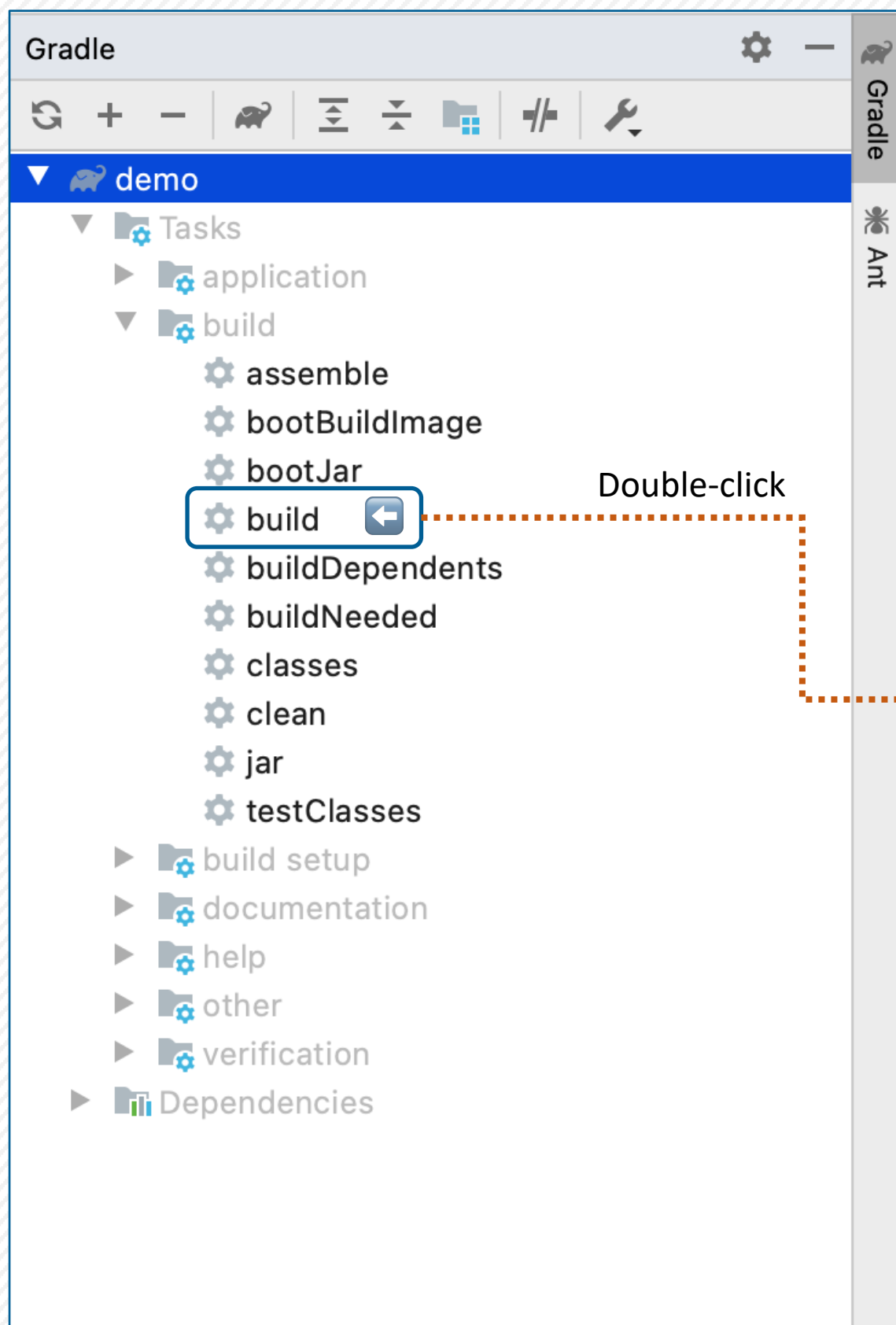
```
package com.techbasevn.demo.controller

import org.springframework.web.bind.annotation.GetMapping
import org.springframework.web.bind.annotation.RestController

@RestController
class TestController {
    data class Response(val message: String, val status: Int)

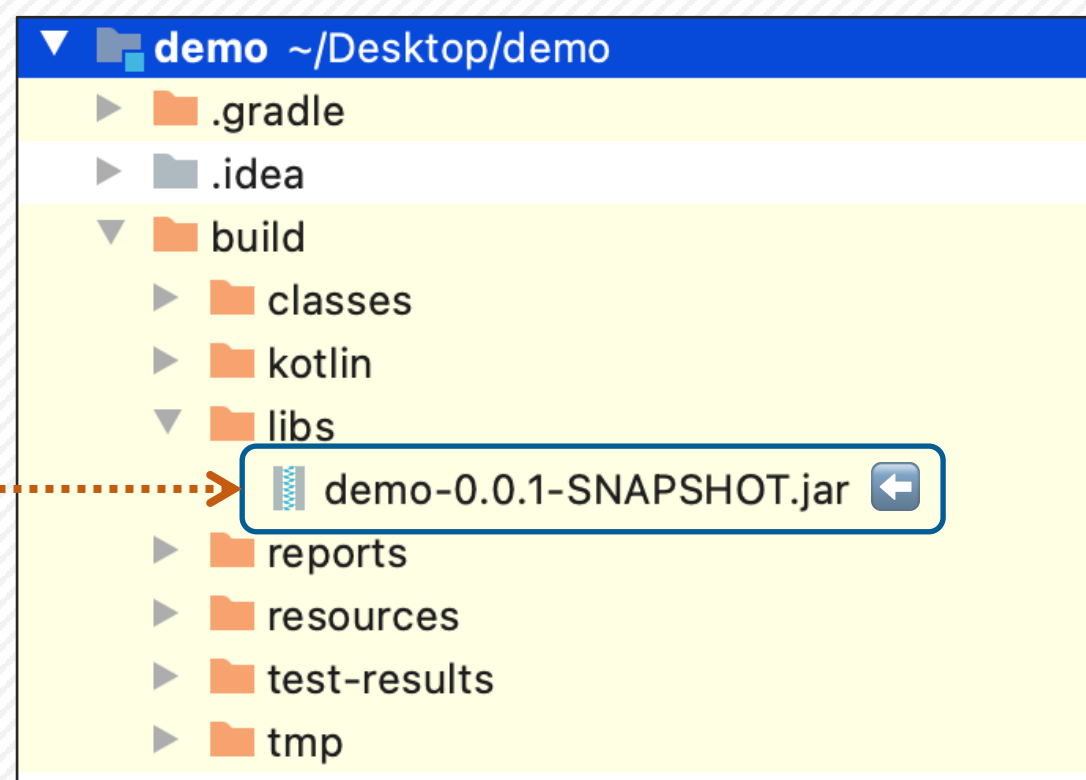
    @GetMapping
    fun index(): Response {
        return Response("Test Success", 200)
    }
}
```

# Build & Launch



Execute chức năng [build] trên cửa sổ [Gradle].

→ Generate \*.jar trong [build/libs]



Trên cửa sổ Terminal, thực hiện lệnh launch file jar

**java -jar [jar\_file]**

