

บทที่ 1

Introduction to Backend Programming

หัวข้อที่จะเรียนรู้

- ความแตกต่างระหว่าง Frontend vs Backend
- สถาปัตยกรรม Web Application
- เทคโนโลยีและเครื่องมือที่ใช้ใน Backend
- แนะนำ Spring Boot และการเตรียมสภาพแวดล้อม

Frontend vs Backend

Frontend (Client-side)

- ส่วนที่ผู้ใช้มองเห็นและโต้ตอบ
- HTML, CSS, JavaScript, React, Angular

Backend (Server-side)

- ทำงานเบื้องหลัง ประมวลผลข้อมูล
- API, Database, Authentication, Business Logic

สถาปัตยกรรม Web Application

- Client-Server Model
- MVC (Model-View-Controller)
- Microservices Architecture
- Monolithic vs Microservices

ตัวอย่าง Client–Server Model

[Client] ⇌ [Backend Server] ⇌ [Database]

- Client: Browser, Mobile App
- Backend: Spring Boot (Java)
- Database: MySQL, PostgreSQL, MongoDB

เทคโนโลยีและเครื่องมือ Backend

- ภาษา: Java, Python, Node.js, Go
- Framework: Spring Boot, Django, Express.js
- Database: MySQL, PostgreSQL, MongoDB, Redis
- Tools: Git, Docker, Postman

ทำไมต้อง Spring Boot?

- Framework ขอดนินยอมสำหรับ Java Backend
- Auto Configuration & Convention over Configuration
- รองรับ RESTful API, Security, Database Integration
- มี ecosystem ที่ใหญ่และทันสมัย

เตรียมสภาพแวดล้อม

- ติดตั้ง JDK 17+
- ติดตั้ง Maven/Gradle
- ใช้ Spring Initializr (<https://start.spring.io>)
- IDE: IntelliJ IDEA, Eclipse, VS Code

Lab

1. ใช้ Spring Initializr สร้างโปรเจกต์
 - Dependencies: Spring Web
2. เขียน Controller ง่าย ๆ (/hello)
3. รันและทดสอบผ่าน Postman

Assignment

- สร้าง REST API `/hello` ที่คืนค่า JSON เช่น:

```
{ "message": "Hello Backend Programming" }
```

- ส่งโค้ด + README วิธีการรัน