



# [스파르타코딩클럽] 웹개발의 봄, Spring - 1주차



매 주차 강의자료 시작에 PDF파일을 올려두었어요!

## ▼ 단축키 모음

### ▼ 코드정렬

- Windows: `Ctrl + Alt + L`
- macOS: `option + command + L`

### ▼ 임포트

- Windows: `Alt + Enter`
- macOS: `option + Enter`

### ▼ 주석

- Windows: `Ctrl + /`
- macOS: `command + /`

### ▼ System.out.println()

- 공통: sout 치고 tab 누르기

### ▼ 환경설정

- Windows: `Ctrl + Alt + S`
- macOS: `command + ,`

## [수업 목표]

1. 웹의 기본적인 구조를 이해하고, 5주 동안 만들 프로젝트를 살펴본다.
2. 스프링 부트의 기본 구조를 이해한다.
3. Java 기초를 학습한다.

## [목차]

01. [1주차] 이번 주 배울 것
02. 필수 프로그램 설치
03. [1주차] 일단 스프링 작동시키기
04. [1주차] 웹의 기본 개념
05. [1주차] 자바 기초 문법 - 1
06. [1주차] 자바 기초 문법 - 2
07. [1주차] 자바 기초 문법 - 3
08. [1주차] 자바 기초 문법 - 4
09. [1주차] 자바 클래스
10. [1주차] Getter, Setter
11. [1주차] 브라우저에 바로 나타내보기
12. [1주차] 그레이들(Gradle)이란?
13. [1주차] 끝 & 숙제설명
14. 1주차 숙제 답안 코드



모든 토큰을 열고 닫는 단축키

Windows : `ctrl + alt + t`

Mac : `⌘ + ⌘ + t`

## 01. [1주차] 이번 주 배울 것

### ▼ 우리가 다룰 기술

- 웹서비스?



웹서비스를 만든다는 건, 클라이언트와 서버 모두 만들어야 한다는 것을 뜻합니다.  
앞으로 우린 이 둘을 모두 직접 만들어 웹서비스를 완성해 나갑니다.

#### 클라이언트는 사용자가 보는 화면입니다.

웹사이트로 따지면 크롬 브라우저나 사파리, 익스플로러가 해당되겠고  
앱으로 따지면 여러분들 휴대폰에 많이 존재하는 어플, 즉 앱 자체가 해당됩니다

서버는 여러분들이 원하는 데이터가 있는 곳이고,  
원하는 데이터를 요청하면 이에 응답해주는 곳이기도 합니다.

웹서비스의 기본 구조는, 클라이언트에서 원하는 데이터가 있을 때마다 서버에 요청하고, 응답 데이터를 받아  
사용자에게 보여줍니다

그렇다면 우린 사용자들이 보는 웹 화면을 만들고  
필요한 데이터를 요청할 수 있는 서버를 만들면 되겠네요? 간단합니다!

- 자바란?



컴퓨터에게 명령을 내리기 위한 소통 도구를 "프로그래밍 언어"라고 하는데 그 중의 하나입니다. 위 그림은 <스택오버플로우(Stack Overflow)>라는 대표적인 개발자 커뮤니티에서 조사한 기술의 사용률인데요.  
Javascript, Python과 함께 여전히 최고의 사용률을 보여주고 있습니다.

<https://insights.stackoverflow.com/survey/2019>



## Most Popular Technologies

### Programming, Scripting, and Markup Languages



- 스프링이란?



위에서 말씀드린 "자바"라는 언어로 서버를 만드는데 필요한 녀석입니다. 훌륭한 개발자들이 미리 필요한 것들을 만들어두었고, 우리는 필요한 부분들만 쑥쑥 채워나가면 됩니다 😎

#### ▼ 1~5주차 배울 순서

- 1주차: Java 기초 문법, Spring 기초 사용법
  - 산에 오르려면 기초 체력을 다져야겠죠? 스프링을 다루기 위해서 꼭 필요한 자바 지식만을 배웁니다.
- 2주차: RDBMS, SQL, Lombok, JPA 사용해보기
  - 정보를 저장하는 데이터베이스를 배우고, 이를 자바로 다루는 표준 방법인 JPA를 써먹어봅니다.
- 3주차: 익명 타임라인 페이지 만들어보기
  - 익명의 타임라인 서비스를 만들어봄으로써 Spring 에 익숙함을 느낍니다.
- 4주차: 나만의 셀렉샵 만들어보기
  - 네이버 검색 API를 이용해서 조금 더 고도화된 웹서비스를 만들어봅니다.
- 5주차: 프로젝트 마무리 & AWS 배포
  - 클라우드에 서비스를 배포하고, 친구들이 들어올 수 있도록 주소를 붙인 뒤 카톡으로 공유해봅니다.

#### ▼ 만들 서비스 살펴보기

##### 1. [코드스니펫] 익명 타임라인 주소

<http://spring.spartacodingclub.kr/timeline>

##### 2. [코드스니펫] 나만의 셀렉샵 주소

<http://spring.spartacodingclub.kr/>

▼ 제일 중요한 마음가짐 / 오류가 났을 땐?



### "해봤다"는 경험을 손에 남기자, 모든 지식을 완벽히 머리에 남길 필요가 없다!

(X) 상속, 추상화, 인터페이스에 대해 달달 암기하고, 세세한 부분까지 신경쓴다.

(O) 꼭 필요한 클래스, 메소드의 핵심만 기억한다. 뭘 모르는지만 기억해둔다.

(X) 스프링/자바를 다루다가 아주 조금이라도 모르는게 나오면 복잡한 개념을 파고든다.

(O) 만드는 재미, 오류를 해결하는 재미에 집중하고, 나중에 공백을 채워나간다.



### 실력을 쌓는 길, 스스로 해결해보기

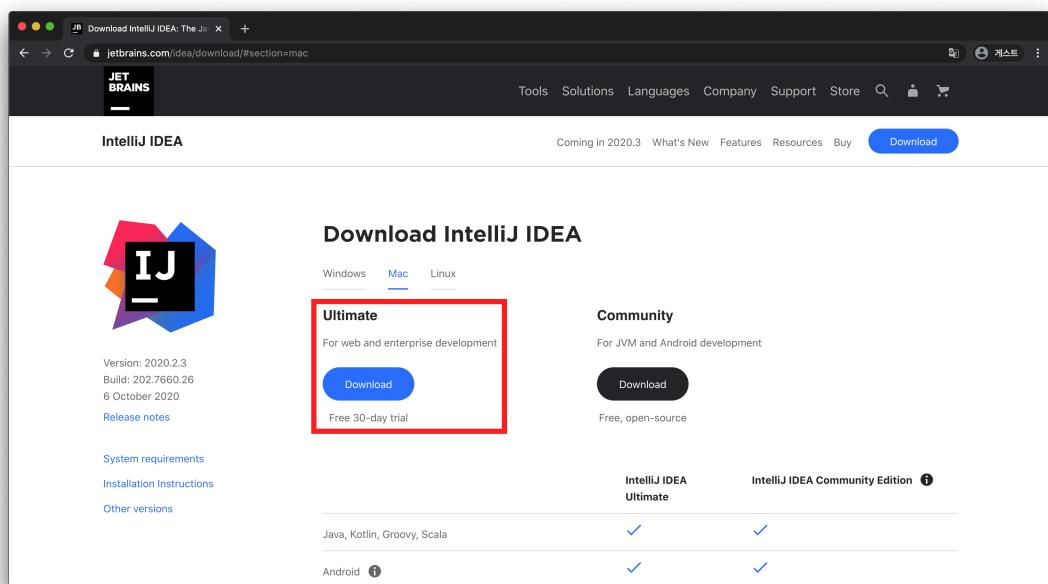
1. 오류가 나면 당황하지 말고, 에러 메시지를 차분하게 읽기!

2. 잘 모르겠다면, 에러 메시지 짚어다 구글링해보는 습관 기르기

## 02. 필수 프로그램 설치

▼ IntelliJ IDEA Ultimate ([다운로드 링크](#))

▼ ([← 놀러보기](#)) IntelliJ 설치 가이드



위 그림과 같이 Ultimate 버전을 다운로드 받은 후 [설치까지 마무리](#) 해주세요.

- "설치"만 해두세요!
- 1주차 강의에서 Activation 하는 방법을 알려드립니다~!

▼ Java

▼ Windows

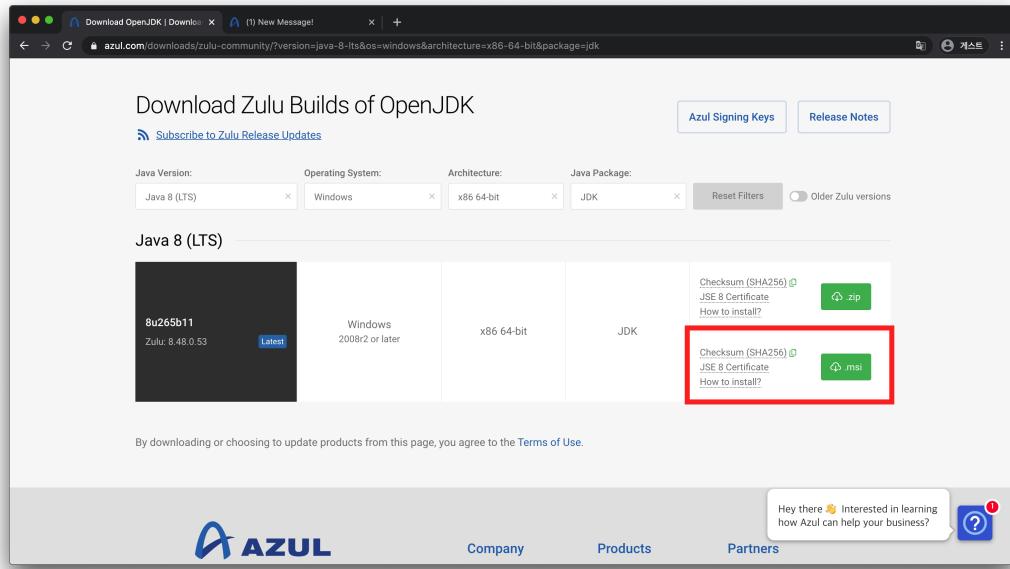
▼ [코드스니펫] [Windows Java 8 다운로드 링크](#)

<https://www.azul.com/downloads/zulu-community/?version=java-8-lts&os=windows&architecture=x86-64-bit&package=jdk>

▼ [코드스니펫] JAVA\_HOME 변수 값

C:\Program Files\Zulu\zulu-8

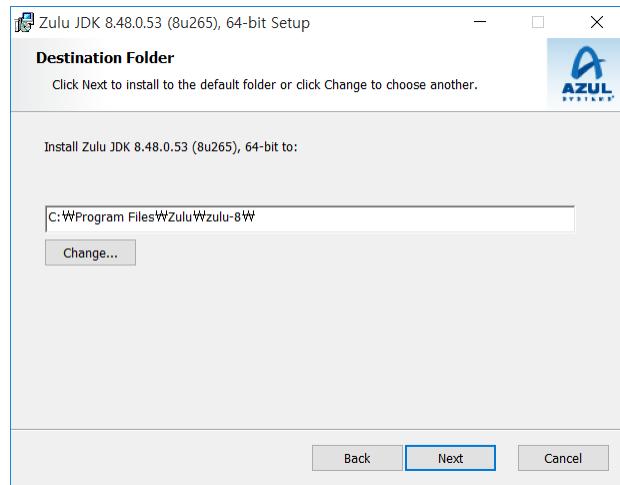
1. (다운로드 링크) 접속 후 [.msi] 버튼을 클릭하여 설치 파일을 다운로드 합니다.



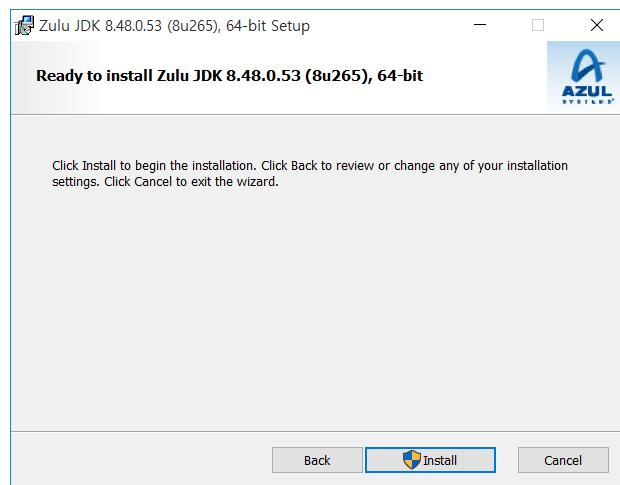
2. 설치 파일을 실행하고 Next를 클릭합니다.



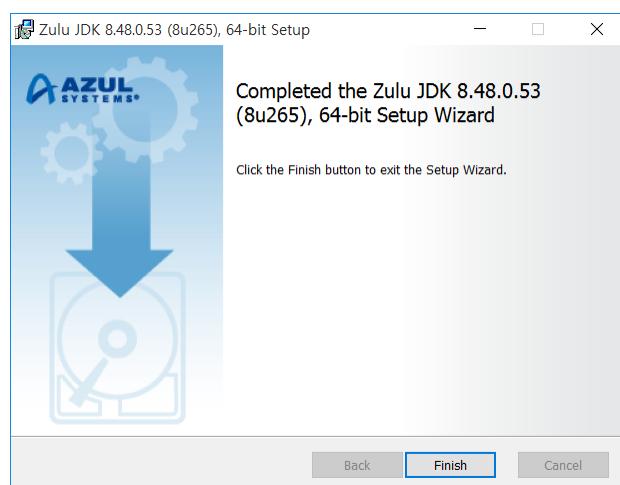
3. Next를 클릭합니다.



4. Install을 클릭합니다.



5. Finish를 클릭합니다.

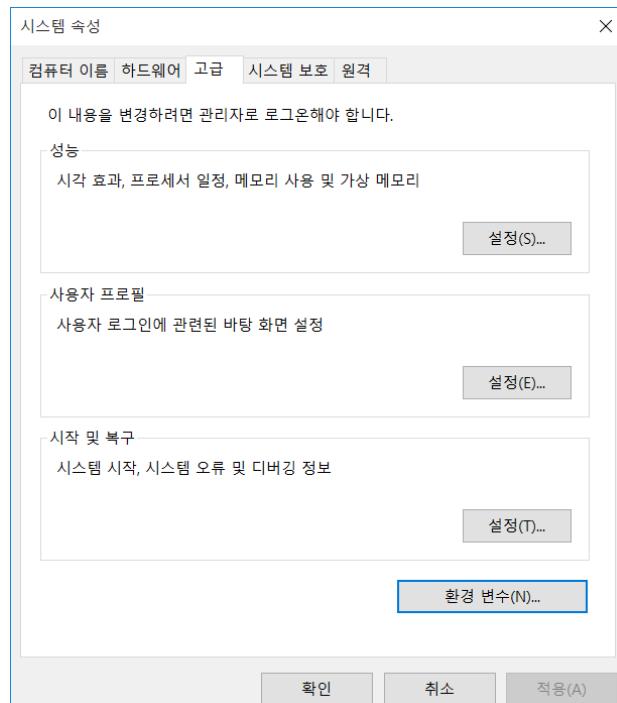


6. 탐색기 > 내 PC 마우스 우클릭 > 속성을 클릭합니다.

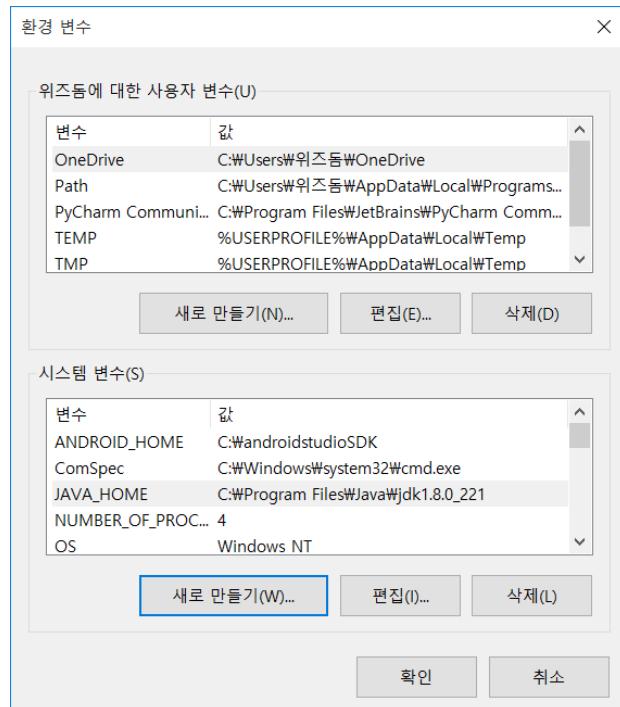
7. 좌측 메뉴 최하단의 "고급 시스템 설정"을 클릭합니다.



8. "환경 변수"를 클릭합니다.



9. 하단 <시스템 변수> 영역의 "새로 만들기"를 클릭합니다.



10. 변수 이름을 JAVA\_HOME, 값을 아래와 같이 입력합니다.

```
C:\Program Files\Zulu\zulu-8
```

11. 마찬가지로 <시스템 변수> 영역의 "Path"를 찾고 "편집"을 클릭합니다. 제일 끝에 아래 내용을 입력합니다.

```
;C:\Program Files\Zulu\zulu-8\bin
```

12. '확인'을 누르고 빠져나온 뒤, 원도우키 + R 을 누르고 cmd 입력 후 엔터를 누릅니다.

13. 아래 내용을 입력했을 때 캡처와 같이 OpenJDK 관련 정보가 표시되면 설치 완료입니다.

```
java -version
```

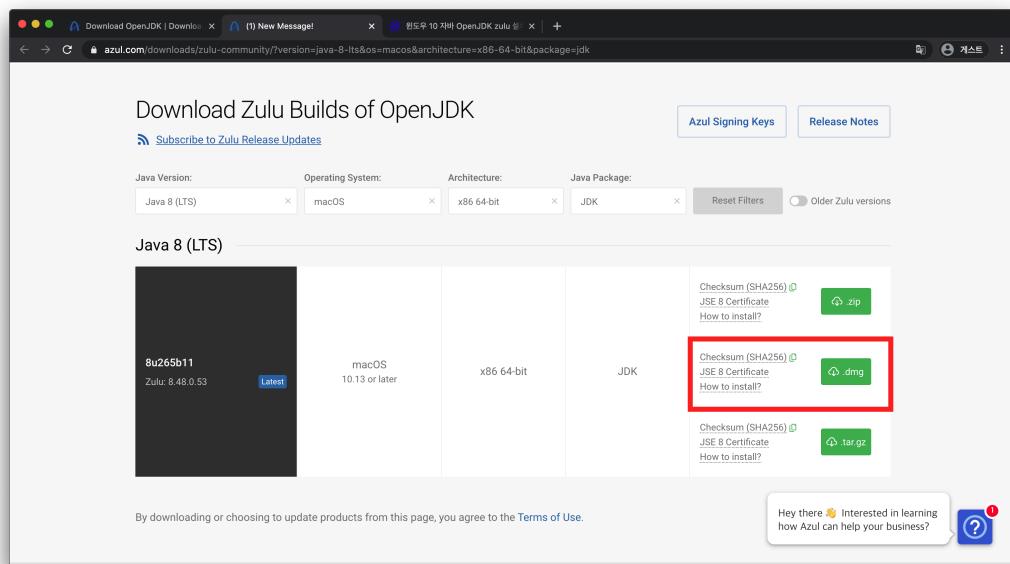
```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\위즈동>java -version
openjdk version "1.8.0_265"
OpenJDK Runtime Environment (Zulu 8.48.0.53-CA-win64) (build 1.8.0_265-b11)
OpenJDK 64-Bit Server VM (Zulu 8.48.0.53-CA-win64) (build 25.265-b11, mixed mode)

C:\Users\위즈동>
```

#### ▼ Mac

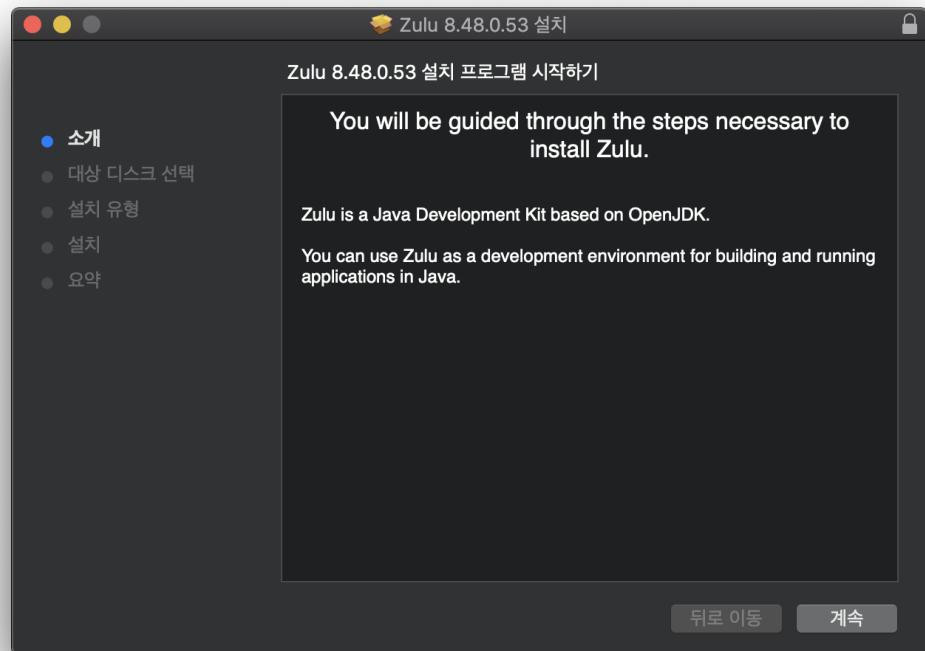
1. ([다운로드 링크](#)) 접속 후 (.dmg 버튼)을 클릭하여 설치 파일을 다운로드 합니다.



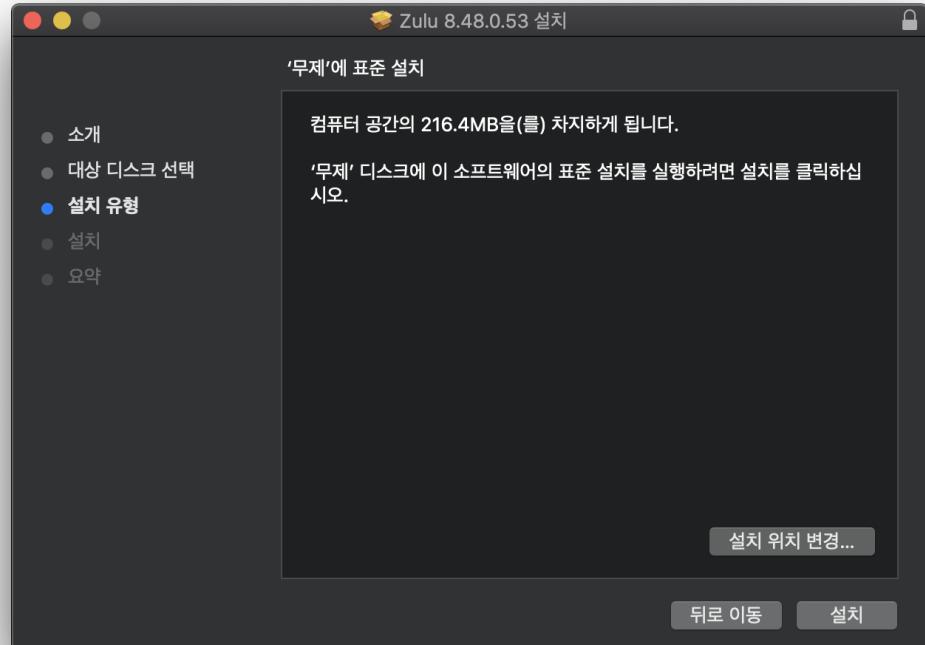
2. 설치파일을 더블클릭하여 실행한 뒤, 오른쪽 상자를 다시 더블클릭합니다.



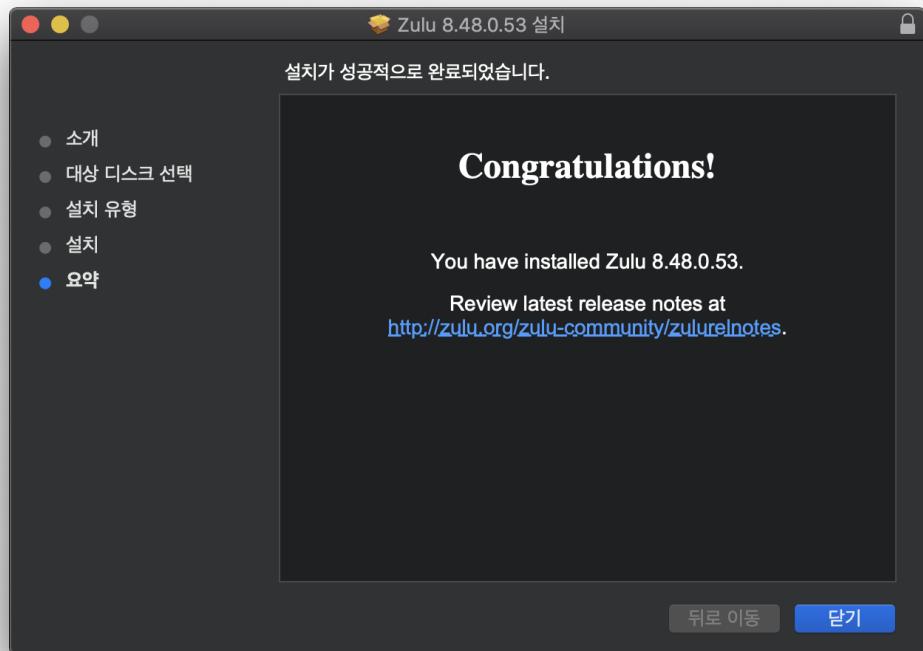
3. '계속'을 클릭합니다



4. '설치'를 클릭합니다



5. '닫기'를 클릭합니다.



6. 터미널을 켜고, 아래 코드를 입력해서 아래 창과 결과가 나오면 설치 완료입니다.

```
java -version
```

A screenshot of a terminal window titled "byeongkwan — byeongkwan@Byeongkwan-MacBook-Pro — ~ — zsh — 85x13". It shows the command "java -version" being run and its output: "openjdk version "1.8.0\_265"" followed by details about the OpenJDK Runtime Environment and Server VM. The prompt "(base)" appears twice.

- Advanced REST Client ([다운로드 링크](#))

### 03. [1주차] 일단 스프링 작동시키기

#### ▼ 인텔리제이(IntelliJ)란?

- MS Word 가 문서작업하는데 특화된 메모장인 것처럼, 자바로 코드를 작성하는데 특화된 메모장입니다.

#### ▼ 스프링 프로젝트 준비하기

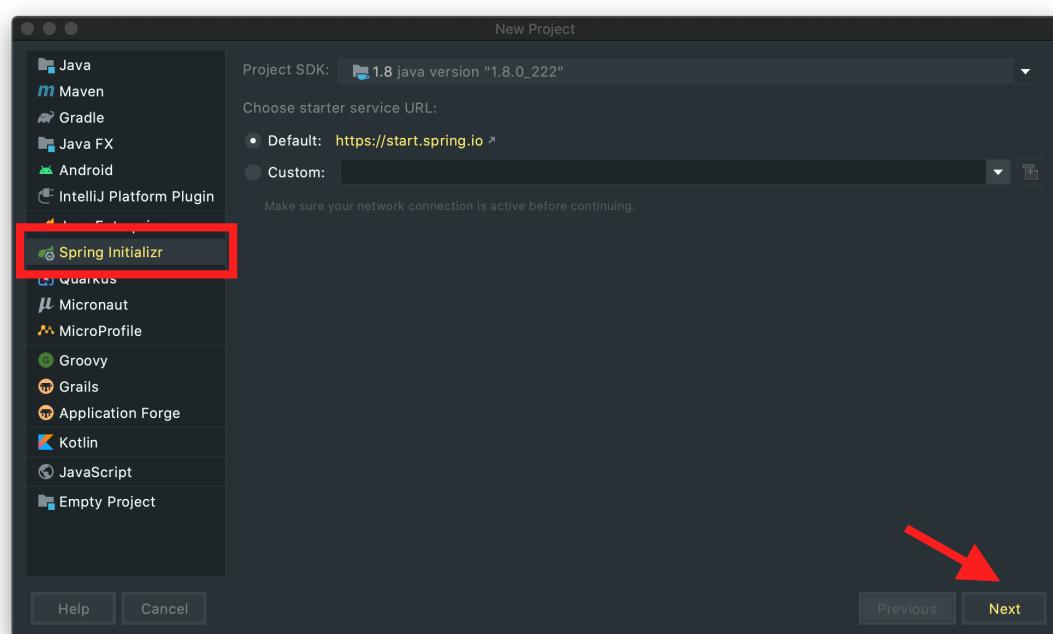


팁) 이미지를 더블클릭하시면 크게 보실 수 있어요 😎

1. 인텔리제이를 실행합니다.
2. New Project를 클릭합니다.



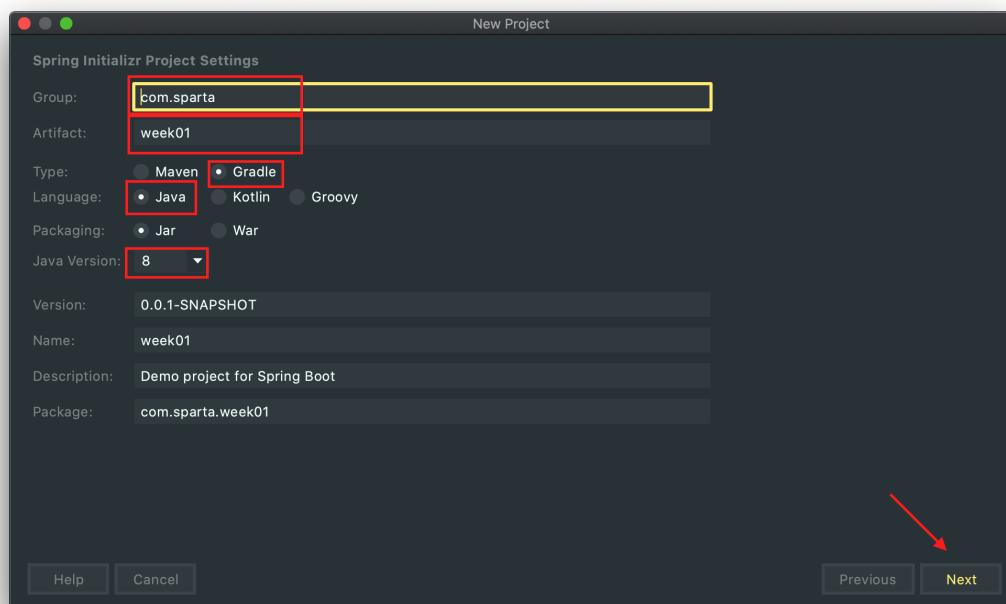
3. 왼쪽 메뉴에서 "Spring Initializr"를 클릭하고 "Next"를 클릭합니다.



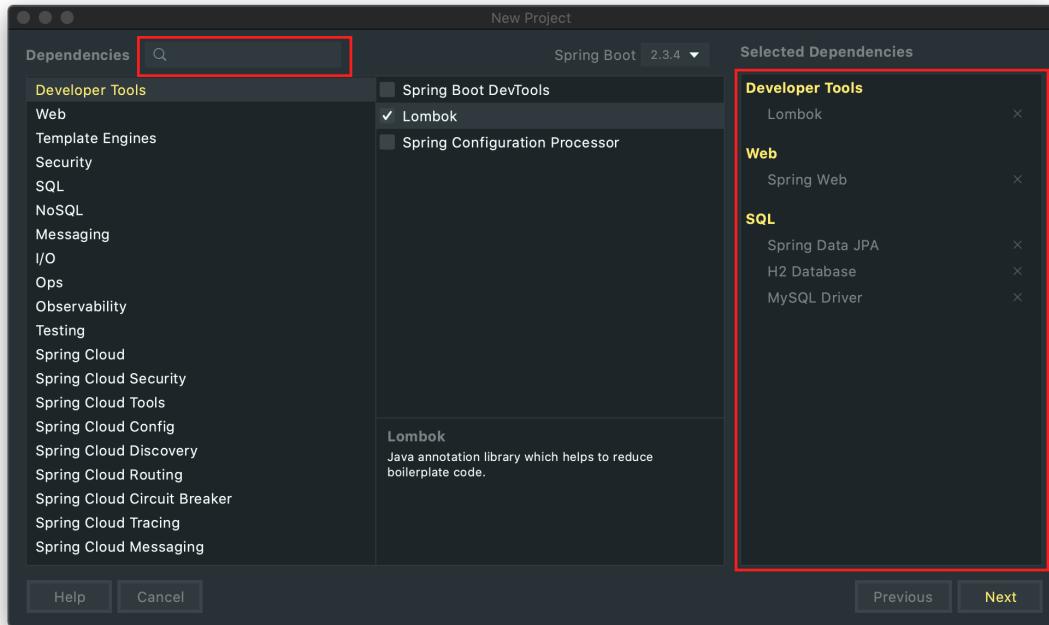
4. 🚨[중요] 꼭 다음 사항을 확인해주세요.

- Group: com.sparta
- Artifact: week01
- **Type: Gradle**
- **Language: Java**
- **Java Version: 8**

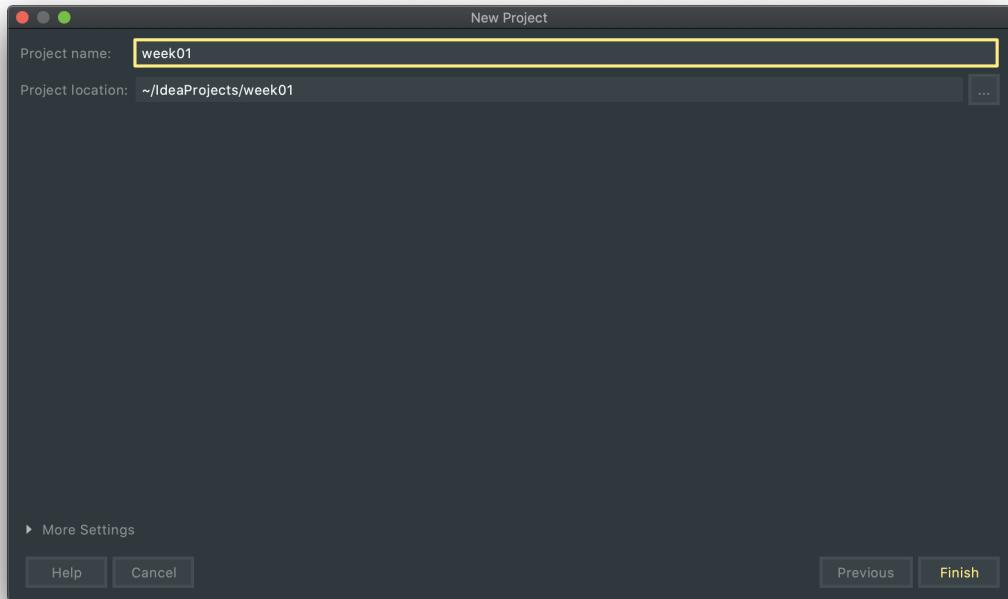
특히 Type, Language, Java version 중 하나라도 다르면 정상적으로 실행이 되지 않습니다. 꼭 정상입력 확인해주세요!



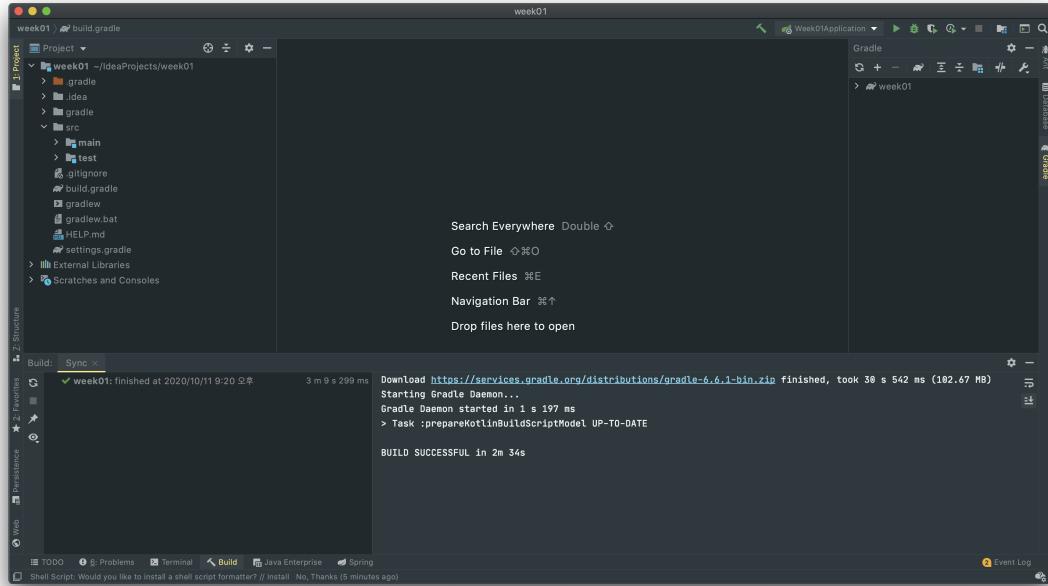
5. 검색창을 클릭하고, 다음을 차례대로 검색한 뒤 엔터를 눌러주세요. 검색이 끝난 뒤 아래 캡처의 우측과 같이 5개의 요소가 포함되어있으면 완료된 것입니다.
- Lombok
  - Spring Web
  - Spring Data JPA
  - H2 Database
  - MySQL Driver



6. Finish를 클릭합니다.



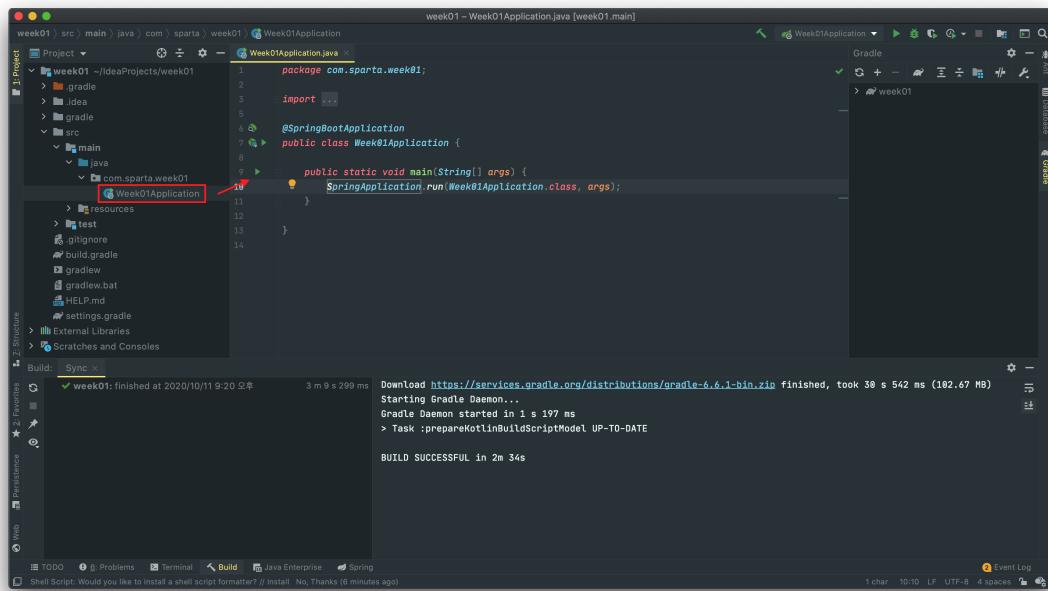
7. 잠시 기다리면, 아래 화면이 완성될 것입니다. 축하합니다! 스프링 개발을 시작할 준비가 되었습니다. 🎉



## ▼ 스프링 작동시키기

**팁)** 이미지를 더블클릭하시면 크게 보실 수 있어요 😎

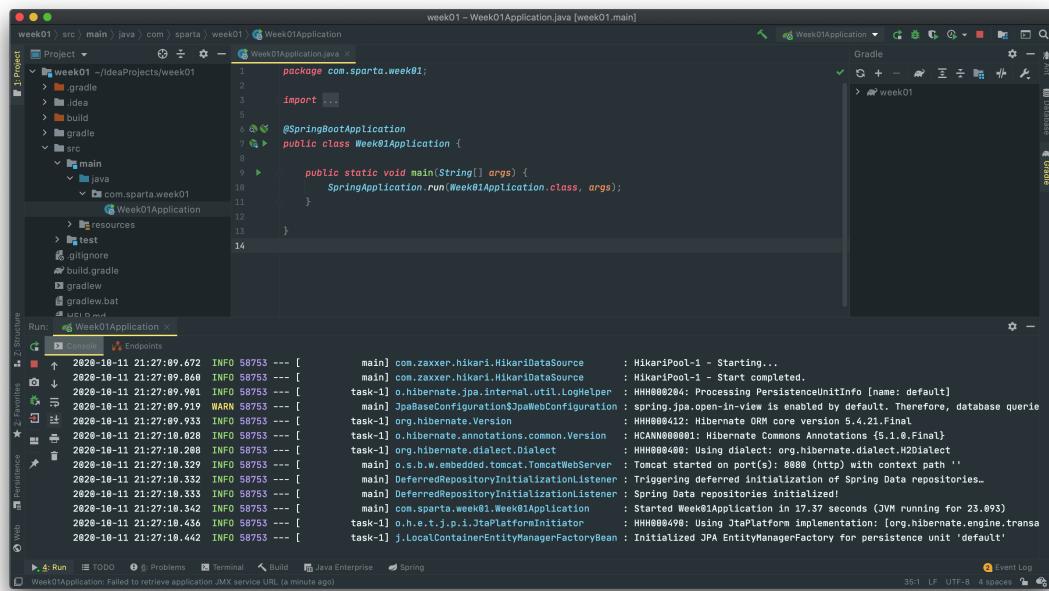
1. main > java > com.sparta.week01 > Week01Application.java 파일을 더블 클릭합니다.



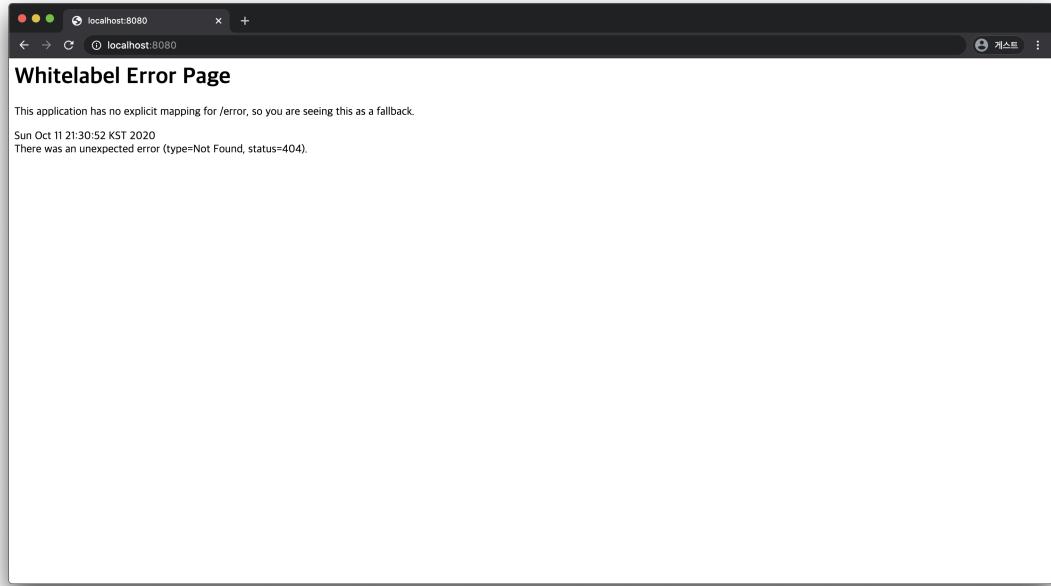
2. main 함수 제일 왼쪽의 run 버튼을 클릭하고, Run "Week01Application" 버튼을 클릭합니다.



3. 잠시 기다리면 아래와 같은 화면이 나옵니다. 크롬 브라우저를 켜고 <http://localhost:8080> 주소로 접속합니다.



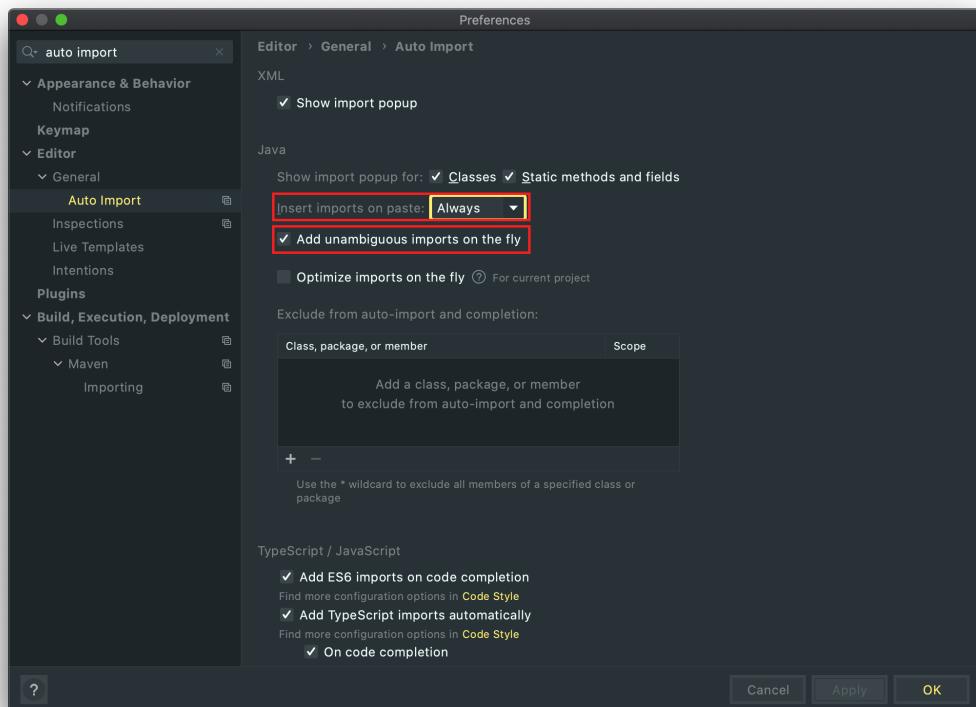
4. 와우! 스프링 작동에 성공하였습니다 😎



▼ 인텔리제이 세팅하기

▼ Auto import

1. 환경설정에 들어갑니다.
  - Windows: Ctrl + Alt + S
  - macOS: command + ,
2. 검색창에 "auto import" 입력
3. 옵션을 다음처럼 체크하고 OK 클릭하여 종료
  - Insert imports on paste: Always
  - Add unambiguous imports on the fly: 체크



## ▼ Plugins



1. 아름다운 코딩화면
2. 단축키 익히기

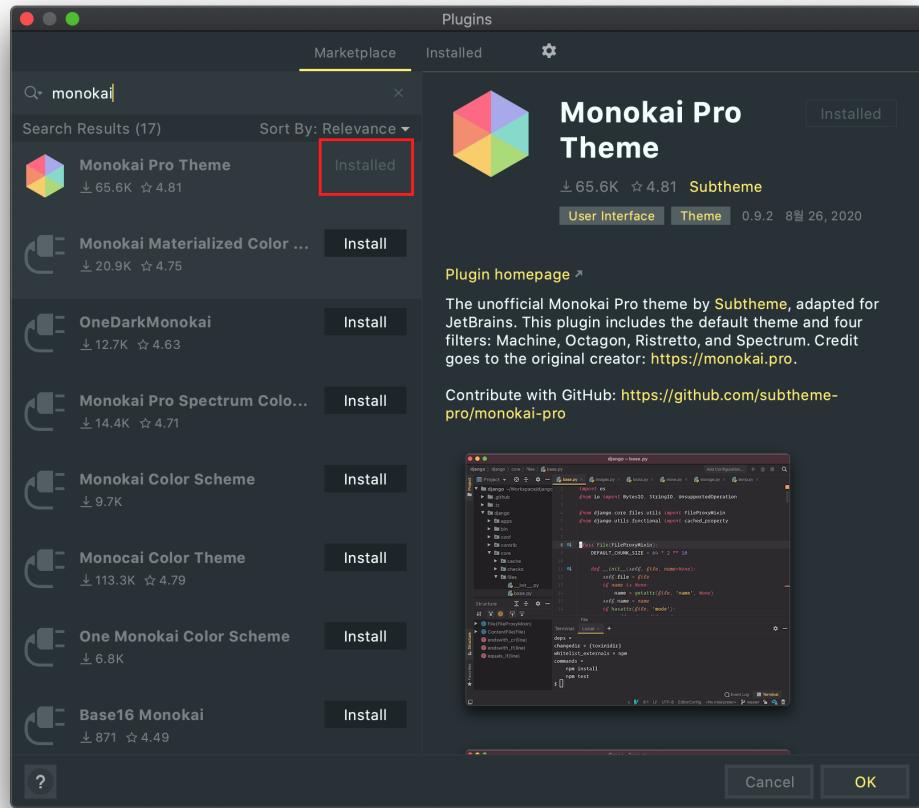
두 가지가, IDE의 꽃이 아닐까 싶습니다 😎

## ▼ Theme

1. Shift 두 번 입력
2. Actions 탭에서 plugins 입력 후 제일 위 옵션 선택



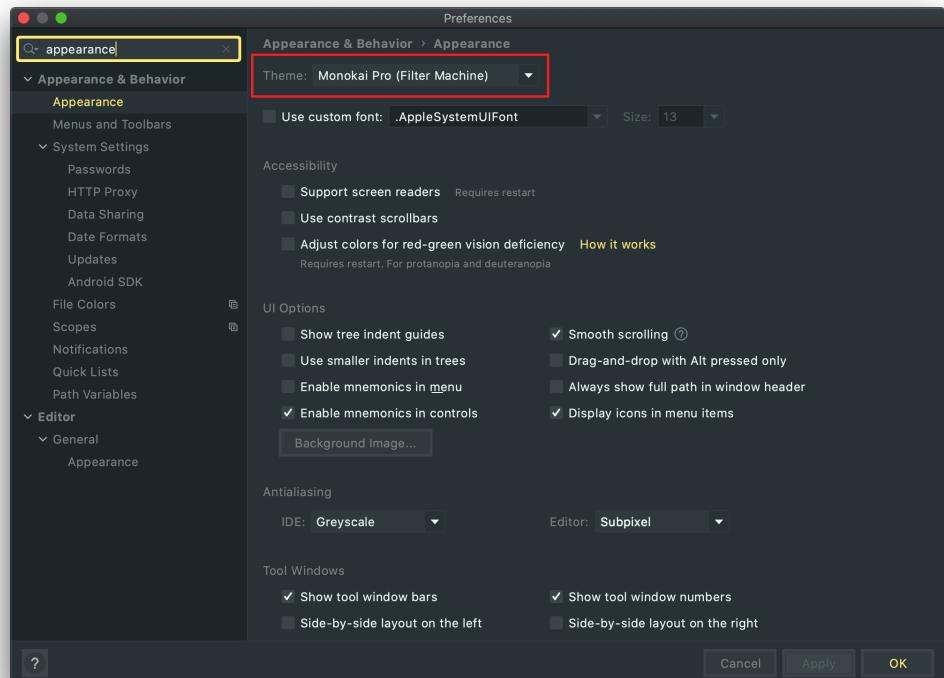
3. 검색창에 "monokai" 입력하고 Monokai Pro Install 버튼 클릭



#### 4. 환경설정 들어가기

- Windows: Ctrl + Alt + S
- macOS: command + ,

5. Appearance 입력 후 선택한 뒤 Monokai Pro (Filter Machine) 선택하고 OK 클릭하여 종료



### ▼ Shortcuts

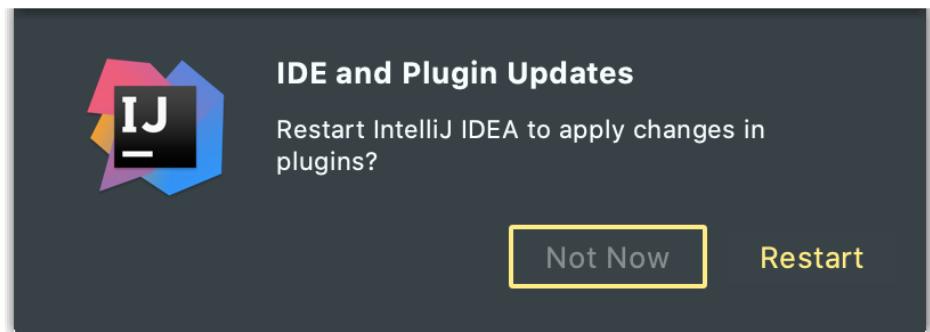
1. Shift 두 번 입력
2. Actions 탭에서 plugins 입력 후 제일 위 옵션 선택



3. Key Promoter X 검색 후 Install 클릭. 설치가 완료되면 Restart IDE 클릭!



#### 4. Restart 클릭



#### 5. 앞으로 단축키를 자동으로 알려준답니다.

##### ▼ 이번 주 배울 내용 확인하기

- 웹의 기본 개념
- 자바(Java) 기본 문법
- 클래스(Class), 메소드(Method), 멤버(Member)
- 스프링 작동 방법

- 컨트롤러(Controller)
- 그레이들(Gradle)

## 04. [1주차] 웹의 기본 개념



클라이언트, 서버, api를 간단히 설명하실 수 있다면 다음 강의로 넘어가셔도 괜찮습니다.  
(ex. 웹개발 종합반 기수강생, 컴퓨터공학과 학생 등)

▼ 네이버 웹페이지를 해킹해보자!



크롬 창에서 보이는 웹페이지는 어떤 원리로 보여지는 것일까요?  
원리를 알기 위해서, 일단 해킹부터 해보죠! (응?)

함께 해보아요 → '네이버뉴스'라고 쓰인 곳을 원하는 문구로 바꾸어보죠!

The screenshot shows the main Naver homepage. At the top, there's a search bar with a magnifying glass icon and a dropdown menu. Below the search bar, there are several navigation links: 메일 (Email), 카페 (Cafe), 블로그 (Blog), 지식IN (Knowledge IN), 쇼핑 (Shopping), Pay, TV, 사전 (Dictionary), 뉴스 (News), 증권 (Finance), 부동산 (Real Estate), 지도 (Map), 영화 (Movies), 뮤직 (Music), 책 (Books), 웹툰 (Webtoon). To the right of these links, there's a "더보기" button and the date "1 2020 3월 모의고사". On the left side of the main content area, there's a large banner for the "100년 후대에게 전달할 꿈과 희망의 손편지를 보내주세요" campaign. The right side features a login button for "NAVER 로그인" and a "LIVE" broadcast link for "저녁방송 메인뉴스 보기". Below the main banner, there's a section for news from various media outlets like YTN, MBC, and EBS.

👉 앗! 바뀌었다!

The screenshot shows the Naver homepage with a search bar at the top. Below the search bar, there's a banner for the 100th anniversary of the Korean National Olympic Committee. The main content area displays news headlines and links from various sources like KBS WORLD, sportalkorea, NewDaily, and JTBC. On the right side, there's a weather forecast (11.8°) and a promotional box for the 'IX TOWER' building.

### ▼ API

은행 창구와 같은, 하나의 "약속" 입니다.  
정해진대로 요구를 하면, 정해진 결과물을 돌려주는 창구이죠.

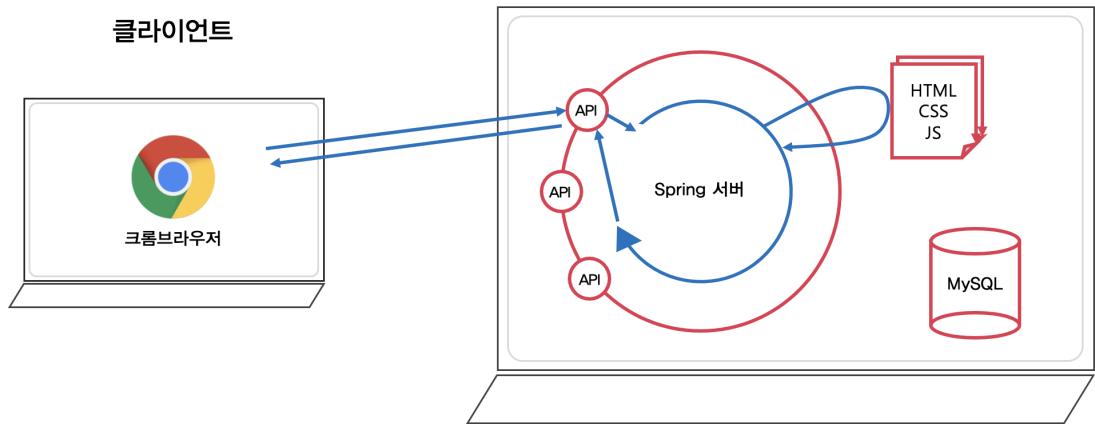
### ▼ 웹의 동작 개념 (HTML을 받는 경우)

네! 우리가 보는 웹페이지는 모두 서버에서 미리 준비해두었던 것을 "받아서",  
"그려주는" 것입니다. 즉, 브라우저가 하는 일은 1) 요청을 보내고, 2) 받은 HTML 파일을 그려주는 일 뿐이죠.

근데, 1)은 어디에 요청을 보내느냐구요? 좋은 질문입니다. 서버가 만들어 놓은 "API"라는 창구에 미리 정해진 약속  
대로 요청을 보내는 것이라니.

예) <https://naver.com/>  
→ 이것은 "naver.com"이라는 이름의 서버에 있는, "/" 창구에 요청을 보낸 것!

## 서버



### ▼ 웹의 동작 개념 (데이터만 받는 경우)

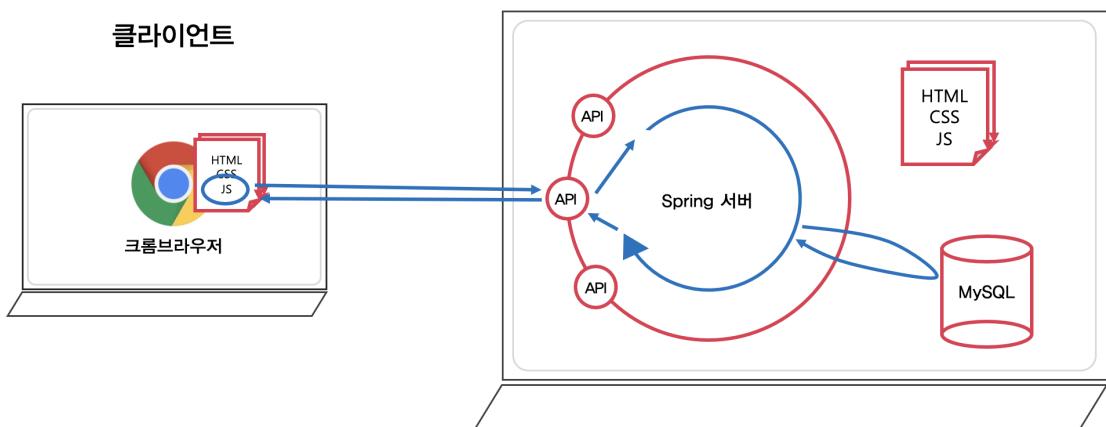
👉 앗, 그럼 항상 이렇게 HTML만 내려주냐구요?  
아뇨! 데이터만 내려 줄 때가 더~ 많아요.

사실 HTML도 줄글로 쓰면 이게 다 '데이터' 아닌가요?

👉 자, 공연 티켓을 예매하고 있는 상황을 상상해봅시다!  
좌석이 차고 깨질때마다 보던 페이지가 리프레시 되면 난감하겠죠ㅠ?

이럴 때! 데이터만 받아서 받아 끼우게 된답니다.

## 서버



👉 데이터만 내려올 경우는, 이렇게 생겼어요!  
(소곤소곤) 이런 생김새를 JSON 형식이라고 한답니다.

The screenshot shows a browser window with the URL [openapi.seoul.go.kr:8088/6d4d776b466c656533356a4b4b5872/json/RealtimeCityAir/1/99](http://openapi.seoul.go.kr:8088/6d4d776b466c656533356a4b4b5872/json/RealtimeCityAir/1/99). The page displays a JSON object representing air quality data. The structure includes a 'RESULT' object with a 'CODE' of 'INFO-000' and a message '정상 처리되었습니다'. The 'list\_total\_count' is 25, and there is a 'row' array containing one element. This element has properties like 'MSRDT' (202004241900), 'MSRRGN\_NM' (도심권), 'MSRSTE\_NM' (중구), and various pollutant levels (PM10: 44, PM25: 20, O3: 0.039, NO2: 0.02, CO: 0.4, SO2: 0.003). It also contains 'INDEX\_NM' (보통), 'INDEX\_MYL' (59), and 'ARPLT\_MAIN' (PM10).

```
{
  "RealtimeCityAir": {
    "list_total_count": 25,
    "RESULT": {
      "CODE": "INFO-000",
      "MESSAGE": "정상 처리되었습니다"
    },
    "row": [
      {
        "MSRDT": "202004241900",
        "MSRRGN_NM": "도심권",
        "MSRSTE_NM": "중구",
        "PM10": 44,
        "PM25": 20,
        "O3": 0.039,
        "NO2": 0.02,
        "CO": 0.4,
        "SO2": 0.003,
        "INDEX_NM": "보통",
        "INDEX_MYL": 59,
        "ARPLT_MAIN": "PM10"
      }
    ]
  }
}
```

▼ 스프링은 어디를 담당하는 걸까?



스프링은 "서버"를 담당합니다. 요청에 따라 눈에 보이는 것들 또는 데이터를 제공해주는 것이지요. 그리고 스프링은 자바 언어 바탕으로 만드는 것이구요.

⇒ **스프링으로 서버 만들어보기** 가 이번 강의의 핵심이군요!  
그럼, 5주 간 힘차게 달려볼까요? 🔥

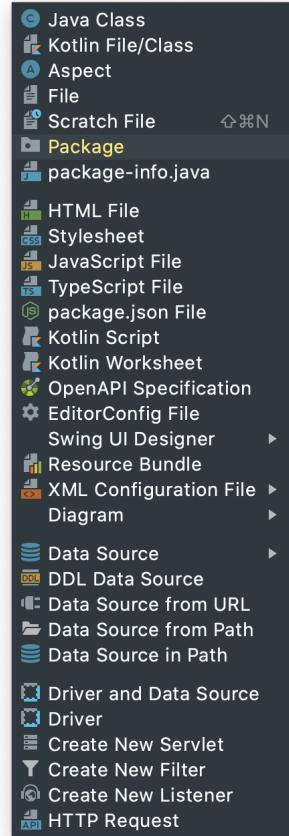
## 05. [1주차] 자바 기초 문법 - 1

▼ 화면에 나타내보기 - `System.out.println();`

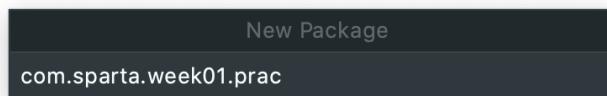


화면에 인쇄하고 싶을 때, `System.out.println();` 을 이용하시면 된답니다 😎 바로 같이 해볼까요?

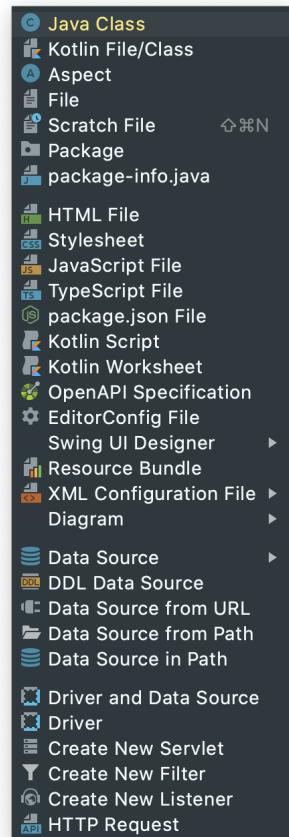
- src > main > java > com.sparta.week01 를 우클릭하고 New > Package를 선택합니다.



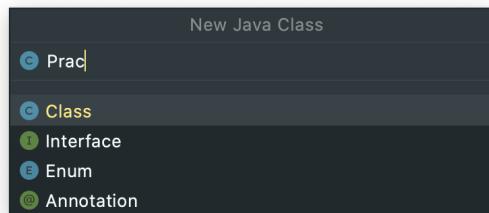
2. prac 이라고 입력한 뒤 엔터를 누릅니다.



3. prac 패키지를 우클릭하고 New > Java Class 를 클릭합니다.



4. Prac 이라고 입력하고 엔터를 누릅니다.



5. main 함수를 작성합니다. 앞으로 직접 실행시킬 코드는 main 함수 안에 작성합니다

#### ▼ [코드스니펫] 메인 함수

```
public static void main(String[] args) {  
    System.out.println("안녕, 스파르타!");  
}
```

6. main 함수를 실행시키면, "안녕, 스파르타!"가 인쇄된 걸 보실 수 있습니다! 🔥

```

package com.sparta.week01.prac;
public class Prac {
    public static void main(String[] args) {
        System.out.println("안녕, 스파르타!");
    }
}

```

10:47:04 오류: Executing task 'Prac.main()'...

> Task :compileJava  
> Task :processResources UP-TO-DATE  
> Task :classes  
> Task :Prac.main()  
안녕, 스파르타!

BUILD SUCCESSFUL in 2s  
3 actionable tasks: 2 executed, 1 up-to-date  
10:47:07 오류: Task execution finished 'Prac.main()'.

👉 앞으로 화면에 인쇄하고 싶을 때는, `System.out.println();` 을 이용하시면 됨답니다 😎 참 쉽죠?

### ▼ 변수 & 초기 문법

- 변수는 "값을 담는 상자"입니다.
- 변수를 선언할 때는 (자료의 종류) (변수명) = (값) 의 순서를 따릅니다.
- 변수명은 일반적으로 1)영어, 2)소문자 시작, 3)구분은 대문자로 합니다.
- 자바는 항상 코드를 작성하고 제일 마지막에 세미콜론(:)을 넣습니다.

```
String myName = "내이름";
```

### ▼ 숫자 - int, float, Long

👉 숫자를 나타내는 자료형은 int, float 가 있습니다.

int - 정수(-1, 0, 1, 2, 3, 4 ...)  
float - 소수점 포함(1.0, 2.4, -3.78, ...)  
Long - 정수 범위가 훨씬 넓은 녀석, 그것 빼고는 int와 같음

```
int a = 1;
float b = 1.2f;

System.out.println("a: " + a);
System.out.println("b: " + b);
```

👉 1 + 1.2의 결과는 소수니까 float 로 지정해주면 됩니다.

```
float c = a + b;
System.out.println("c: " + c);
```

int = int + float 는 오류!

```
int d = a + b;
```

▼ 문자 - String

 문자열은 대문자 S로 시작합니다. 반드시 쌍따옴표("")를 써주셔야 해요!

String 선언

```
String e = "자바 스프링, 5주 안에 끝낸다!";
```

## 06. [1주차] 자바 기초 문법 - 2

▼ 참거짓 - boolean

 참거짓 타입은 boolean으로 표시합니다.

boolean 선언

```
int age = 20;
boolean isAdult = age > 19;
boolean isKid = age <= 19;

// 같은지는 == 으로 확인, 다른지는 != 으로 확인
boolean isEq = 20 == 20;
boolean isDiff = 20 != 20;
```

▼ 배열 - List

 정보를 뭉을 때 List라는 녀석을 사용합니다. 세 가지를 기억하시면 돼요.

1. 순서가 중요한 녀석이에요!
2. 같은 자료형만 넣을 수 있어요
3. 넣을 때는 add, 조회할 때는 get, 제거할 때는 remove를 사용합니다.

```
List<String> newList = new ArrayList<>();
newList.add("웹개발의 봄 Spring");
newList.add("프론트엔드의 꽃 React");
System.out.println(newList);
System.out.println(newList.get(0));
System.out.println(newList.get(1));
newList.remove(0);
System.out.println(newList);
```

▼ 임포트

- List처럼, 미리 작성된 코드를 불러오는 것을 "임포트(import)"라고 합니다.
- 자바는 기본적인 임포트 파일들이 있는데, 임포트 단축키는 다음과 같습니다.
  - 원도우: Alt + Enter

- 맥: Option + Enter

#### ▼ 주석, 정렬 단축키

- 주석은, "나만 볼 수 있고 컴퓨터는 알아듣지 못하는 메모" 입니다. 개발자들끼리 소통하거나, 내가 개발하다가 메모를 남기고 싶을 때 주석을 사용합니다.
  - 윈도우: Ctrl + /
  - 맥: Cmd + /
- 코드를 작성하거나 복사/붙여넣기를 할 때, 정렬을 해주지 않으면 가독성이 무척 떨어집니다. 코드스니펫을 복사/붙여넣기 하신 다음에는 꼭 코드 정렬을 해주세요😎
  - 윈도우: Ctrl + Alt + L
  - 맥: Option + Cmd + L

#### ▼ 연습 퀴즈

?  
course1이라는 이름의 변수에 값을 "웹개발 종합반",  
course2라는 이름의 변수에 값을 "앱개발 종합반",  
넣고, course1과 course2를 courseList라는 배열에 순서대로 넣으려면 어떻게 해야 할까요?

#### ▼ 정답

```
// 선언
String course1 = "웹개발 종합반";
String course2 = "앱개발 종합반";
List<String> courseList = new ArrayList<>();

// 삽입
courseList.add(course1);
courseList.add(course2);

// 확인
System.out.println(courseList);
```

## 07. [1주차] 자바 기초 문법 - 3

#### ▼ 메소드

- 반복적인 행동을 줄여서 어디든 갖다쓸 수 있는 명령 모음입니다.

#### ▼ [코드스니펫] 강의 정보 인쇄하기 - 메소드가 없을때

```
String title = "웹개발의 봄 Spring";
String tutor = "남병관";
int weeks = 5;
float ratings = 5.0f;

System.out.println("제목: " + title);
System.out.println("튜터: " + tutor);
System.out.println("주차: " + weeks);
System.out.println("별점: " + ratings);
```

#### ▼ [코드스니펫] 강의 정보 인쇄하기 - 메소드가 있다면

```
// public, static void 에 대해서는 곧 배웁니다! 우선은 넘어갈게요.
public static void printInfo() {
    String title = "웹개발의 봄 Spring";
    String tutor = "남병관";
    int weeks = 5;
    float ratings = 5.0f;
```

```
    System.out.println("제목: " + title);
    System.out.println("튜터: " + tutor);
    System.out.println("주차: " + weeks);
    System.out.println("별점: " + ratings);
}
```

- 실행하는 방법

메소드명() ← 처럼 작성해줍니다!

- 메소드의 구조

```
public (반환타입) 메소드명(파라미터) {
    // 명령 내용
    return (반환값)
}
```



메소드명은 변수명과 규칙(영어사용, 소문자시작, 공백없음, 대문자구분)이 같습니다.

- 컴퓨터에게 명령을 하려면 세 가지 요소를 주어야 합니다.

1. 재료 = 파라미터 (없을 수도 있음)
2. 명령 = 실행코드
3. 결과 = 반환값 (없을 수도 있음)

- 파라미터와 반환값이 없을 수도 있다구요?

#### ▼ [코드스니펫] 파라미터, 반환값의 다양한 경우

```
// 파라미터 X, 반환값 X
public void simplePrint() {
    System.out.println("파라미터도 없고, 반환값도 없어요!");
}

// 파라미터 0, 반환값 X
public void simpleSum(int num1, int num2) {
    System.out.println("num1 :" + num1 + ", num2: " + num2);
}

// 파라미터 X, 반환값 0
public int simpleReturn() {
    return 3;
}

// 파라미터 0, 반환값 0
public int sum(int num1, int num2) {
    return num1 + num2;
}
```

#### ▼ 연습퀴즈 - 메소드



두 정수를 받아서, 뺀 값을 반환하는 메소드를 만들고, 그 값을 인쇄해보세요!  
(메소드명은 자유롭게 지으세요)

#### ▼ 정답

```
public static int sub(int num1, int num2) {
    return num1 - num2;
}

public static void main(String[] args) {
```

```
    int result = sub(1, 2);
    System.out.println(result);
}
```

## 08. [1주차] 자바 기초 문법 - 4

### ▼ 반복문



컴퓨터에게 일일이 명령문을 치고 있다면, 누가 봐도 효율적으로 일하는 모습은 아니죠?  
업무에서 반복되는 부분을 줄여주는 문법의 꽃, 반복문을 배워보겠습니다.

### ▼ [코드스니펫] 과일 목록 준비하기

```
List<String> fruits = new ArrayList<>();
fruits.add("감");
fruits.add("배");
fruits.add("감");
fruits.add("딸기");
fruits.add("수박");
fruits.add("메론");
fruits.add("수박");
fruits.add("딸기");
fruits.add("메론");
fruits.add("수박");
fruits.add("메론");
fruits.add("수박");
fruits.add("감");
System.out.println(fruits);
```

### ▼ [코드스니펫] 과일 목록 차례대로 인쇄하기 - 반복문 없이

```
System.out.println(fruits.get(0));
System.out.println(fruits.get(1));
System.out.println(fruits.get(2));
System.out.println(fruits.get(3));
System.out.println(fruits.get(4));
System.out.println(fruits.get(5));
System.out.println(fruits.get(6));
System.out.println(fruits.get(7));
System.out.println(fruits.get(8));
System.out.println(fruits.get(9));
System.out.println(fruits.get(10));
System.out.println(fruits.get(11));
System.out.println(fruits.get(12));
```

### ▼ [코드스니펫] 과일 목록 차례대로 인쇄하기 - 반복문과 함께

```
for (int i=0; i<13; i++) {
    String fruit = fruits.get(i);
    System.out.println(fruit);
}
// 매번 13개, 14개, 12개.. 개수 세어줘야 할까요?
```

### • 반복문의 구조

```
for (시작조건; 계속하는조건; 더하기) {
    명령
}
```

### ▼ [코드스니펫] 과일 목록 차례대로 인쇄하기 - 반복문 완성버전

```
for (int i=0; i<fruits.size(); i++) {  
    String fruit = fruits.get(i);  
    System.out.println(fruit);  
}
```

### ▼ 연습퀴즈 - 반복문



주어진 예제 목록을 차례대로 하나씩 인쇄하는 반복문을 작성해보세요.

### ▼ [코드스니펫] 연습퀴즈 반복문

```
List<String> celebs = new ArrayList<>();  
celebs.add("아이유");  
celebs.add("린다G");  
celebs.add("은비");  
celebs.add("금비");  
celebs.add("비");  
celebs.add("치은우");  
celebs.add("남주혁");  
celebs.add("수지");  
celebs.add("정우성");  
celebs.add("제니");  
celebs.add("정국");
```

### ▼ 정답

```
for (int i=0; i<celebs.size(); i++) {  
    String celeb = celebs.get(i);  
    System.out.println(celeb);  
}
```

### ▼ 조건문



boolean 타입을 활용하여 특정 경우일 때 명령을 내리는 문법입니다.

### ▼ [코드스니펫] 미성년자 여부 판별하기

```
int age = 20;  
if (age > 19) {  
    System.out.println("성인입니다.");  
} else {  
    System.out.println("미성년자입니다.");  
}
```

### ▼ [코드스니펫] "감" 개수 세는 반복문 만들기

```
int count = 0;  
for (int i=0; i<fruits.size(); i++) {  
    String fruit = fruits.get(i);  
    if (fruit == "감") {  
        count += 1;  
    }  
}  
System.out.println(count);
```

### ▼ 연습퀴즈 - 조건문 + 반복문

?

주어진 파일의 개수를 세고 그 수를 반환하는 메소드를 만들어보세요.

▼ [코드스니펫] 퀴즈) 파일 목록 준비하기

```
public static int countFruit(String fruit) {  
    List<String> fruits = new ArrayList<>();  
    fruits.add("감");  
    fruits.add("배");  
    fruits.add("감");  
    fruits.add("딸기");  
    fruits.add("수박");  
    fruits.add("메론");  
    fruits.add("수박");  
    fruits.add("딸기");  
    fruits.add("메론");  
    fruits.add("수박");  
    fruits.add("메론");  
    fruits.add("수박");  
    fruits.add("감");  
}
```

▼ 정답

```
public static int countFruit(String given) {  
    List<String> fruits = new ArrayList<>();  
    fruits.add("감");  
    fruits.add("배");  
    fruits.add("감");  
    fruits.add("딸기");  
    fruits.add("수박");  
    fruits.add("메론");  
    fruits.add("수박");  
    fruits.add("딸기");  
    fruits.add("메론");  
    fruits.add("수박");  
    fruits.add("메론");  
    fruits.add("수박");  
    fruits.add("감");  
  
    int count = 0;  
    for (int i=0; i<fruits.size(); i++) {  
        String fruit = fruits.get(i);  
        if (given == fruit) {  
            count += 1;  
        }  
    }  
    return count;  
}
```

## 09. [1주차] 자바 클래스

▼ 클래스란?



클래스는 정보를 묶는 것입니다!

▼ 객체지향 프로그래밍

- 조건문, 반복문, 메소드만으로 무질서한 코드에 규칙이 생겼던 것, 기억하시죠?
- 조금 더 프로그래밍을 현실 세계에 빗대어 체계적으로 해보자는 발상이, 객체지향 프로그래밍의 핵심입니다.

▼ 클래스

- 현실과 비슷한 개념(객체)을 나타내기 위한 자바의 도구를 클래스라고 부릅니다.

- 클래스 내 정보를 멤버 변수라고 합니다.

▼ [코드스니펫] 클래스 바로 연습해보기 Course.java

```
public class Course {
    // title, tutor, days 가 Course라는 맥락 아래에서 의도가 분명히 드러나죠!
    public String title;
    public String tutor;
    public int days;
}
```

▼ [코드스니펫] 클래스 바로 연습해보기 Prac.java

```
Course course = new Course();
course.title = "웹개발의 봄, Spring";
System.out.println(course.title);
System.out.println(course.tutor);
```

- 변수 course는 실제 빵, Course 클래스는 빵틀



▼ 생성자(Constructor)

❓ course.tutor를 인쇄해보면 null(아무 값도 없음)이 보입니다.  
여기에 값을 부여하려면 어떻게 해야 할까요?

▼ [코드스니펫] Course 생성자

```
public class Course {
    public String title;
    public String tutor;
```

```
public int days;  
  
// 이렇게 아무런 파라미터가 없는 생성자를 기본생성자 라고 부릅니다.  
public Course() {  
  
}  
  
public Course(String title, String tutor, int days) {  
    // this 는 "이것" 이죠? 클래스 변수를 가리킵니다.  
    this.title = title;  
    this.tutor = tutor;  
    this.days = days;  
}  
}
```

👉 위와 같이 "클래스명과 똑같은 이름을 가진 메소드를 "생성자"라고 부릅니다.  
클래스 변수를 새롭게 만들 때 사용되는 녀석인데요, 아래와 같이 사용할 수 있습니다.

👉 this?

아까 말씀드린, 빵틀(Course)과 빵(course)의 관계 기억나시나요?  
우리가 바꾸고 싶은 건 빵틀의 값이 아니라 실제 빵의 값입니다.  
this 라고 표시함으로써, 빵틀 전체의 값을 바꾸는게 아니라 빵 하나의 값만 바꾸는 것이죠!



#### ▼ [코드스니펫] 생성자 사용해보기

```
Course course = new Course();  
System.out.println(course.title);  
System.out.println(course.tutor);  
System.out.println(course.days);
```

```
Course course2 = new Course("웹개발의 봄 스프링", "남병관", 35);
System.out.println(course2.title);
System.out.println(course2.tutor);
System.out.println(course2.days);
```

## 10. [1주차] Getter, Setter

▼ private, public



클래스의 모든 정보를 손쉽게 바꿀 수 있으면 안되겠죠~!

왜냐하면 클래스는 고객의 비밀번호/결제정보와 같이 민감한 정보들도 담고 있기 때문입니다.

그래서 자바의 클래스는, 밖에 드러내도 되는 것들을 public, 함부로 바꾸면 안되는 것들을 private 으로 구분해서 나타낸답니다. 😎

- private 일 경우 발생하는 오류

```
public static void main(String[] args) {
    Course course = new Course();
    course.title;|
}
'title' has private access in 'com.sparta.week01.prac.Course'
Create field 'title' in 'Course' More actions...
com.sparta.week01.prac.Course
private String title
```

- public 이면? 아무런 오류도 발생하지 않습니다.

```
public static void main(String[] args) {
    Course course = new Course();
    course.title = "웹개발의 봄 스프링";
}
```

▼ Getter / Setter



정보를 가져오는 메소드를 Getter,  
정보를 바꾸는 메소드를 Setter 라고 부른답니다.

▼ [코드스니펫] Getter, Setter 만들기

```

// Getter
public String getTitle() {
    return this.title;
}
// Getter
public String getTutor() {
    return this.tutor;
}
// Getter
public int getDays() {
    return this.days;
}

// Setter
public void setTitle(String title) {
    this.title = title;
}
// Setter
public void setTutor(String tutor) {
    this.tutor = tutor;
}
// Setter
public void setDays(int days) {
    this.days = days;
}

```

#### ▼ [코드스니펫] Getter, Setter 사용하기

```

Course course = new Course();

System.out.println(course.getTitle());
System.out.println(course.getTutor());
System.out.println(course.getDays());

course.setTitle("웹개발의 봄 스프링");
course.setTutor("남병관");
course.setDays(35);

System.out.println(course.getTitle());
System.out.println(course.getTutor());
System.out.println(course.getDays());

```

#### ▼ 연습퀴즈 - 클래스 & 메소드

- ?
- 1) Tutor 클래스를 만들고, 이름(name)과, 경력(bio) 멤버 변수를 추가하세요.
  - 2) 그리고 각 변수를 private 으로 선언하고, Getter와 Setter 를 만들어보세요.
  - 3) 마지막으로, 기본 생성자와, name/bio 입력받는 생성자 두 개를 만들어보세요.

#### ▼ 정답

```

public class Tutor {
    // 멤버 변수
    private String name;
    private String bio;

    // 기본생성자
    public Tutor() {
    }

    // 생성자
    public Tutor(String name, String bio) {
        this.name = name;
        this.bio = bio;
    }

    // Getter
    public String getName() {
        return this.name;
    }
}

```

```

public String getBio() {
    return this.bio;
}

// Setter
public void setName(String name) {
    this.name = name;
}
public void setBio(String bio) {
    this.bio = bio;
}

}

```

## 11. [1주차] 브라우저에 바로 나타내보기

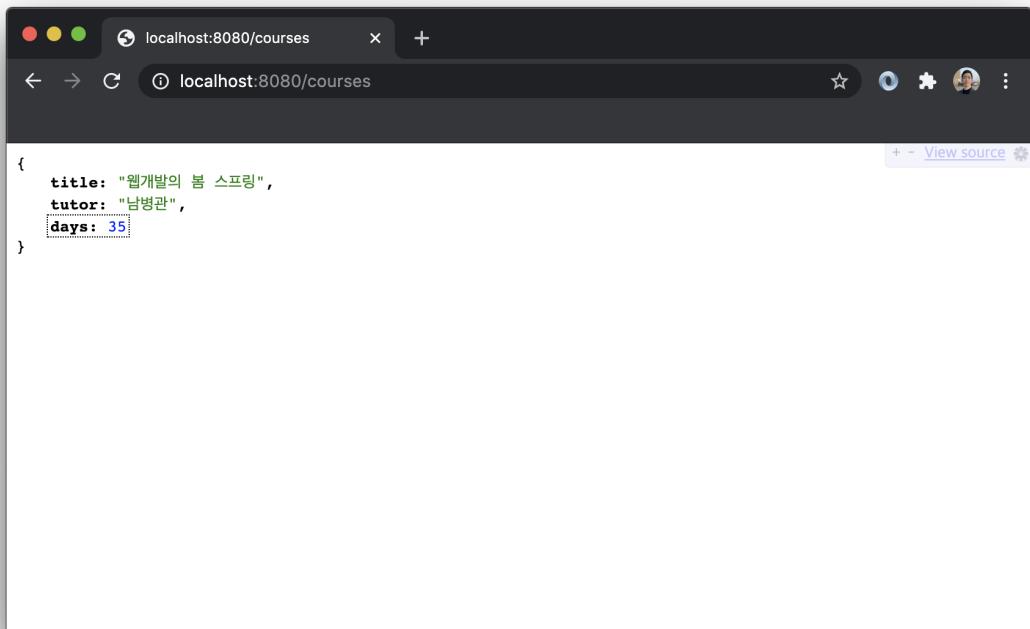
▼ 인트로

▼ [코드스니펫] 예시화면

<http://spartacodingclub.shop/order>

위와 같이 데이터를 서버에서 전달받는 형식을 JSON이라고 부른다고 했던 것, 기억하시죠?

이번 시간에는 스프링 서버를 띄워서 클래스 정보를 JSON으로 브라우저에 나타내보겠습니다.



▼ [코드스니펫] JSONView 설치하기

<https://chrome.google.com/webstore/detail/jsonview/chklaanhfefbnpoihckbnefhakgolnmc?hl=en>

▼ 화면에 클래스 정보 띄워보기



데이터로 응답하려면, RestController 를 사용해야 합니다.

▼ Rest ?

- 서버의 응답이 JSON 형식임을 나타냅니다.
- HTML, CSS 등을 주고받을 때는 Rest 를 붙이지 않습니다.

▼ Controller ?



자동 응답기!

누군가 말을 걸면 응답하는 녀석이에요.

- 클라이언트의 요청(Request)을 전달받는 코드를 Controller 라고 부릅니다.  
→ JSON 만을 돌려주는 것은 RestController 라고 부릅니다.
- 우리는 이런 역할을 하는 뺑틀을 새롭게 만들면 되는 것이군요!

▼ RestController 만들기

1. src > main > com.sparta.week01 에 controller 패키지를 만듭니다.
2. CourseController.java 파일을 만듭니다.
3. [코드스니펫] CourseController.java 를 복사/붙여넣기 합니다.

```
@RestController  
public class CourseController {  
  
    @GetMapping("/courses")  
    public Course getCourse() {  
        Course course = new Course();  
        course.setTitle("웹개발의 봄 스프링");  
        course.setDays(35);  
        course.setTutor("남병관");  
        return course;  
    }  
}
```

▼ CourseController.java 이해하기

- @GetMapping

브라우저에서 주소를 치는 행위를 GET 방식으로 정보를 요청한다고 합니다. 스프링 주소(<http://localhost:8080>) 뒤의 주소가 /courses 일 경우, getCourse 메소드를 실행함을 나타냅니다.

## 12. [1주차] 그레이들(Gradle)이란?

▼ [코드스니펫] 개발자들은 얼마나 남의 코드에 의존할까?

<http://www.bloter.net/archives/253447>

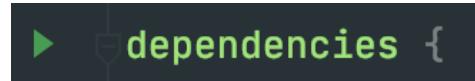
▼ 다른 사람들이 만들어둔 도구 내려받기

- Javascript - NPM
- Python - pip
- Java - mavenCentral, jcenter

⇒ 다운로드받고 적용하는 과정을 보다 편하게!

▼ 추가해보기

1. Maven Repository에서 원하는 라이브러리를 찾는다.
2. build.gradle에 원하는 프로젝트 파일을 넣는다.
3. dependencies 옆 Run 버튼을 누른다.



4. 우측 Gradle 탭의 새로고침 버튼을 누른다.
5. 대상 프로젝트가 추가된 것을 확인한다.

## 13. [1주차] 끝 & 숙제설명

▼ 잠깐 복습



아래 내용을 빠르게 훑어보고 바로 숙제하러 가볼까요? 교재를 다시 보는 것보다, 바로 숙제하길 추천드려요. 손으로 자꾸 만들고 오류내고 해봐야 실력이 빠르게 성장한답니다 😎

- 웹의 기본 개념에 대해 배웠습니다. 서버, 클라이언트를 익혔습니다.
- 인텔리제이를 사용해 스프링 프로젝트를 만들어 보았습니다. 스프링, 그레이들을 익혔습니다.
- 자바 언어의 기초 문법에 대해 배웠습니다. int, float, String, List를 익혔습니다.
- 클래스를 만들어 보았습니다. 클래스, 메소드, 멤버를 익혔습니다.
- 클래스 정보를 브라우저에 나타내보았습니다. JSON, 컨트롤러를 익혔습니다.

▼ 숙제 설명



1. 새로운 스프링 프로젝트를 만든다. (New Project ...)
2. Person 클래스를 만든다.
3. 3개 이상의 멤버 변수를 만든다. (name, age, address, job ...)
4. 멤버 변수는 모두 private이다.
5. Getter, Setter를 만든다.
6. PersonController를 만들고, <http://localhost:8080/myinfo>에 나의 정보가 뜨도록 한다.
7. Person.java, PersonController.java를 제출한다.

## 14. 1주차 숙제 답안 코드

▼ [코드스니펫] - 1주차 숙제 답안 코드

전체 코드

▼ Person.java

```
public class Person {  
    private String name;  
    private String job;  
    private int age;
```

```
private String address;

public String getName() {
    return this.name;
}

public String getJob() {
    return this.job;
}

public int getAge() {
    return this.age;
}

public String getAddress() {
    return this.address;
}

public void setName(String name) {
    this.name = name;
}

public void setJob(String job) {
    this.job = job;
}

public void setAge(int age) {
    this.age = age;
}

public void setAddress(String address) {
    this.address = address;
}
}
```

#### ▼ PersonController.java

```
@RestController
public class PersonController {

    @GetMapping("/myinfo")
    public Person getPerson() {
        Person person = new Person();
        person.setName("손흥민");
        person.setAddress("런던");
        person.setAge(28);
        person.setJob("대한민국 축구선수");
        return person;
    }
}
```

Copyright © TeamSparta All rights reserved.



# [스파르타코딩클럽] 웹개발의 봄, Spring - 5 주차



매 주차 강의자료 시작에 PDF파일을 올려두었어요!

## ▼ 단축키 모음

### ▼ 코드정렬

- Windows: `Ctrl + Alt + L`
- macOS: `option + command + L`

### ▼ 임포트

- Windows: `Alt + Enter`
- macOS: `option + Enter`

### ▼ 주석

- Windows: `Ctrl + /`
- macOS: `command + /`

### ▼ System.out.println()

- 공통: sout 치고 tab 누르기

### ▼ 환경설정

- Windows: `Ctrl + Alt + S`
- macOS: `command + ,`

## [수업 목표]

1. 나만의 셀렉샵을 AWS RDS의 MySQL을 연결한다.
2. 나만의 셀렉샵을 AWS EC2에 배포한다.
3. 나만의 셀렉샵에 도메인을 붙이고 친구들에게 카톡으로 자랑한다.

## [목차]

01. [5주차] 이번 주 배울 것
02. [5주차] RDS 구매하고 MySQL 세팅하기
03. [5주차] OG 태그 작업하기
04. [5주차] EC2 구매하기
05. [5주차] EC2 준비하고 배포하기
06. [5주차] 포트포워딩 & nohup 적용하기

07. [5주차] 도메인 붙이고 카카오톡에 공유해보기

08. [5주차] 끝 & 숙제설명



모든 토글을 열고 닫는 단축키

Windows : `ctrl + alt + t`

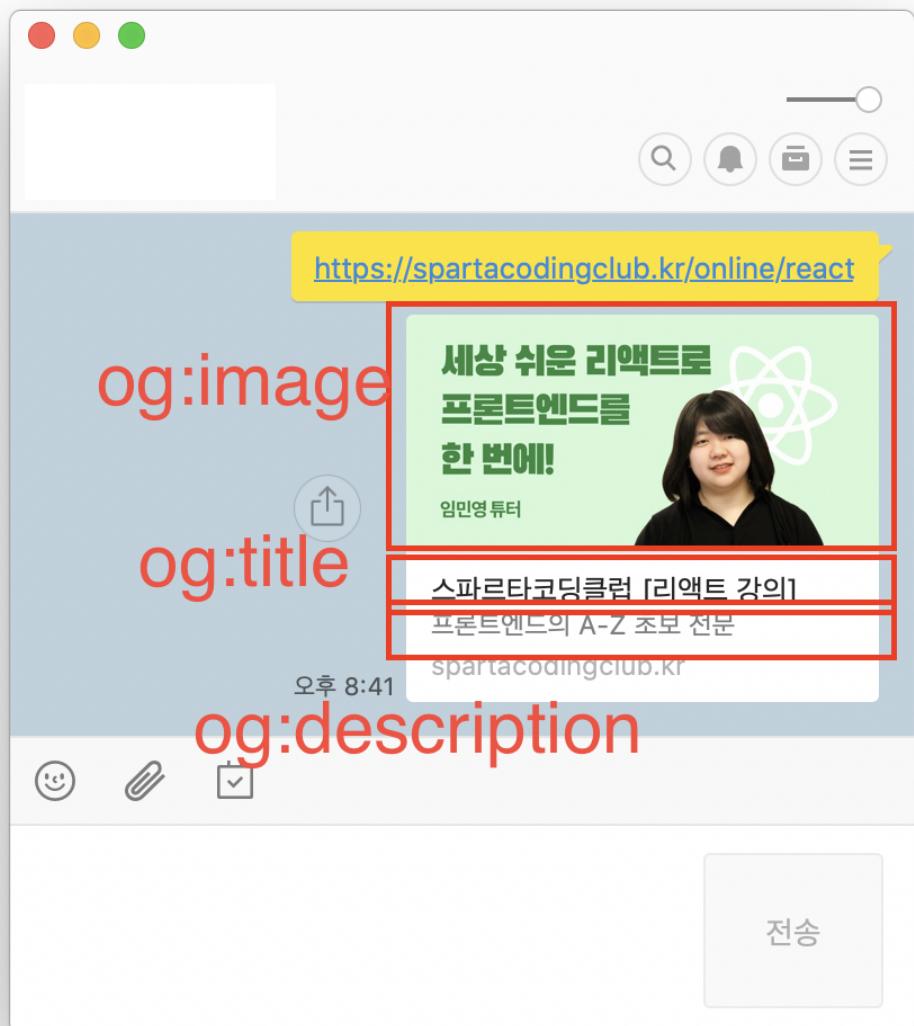
Mac : `⌘ + ⌘ + t`

## 01. [5주차] 이번 주 배울 것

▼ AWS RDS 와 로컬 서비스를 연결해봅니다.

- 그 동안 H2를 사용했기 때문에 서비스를 내렸다 올리면 모든 데이터가 사라졌습니다.
- RDBMS 중 가장 많이 쓰이는 MySQL을 AWS RDS 서비스에 설치 후, 우리 컴퓨터에 연결해봅니다.
- 이제는 데이터를 클라우드에 저장하기 때문에 꺼도 켜도 아무 문제 없을 거에요! 😎

▼ OG 태그를 이용하여 카톡방에 공유할 준비를 합니다.



- 카톡방 사용하다보면, 저렇게 자동으로 이미지, 제목, 내용을 가져오는게 신기하지 않으셨나요?
- 우리가 만든 서비스를 친구들에게 자랑할 때, 저렇게 멋지게 공유되면 훨씬 멋지지 않을까요?
- 이렇게 멋지게 카드를 만들기 위한 필수, OG 태그에 대해 배워보겠습니다.

▼ AWS EC2 를 세팅하고, 나만의 셀렉샵을 배포합니다.

- 우리 컴퓨터에서는 우리만 접속할 수 있다는 한계가 있습니다.
- 누구나 서비스에 접속할 수 있게 하려면 원격으로 항상 작동하는 컴퓨터가 필요합니다.
- AWS에서는 EC2라는 컴퓨터를 1년 간 무료로 제공합니다. 이 녀석을 이용하여 우리 프로젝트를 배포해 보겠습니다.

▼ 나만의 셀렉샵에 도메인을 붙이고 친구들에게 카톡으로 자랑한다.

- 지난 시간 숙제로 도메인 구매하기를 드렸습니다!

- 구매한 도메인을 우리 서비스에 붙여서, 누구나 접속할 수 있게 만들어볼게요.

## 02. [5주차] RDS 구매하고 MySQL 세팅하기

### ▼ RDS 구매하기

- 미리 가입한 AWS에 로그인합니다.
- (링크)에 접속합니다.

#### ▼ [코드스니펫] AWS 대시보드 링크

```
https://ap-northeast-2.console.aws.amazon.com/console/home?region=ap-northeast-2#
```

- RDS를 검색하고 결과를 클릭합니다.

The screenshot shows the AWS Management Console homepage. At the top, there's a search bar with the placeholder "AWS 서비스" (AWS Services) and a dropdown menu showing "RDS" selected. Below the search bar, there's a list of services under "AWS 서비스". On the left, there's a sidebar with "전체 서비스" (All Services) expanded, showing various AWS services like Lambda, S3, and CloudWatch. On the right, there are several promotional boxes: one for "AWS 리소스와 연결" (Connect to AWS resources), one for "AWS 팀색" (AWS Team Colors), one for "Amazon Redshift", one for "AWS Fargate", and one for "Amazon S3". The URL in the browser is https://ap-northeast-2.console.aws.amazon.com/console/home?region=ap-northeast-2#.

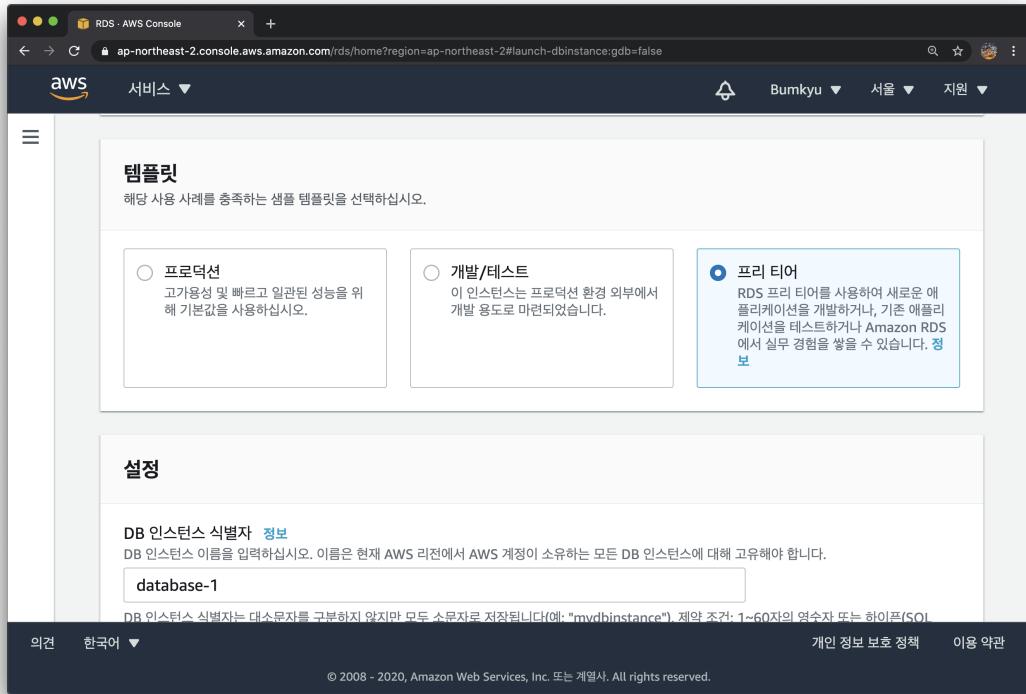
- "데이터베이스 생성"을 클릭합니다.

The screenshot shows the AWS RDS console for the Asia Pacific (Seoul) region. On the left, there's a sidebar with links like Dashboard, 데이터베이스 (Database), 성능 개선 도우미 (Performance Advisor), 스크립트 (Scripts), Automated backups, 예약 인스턴스 (Scheduled Instances), Proxies, 서브넷 그룹 (Subnet Groups), 파라미터 그룹 (Parameter Groups), and 온서브 그룹 (OnSubscriptions). The main content area features a section for Amazon Aurora, which is described as supporting MySQL and PostgreSQL with 6-node replication and 15-second point-in-time recovery. It includes a large orange "데이터베이스 생성" (Create Database) button and a link to "S3에서 Aurora DB 클러스터 복원" (Restore from S3). Below this is a "리소스" (Resources) section showing 2/40 DB instances and 1 parameter group.

5. "표준생성"과 "MySQL"을 클릭합니다.

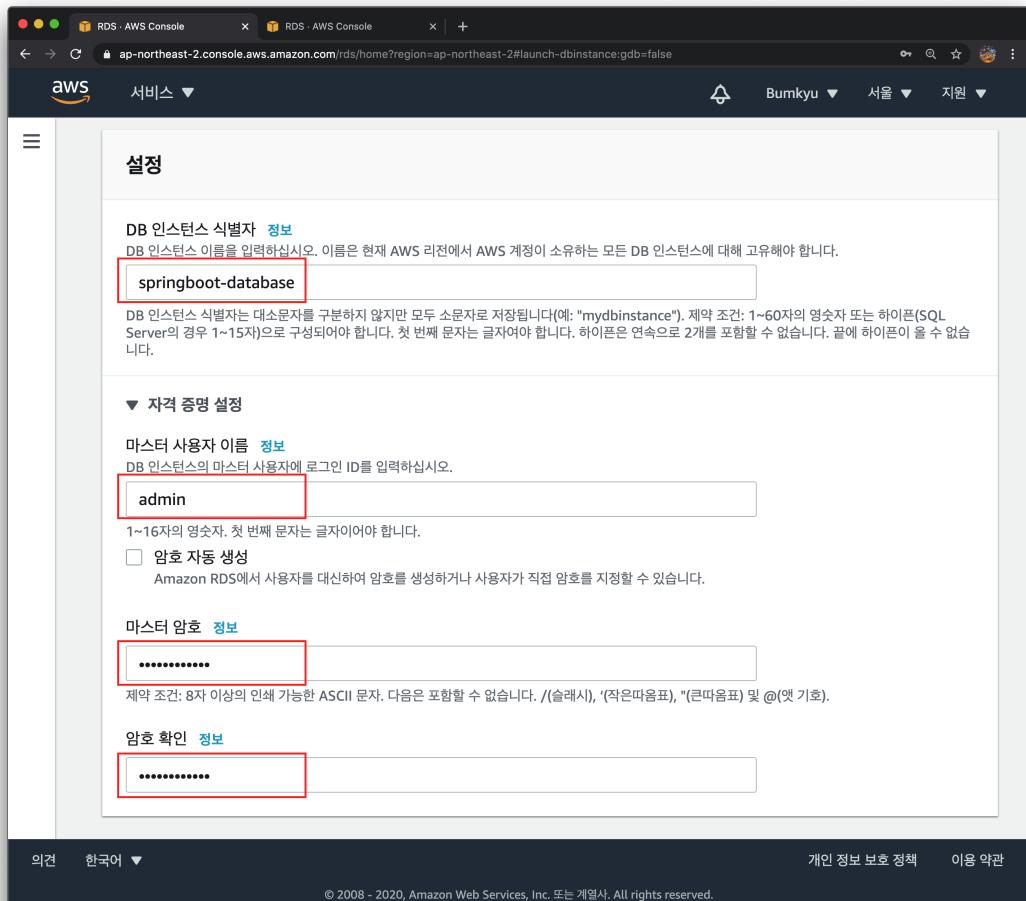
This screenshot shows the "Create Database" wizard on the "Standard Creation" tab. It asks for the engine type. Under the "엔진" (Engine) heading, the "MySQL" option is selected, indicated by a blue circle and highlighted with a blue border. Other options shown are "Amazon Aurora" and "MariaDB". The "Amazon Aurora" icon features three stacked cylinders with asterisks, while the "MariaDB" icon features a seal.

6. "프리 티어"를 선택합니다.



## 7. 다음과 같이 입력합니다.

- DB 인스턴스 식별자에 "springboot-database" 라고 입력합니다. (원하는 이름으로 바꾸셔도 돼요!)
- 마스터 사용자 이름과 암호에 만들고 싶은 계정의 아이디, 비밀번호를 입력합니다. DB 접속용으로 사용해야 하니 꼭 기억해주세요!



8. DB 인스턴스 크기와 스토리지는 설정 그대로 두고 넘어갑니다.

**DB 인스턴스 크기**

**DB 인스턴스 클래스 정보**  
처리 성능 및 메모리 요구 사항을 충족하는 DB 인스턴스 클래스를 선택하십시오. 아래의 DB 인스턴스 클래스 옵션은 위에서 선택한 엔진에서 지원하는 옵션으로 제한됩니다.

스텠다드 클래스(m 클래스 포함)  
 메모리 최적화 클래스(r 및 x 클래스 포함)  
 버스터블 클래스(t 클래스 포함)

db.t2.micro  
1 vCPUs 1 GiB RAM Not EBS Optimized

이전 세대 클래스 포함

**스토리지**

**스토리지 유형 정보**  
범용(SSD)

할당된 스토리지  
20 GiB  
(최소: 20GiB, 최대: 16,384GiB) 할당된 스토리지가 많으면 IOPS 성능이 개선될 수 있습니다.

**스토리지 자동 조정 정보**  
애플리케이션의 필요에 따라 데이터베이스 스토리지의 동적 조정 지원을 제공합니다.

스토리지 자동 조정 활성화  
이 기능을 활성화하면 지정한 임계값 초과 시 스토리지를 늘릴 수 있습니다.

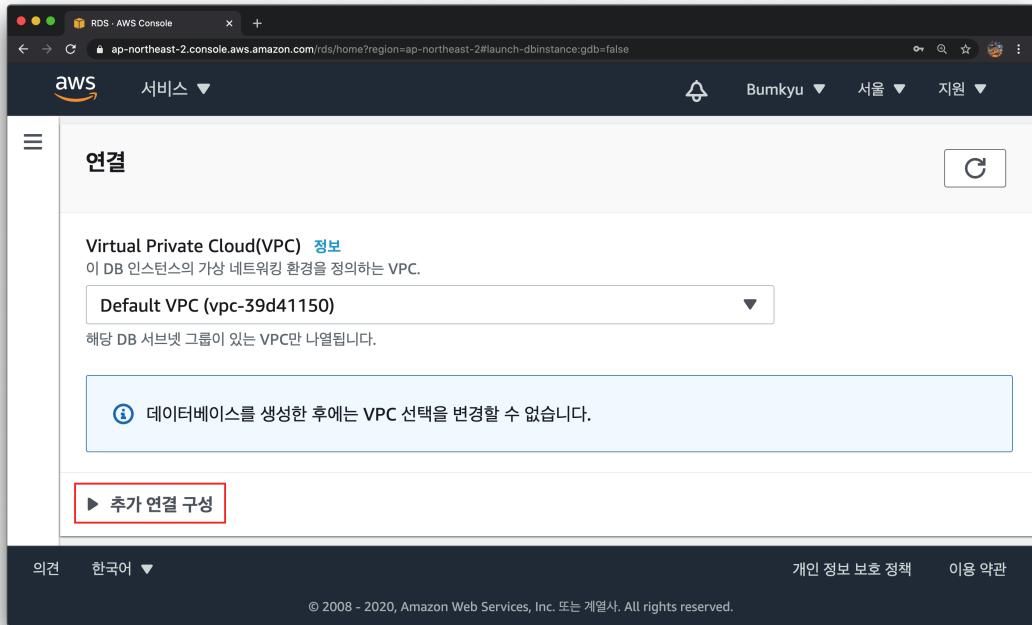
**최대 스토리지 임계값 정보**  
데이터베이스를 지정된 임계값으로 자동 조정하면 요금이 부과됩니다.

1000 GiB  
최소: 21 GiB, 최대: 16,384 GiB

의견 한국어 ▼ 개인 정보 보호 정책 이용 약관

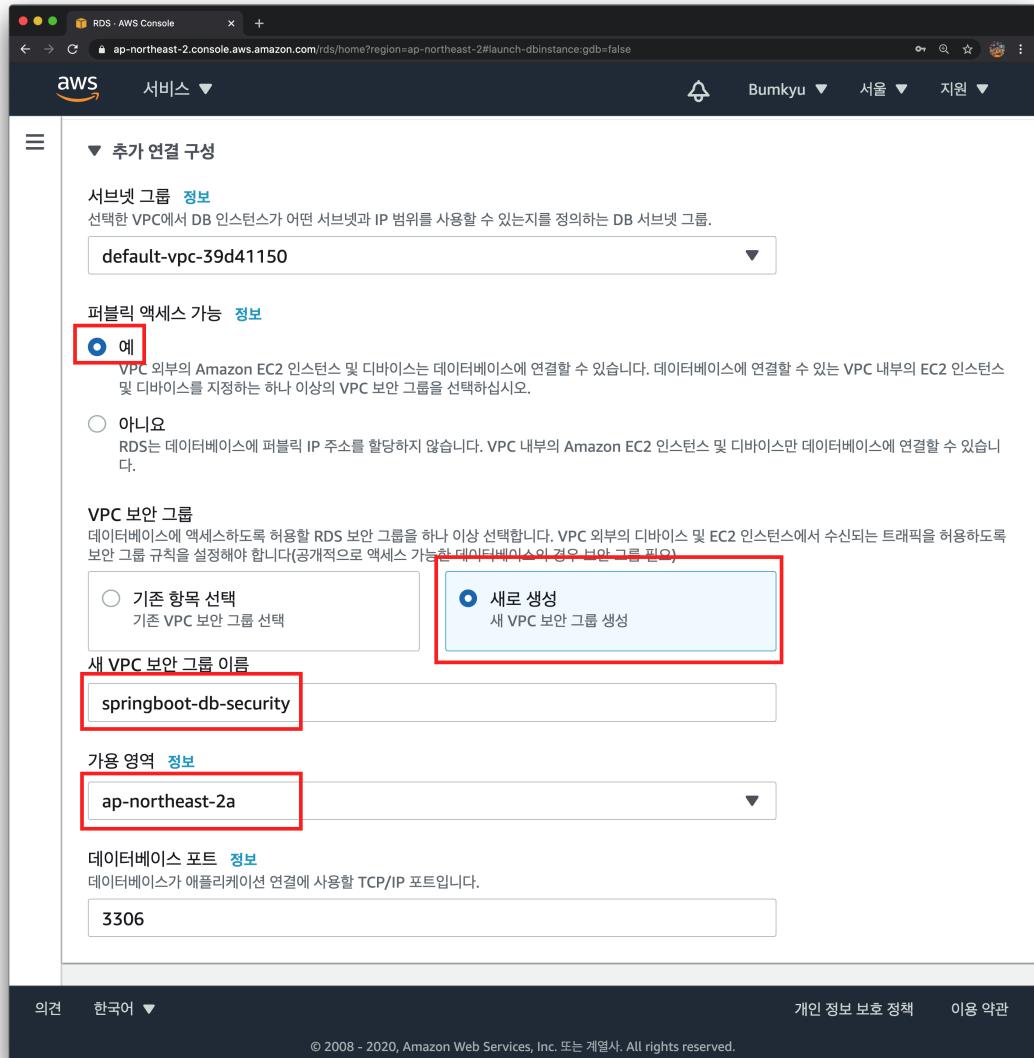
© 2008 – 2020, Amazon Web Services, Inc. 또는 계열사. All rights reserved.

9. [중요] 연결 > 추가 연결 구성 탭을 클릭합니다.

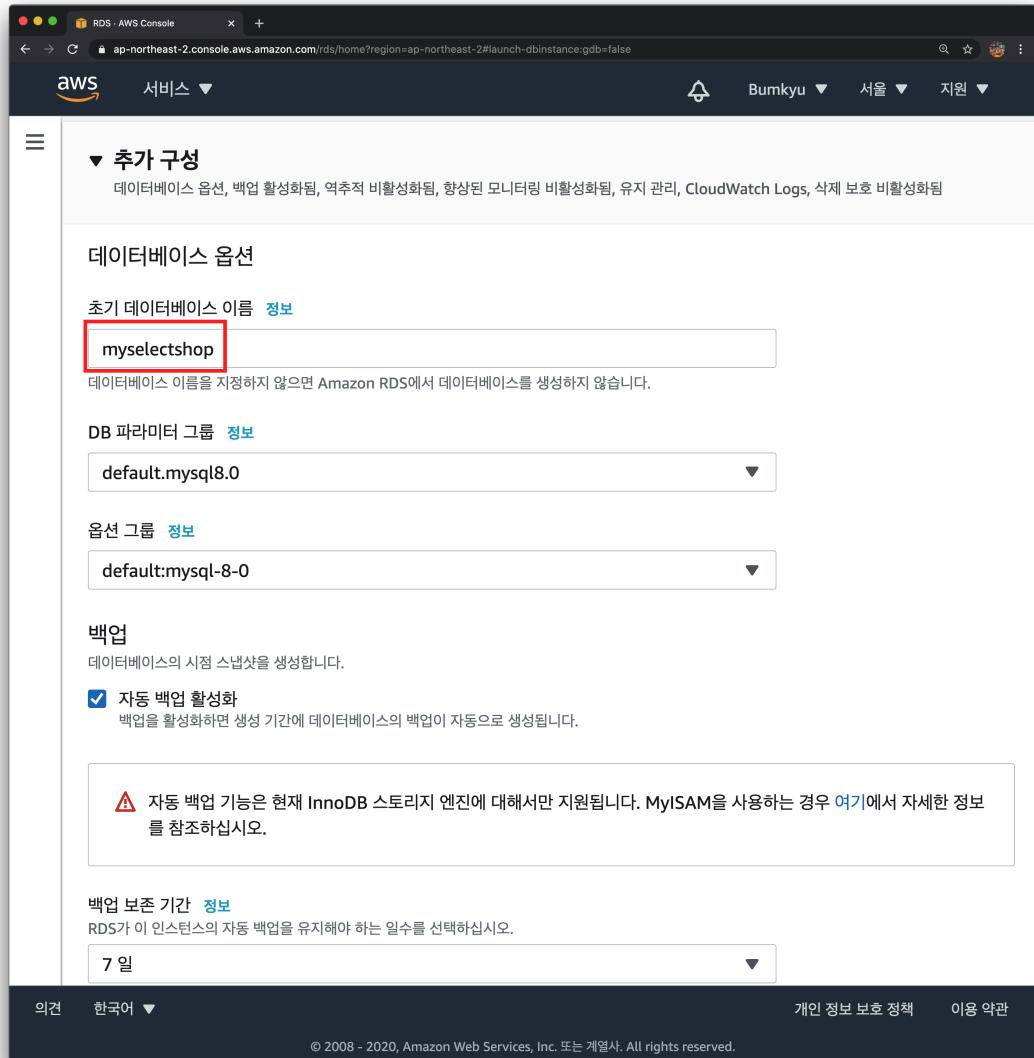


#### 10. 다음과 같이 설정합니다.

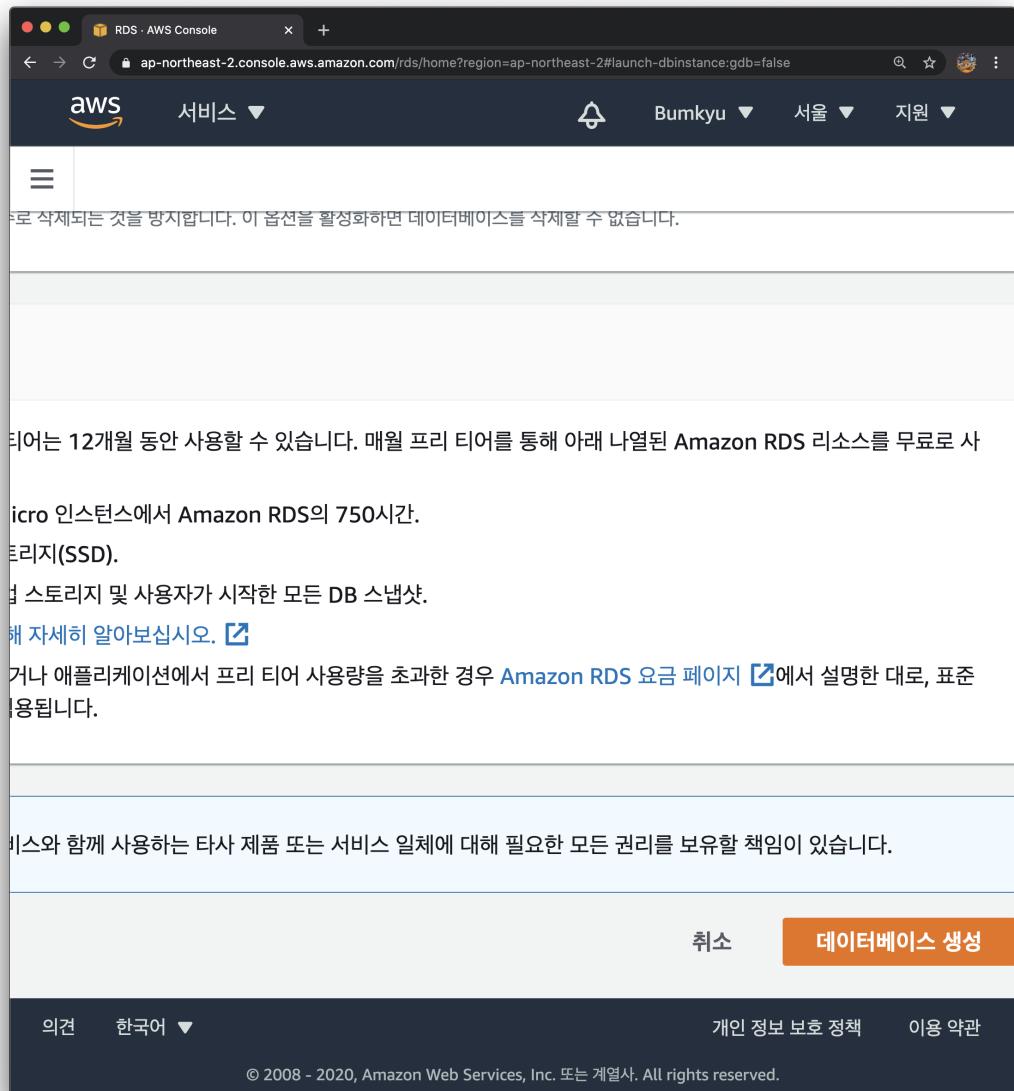
- **퍼블릭 액세스 기능: "예" 선택**  
→ 이 설정이 되어있어야 우리 컴퓨터에서 AWS RDS 의 MySQL과 연결이 가능합니다. 꼭 퍼블릭 액세스 설정을 확인해주세요!
- VPC 보안 그룹: "새로 생성" 선택
- 새 VPC 보안 그룹 이름: springboot-db-security
- 가용 영역: 옵션 중 아무거나 선택하셔도 괜찮아요.



11. 추가 구성 > 초기 데이터베이스 이름에 "myselectshop"을 입력한다.



## 12. 데이터베이스 생성 클릭



## ▼ RDS 포트 열어주기

1. `springboot-database` 클릭

The screenshot shows the AWS RDS console interface. On the left, there's a sidebar with various options like Dashboard, Database, Automated backups, and Proxies. The main area displays a table of databases. A message at the top says '데이터베이스 springboot-database 생성. 데이터베이스를 시작하는 데 몇 분 정도 걸릴 수 있습니다.' (Creating database springboot-database. It may take several minutes to start the database.) Below this, another message says 'In progress of creating a final snapshot, retaining automated backups and deleting database boot-database.' The table lists three databases: 'boot-database' (inactive, db.t2.micro), 'sparta' (active, db.t2.micro), and 'springboot-database' (creating, db.t2.micro).

## 2. 연결 & 보안 > 보안 > VPC 보안 그룹 의 springboot-db-security 클릭

This screenshot shows the detailed view of the 'springboot-database' database in the AWS RDS console. The '연결 & 보안' tab is active. In the '보안' section, it shows the VPC security group 'springboot-db-security' (sg-0a2c9bb0bbdd0f6d2) is associated with the database. Other details shown include the engine (MySQL Community), instance type (db.t2.micro), and region (ap-northeast-2a). The '연결 & 보안' section is highlighted with a red box.

## 3. 보안 그룹 ID 클릭

#### 4. 인바운드 규칙 편집 클릭

유형	프로토콜	포트 범위	소스	설명 - 선택 사항
MySQL/Aurora	TCP	3306	106.246.248.218/32	

#### 5. 소스 > 위치 무관 클릭해서 0.0.0.0/0, ::/0 생성 확인 후 "규칙 저장" 클릭

Inbound Security Group Rules configuration for sg-0a2c9bb0bdd0f6d2 - springboot-db-security

**Source:** Anywhere (0.0.0.0/0)

**Port Range:** 3306

### ▼ IntelliJ에서 확인하기

1. (링크) 이동 후 `springboot-database` 클릭

#### ▼ [코드스니펫] RDS 대시보드 링크

```
https://ap-northeast-2.console.aws.amazon.com/rds/home?region=ap-northeast-2#databases:
```

2. 엔드포인트 확인 후 드래그하여 복사

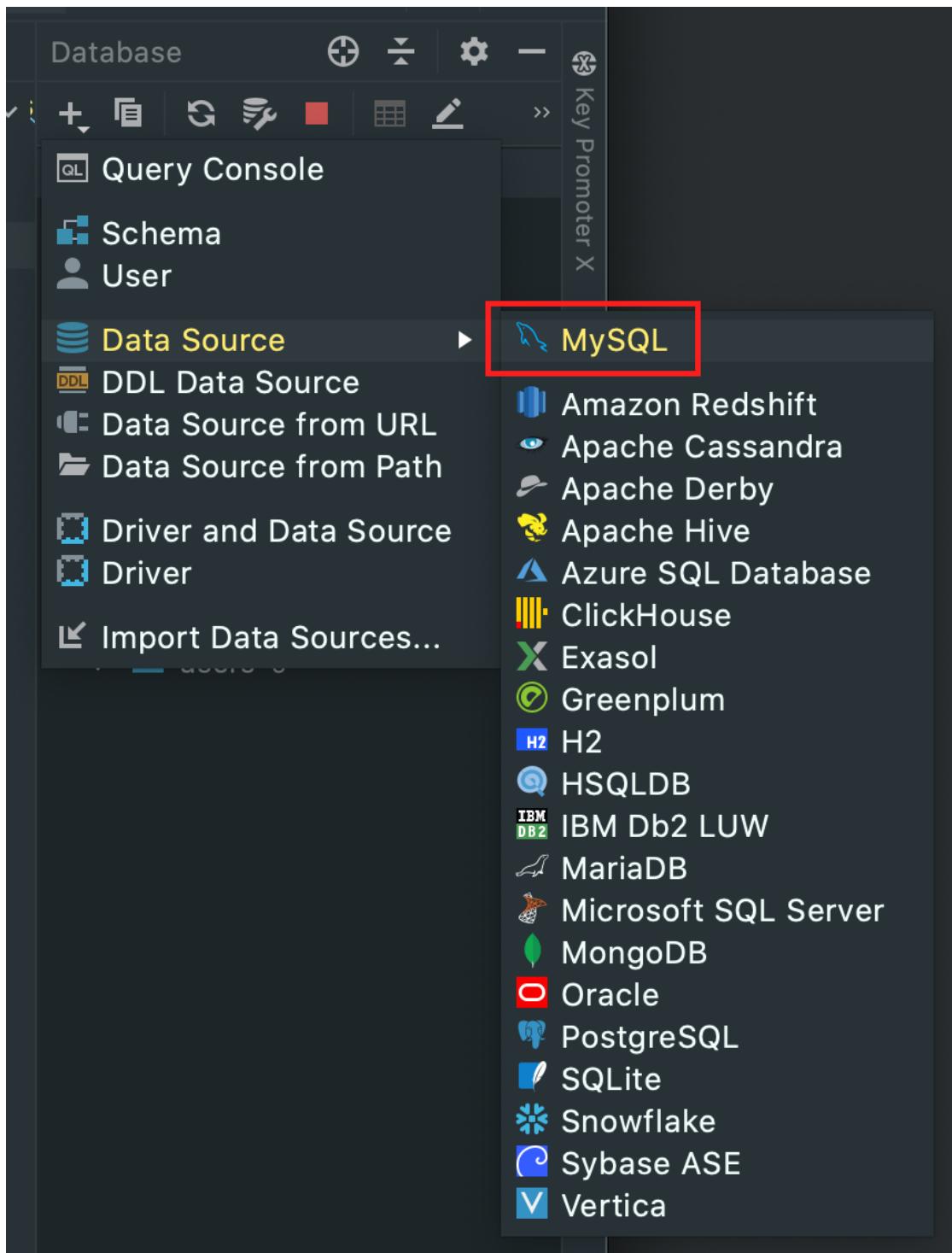
springboot-database

**Endpoint:** springboot-database.cbt9ceqjwlr9.ap-northeast-2.rds.amazonaws.com

3. week04 프로젝트를 IntelliJ에서 열기

4. 우측 Database 탭을 클릭합니다.

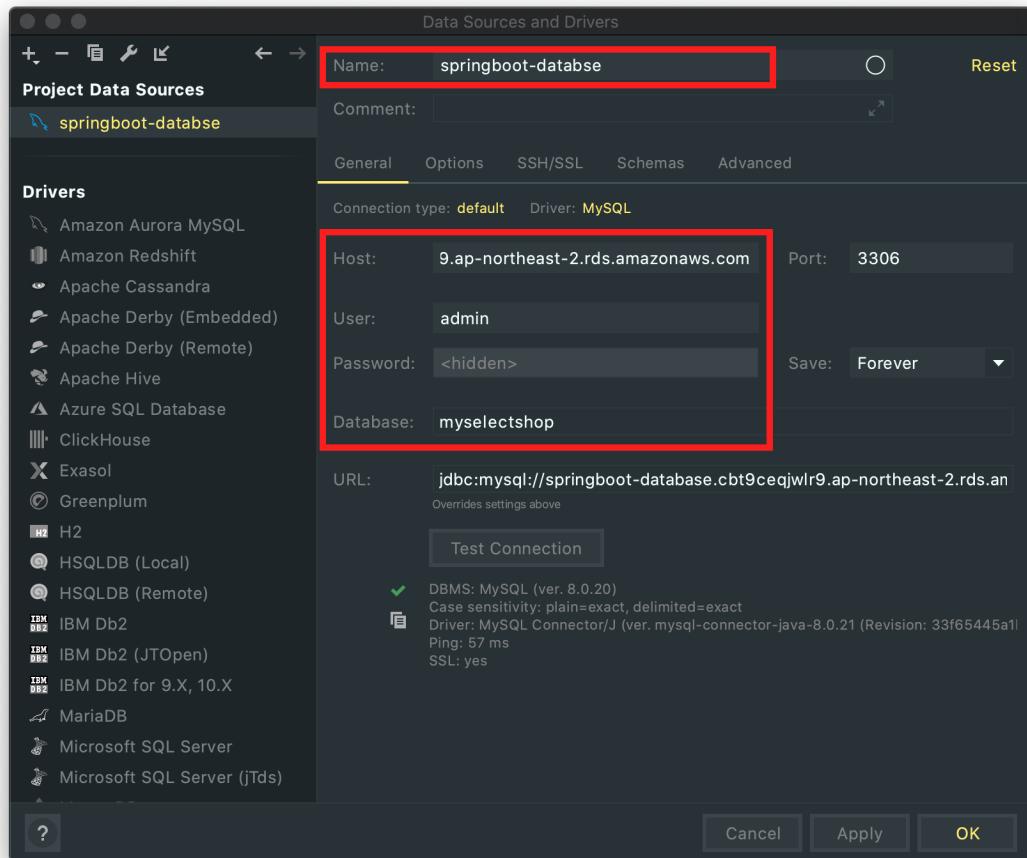
5. Data Source > MySQL 을 클릭합니다. (제일 상단이 아니라면 중간 즈음 있을 거에요!)



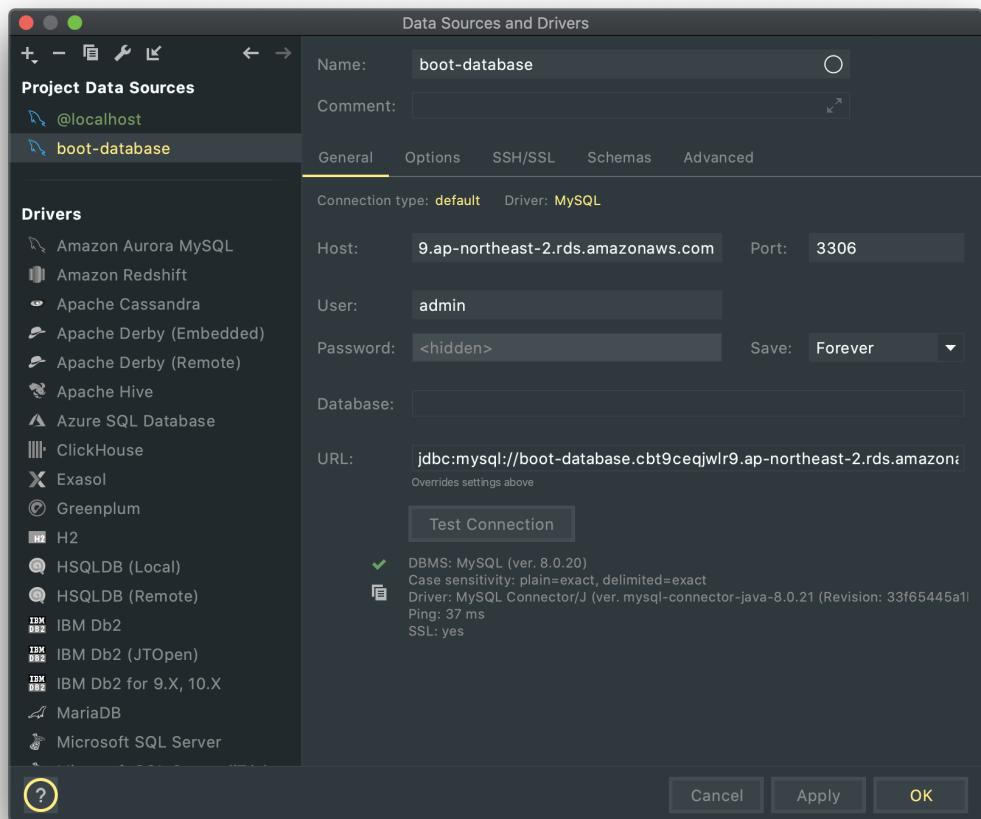
6. 다음과 같이 입력합니다.

- Name: springboot-database
- Host: 나의 엔드포인트
- User: 나의 Username
- Password: 나의 비밀번호

- Database: myselectshop



7. Test Connection 을 클릭하고, 아래와 같이 성공적으로 연결되면 OK 를 클릭합니다.



#### ▼ 스프링 부트를 MySQL과 연결하기

- 스프링 부트 설정의 대부분은 application.properties에서 관리합니다.

#### ▼ [코드스니펫] MySQL 연결하기

```
spring.datasource.url=jdbc:mysql://나의엔드포인트:3306/myselectshop
spring.datasource.username=나의USERNAME
spring.datasource.password=나의패스워드
spring.jpa.hibernate.ddl-auto=update
```

#### ▼ 연결 확인하기

- 프로젝트를 작동시키고, 관심 상품 등록 후, 서버를 내린 다음에 다시 작동시켰을 때 관심 상품 목록이 잘 나타남을 확인합니다.
- 쿼리 콘솔에서 아래 내용을 입력해서 product 테이블에 내용이 잘 들어가 있음을 확인합니다.

```
select * from product;
```

## 03. [5주차] OG 태그 작업하기

#### ▼ OG 이미지 다운로드

### ▼ [코드스니펫] OG 이미지 다운로드

```
https://s3.ap-northeast-2.amazonaws.com/materials.spartacodingclub.kr/spring/week05/og_selectshop.png
```

### ▼ OG title, description 작업하기

👉 OG title과 OG description 은, "나의 이름"을 넣어서 자유롭게 작성해보겠습니다 😎

### ▼ [코드스니펫] OG 태그

```
<meta property="og:title" content="00만의 셀렉샵">
<meta property="og:description" content="관심상품을 선택하고, 최저가 알림을 확인해보세요!">
<meta property="og:image" content="images/og_selectshop.png">
```

## 04. [5주차] EC2 구매하기

### ▼ EC2 서버 구매하기

#### ▼ AWS EC2 서버 사기

##### ▼ [코드스니펫] EC2 구매 주소

```
https://ap-northeast-2.console.aws.amazon.com/ec2/v2/home?region=ap-northeast-2
```

### ▼ 구매 화면들 따라하기

The screenshot shows the AWS EC2 Instances page. At the top, there's a navigation bar with 'Launch Instance', 'Connect', and 'Actions'. Below it is a search bar and a filter section. The main table lists two instances:

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP
sparta_websi...	i-02cfa0a8b7514ebbe	t2 micro	ap-northeast-2c	running	2/2 checks ...	None	ec2-13-209-143-91.ap...	13.209.143.91
수업용서버	i-095f6d46769d261c	t2 micro	ap-northeast-2c	running	2/2 checks ...	None	ec2-13-209-89-12.ap...	13.209.89.12

At the bottom of the page, there's a message: 'Select an instance above'.

**Step 1: Choose an Amazon Machine Image (AMI)**

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Search for an AMI by entering a search term e.g. "Windows"

Quick Start	AMIs	Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0d097db2fb6e0f05e	Select
My AMIs	Amazon Linux	Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras.	64-bit (x86)
AWS Marketplace	Amazon Linux	The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.	Select
Community AMIs	Amazon Linux	Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-0fd02cb7da42ee5e0	64-bit (x86)
Free tier only	Amazon Linux	Ubuntu Server 18.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical ( <a href="http://www.ubuntu.com/cloud/services">http://www.ubuntu.com/cloud/services</a> ).	Select

**Step 2: Choose an Instance Type**

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All instance types Current generation ShowHide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
General purpose	<b>t2.micro</b> Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes
General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes
General purpose	t2.2xlarge	8	32	EBS only	-	Moderate	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

**Step 7: Review Instance Launch**

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details

**Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-0fd02cb7da42ee5e0**  
 Ubuntu Server 18.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).  
 Root Device Type: ebs Virtualization type: hvm

Instance Type

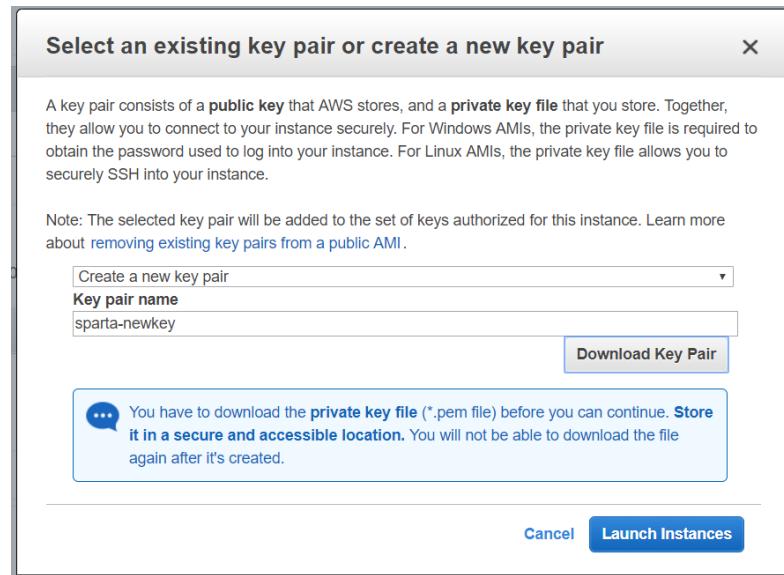
Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Security Groups

Security group name: launch-wizard-10  
 Description: launch-wizard-10 created 2019-10-03T15:25:50.845+09:00

Type	Protocol	Port Range	Source	Description
This security group has no rules				

Cancel Previous Launch

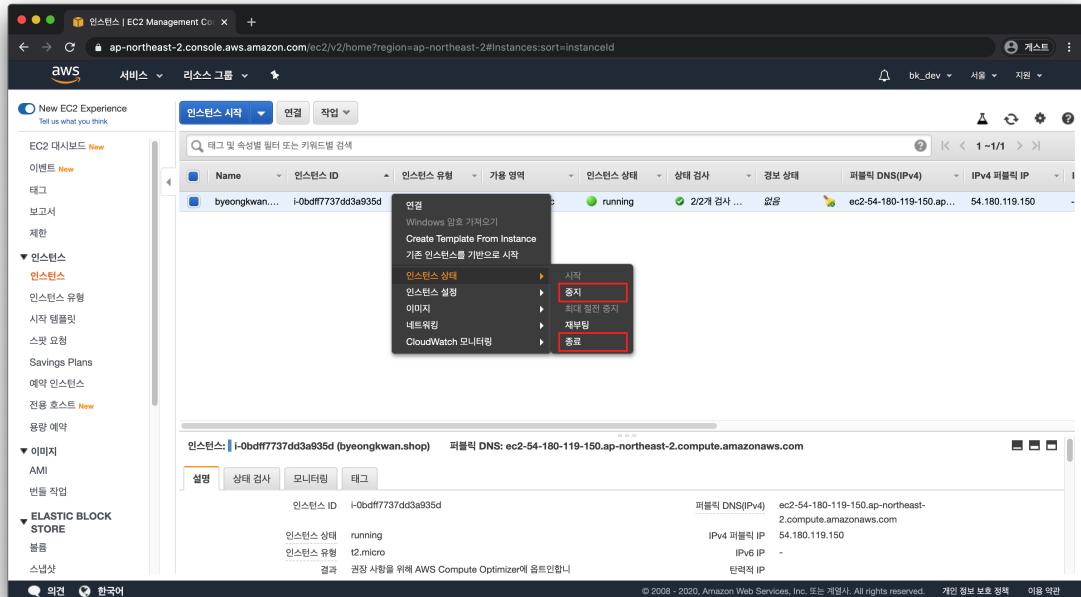


### ▼ EC2 서버 종료하는 방법 (1년 후 자동결제 방지!)



**중지 또는 종료하는 법. 무료 기간(1년) 후 결제가 되기 전에, 이렇게 종료하세요!**

대상 인스턴스에 마우스 우클릭 > '인스턴스 상태' 를 클릭합니다. 중지 또는 종료 중 하나를 클릭하면 명령을 실행합니다.



### ▼ EC2에 접속하기

#### ▼ AWS EC2에 접속하기 (준비 상식 편)

- SSH(Secure Shell Protocol)

- 다른 컴퓨터에 접속할 때 쓰는 프로그램입니다. 다른 것들 보다 보안이 상대적으로 뛰어납니다.

- 접속할 컴퓨터가 22번 포트가 열려있어야 접속 가능합니다. AWS EC2의 경우, 이미 22번 포트가 열려있습니다. 확인해볼까요?

The screenshot shows the AWS CloudWatch Metrics interface. At the top, there are tabs for 'Metrics' and 'CloudWatch Metrics'. Below that, a search bar and a filter section with dropdowns for 'Metric Name', 'Dimensions', and 'Time Range'. A table lists metrics with their names, dimensions, values, and unit. One metric is highlighted:

Metric Name	Dimensions	Value	Unit
AWS/EC2/Instances		1	Count

Below the table, a message says 'No data found for this time range'.

## ▼ AWS EC2에 접속하기

### ▼ Mac OS: Mac은 ssh가 있어서, 명령어로 바로 접근 가능!

- 터미널을 열기 (spotlight에 terminal 입력)
- 방금 받은 내 Keypair의 접근 권한을 바꿔주기

```
sudo chmod 400 받은키페어를끌어다놓기
```

- SSH로 접속하기

```
ssh -i 받은키페어를끌어다놓기 ubuntu@AWS에적힌내아이피
```

예) 아래와 비슷한 생김새!

```
ssh -i /path/my-key-pair.pem ubuntu@13.125.250.20
```

### ▼ Window: ssh가 없으므로, git bash라는 프로그램을 이용!

- gitbash를 실행하고, 아래를 입력!

```
ssh -i 받은키페어를끌어다놓기 ubuntu@AWS에적힌내아이피
```

예) 아래와 비슷한 생김새!

```
ssh -i /path/my-key-pair.pem ubuntu@13.125.250.20
```

- Key fingerprint 관련 메시지가 나올 경우 Yes를 입력해주세요!
- git bash를 종료할 때는 exit 명령어를 입력하여 ssh 접속을 먼저 끊어주세요.

▼ 간단한 리눅스 명령어 연습하기

- 리눅스는 윈도우 같지 않아서, '쉘 명령어'를 통해 OS를 조작한다. (일종의 마우스 역할)

[가장 많이 쓰는 몇 가지 명령어]

- 팁! 리눅스 커널에서 윗화살표를 누르면 바로 전에 썼던 명령어가 나옵니다.

```
ls: 내 위치의 모든 파일을 보여준다.
```

```
pwd: 내 위치(폴더의 경로)를 알려준다.
```

```
mkdir: 내 위치 아래에 폴더를 하나 만든다.
```

```
cd [갈 곳]: 나를 [갈 곳] 폴더로 이동시킨다.
```

```
cd .. : 나를 상위 폴더로 이동시킨다.
```

```
cp -r [복사할 것] [붙여넣기 할 것]: 복사 붙여넣기
```

```
rm -rf [지울 것]: 지우기
```

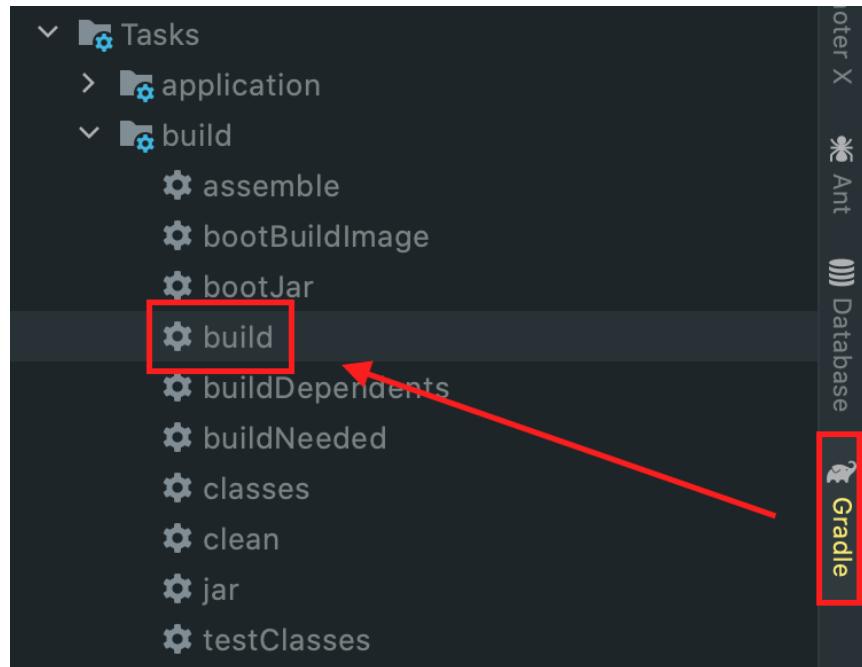
```
sudo [실행 할 명령어]: 명령어를 관리자 권한으로 실행한다.
```

```
sudo su: 관리자 권한으로 들어간다. (나올때는 exit으로 나옴)
```

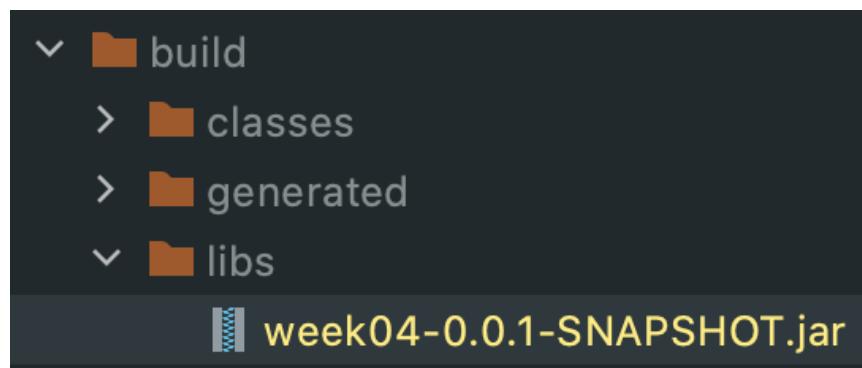
## 05. [5주차] EC2 준비하고 배포하기

▼ 배포 파일 빌드하기

- 우측 탭 중에서 Gradle 을 선택합니다.
- Tasks > build > build 를 더블 클릭합니다.



- 좌측 build 폴더 > libs 아래에 .jar 확장자로 끝나는 파일이 생겼을 거예요! 빌드에 성공하였습니다.

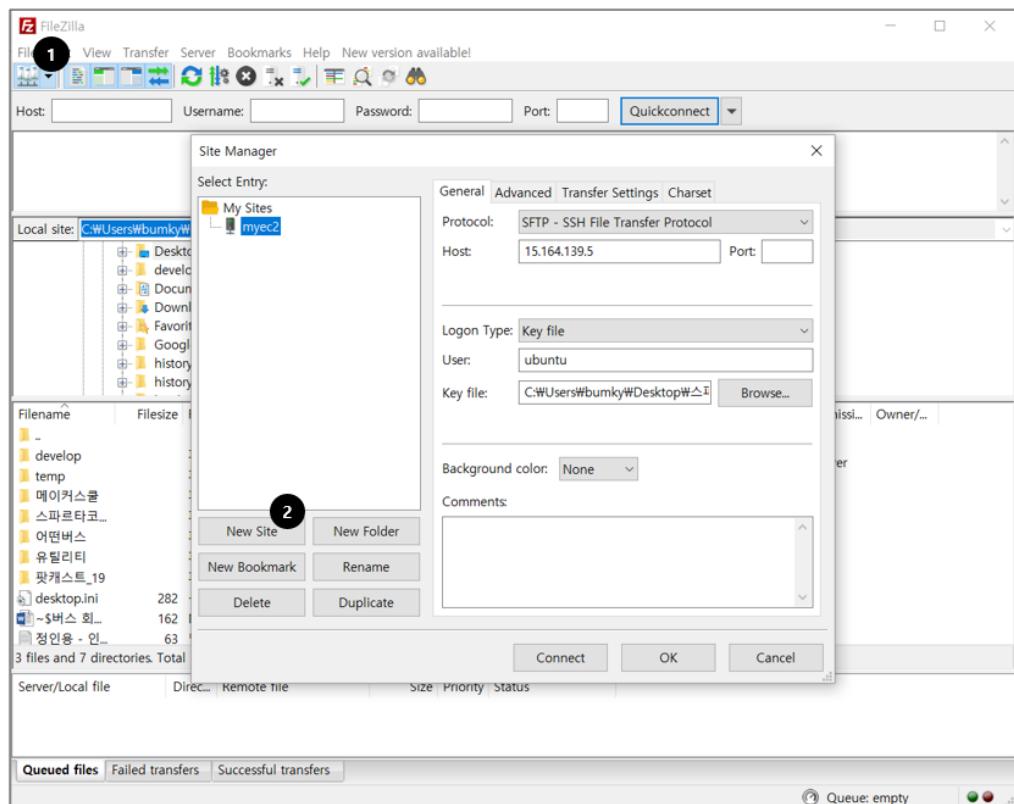


▼ OpenJDK 설치하기

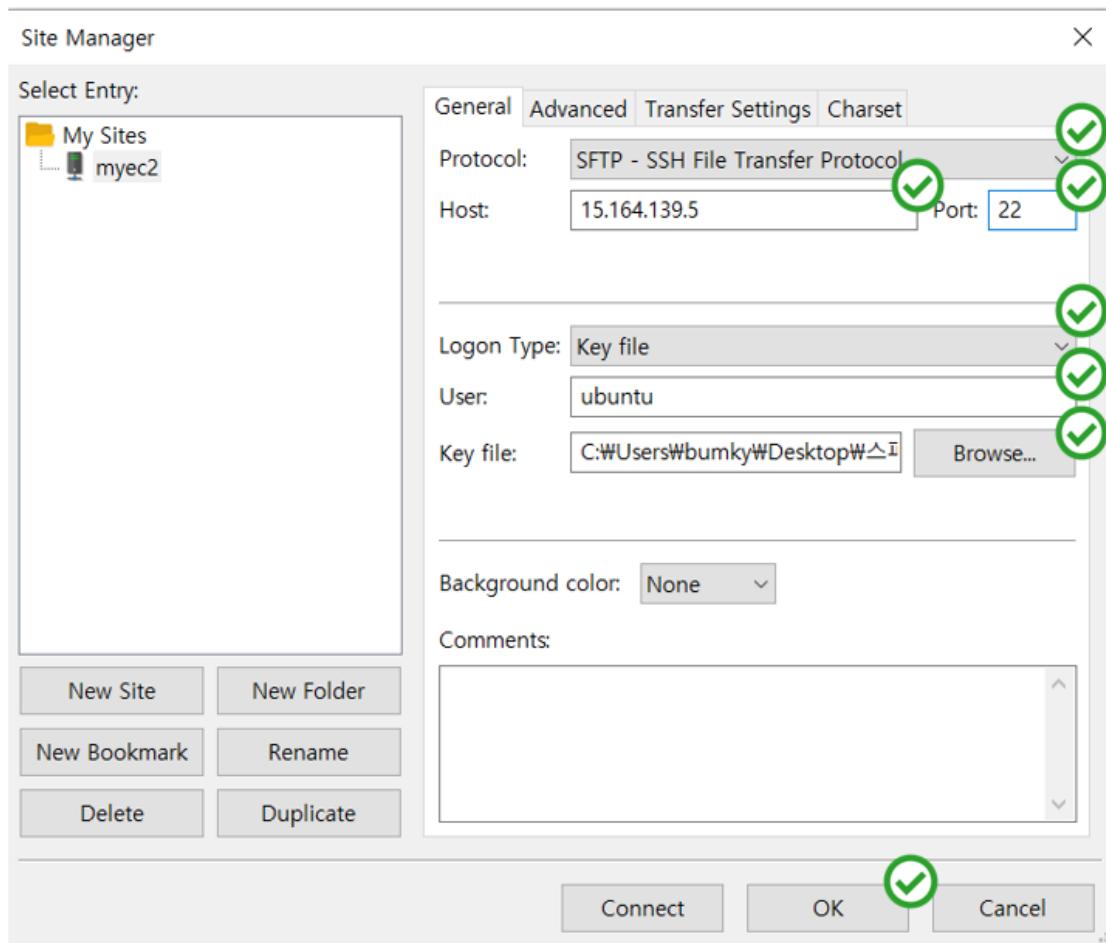
```
sudo apt-get update  
sudo apt-get install openjdk-8-jdk  
java -version
```

▼ Filezilla를 이용해서 배포 파일을 업로드하기

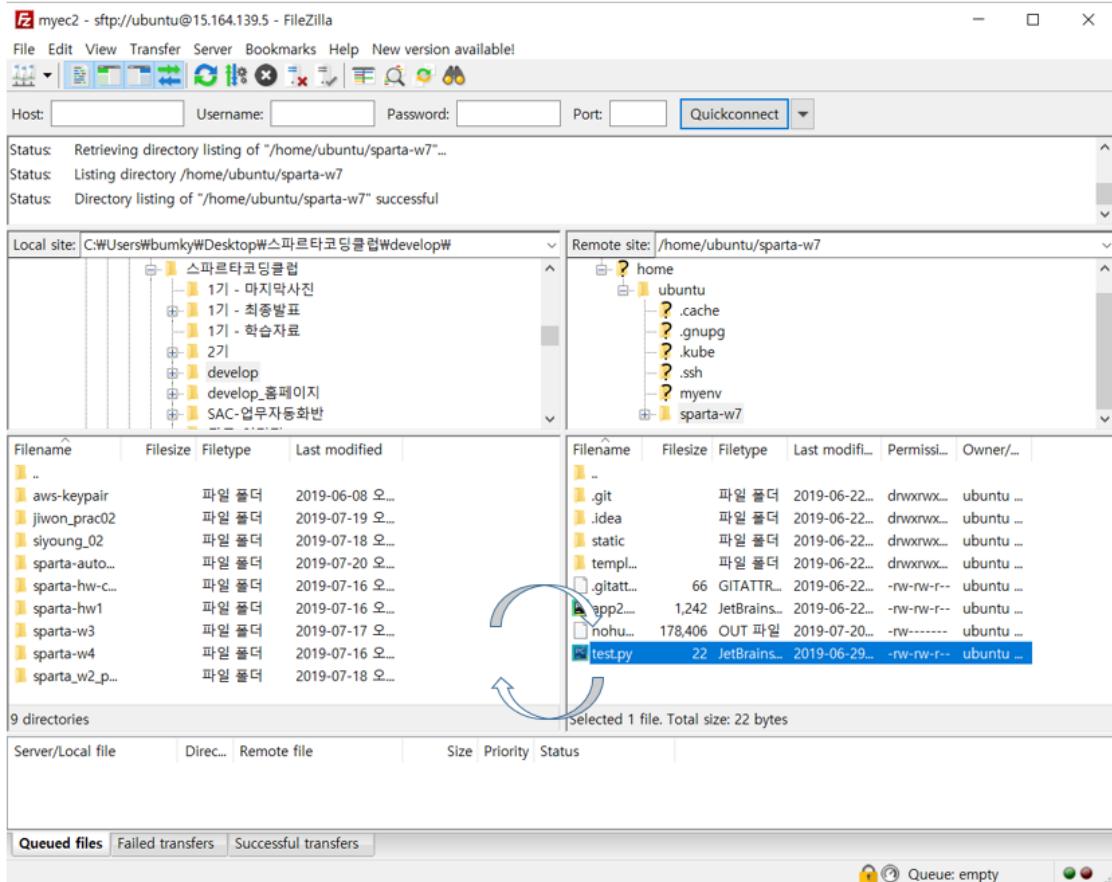
- 파일질라 실행, 다음과 같이 설정



- 정보들을 입력하고, ok 누르면 서버의 파일들을 볼 수 있음  
(Host: 내 EC2서버의 ip // User: ubuntu 로 입력)



- 마우스로 드래그 해서 파일을 업로드/다운로드하면 됩니다!



## ▼ 스프링 부트 작동시키기

```
java -jar JAR파일명.jar
```

## ▼ AWS에서 80, 8080 포트를 열어주기

- EC2 서버(=가상의 내 컴퓨터)에서 포트를 따로 설정하는 것 외에도, AWS EC2에서도 자체적으로 포트를 열고/닫을 수 있게 관리를 하고 있습니다.  
→ 그래서 AWS EC2 Security Group에서 인바운드 요청 포트를 열어줘야 합니다.
- 일단, EC2 관리 콘솔로 들어갑니다. 그리고 보안그룹(영문: Security Group)을 눌러 들어갑니다. 여기선 launch-wizard-1이라고 쓰여 있네요

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like '서비스', '리소스 그룹', 'AWS Lambda', 'Amazon S3', etc. Under 'EC2 대시보드', there are sections for '이벤트', '태그', '보고서', '제한', and '인스턴스'. A red checkmark is placed next to '인스턴스'. Below these are sections for '시작 템플릿', '스프 요청', '예약 인스턴스', '전용 호스트', '용량 예약', '이미지', 'AMI', '변동 작업', 'ELASTIC BLOCK STORE', '블룸', '스냅샷', '수명 주기 관리자', '네트워크 및 보안', '보안 그룹', '탄력적 IP', and '네트워크 카페'. The main content area displays two instances:

Name	인스턴스 ID	인스턴스 유형	가용 영역	인스턴스 상태	상태 검사	경보 상태	퍼블릭 DNS(IPv4)	IPv4 퍼블릭 IP	IPv6 IP
sparta_website_new	i-02cfa0a8b7514ebbe	t2.micro	ap-northeast-2c	running	2/2 검사 통과	없음	ec2-13-209-143-91.ap...	13.209.143.91	-
sparta_website	i-0a0fc904b368d003a	t2.micro	ap-northeast-2c	stopped	-	-	-	-	-

Below the table, a detailed view of the selected instance 'sparta\_website\_new' is shown. It includes fields for '설명', '상태 검사', '모니터링', and '태그'. The '설명' tab is active. The instance details are as follows:

- 인스턴스 ID: i-02cfa0a8b7514ebbe
- 인스턴스 상태: running
- 인스턴스 유형: t2.micro
- 탄력적 IP: 13.209.143.91\*
- 가용 영역: ap-northeast-2c
- 보안 그룹: launch-wizard-7
- 예약된 이벤트: 없음
- AMI ID: ubuntu/images/hvm-ssd/ubuntu-bionic-18.04-amd64-server-20190722.1 (ami-0fd02cb7d42ee5e0)
- 플랫폼: -
- IAM 역할: -
- 기 페어 이름: sparta\_bk\_190805
- 퍼블릭 DNS(IPv4): ec2-13-209-143-91.ap-northeast-2.compute.amazonaws.com
- IPv4 퍼블릭 IP: 13.209.143.91
- IPv6 IP: -
- 프라이빗 DNS: ip-172-31-17-74.ap-northeast-2.compute.internal
- 프라이빗 IP: 172.31.17.74
- 보조 프라이빗 IP: -
- VPC ID: vpc-39d41150
- 서브넷 ID: subnet-1cf8f156
- 네트워크 인터페이스: eth0
- 소스/대상 확인: 예
- T2/T3 무제한 비활성: -

- 해당 보안그룹을 클릭합니다.

The screenshot shows the AWS Security Groups page. The left sidebar has the same navigation as the previous screenshot. The main content area shows a table of security groups:

Security group ID	Security group name	VPC ID	Description	Owner	Inbound rules count	Outbound rules count
<a href="#">sg-Debf753ec1e1150d8</a>	launch-wizard-7	vpc-5b3df430	launch-wizard-7 create...	781639735494	9 Permission entries	1 Permission entry

A red box highlights the row for 'sg-Debf753ec1e1150d8'. Below the table, a detailed view of the selected security group 'sg-Debf753ec1e1150d8 - launch-wizard-7' is shown. It includes tabs for 'Details', 'Inbound rules' (which is selected), 'Outbound rules', and 'Tags'. The 'Inbound rules' tab shows a single rule:

Source	Port range	Protocol	Action
0.0.0.0/0	22	tcp	Allow

- Edit inbound rules를 선택합니다.

- 두 가지 포트를 추가해봅니다.

Type	Protocol	Port range	Source	Description - optional
HTTP	TCP	80	0.0.0.0/0	-
HTTP	TCP	80	::/0	-
SSH	TCP	22	0.0.0.0/0	-
Custom TCP	TCP	5000	0.0.0.0/0	-
Custom TCP	TCP	5000	::/0	-
Custom TCP	TCP	27017	0.0.0.0/0	-
Custom TCP	TCP	27017	::/0	-

**인바운드 규칙 편집 정보**

인바운드 규칙은 인스턴스에 도달하도록 허용된 수신 트래픽을 제어합니다.

**인바운드 규칙 정보**

유형: 사용자 지정 TCP | 프로토콜: TCP | 포트 범위: 8080 | 소스: 사용자 ... | 설명: -

수신 트래픽 설정: 0.0.0.0/0 | 삭제

수신 트래픽 설정: ::/0 | 삭제

수신 트래픽 설정: 22 | 삭제

수신 트래픽 설정: 0.0.0.0/0 | 삭제

수신 트래픽 설정: 80 | 삭제

**규칙 추가**

**주의:** 기존 규칙을 편집하면 편집된 규칙이 삭제되고 새 세부 정보로 새 규칙이 생성됩니다. 이렇게 하면 새 규칙이 생성될 때까지 해당 규칙에 의존하는 트래픽이 잠시 중단될 수 있습니다.

**취소** | **변경 사항 미리 보기** | **규칙 저장**

→ 80포트: HTTP 접속을 위한 기본포트

→ 8080포트: 스프링 부트 기본포트

#### ▼ 접속해보기

`http://내|AWS아이피:8080`

## 06. [5주차] 포트포워딩 & nohup 적용하기

#### ▼ 포트 번호 없애기 - 기본 개념

- 지금은 8080 포트에서 웹 서비스가 실행되고 있습니다. 그래서 매번 :8080 이라고 뒤에 붙여줘야 하죠. 뒤에 붙는 포트 번호를 없애려면 어떻게 해야할까요?
- http 요청에서는 80포트가 기본이기 때문에, 굳이 :80을 붙이지 않아도 자동으로 연결이 됩니다.
- 포트 번호를 입력하지 않아도 자동으로 접속되기 위해, 우리는 80포트로 오는 요청을 8000 포트로 전달하게 하는 포트포워딩(port forwarding) 을 사용하겠습니다.

▼ 포트 번호 없애기 - 리눅스 자체 포트포워딩을 작동시키기

- 띄워둔 터미널에서 새롭게 설정을 적용하기 전에 돌아가고 있던 서비스는 일단 끄고! (터미널에서 ctrl + c)
- 포트포워딩 룰을 입력합니다.

▼ [코드스니펫] 포트포워딩

```
sudo iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j REDIRECT --to-port 8080
```

- 다시 서비스 시작

```
java -jar JAR파일명.jar
```

▼ 포트 번호를 떼고 접속해봅니다.

- 웹 브라우저에서 접속하기

```
http://내AWS아이피/
```

▼ SSH 접속을 끊어도 서버가 계속 돌아가기

- 현재 상황

Git bash 또는 맥의 터미널을 종료하면 (=즉, SSH 접속을 끊으면) 프로세스가 종료되면서, 서버가 돌아 가지 않고 있습니다. 그러나 우리가 원격접속을 끊어도, 서버는 계속 동작해야겠죠?

- 원격 접속을 종료하더라도 서버가 계속 돌아가게 하기

```
# 아래의 명령어로 실행하면 된다
nohup java -jar JAR파일명.jar &
```

- 서버 종료하기 - 강제종료하는 방법

```
# 아래 명령어로 미리 pid 값(프로세스 번호)을 본다
ps -ef | grep java

# 아래 명령어로 특정 프로세스를 죽인다
kill -9 [pid값]
```

- 다시 켜기

```
# 아래의 명령어로 실행하면 된다
nohup java -jar JAR파일명.jar &
```

▼ SSH 접속을 종료한 뒤, 접속해봅니다!

- 브라우저에서 접속하기

```
http://내AWS아이피/
```

## 07. [5주차] 도메인 붙이고 카카오톡에 공유해보기

▼ 도메인 구입/연결

- 도메인을 구매한다는 것은, 네임서버를 운영해주는 업체에, IP와 도메인 매칭 유지비를 내는 것입니다. 한국 또는 글로벌 업체 어디든 상관 없지만, 우리는 한국의 '가비아'라는 회사에서 구입해보겠습니다.

▼ 구입 후 아래 화면을 띄워주세요

▼ [코드스니펫] 가비아 링크

```
https://my.gabia.com/service#/
```

The screenshot shows a table of domain registrations. The columns are: 도메인 (Domain), 서비스 관리 (Service Management), 비용 결제 (Cost Settlement), 결제 관리 (Payment Management), and My 정보 관리 (My Information Management). The first row shows a domain registered from May 24, 2020, to May 24, 2021, at a yearly cost of 53,900원. A blue button labeled '관리툴' (Management Tools) is highlighted with a red box. At the bottom of the table, there are navigation arrows and a page number '1'. A blue circular button with a white upward arrow and the text 'TOP' is located in the bottom right corner.

도메인	서비스 관리	비용 결제	결제 관리	My 정보 관리
spart@example.shop		2020-05-24	2021-05-24	53,900원/년

(도메인 연결 클릭)

전체 도메인

도메인 정보 변경

DNS 정보

도메인 보안

예약 도메인 관리

관심 도메인

spartaexample.shop

등록일: 2020-05-24 만기일: 2021-05-24 (남은 기간: 364일)

인증 코드

만기일 맞춤 도메인 연장 연장 알림

소유자	수정	소유권 이전	관리자	수정
이범규 bumkyu.lee@gmail.com 010-7127-4503			이범규 bumkyu.lee@gmail.com 010-7127-4503	

도메인 카테고리

1544-4370

도메인 고객을 위한 특별한 혜택  
마이크로사이트  
무료 제공!

네임서버 설정

1차 ns.gabia.co.kr	2차 ns1.gabia.co.kr	3차 ns.gabia.net	4차 데이터 없음	5차 데이터 없음	6차 데이터 없음
-------------------	--------------------	-----------------	-----------	-----------	-----------

보안 서비스

등록정보 숨김 설정

안전 장금 설정

기과 이전 차단 설정

DNS 정보

도메인 연결 설정

포워딩 설정

파킹 설정

ICANN 60일 정책

(DNS 설정 클릭)

DNS 설정

DNS 설정 도메인 연결 포워딩 웹 파킹 모바일 파킹

도메인 명

DNS 정보

<input type="checkbox"/>	도메인 명	설정된 DNS 레코드 정보가 없습니다.	설정
<input type="checkbox"/>		TXT	설정
<input type="checkbox"/>		A	설정
<input checked="" type="checkbox"/>	spartaexample.shop	A	설정
<input type="checkbox"/>		TXT / MX	설정
<input type="checkbox"/>		설정된 DNS 레코드 정보가 없습니다.	설정

(호스트 이름에 @, IP주소에 IP주소를 입력합니다)

레코드 개수: 0개 최근 업데이트: - 네임서버: ns0.wixdns.net

타입 호스트 값 TTL 우선 순위 서비스 상태

A 호스트 이름 IP 주소 180 DNS 설정 확인 삭제

+ 레코드 추가

DNS 설정 목록 저장

(이렇게!)

spartaexample.shop

레코드 개수: 1개 최근 업데이트: 2020-05-24 19:16:33 네임서버: ns.gabia.co.kr

이력 확인 엑셀 다운로드

DNS 설정 레코드 수정

타입	호스트	값/위치	TTL	우선 순위	서비스
A	@	13.125.61.109	1800		DNS 설정

▼ 10분 정도 기다려주세요!

- 네임서버에 내 도메인-IP가 매칭되는 시간이 필요합니다.

▼ IP주소로 접근해보기

- 내 IP주소로 지금 스프링 서버가 잘 돌고 있나요? 먼저 확인해봅니다.

http://내AWS아이피/

- 약간의 시간을 가진 후, 내 도메인으로 접근하면, 접속이 됩니다.

http://내도메인/

▼ 카카오톡 공유하기



드디어 우리가 애써 만든 작품을 공유할 시간입니다! 🎉

가족, 친구, 동료 단톡방 등에 나의 URL을 공유해보세요. (OG 태그가 짠! 😎)

## 08. [5주차] 끝 & 숙제설명



### 내 도메인을 제출해주세요!

착착착 잘 따라오셨다면, 나만의 셀렉샵을 EC2에 잘 올려두셨을 거예요.

- 열심히 돌아가고 있는 스프링 웹서비스를 제출하시면, 이번 숙제는 끝이랍니다~! 😊

Copyright © TeamSparta All rights reserved.