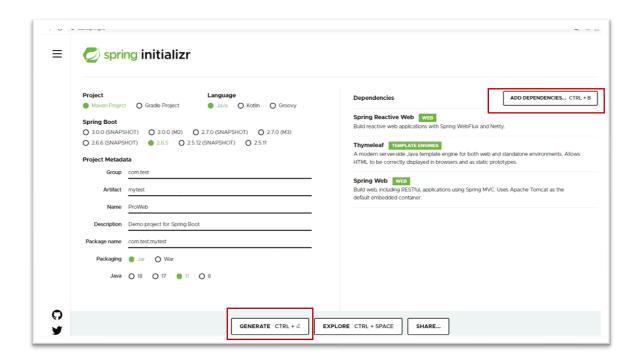
## Spring-boot lab01 \_ProWeb

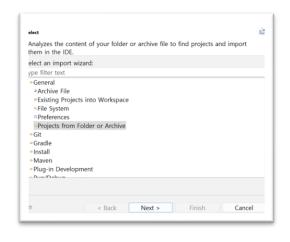
[STEP 01] start.spring.io 에서 아래와 같이 선택한다.

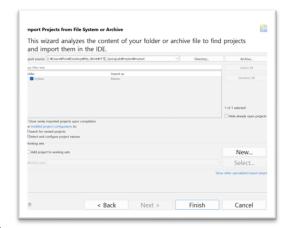
- 1-1. ADD DEPENDENCIES 클릭 -> Dependencies를 추가한다.
- 1-2. Project Metadata를 화면과 같이 채운다.
- 1-3. GENERATE 클릭 -> 다운로드 한 후 압축을 푼다.



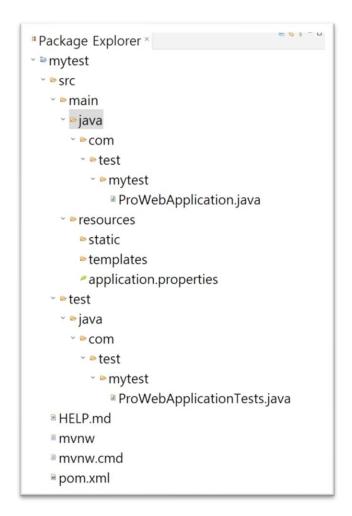
[STEP 02] 프로젝트를 임포트 한다.

2-1. STS에 임포트 한다.

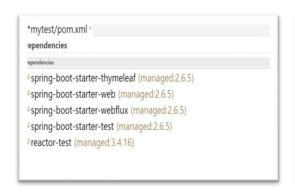




## 2-2. 완성형 구조 폴더를 확인한다.



# 2-3. pom.xml 확인





## [STEP 03] 클래스를 추가한다.

3-1. com.test.mytest.Address 추가 \_일반적인 setter&getter (POJO)형식의 클래스

```
*Package Explorer ×
~ ≥ mytest [boot]
 src/main/java
   com.test.mytest
     Address.java
       <sup>o</sup> Address
         · addr
         • name
         • tel
         Address(String, String, String)
         • getAddr() : String
          • getName() : String
         • getTel() : String
         * setAddr(String) : void
         * setName(String) : void
          setTel(String) : void
         *toStrina(): Strina
```

3-2 . com.test.mytest.MyService 추가

```
*MyService.java *
 1 package com.test.mytest;
 2-import java.time.Duration;
 3 import java.util.Arrays;
 4 import java.util.List;
 5 import java.util.Random;
6 import org.springframework.stereotype.Service;
 7 import reactor.core.publisher.Flux;
 8
 9 @Service
10 public class MyService {
11
       private Random my_ran=new Random();
12
13
       public Flux<Address> getAllAddress() {
14
            return Flux.<Address> generate(sink -> sink.next(randomAddress()))
15
                      .delayElements(Duration.ofMillis(250));
16
       }
17
18-
       private Address randomAddress() {
19
            return all.get(my_ran.nextInt(all.size()));
20
21
22
       private List<Address> all = Arrays.asList(
            new Address("Hong-gil dong", "seoul", "02-000,000"),
new Address("jung-gil dong", "inchon", "031-000,000"),
new Address("Pak-gil dong", "pusan", "051-000,000")
23
24
25
26
       );
27
28 }
29
```

## 3-3. com.test.mytest.MyController추가

```
Package Explorer ×
                                   MvController.java ×
                                    1 package com.test.mytest;
<sup>⋄</sup> src/main/java
                                     3 import org.springframework.beans.factory.annotation.Autowired;

<sup>™</sup> com.test.mytest
  Address.java
                                    10 @RestController
   MyController.java
                                    11 public class MyController {
  MvService.iava
                                    12°
                                            @Autowired
  ProWebApplication.java
                                    13
                                             private MyService myservice;
src/main/resources
                                    14
  static
                                    15
                                             @GetMapping(value="/server",produces=MediaType.TEXT_EVENT_STREAM_VALUE)
  templates
                                    16
                                             Flux<Address> serviceAddress(){
  application.properties
                                    17
                                                   return this.myservice.getAllAddress();
src/test/java
                                    18
→ ■JRE System Library [JavaSE-11]
                                    19 }
Maven Dependencies
                                    20
```

# 3-4. 만들어진 ProWebApplication.java -> 오.마 -> Run As -> SpringBoot APP 실행

#### 이때 Console에 port 확인

```
Package Explorer
                                  MyController.java ProWebApplication.java ×
1 package com.test.mytest;
src/main/java
                                    3*import org.springframework.boot.SpringApplication;
~ 

    com.test.mytest

  Address.java
                                   6 @SpringBootApplication
  MyController.java
                                   7 public class ProWebApplication {
  MvService.iava
  ProWebApplication.java
                                          public static void main(String[] args) {
                                   9.
src/main/resources
                                   10
                                              SpringApplication.run(ProWebApplication.class, args);
 static
                                  11
 templates
                                  12
  application.properties
src/test/java
                                  Problems Javadoc Declaration Console ×
* JRE System Library [JavaSE-11]
                                                   [Spring Boot App] D:#sts-4.13.1.RELEASE#plugins#ora.ed
Maven Dependencies
                                  2022-03-28 21:05:51.769 INFO 17344 --- [
                                                                                    mainl com.test.mv
bin
                                  2022-03-28 21:05:51.772 INFO 17344 --- [
                                                                                        main] com.test.my
src
                                                                                         main] o.s.b.w.emb
main] o.apache.ca
                                  2022-03-28 21:05:52.386 INFO 17344 --- [
target
                                  2022-03-28 21:05:52.392 INFO 17344 --- [
B HELP.md
                                                                                        main] org.apache.
                                  2022-03-28 21:05:52.393 INFO 17344 --- [
mvnw
                                                                                         main] o.a.c.c.C.[
main] w.s.c.Servl
                                  2022-03-28 21:05:52.504 INFO 17344 --- [
mvnw.cmd
                                  2022-03-28 21:05:52.504 INFO 17344 --- [
 ■ pom.xml
                                  2022-03-28 21:05:52.773 WARN 17344 --- [
                                                                                         main] ion$Default
                                  2022-03-28 21:05:52.862 INFO 17344 --- [
SBJdbc
                                                                                        main] o.s.b.w.emb
■ test
                                  2022-03-28 21:05:52.870 INFO 17344 --- [
                                                                                          main] com.test.my
                                  2022-03-28 21:05:59.370 INFO 17344 --- [nio-8080-exec-2] o.a.c.c.C.[
                                  2022-03-28 21:05:59.370 INFO 17344 --- [nio-8080-exec-2] o.s.web.ser
                                  2022-03-28 21:05:59.370 INFO 17344 --- [nio-8080-exec-2] o.s.web.ser
3oot Dashboard ×
                                  2022-03-28 21:06:13.136 INFO 17344 --- [on(2)-127.0.0.1] inMXBeanReg
```

## 3-5 . 브라우저 확인

```
← → C i localhost:8080/server
lata:{"name":"Pak-gil dong","addr":"pusan","tel":"051-000,000"}
lata:{"name":"Pak-gil dong","addr":"pusan","tel":"051-000,000"}
lata: {"name": "Pak-gil dong", "addr": "pusan", "tel": "051-000,000"}
lata:{"name":"Pak-gil dong","addr":"pusan","tel":"051-000,000"}
lata:{"name":"Pak-gil dong", "addr":"pusan", "tel":"051-000,000"}
lata:{"name":"jung-gil dong","addr":"inchon","tel":"031-000,000"}
lata: {"name": "Pak-gil dong", "addr": "pusan", "tel": "051-000,000"}
lata:{"name":"Pak-gil dong","addr":"pusan","tel":"051-000,000"}
lata:{"name":"Hong-gil dong","addr":"seoul","tel":"02-000,000"}
lata:{"name":"Pak-gil dong", "addr":"pusan", "tel":"051-000,000"}
lata:{"name":"Hong-gil dong","addr":"seoul","tel":"02-000,000"}
lata:{"name":"Hong-gil dong","addr":"seoul","tel":"02-000,000"}
lata:{"name":"Pak-gil dong","addr":"pusan","tel":"051-000,000"}
lata: {"name": "Pak-gil dong", "addr": "pusan", "tel": "051-000,000"}
lata:{"name":"jung-gil dong","addr":"inchon","tel":"031-000,000"}
lata:{"name":"Hong-gil dong","addr":"seoul","tel":"02-000,000"}
lata: {"name": "Hong-gil dong", "addr": "seoul", "tel": "02-000,000"}
lata:{"name":"Hong-gil dong","addr":"seoul","tel":"02-000,000"}
lata:{"name":"Hong-gil dong","addr":"seoul","tel":"02-000,000"}
lata:{"name":"jung-gil dong","addr":"inchon","tel":"031-000,000"}
lata: {"name": "Hong-gil dong", "addr": "seoul", "tel": "02-000,000"}
lata:{"name":"Hong-gil dong","addr":"seoul","tel":"02-000,000"}
lata:{"name":"jung-gil dong","addr":"inchon","tel":"031-000,000"}
```

#### [STEP 04] 클래스를 추가한다.

```
Package Explorer *
                                     ServletController.java ×
                                      1 package com.test.mytest;
≥ mytest [boot]
src/main/java
                                      3 import org.springframework.stereotype.Controller;
 com.test.mytest
                                      4 import org.springframework.web.bind.annotation.GetMapping;
  Address.java
  MyController.java
                                      6 import reactor.core.publisher.Mono;
  MyService.java
  ProWebApplication.java
                                      8 @Controller
   ServletController.java
                                      9 public class ServletController {
src/main/resources
                                     10
                                               @GetMapping
  static
                                     11
                                               Mono<String> Indexhome(){
 · etemplates
                                     12
                                                   return Mono.just("home");
   home.html
                                     13
  application.properties
                                     14 }
src/test/java
                                     15

■JRE System Library [JavaSE-11]
Maven Dependencies
```

# [STEP 05] 웹페이지를 추가한다. (오타조심)

5-1. /src/main/resources/templates/home.html 로 페이지 추가 후 내용 작성

```
home.html ×
Package Explorer ×
                                     1 <! DOCTYPE html>
≥ mytest [boot]
                                     2-<html lang="en" xmlns: th ="http://www.thymeleaf.org">
src/main/java
 * * com.test.mytest
                                    4 <title> My _home </title>
  Address.java
                                    5 </head>
  MyController.java
                                    6-<body>
  MyService.java
                                          <h1>My_First Spring boot Page</h1>
  ProWebApplication.java
                                    8

■ ServletController.java

                                    9- 
src/main/resources
                                   10
                                           Spring -Boot is ?
  static
                                   11
 • templates
                                   12
   home.html
                                    13 </body>
  application.properties
                                    14 </html>
src/test/java
                                   15
► JRE System Library [JavaSE-11]
                                   16
 Maven Dependencies
```

# [STEP 06] 실행 후 브라우저 확인

6-1. 만들어진 ProWebApplication.java -> 오.마 -> Run As -> SpringBoot APP 실행

