

เอกสารนี้จัดทำขึ้นเพื่อให้ นศ. สามารถ ทดลองทำตามและทำ assignment 1 ได้ ด้วยการทดสอบและเรียนรู้
แก้ปัญหาด้วยตัวเอง ไม่ได้ทำขึ้นเพื่อทดแทน lecture session

วัตถุประสงค์

1. จัดตั้ง env สำหรับการพัฒนาโครงการ ส่วนที่ 1 : Java Spring Boot Restful Web Services
2. อธิบายการเขียนโค้ดเพิ่มเติมจาก session เนื่องจากใน session มีเวลาจำกัด

เป้าหมายต้องส่งงาน

เนื้อเรื่อง (Context): เมื่อเราซื้อของมา เราต้องเก็บไว้ในสถานที่ และต้องจดจำว่าได้เก็บไว้ที่ไหน เมื่อมีการนำไปใช้โดย
เรา หรือคนอื่นๆ เราก็ต้องจำว่าได้นำไปใช้ที่ไหน และใครนำไปใช้ (หากรู้) ในกรณีที่มีคนยืมของที่เราเก็บไว้ไป และได้นำ
มาคืน เราก็ควรต้องทราบว่าเขาได้คืนเรามาแล้ว และเราเอาไปเก็บไว้ที่ไหน ทั้งนี้หากเราต้องจำ ต้องจด ต้องตาม เราอาจ
จะไม่อยากทำเอง

คำสั่ง จงเขียน Java RESTful web service เพื่อประกอบการให้บริการข้างต้น

การตรวจสอบ นศต้องเขียน test_<id ของตนเอง>.rest และนำส่งมาพร้อมกับ code ที่ทำ และได้ทดสอบเรียนรู้และออก
แบบด้วยตนเอง

Table of Contents

Code Lab A

PART 1 Development Environment Setup

PART 2 ใช้โปรแกรมที่ demo บน VS code

Notes: คำสั่งที่เป็นประโยชน์

1
1
6
9

Code Lab A

เนื้อหา เพื่อให้ทำ assignment 1 ได้

PART 1 Development Environment Setup

ผู้สอนใช้ Notebook windows 11 และทำ code บน Ubuntu 22.04.5 TLS จึงมีคำแนะนำดังนี้

1. นศ ที่ใช้ windows ต้องใช้ wsl (version 2) เพื่อ develop งาน สำหรับทั้งวิชา CS367 การใช้ wsl2 มีประโยชน์
ในเรื่องการจัดการและการช่วยแก้ปัญหาเมื่อจำเป็น
2. นศ ที่ใช้ linux หรือ MacOS เนื่องจากทั้งคู่เป็น Unix แบบ native หากต้องการได้ความยืดหยุ่นในการสำรอง
หรือช่วยเหลือ อาจจะต้องพิจารณาใช้ Virtual Box และ download image มาใช้แทนการติดตั้ง software ตาม
ขั้นตอนเหล่านี้ของตนเอง แต่มีข้อเสียเรื่องประสิทธิภาพจึงไม่แนะนำหากมีเครื่องเก่าและช้า สำหรับคนสนใจ
MacOS ให้ติดตั้ง Virtual Box เพื่อเข้าถึง Ubuntu Linux ได้ตาม web นี้
<https://ubuntu.com/tutorials/how-to-run-ubuntu-desktop-on-a-virtual-machine-using-virtualbox>
3. สำหรับนศที่ใช้ MacOS การติดตั้งซอฟต์แวร์จะคล้ายๆ กับขั้นตอน แต่เนื่องจากไม่ใช่ Ubuntu Linux คำสั่ง apt /
apt-get ไม่มี อาจจะต้องใช้วิธีอื่นแทน เช่น brew นศ สามารถหาวิธีติดตั้งได้จาก internet

Note: เนื่องจากเป็นวิชาสาขาคอม ผู้สอนไม่เตรียม Env ให้ (แต่มีทำไว้) การติดตั้งและ Setup ต่างๆ ถือเป็นส่วนหนึ่งของการ
การเรียนรู้และการทำการแก้ปัญหา Trouble Shooting

ระบบและเครื่องมือที่จะใช้ในวิชา CS367 part 1 (assignment 1)

เตรียมระบบ

สำคัญ สำหรับ windows ต้องเปิดใช้ wsl2 ก่อน วิธีการตาม web นี้

**<https://learn.microsoft.com/en-us/windows/wsl/install> ถ้าคุณมี wsl แล้ว พิมพ์ wsl ที่ command
prompt ก็จะใช้ Linux distribution ได้**

**สำหรับ MacOS / Linux owner คุณมี shell ที่รองรับอยู่แล้ว หากต้องการใช้ virtual box ต้องบริหารจัดการ
เองตามสะดวก**

1. Windows : เข้า WSL ด้วย wsl command จาก command prompt
อื่นๆ: เปิดจาก terminal โปรแกรมบน OS ของตน (windows ก็มี)
2. Java OpenJDK (ที่ complied โดย Microsoft) ใช้เป็นตัวแปลภาษาหลัก ในการติดตั้ง software ทุกตัว หากมี
การทำงานเป็นทีมที่ช่วยกันพัฒนา นศ ควรต้องระมัดระวัง เรื่องการเลือกใช้ version ให้ดี เพราะความผิดพลาด
อาจจะเกิดจากการใช้ version ที่ไม่เหมือนกัน ผู้สอนใช้ version 21 LTS (long term service) และทดสอบกับ
17 LTS ก็ใช้งานได้

Release	Released	Security Support	Latest
21 (LTS)	1 year and 4 months ago (19 Sep 2023)	Ends in 3 years and 7 months (30 Sep 2028)	21.0.6 (04 Jan 2025)
17 (LTS)	3 years and 4 months ago (14 Sep 2021)	Ends in 2 years and 7 months (30 Sep 2027)	17.0.14 (04 Jan 2025)
11 (LTS)	6 years ago (21 Jan 2019)	Ends in 2 years and 7 months (30 Sep 2027)	11.0.25 (10 Oct 2024)

3. Spring Boot version 3.4 หรือมากกว่า หลายๆ features ที่จะใช้เริ่มมีตอน 3.2 เมื่อנס เลือก version กันภายใน
ทีมแล้วการ update / upgrade อาจจะมีผลต่อการทำงานในโครงการร่วมกัน สำหรับ assignment 1 เป็นงาน
เดี่ยว (Note: Spring Foundation version - `SpringVersion.getVersion()` = 6.2.2)
4. เลือกระหว่าง Maven หรือ Gradle! ตัว code project ที่จะทำจำเป็นต้องมีตัวบริหารจัดการโครงการที่กำหนด
dependencies ต่างๆ เพื่อเป็นเครื่องมือดูแลโค้ดระบบงานที่มีซับซ้อน (ผู้สอนใช้ Maven 3.6.3 หากเลือกใช้
Gradle ก็ต้องดูแล dependencies เองนะครับ โดย concept คล้ายๆ กัน แต่ syntax ต่างกัน)

ขั้นตอนการเตรียม Env ที่ Linux Prompt (CS367 ใช้ Linux ในการทำโครงการ)

1. สร้าง directory เพื่อใช้ทำงานใน class CS367 เลือกชื่อ directory ตามใจชอบ สมมติว่าชื่อ X
> mkdir <X>
2. Change directory ไปที่ที่สร้าง
> cd <X>
3. ถ้ายังไม่มี Java SDK version ที่กำหนด ก็นำมาลง เช่น
> sudo apt install openjdk-21-jdk
4. ถ้ายังไม่มี Maven ตามเวอร์ชันที่กำหนด ก็นำมาลง เช่น
> sudo apt install maven
หลังติดตั้งเรียกดู maven version ได้ด้วยคำสั่ง
> mvn -version
5. ติดตั้ง sdk managing version tool เพื่อติดตั้ง Spring Framework และ Spring Boot
> curl -s "https://get.sdkman.io" | bash
6. ทำการ refresh ค่าเริ่มต้นของ sdk script ด้วยคำสั่ง
> source \$HOME/.sdkman/bin/sdkman-init.sh
7. ตรวจสอบว่าท่านสั่งคำสั่ง sdk แล้วไม่ error
> sdk
(ควรพบว่าทำงานได้ ไม่ error)
8. ทำการติดตั้ง spring boot ด้วย คำสั่ง
> sdk install springboot
9. เมื่อสำเร็จ ลองตรวจดู version ว่าเกิน 3.4 หรือไม่ด้วยคำสั่ง
> spring --version

สำคัญ ขั้นตอนต่อไปต้อง ติดตั้ง Virtual Studio Code บนระบบปฏิบัติการของท่าน ก่อนทำการสร้าง project folder เปิด browser และไปที่ <https://code.visualstudio.com/> เพื่อ download software และทำการติดตั้งไฟล์ อ้างอิงตามขั้นตอนการ install virtual studio code ได้ที่ <https://code.visualstudio.com/docs/setup/setup-overview>

ขั้นตอนการใช้งาน vscode หลังติดตั้ง

VScode ที่อยู่บน Windows สามารถใช้ WSL2 integration กับ VScode ได้เลย ด้วยการใช้คำสั่ง “code” ที่ wsl2 linux prompt สำหรับคนที่อยู่บน Linux ก็สามารถเรียนผ่าน GUI ได้ ส่วน MacOS ขึ้นกับว่าใช้แบบ Native หรือผ่าน Virtual Box อีกที ดังนั้น vscode มีให้ใช้งานได้แน่นอน ดูขั้นตอนการติดตั้งได้จาก web link ข้างต้น

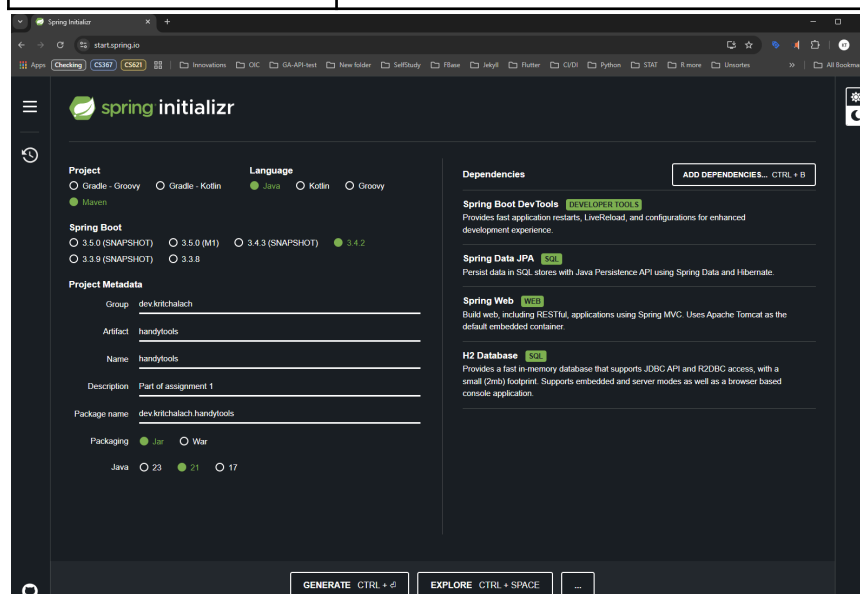
การสร้างโครงการ Java Spring Boot ตามที่อธิบายใน Session

หลังจากนักศึกษา มี Java JDK, Maven และ VScode แล้ว และได้เข้าใจว่าจะออกแบบอย่างไรแล้ว เพื่อให้สามารถเริ่มเขียนโปรแกรมได้ ณ จุด สามารถทดลองทำตามเอกสาร และไฟล์ตัวอย่างเพื่อเขียน Java Restful Web Service ได้ตามที่บรรยาย ก่อนจะเริ่มทำ assignment 1 ด้วยตัวเอง

ตัวอย่างการแก้ไขโจทย์ที่ให้ใน assignment 1 บางส่วน (โดยผู้สอน)

1. หลังจากผู้สอนทำการออกแบบด้วยวิธีต่างๆ ตาม assumption ที่ตั้งขึ้นจากโจทย์ ก็เริ่มต้นสร้างโครงการ Spring Boot ด้วย Tool ที่ชื่อว่า Spring Initializer
2. ไปที่ <https://start.spring.io/>
3. ใส่ข้อมูลที่กำหนดไว้เป็นโครงการ Restful Web Service ดังนี้

ตัวเลือก	ค่าที่เลือก
Project	Maven
Language	Java
Spring Boot	3.4.2
Project Metadata	เปลี่ยนตามที่มี (ติดตั้ง) <ol style="list-style-type: none"> 1. Group เลือกชื่อ package 2. Artifact & Name : handytools 3. Packaging jar 4. Java แล้วแต่ที่ใช้ 17 หรือ 21
Add Dependencies	Spring Boot DevTools Spring Web Spring Data JPA (SQL) H2 database



รายละเอียดการใช้งาน initializr เป็นไปตามที่บรรยาย

- กด Generate เพื่อทำการ download zipped file ที่เป็น maven project / pom (project object model) เมื่อกดแล้วคุณจะได้ zip ไฟล์ตามชื่อของ Artifact
- โดยปกติ ไฟล์จะถูก download มาที่ download directory ของระบบของคุณ เช่นใน windows ก็อยู่ที่ directory download หรือที่คุณกำหนดให้หาว่าได้ download ไฟล์ดังกล่าวไว้ที่ directory ไหน
- ไปที่ project directory ที่คุณสร้างไว้ (windows คืออยู่ใน wsl2 Linux shell / อื่นๆ ก็ตามที่กำหนดไว้เอง)
- คุณสามารถเข้าถึงไฟล์บน windows จาก wsl linux shell ได้ ผ่าน /mnt/c (mount point) เช่น เมื่อผม download handytools.zip ปรากฏว่าผมเก็บไว้ที่ C:\Download\handytools.zip ผมสามารถที่จะ unzip ไฟล์ดังกล่าวได้ด้วยคำสั่ง

```
> cd $HOME/demo/
```

```
> unzip /mnt/c/Downloads/handytools.zip
```

```
cs367@FlipS:~/demo$ unzip /mnt/c/Downloads/handytools.zip
Archive: /mnt/c/Downloads/handytools.zip
  creating: handytools/
  inflating: handytools/.gitignore
  inflating: handytools/HELP.md
  creating: handytools/.mvn/
  creating: handytools/.mvn/wrapper/
  inflating: handytools/.mvn/wrapper/maven-wrapper.properties
  inflating: handytools/.gitattributes
  creating: handytools/src/
  creating: handytools/src/test/
  creating: handytools/src/test/java/
  creating: handytools/src/test/java/dev/
  creating: handytools/src/test/java/dev/kritchalach/
  creating: handytools/src/test/java/dev/kritchalach/handytools/
  inflating: handytools/src/test/java/dev/kritchalach/handytools/HandytoolsApplicationTests.java
  creating: handytools/src/main/
  creating: handytools/src/main/resources/
```

จากภาพ directory โครงการที่มีการกำหนด POM file ไว้จะเป็นชื่อ \$HOME/demo/handytools ขั้นตอนต่อไปคือไปที่ directory นี้และเรียก vscode เพื่อเริ่มการเขียนโปรแกรม ตามที่ได้ design ไว้

- > cd \$HOME/demo/handytools

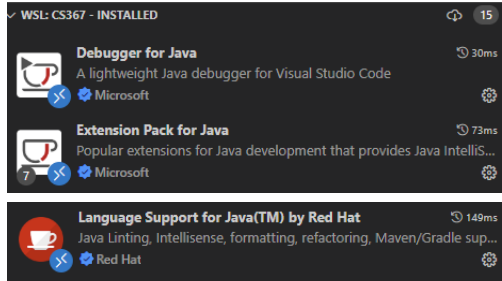
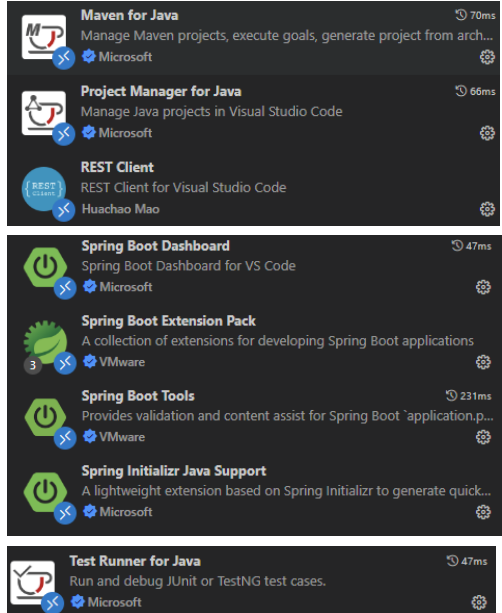
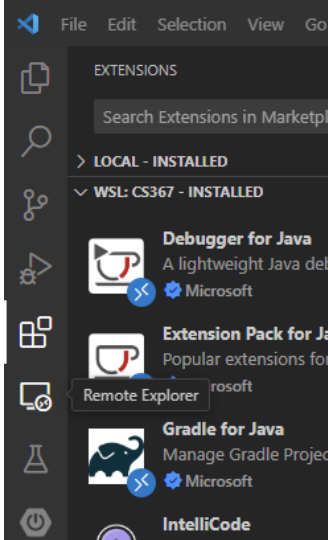
```
> code .
```

Note: . คือการอ้างถึง directory ที่คุณกำลังอยู่ที่โครงการ vscode จะเริ่มถามเกี่ยวกับการติดตั้ง ซึ่งเราจะตรวจสอบว่ามีครบ พร้อมในการใช้งานตามที่แสดงได้หรือไม่ ในขั้นตอนต่อไป

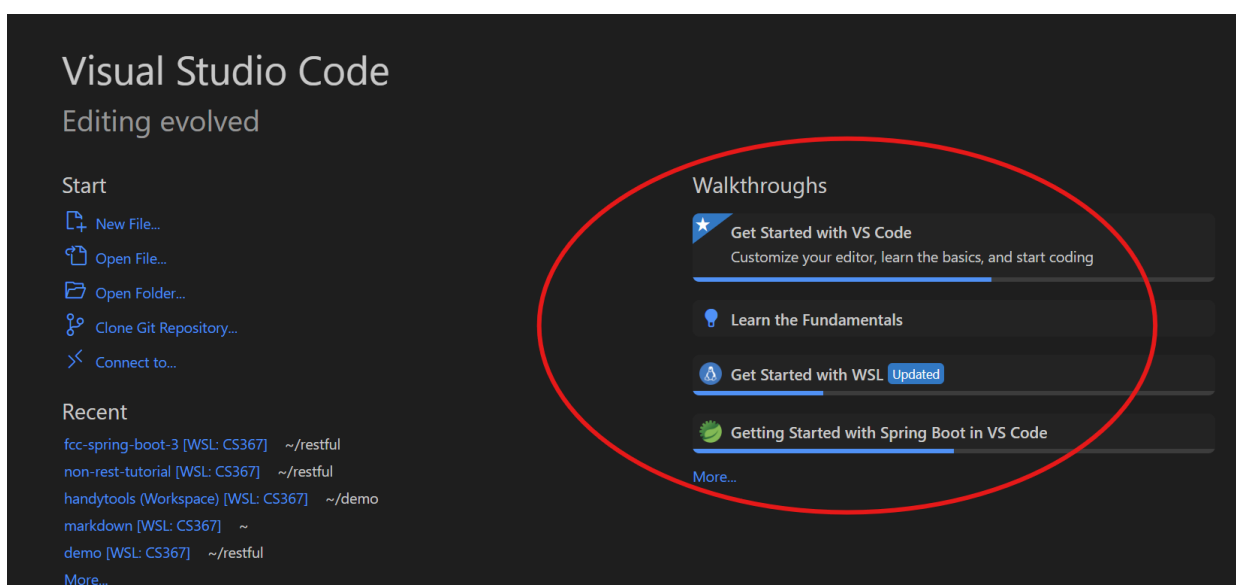
ไฟล์ที่ปรากฏในภาพข้างต้น มีนัยสำคัญดังนี้

- handytools/src/main/java/dev/kritchalach/handytools เป็นตำแหน่งที่คุณใช้เขียนโปรแกรมไฟล์ หรือ src directory ภายใต้ package dev.kritchalach.handytools เป็นไปตามที่ให้ข้อมูลใน Spring Initializer
 - handytools/src/test/xxxx ภายใต้ directory นี้จะเป็น core feature เกี่ยวกับการ unit test ที่ Spring Boot supports
 - handytools/mvnw.cmd เป็น maven script ที่ใช้เรียก application ที่เราเขียนตามการใช้งานของ Maven
 - handytools/pom.xml เป็น project object model ของ maven ที่ generate dependencies ต่างๆ ตามที่เราได้ให้ข้อมูลผ่าน initializer
 - handytools/mvnw เป็นคำสั่งที่ใช้ execute ด้วย parameter ที่เหมาะสมตามที่กำหนดใน .cmd
- ในขั้นตอนนี้หากเป็นการใช้งาน vscode ครั้งแรก ควรต้องตรวจสอบว่าได้มีการติดตั้ง extension ที่เหมาะสมจะช่วยให้ผู้ใช้งานได้เร็วขึ้น สะดวกมากขึ้นหรือไม่
มี extension ที่จำเป็นสำหรับ CS367 เบื้องต้นตามนี้ (หากสะดวกใช้ extension อื่นก็แล้วแต่ชอบ)

Extension	Notes
-----------	-------

	<p>ในกลุ่มนี้โดยปกติ vscode เมื่อตรวจพบว่าโครงการเป็นการ setup Java Dev Environment จะทำการติดตั้งให้อัตโนมัติ</p> <p>หากไม่พบ คือบางครั้งอาจจะเฟลออกด no หรือไม่ทันสังเกต</p>
	<p>หากไม่พบ extension ใด ก็ให้ติดตั้งด้วยตนเอง โดย</p> <ol style="list-style-type: none"> 1. กดไปที่เครื่องหมาย  <ol style="list-style-type: none"> 2. Search หา extension ตามชื่อที่ต้องการอย่างถูกต้อง 3. กดเลือกและทางหน้าจอด้านขวาจะมีปุ่มสีฟ้าที่สามารถเลือกที่จะ install หรือ uninstall ก็ได้ 4. การใช้งาน extension จะเฉพาะที่จำเป็นเพื่อการ demo

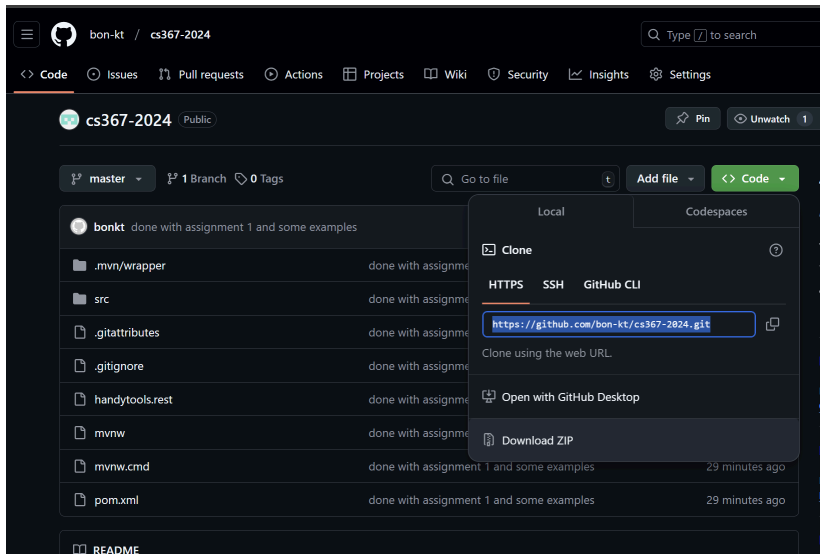
10. ความสำคัญของหน้า welcome คือการที่คุณสามารถกดศึกษา VS code, WSL, Spring Boot และอื่นๆได้ด้วยตัวเอง



PART 2 ใช้โปรแกรมที่ demo บน VS code

Note: โปรแกรมไฟล์ที่เขียนและได้อธิบายใน session ด้วย assumption แบบง่าย มีดังนี้ ก่อนจะดำเนินการต่อควรตรวจสอบทำความเข้าใจ

นศ สามารถไป download ได้จาก <https://github.com/bon-kt/cs367-2024/>



ขั้นตอนการ run Spring Boot ด้วยตัวอย่างที่ให้

*** เอกสารนี้ไม่ได้เขียนเพื่ออธิบายทีละขั้นตอน หรือ เขียนโปรแกรม Java แต่เขียนเพื่อให้ นศ. สามารถที่จะนำ Code ที่ใช้ demo ก่อนที่จะบรรยายไปรัน เพื่อศึกษาและดูผลได้ (ลดการ key หรือทำซ้ำ)

1. ตามภาพข้างต้น หากเลือก Download zip แล้ว unzip หรือ “git clone” จะได้ folder handytool ที่เป็น project directory นี้สัปดาห์ที่ 1 ศึกษาโครงสร้างการออกแบบ โปรแกรม ตามที่ได้บรรยาย เช่น กำหนดให้เป็น design แบบ quick & simple และ เข้าใจว่า code project อันนี้พยายามทำอะไร (ตามที่ได้บรรยาย)
2. อธิบายส่วนที่มีการเขียนโค้ด เป็นไฟล์ๆ ดังนี้

	<p>HandytoolApplication.java</p> <ul style="list-style-type: none">- Code นี้ generate มาให้โดย initializer <p>HandytoolController.java</p> <ul style="list-style-type: none">- เป็น RestController class โดย component นี้จะถูกดูแลจัดการใน context- เรา implement uri endpoint ที่นี้ โดยใช้ DI กับ Spring framework โดยมี configuration managed ด้วย Spring Boot <p>Storage.java</p> <ul style="list-style-type: none">- Entity bean ที่อธิบายรายละเอียดของ record ที่เราเก็บข้อมูลเกี่ยวกับของ ว่าของนั้นอยู่ที่ไหน ใครยืม <p>StorageRepository.java</p> <ul style="list-style-type: none">- Class ที่เรา extend JpaRepository เพื่อให้ใช้ได้กับ Storage record. Spring Framework manage ทั้งหมดให้ <p>LoadDatabase.java</p> <ul style="list-style-type: none">- กำหนดให้เป็น configuration component ที่จะถูก run ครั้งเดียวเมื่อ Spring เริ่มทำงาน (เป็น bean ที่ method interface เรียก StorageRepository เพื่อเก็บ record เริ่มต้น (เนื่องจากเราใช้ H2 in memory database ที่ Framework มีมาให้ ข้อมูลจะหายทุกครั้งที่เมื่อเริ่มรันโปรแกรมใหม่ เราเองก็ไม่ต้องบริหารจัดการข้อมูล jdbc เอง) <p>Handytools.rest</p> <ul style="list-style-type: none">- เราใช้ extension RestClient (คล้ายๆ กับที่มึบน intelliJ) ในการทดสอบและให้ง่ายต่อการจดจำลักษณะ uniform interface ที่เราออกแบบ
--	---

- เมื่อคุณรันโปรแกรมนี้แล้ว Spring Boot configure components ที่เรากำหนดให้เป็น dependencies ดังนั้นเราจึงสามารถแทรก code ด้วยการบอก framework ลงไป เช่น @GetMapping @Configuration เป็นต้น ข้อมูลที่เห็นจะทำให้เราทราบว่า
- เมื่อ HandytoolsApplication เริ่มต้น ระบบ config และ run Apache Tomcat ที่เป็นตัว web server ให้เรา ในไฟล์ application.properties เราสามารถบอกค่าที่เราอยากให้ Tomcat ใช้ได้ เช่น service port เป็นต้น

```
2025-02-07T15:54:00.222+07:00 INFO 35972 --- [handytools] [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing
```

- Spring ได้เริ่มใช้ JPA และ H2 database ในแบบ in-memory (ไม่เก็บข้อมูลลง persistence) และมีการสร้าง DataSource ให้ มีการเชื่อมต่อผ่าน jdbc ที่ random ขึ้นมา ในเคสตัวอย่างนี้คือประโยค
- H2 console available at '/h2-console'. Database available at jdbc:h2:mem:db2712cf-2026-49cf-8c92-b0e6fc69561e'

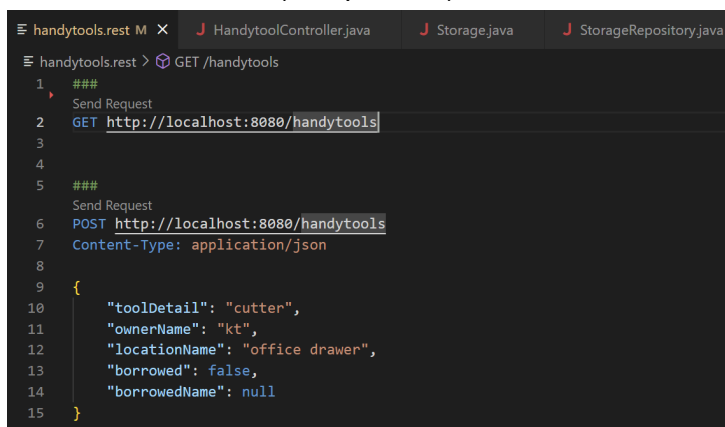
```
2025-02-07T15:54:01.460+07:00 WARN 35972 --- [handytools] [ restartedMain] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.d
open-in-view is enabled by default. Therefore, database queries may be performed during view rendering. Explicitly configure spring.j
pa.open-in-view to disable this warning
2025-02-07T15:54:01.620+07:00 INFO 35972 --- [handytools] [ restartedMain] o.s.b.a.h2.H2ConsoleAutoConfiguration : H2 console a
available at '/h2-console'. Database available at 'jdbc:h2:mem:db2712cf-2026-49cf-8c92-b0e6fc69561e'
2025-02-07T15:54:01.669+07:00 INFO 35972 --- [handytools] [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload s
server is running on port 35729
2025-02-07T15:54:01.690+07:00 INFO 35972 --- [handytools] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat start
ed on port 8080 (http) with context path '/'
2025-02-07T15:54:01.695+07:00 INFO 35972 --- [handytools] [ restartedMain] d.k.handytools.HandytoolsApplication : Started Hand
ytoolsApplication in 2.944 seconds (process running for 3.19)
2025-02-07T15:54:01.763+07:00 INFO 35972 --- [handytools] [ restartedMain] d.kritchalach.handytools.LoadDatabase : loading Stor
```

เห็นได้จาก log ดังกล่าว่า หลังจากนั้น ก็ไปทำ LoadDatabase ให้เรา เพื่อเริ่มเอาข้อมูลเข้าไปใส่ใน in-memory database ดังกล่าว

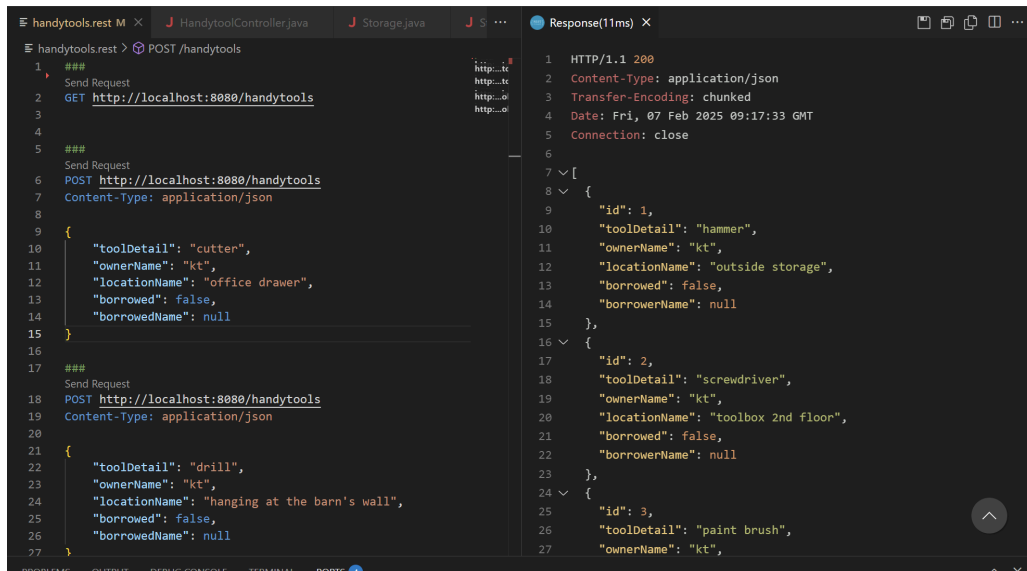
5. เมื่อต้องการหยุดการรัน ท่านสามารถกด **Ctrl - C** ที่ terminal หรือกด stop icon ได้ ขึ้นกับวิธีการที่ใช้ในการรัน

4. การทดสอบด้วย RestClient extension ที่ติดตั้งใน VS Code

- a. ที่ Project folder จะมีไฟล์ handytools.rest ที่ extension ดังกล่าวจะสามารถอ่านและเปิดให้เราทำคำสั่งไปที่ uri / url endpoint RestClient extension support json และ http ดังนั้นจึงสามารถทำอะไรได้อีกหลายอย่าง
- b. การเขียนเคสที่จะทดสอบ
 - i. แต่ละคำสั่ง GET POST PUT DELETE จะกระทำกับ uri ที่เป็น resource โดยท่านสามารถกำหนด parameter header body อะไรต่างๆ ตามที่ได้บรรยายไปแล้ว ตามมาตรฐาน http
 - ii. ในส่วนที่ต้องกำหนด body ของ Request เราต้องกำหนด encoding เพื่อให้สามารถเข้าใจได้ระหว่างผู้ส่งกับผู้รับ
- c. ในไฟล์ handytools.rest หากมี extension ที่เหมาะสมติดตั้ง เมื่อเปิดไฟล์นี้มา edit บน VS code จะมี link “Send Request” เหนือคำสั่งที่ได้กำหนดไว้ ท่านสามารถกดเพื่อทดสอบ โดยผลที่ได้จะไปอยู่ที่ window area ข้างขวา (ส่วน preview)



- d. รูปภาพตัวอย่างเมื่อกด Send Request เหนือบรรทัดที่ 2 การแก้ไขเคสที่จะใช้ทดสอบกับ RESTful web service ที่เขียน สามารถทดสอบด้วยวิธีเหล่านี้ได้ แต่ก็ยังมีวิธีอื่นๆ เช่นใช้ postman, curl หรือ browser ในการทดสอบก็ได้ แต่การส่ง assignment จะบังคับให้ทดสอบด้วยวิธีนี้ ที่สำคัญ นิสิตสามารถเขียนอธิบายหรือ ใส่ประโยคที่ต้องการถามหรือบ่นมาได้ ด้วยการใส่ # ไว้ที่หน้าบรรทัดนั้นๆ



The screenshot shows a VS Code editor with a REST client file named 'handytools.rest'. It contains two requests to 'http://localhost:8080/handytools'. The first is a GET request, and the second is a POST request with a JSON body. The response for the POST request is shown on the right, indicating a 200 status and a JSON array of three tool objects.

```
handytools.rest
1  ###
2  Send Request
3  GET http://localhost:8080/handytools
4
5  ###
6  Send Request
7  POST http://localhost:8080/handytools
8  Content-Type: application/json
9
10 {
11   "toolDetail": "cutter",
12   "ownerName": "kt",
13   "locationName": "office drawer",
14   "borrowed": false,
15   "borrowerName": null
16 }
17
18 ###
19 Send Request
20 POST http://localhost:8080/handytools
21 Content-Type: application/json
22
23 {
24   "toolDetail": "drill",
25   "ownerName": "kt",
26   "locationName": "hanging at the barn's wall",
27   "borrowed": false,
28   "borrowerName": null
29 }
```

```
Response(11ms)
1  HTTP/1.1 200
2  Content-Type: application/json
3  Transfer-Encoding: chunked
4  Date: Fri, 07 Feb 2025 09:17:33 GMT
5  Connection: close
6
7  [
8    {
9      "id": 1,
10     "toolDetail": "hammer",
11     "ownerName": "kt",
12     "locationName": "outside storage",
13     "borrowed": false,
14     "borrowerName": null
15   },
16   {
17     "id": 2,
18     "toolDetail": "screwdriver",
19     "ownerName": "kt",
20     "locationName": "toolbox 2nd floor",
21     "borrowed": false,
22     "borrowerName": null
23   },
24   {
25     "id": 3,
26     "toolDetail": "paint brush",
27     "ownerName": "kt",
```

Notes: คำสั่งที่เป็นประโยชน์

1. curl เป็นคำสั่งบน Linux ที่มีติดตั้งอยู่ในทั้ง Windows 11 และ MacOS ได้ด้วย curl ช่วยทำให้การทดสอบสามารถทำได้ ระหว่างเครื่อง นอก VS code
2. jps - list the instrumented JVMs on the target system เวลาที่รัน รันโครงการหรือ Code ที่ตนเขียนอาจจะรันช้า และไม่ได้ kill หรือหยุด โปรแกรมเหล่านั้นก่อนทำให้ port network ในการใช้งานไม่สามารถเปิดได้ เป็นต้น jps จะบอก process id และ โปรแกรม java ที่กำลังทำงานอยู่
3. Json_pp เป็นโปรแกรมที่ช่วย format jsonให้อ่านได้ง่ายๆ