



# [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". The "RDS" service is highlighted with a red box. To the right of the search bar, there's a sidebar with sections for "이동 중에도 AWS 리소스와 연결 유지" (Stay connected to AWS resources while on the go), "AWS 탐색" (AWS Explorer), and specific details for "Amazon Redshift", "AWS Fargate", and "Amazon S3". The bottom of the page includes standard AWS footer links like "AWS 블로그", "AWS 제품", "AWS 기술", and "AWS 고객센터".

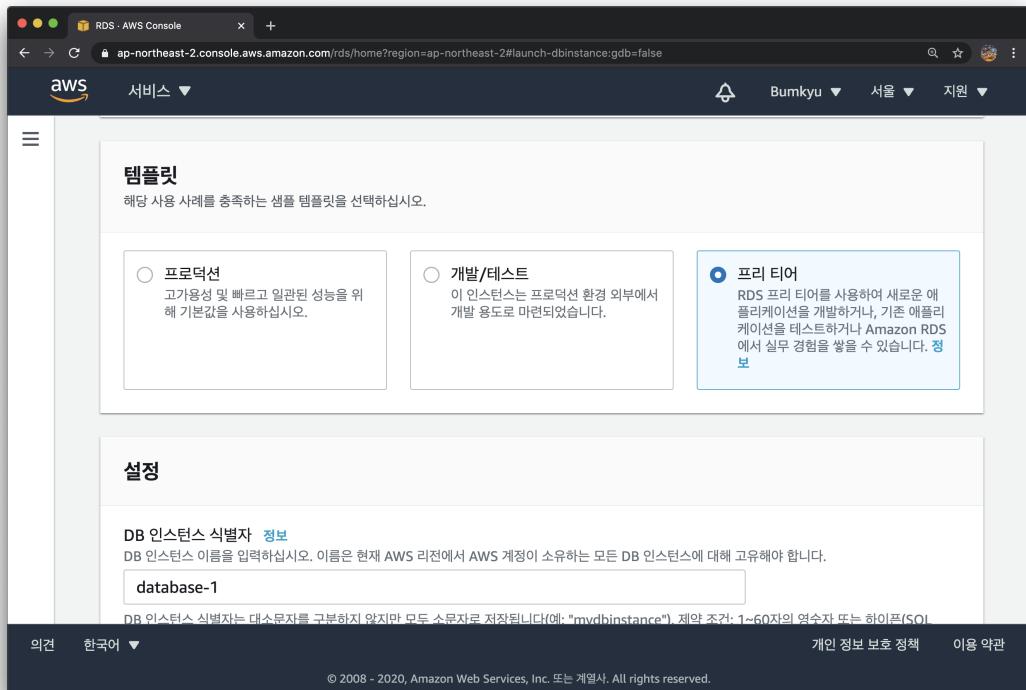
- "데이터베이스 생성"을 클릭합니다.

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.

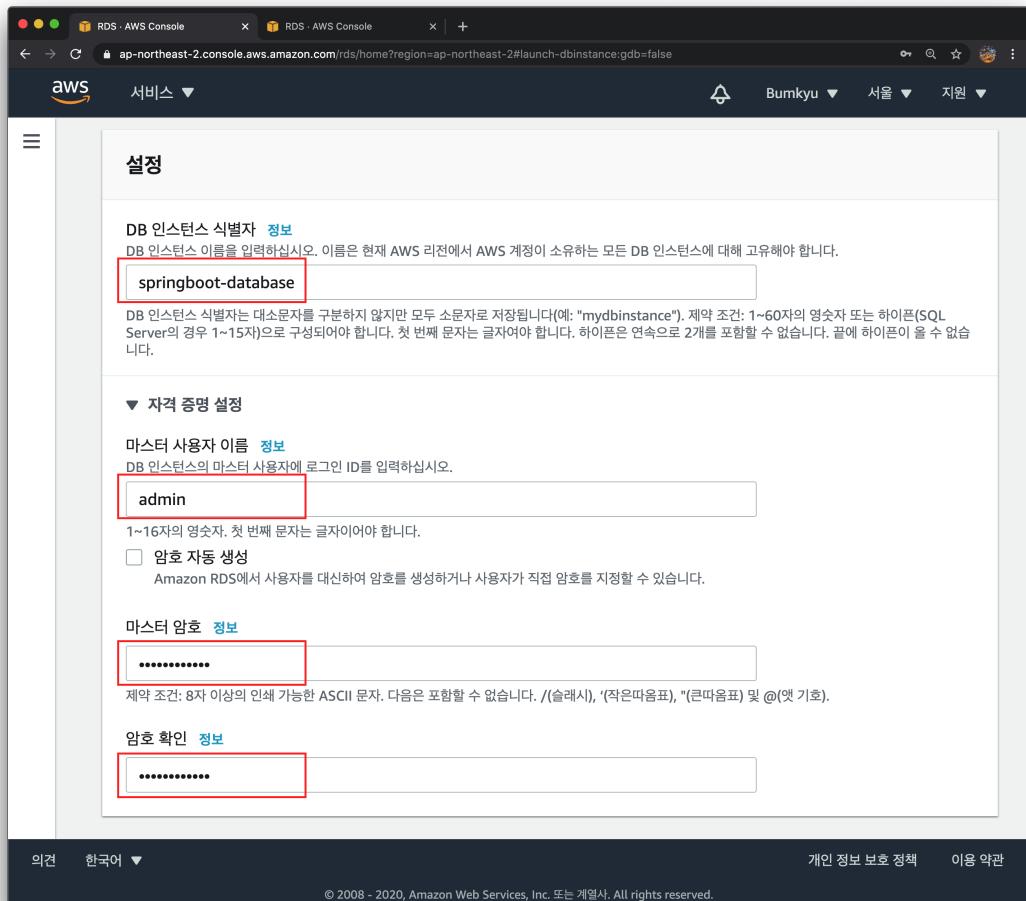
This screenshot shows the "Create Database" wizard on the "Standard Creation" tab. It asks for the engine type. Under the "엔진" (Engine) heading, the "MySQL" option is selected, indicated by a blue circle and highlighted with a blue border. Other options shown are "Amazon Aurora" and "MariaDB". The "Amazon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

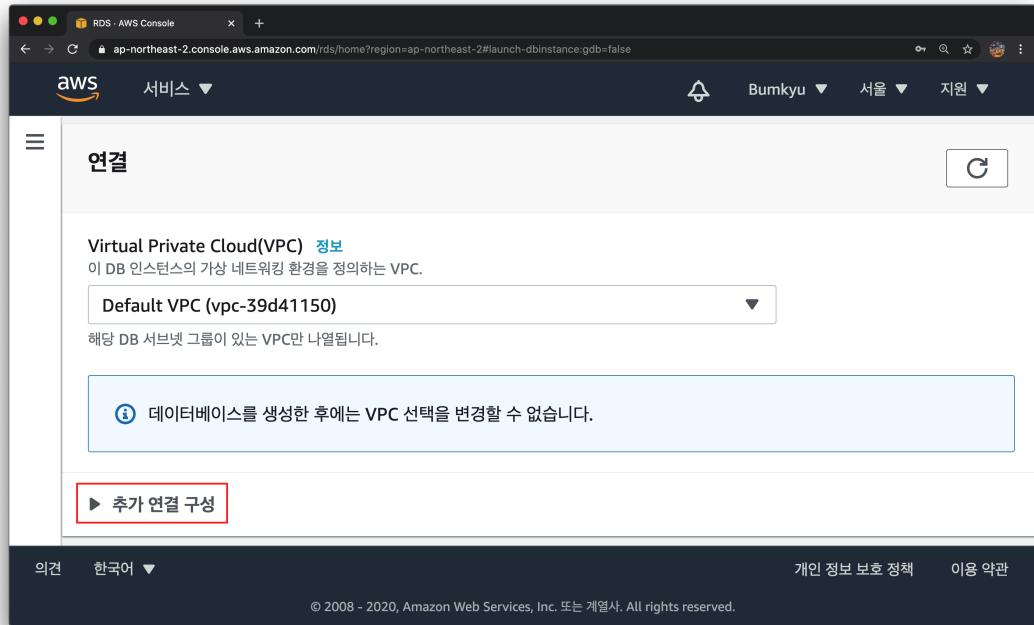
**최대 스토리지 임계값 정보**  
데이터베이스를 지정된 임계값으로 자동 조정하면 요금이 부과됩니다.

1000 GiB  
최소: 21 GiB, 최대: 16,384 GiB

의견 한국어 ▼ 개인 정보 보호 정책 이용 약관

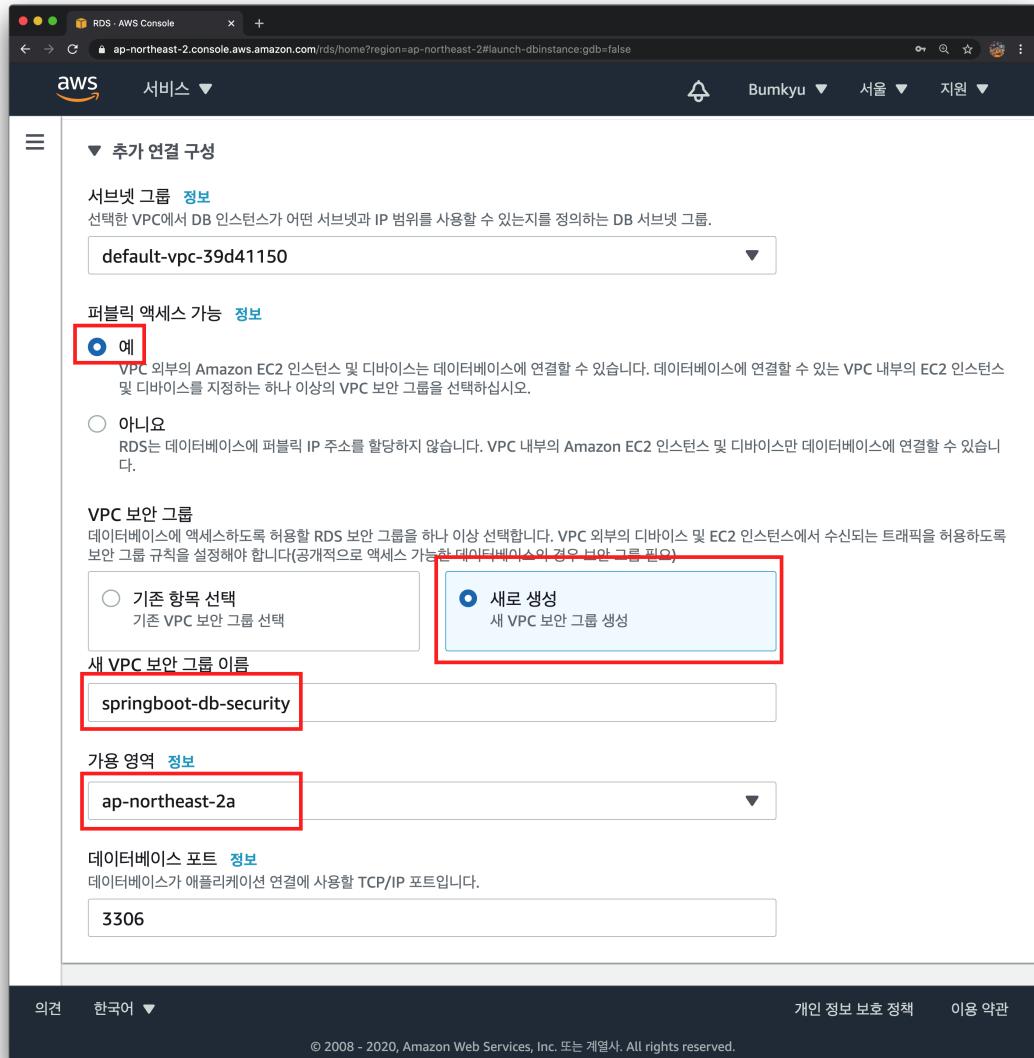
© 2008 – 2020, Amazon Web Services, Inc. 또는 계열사. All rights reserved.

9. [중요] 연결 > 추가 연결 구성 탭을 클릭합니다.

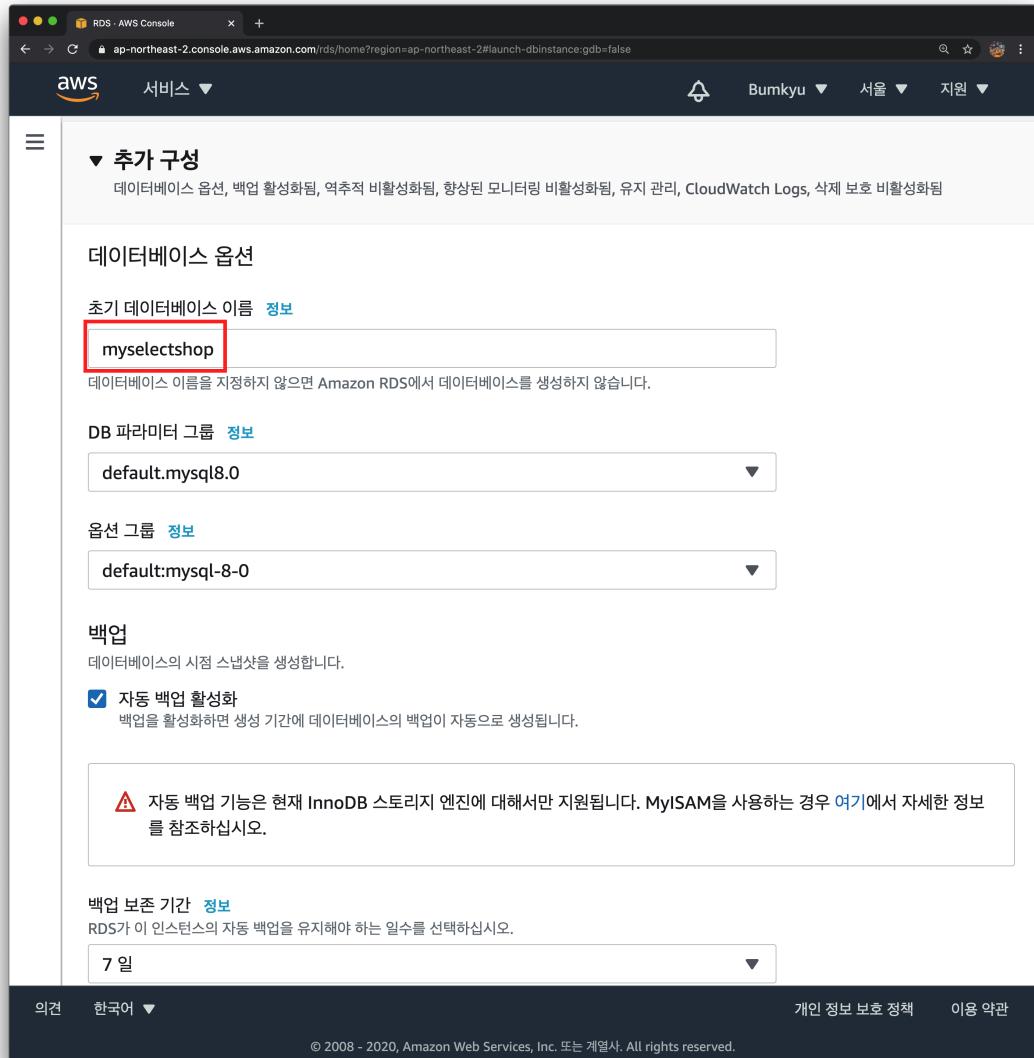


#### 10. 다음과 같이 설정합니다.

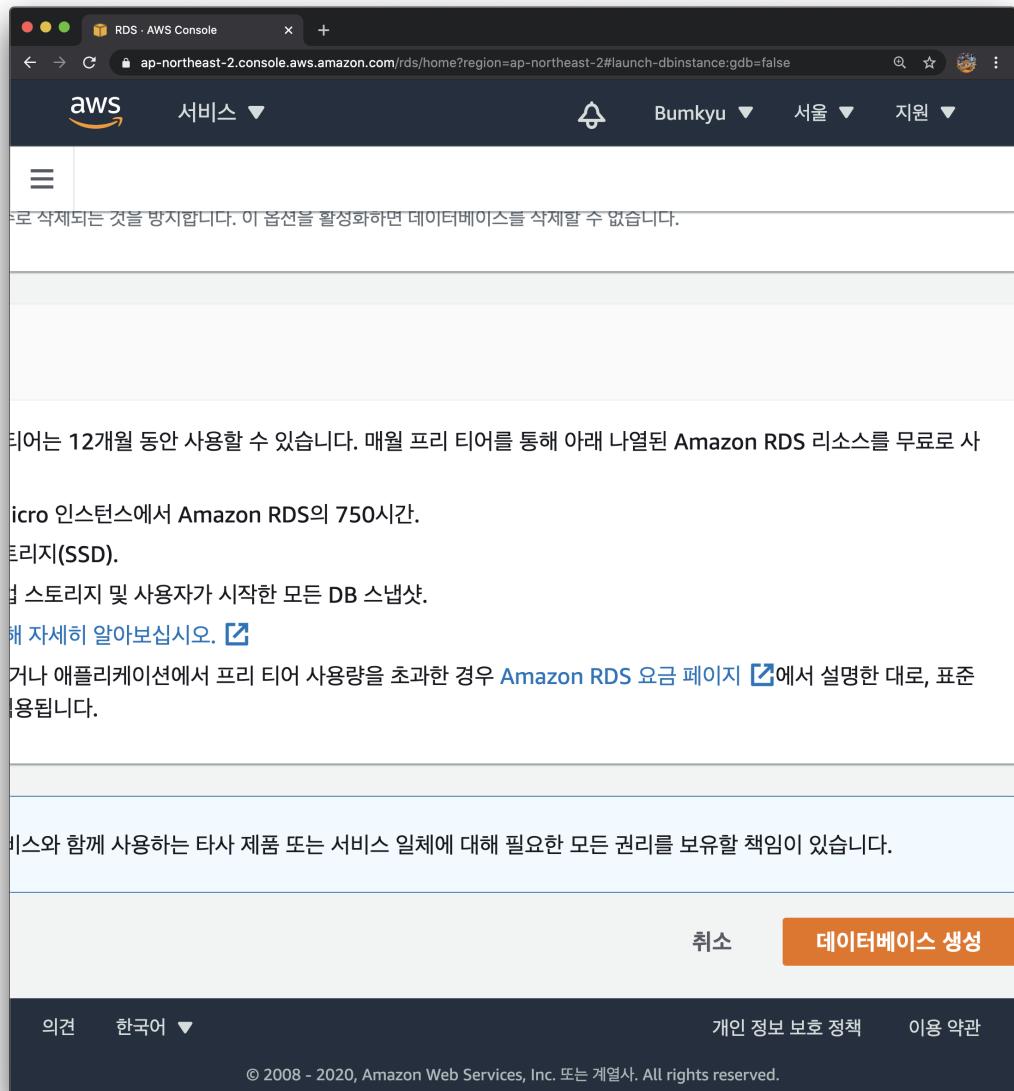
- **퍼블릭 액세스 기능: "예" 선택**  
→ 이 설정이 되어있어야 우리 컴퓨터에서 AWS RDS 의 MySQL과 연결이 가능합니다. 꼭 퍼블릭 액세스 설정을 확인해주세요!
- VPC 보안 그룹: "새로 생성" 선택
- 새 VPC 보안 그룹 이름: springboot-db-security
- 가용 영역: 옵션 중 아무거나 선택하셔도 괜찮아요.



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, Databases, 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 configuration for the 'springboot-database'. The left sidebar has the 'Database' section selected. The main content area shows the database details and the 'Security Groups' tab is active. It lists the security group 'springboot-db-security' assigned to the database. Other tabs include 'Networking', 'Logs & Events', 'Configuration', 'Monitoring', and 'Tags'.

## 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

**Protocol:** TCP

**Type:** MySQL/Aurora

### ▼ 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

Amazon RDS

springboot-database

엔드포인트 및 포트

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

포트: 3306

네트워킹

기술 영역: ap-northeast-2a

VPC: vpc-39d41150

서브넷 그룹

보안

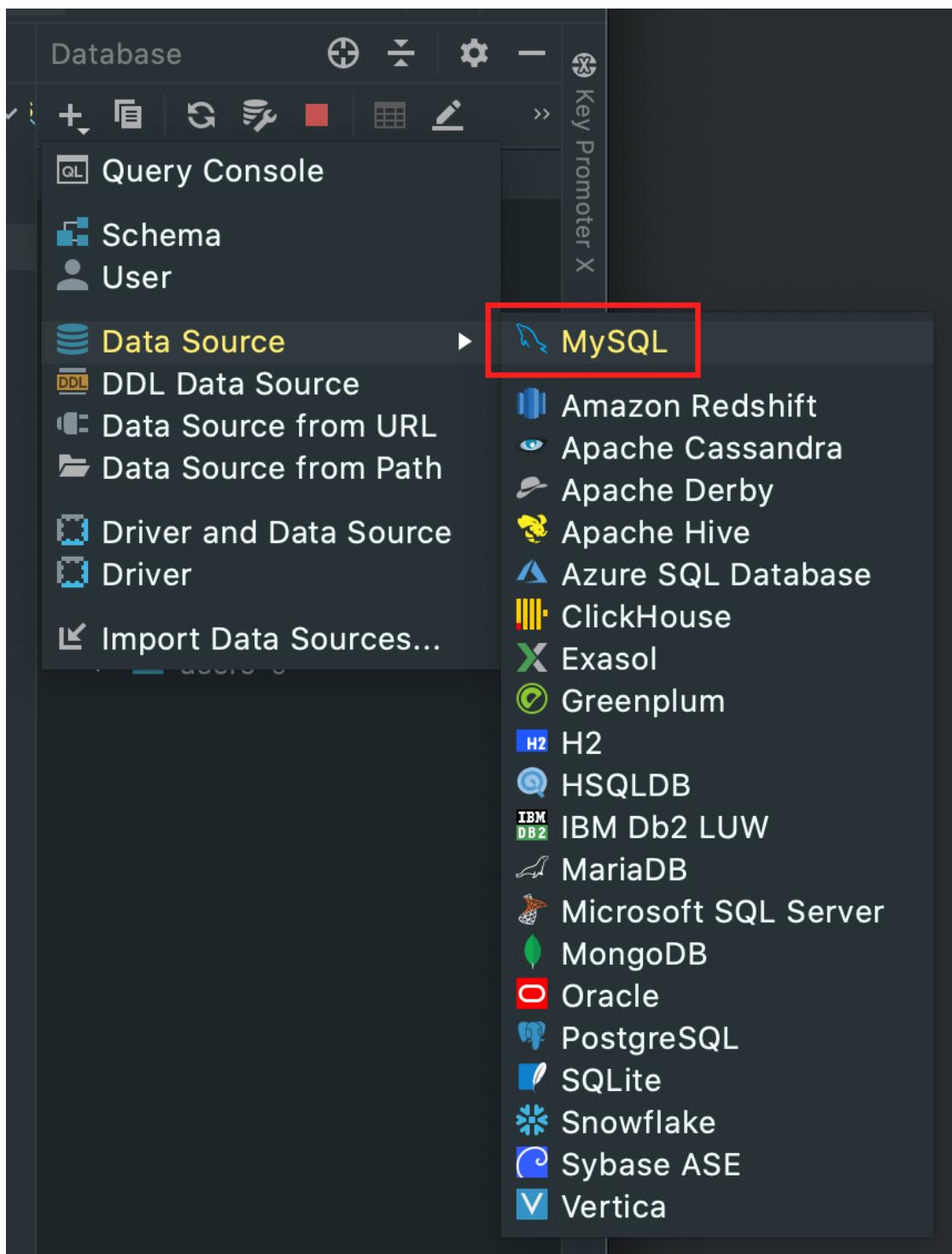
VPC 보안 그룹: **springboot-db-security (sg-0a2c9bb0bdd0f6d2) (활성)**

피블릭 액세스 가능성: 예

3. week04 프로젝트를 IntelliJ에서 열기

4. 우측 Database 탭을 클릭합니다.

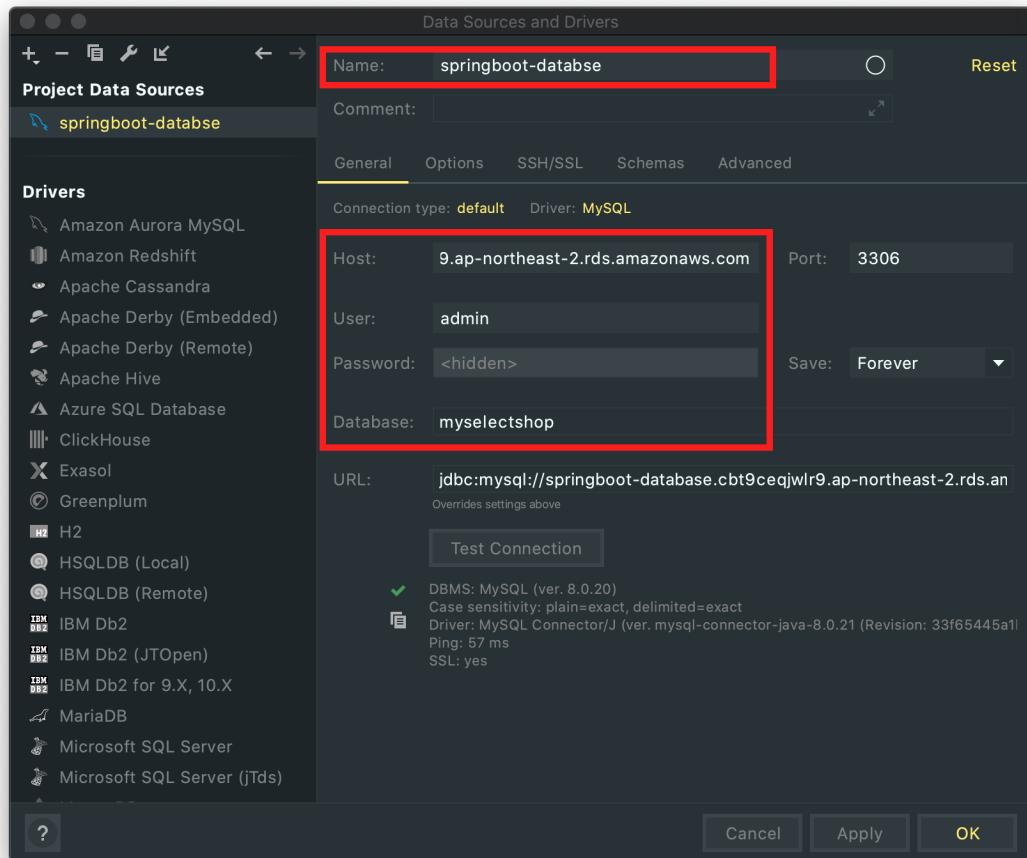
5. Data Source > MySQL 을 클릭합니다. (제일 상단이 아니라면 중간 즈음 있을 거에요!)



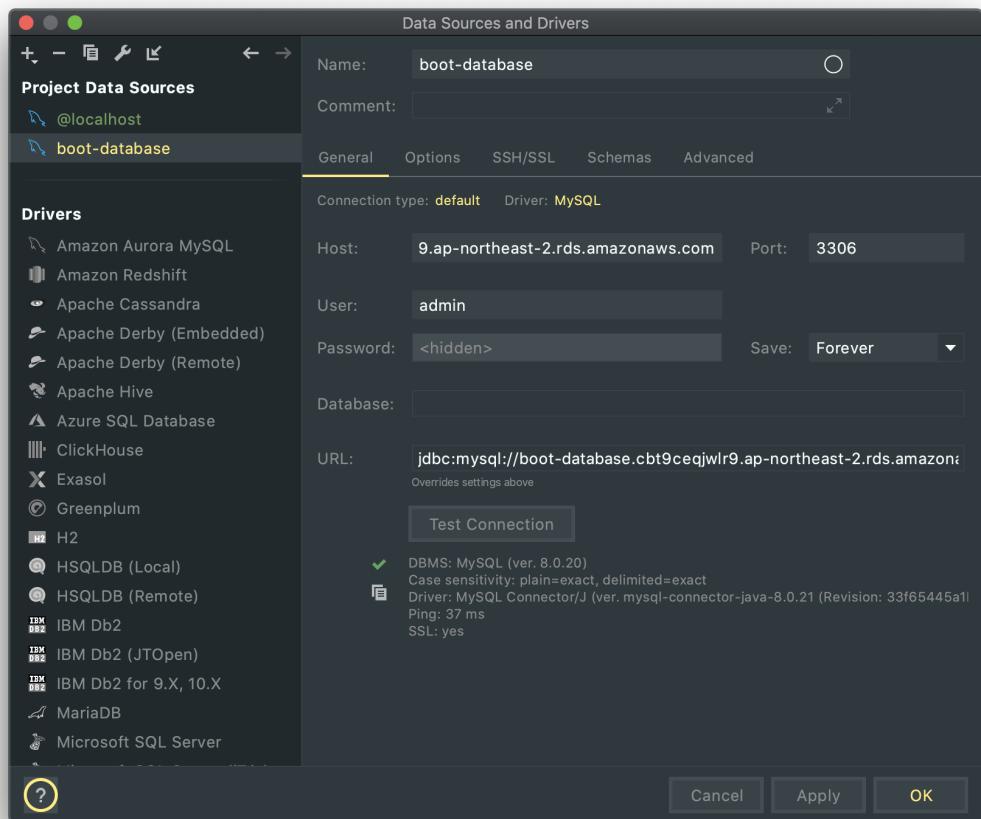
6. 다음과 같이 입력합니다.

- 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		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
<input checked="" type="checkbox"/> General purpose	<input checked="" type="checkbox"/> t2.micro	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

Free tier eligible Root Device Type: ebs Virtualization type: hvm

Instance Type

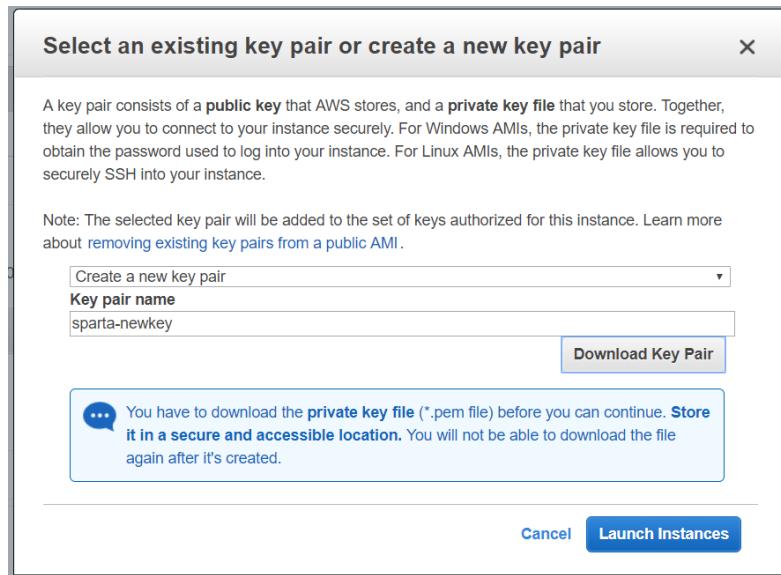
Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

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

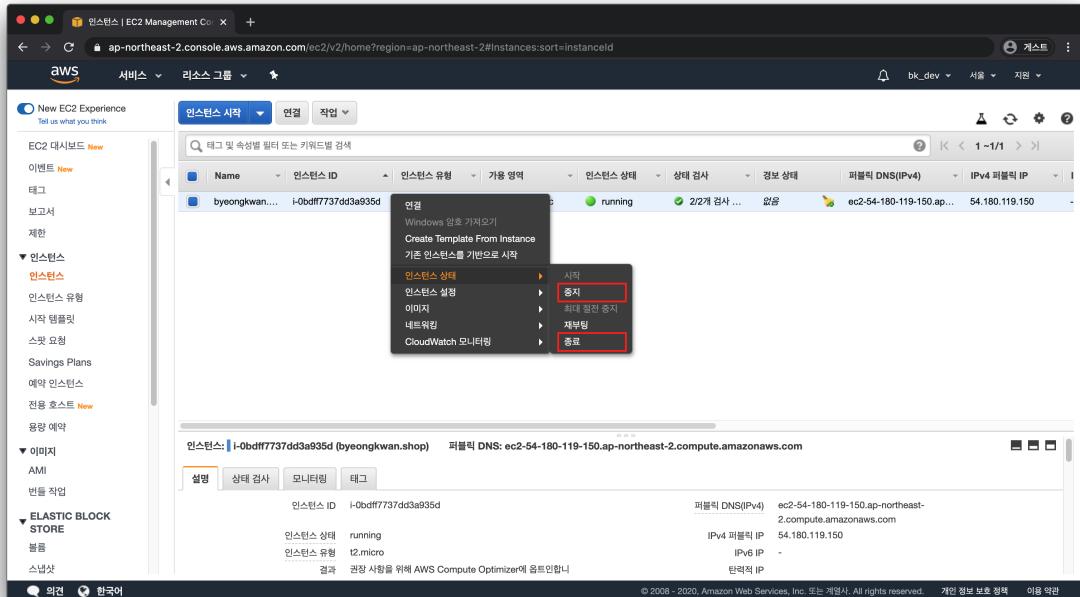


### ▼ EC2 서버 종료하는 방법 (1년 후 자동결제 방지!)



**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. A metric named 'CPU Utilization' is selected. The chart displays data over a period from March 15, 2018, at 09:00 UTC to 10:00 UTC. The utilization starts at approximately 10%, remains relatively stable until about 09:30 UTC, then rises sharply to nearly 100% by 10:00 UTC, before gradually decreasing. The Y-axis represents CPU Utilization from 0% to 100%.

## ▼ 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

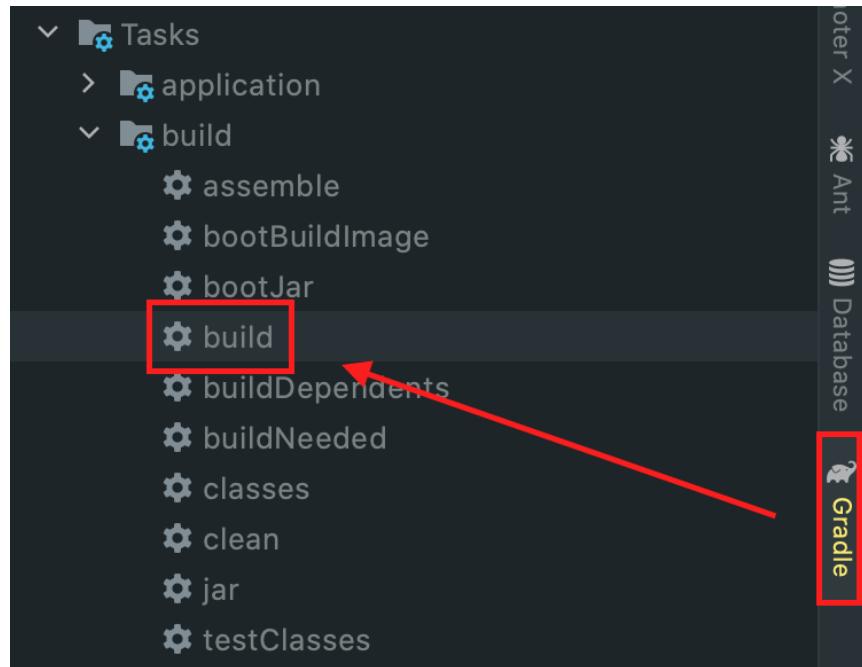
sudo [실행 할 명령어]: 명령어를 관리자 권한으로 실행한다.

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

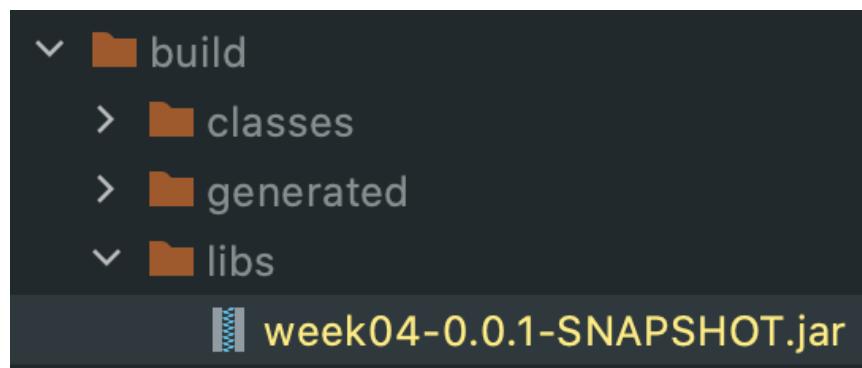
## 05. [5주차] EC2 준비하고 배포하기

▼ 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.

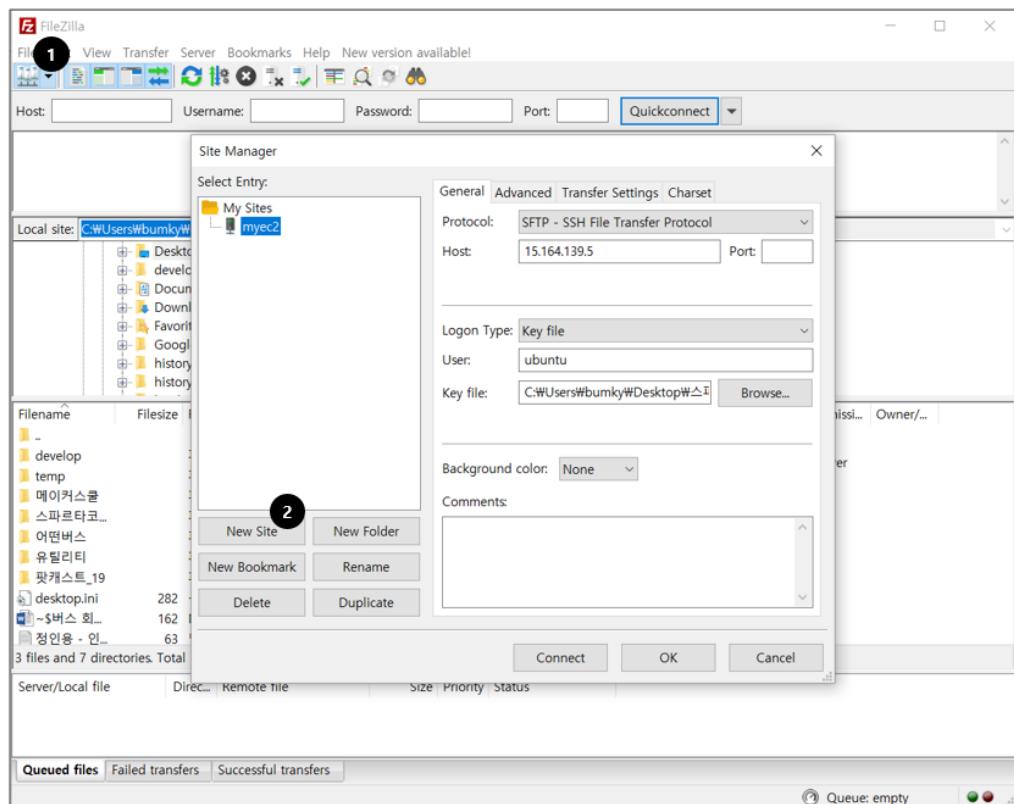


▼ OpenJDK 설치하기

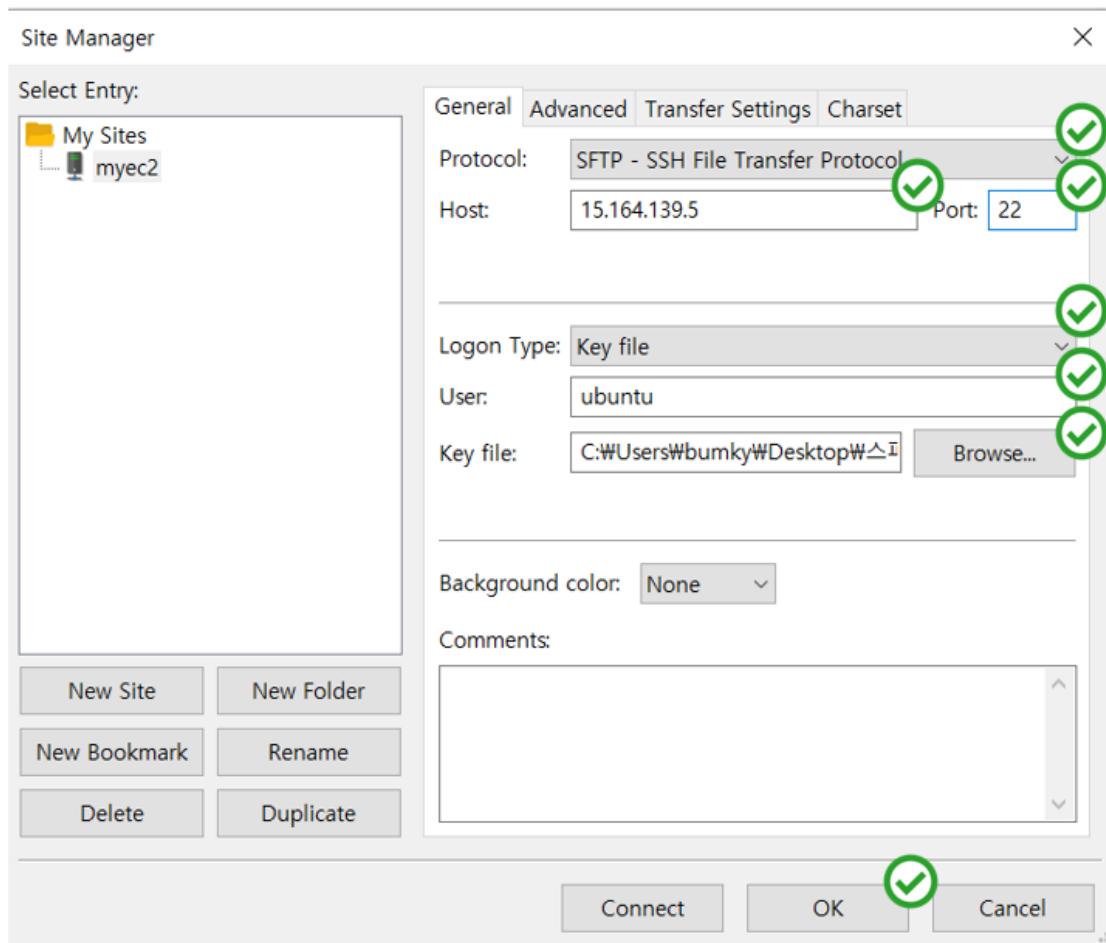
```
sudo apt-get update  
sudo apt-get install openjdk-8-jdk  
java -version
```

▼ Filezilla를 이용해서 배포 파일을 업로드하기

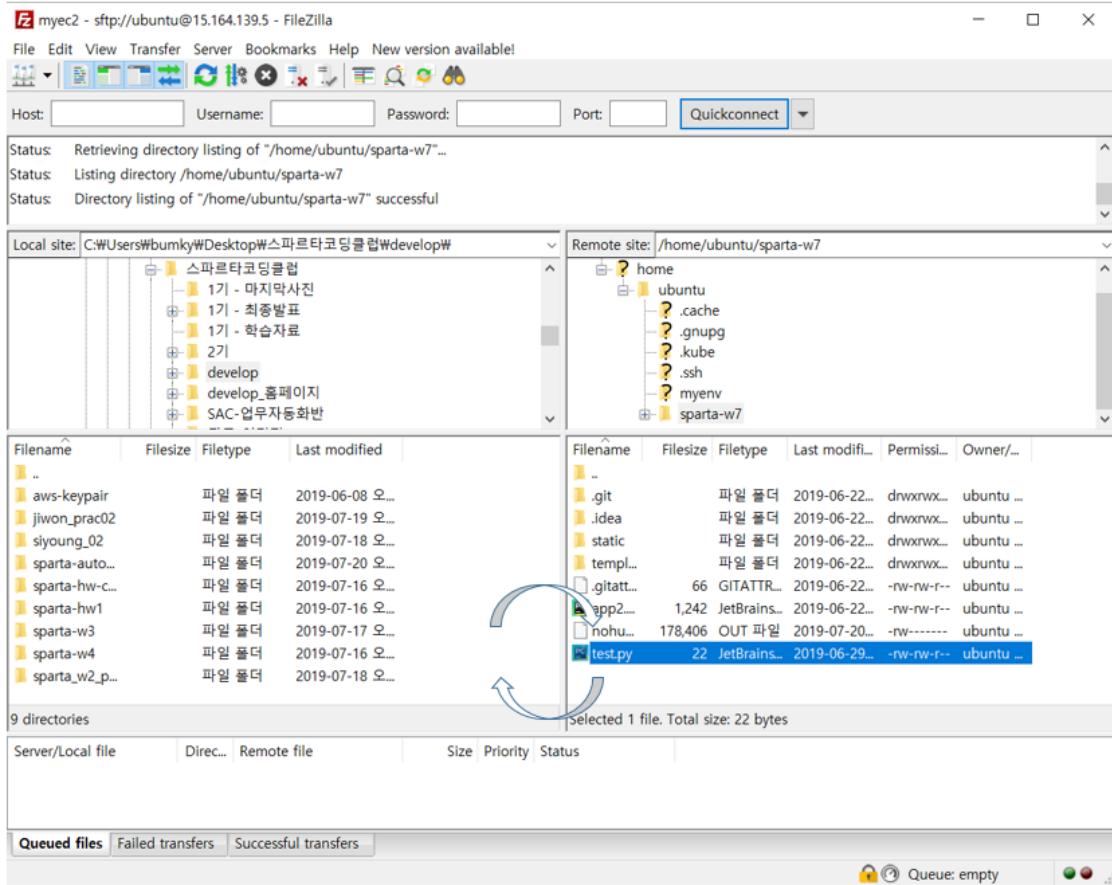
- 파일질라 실행, 다음과 같이 설정



- 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: 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. Under 'EC2', it says 'Security Groups'. The main content 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 'sg-Debf753ec1e1150d8' row. 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 one 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	-

**New rule**

User-defined TCP	TCP	8080	Anywhere	0.0.0.0/0	X
User-defined TCP	TCP	8080	Anywhere	::/0	X
SSH	TCP	22	Anywhere	0.0.0.0/0	X
User-defined TCP	TCP	80	Anywhere	0.0.0.0/0	X
User-defined TCP	TCP	80	Anywhere	::/0	X

**New rule**

△ 참고: 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.

- 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 last column contains the price '53.900원/년' (53,900 won/year) and a blue button labeled '관리툴' (Management Tools) which 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 labeled 'TOP' is located in the bottom right corner of the page.

도메인	서비스 관리	비용 결제	결제 관리	My 정보 관리
spart@example.shop		2020-05-24	2021-05-24	53.900원/년 관리툴

(도메인 연결 클릭)

전체 도메인 | 도메인 정보 변경 | DNS 정보 | 도메인 보안 | 예약 도메인 관리 | 관심 도메인

spartaexample.shop | 인증 코드

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

소유자	<a href="#">수정</a>	소유권 이전	<a href="#">수정</a>
이범규 bumkyu.lee@gmail.com 010-7127-4503		관리자	<a href="#">수정</a>
		이범규 bumkyu.lee@gmail.com 010-7127-4503	

만기일 맞춤 | 도메인 연장 | 연장 알림

도메인 카테고리 | +

고객센터 | 네임서버 | 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 승인 | 설정

안전 장금 | 설정

기과 이전 차단 | 설정

ICANN 60일 정책

DNS 정보

도메인 연결 | 설정

포워딩 | 설정

파킹 | 설정

모바일 | 설정

(DNS 설정 클릭)

DNS 설정 | 도메인 연결 | 포워딩 | 웹 파킹 | 모바일 파킹

도메인 명 | DNS 정보

<input type="checkbox"/>	도메인 명	도메인 명	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 | 이력 확인 | 엑셀 다운로드

타입	호스트	값	TTL	우선 순위	서비스	상태
A	@	IP 주소: 13.125.61.109	1800		DNS 설정	확인   삭제

+ 레코드 추가 | DNS 설정 목록 | 저장

(이렇게!)

spartaexample.shop | 이력 확인 | 엑셀 다운로드

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

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.