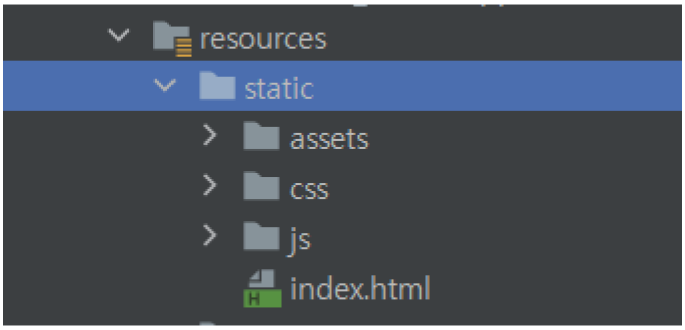
# Spring Boot 3일차 - QueryDSL 적용

## Bootstrap 적용하기

강사님이 주신 폴더 안에 내용 파일 4개를 demo3 에 resources - static 디렉토리 안에 넣는다!!



```
resources
  static
    assets
    css
    js
    index.html
```

그다음 index.html 전체를 templates - layout - layout2.html 만든다음 그 안에 전체 복붙한다.

그다음

```
<html xmlns:th="http://www.thymeleaf.org"
 xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout">
```

이것을 layout2.html 맨위에 복붙하여 교체한다.

그다음

layout2.html ㄱㄱ

```
<!-- Page content-->
<div class="container-fluid" layout:fragment="content">

</div>
```

layout:fragment="content" 이거 추가

그다음

ex1.html ㄱㄱ

```
<div layout:fragment="content">
    <h1 class="mt-4">Simple Sidebar</h1>
    <p>The starting state of the menu will appear collapsed on smaller screens,
and will appear non-collapsed on larger screens. When toggled using the button
below, the menu will change.</p>
    <p>
        Make sure to keep all page content within the
        <code>#page-content-wrapper</code>
        . The top navbar is optional, and just for demonstration. Just create an
element with the
        <code>#sidebarToggle</code>
        ID which will toggle the menu when clicked.
    </p>
   <h1>Ex1</h1>
   <h2 th:text="${'Hello1'}"></h2>
   <h2>[[${'Hello2'}]]</h2>
</div>
```

layout2.html 에 있는 부분을 위에처럼 ex1.html 안에 복붙한다.

그다음

ex1.html 맨위에

```
<html xmlns:th="http://www.thymeleaf.org"
xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
      layout:decorator="layout/layout2">
```

이 부분에서 layout2 로 바꿔준다.

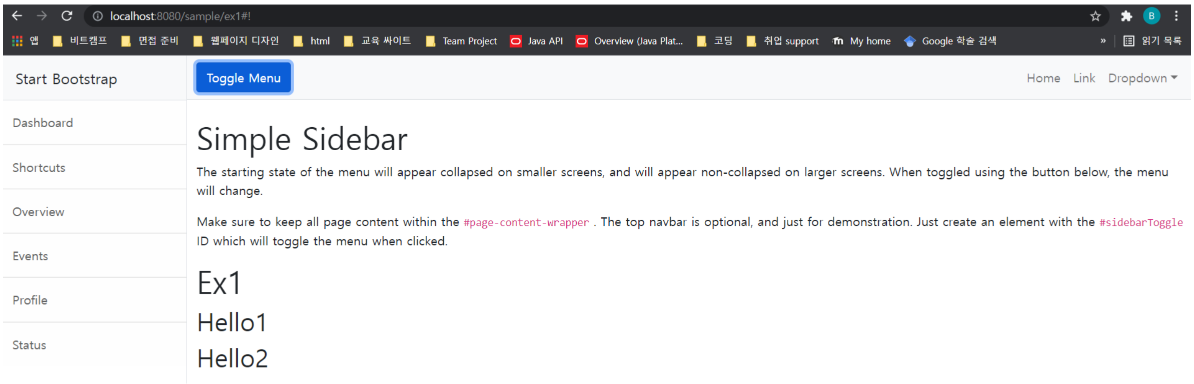
그다음

layout2.html ㄱㄱ

```
<link th:href="@{/css/styles.css}" rel="stylesheet" />
.
.
.
<script th:src="@{/js/scripts.js}"></script>
```

저렇게 두 부분에 th: @{ } 를 붙어준다.

그다음 실행하면



Simple Sidebar

The starting state of the menu will appear collapsed on smaller screens, and will appear non-collapsed on larger screens. When toggled using the button below, the menu will change.

Make sure to keep all page content within the `#page-content-wrapper` . The top navbar is optional, and just for demonstration. Just create an element with the `#sidebarToggle` ID which will toggle the menu when clicked.

Ex1

Hello1

Hello2

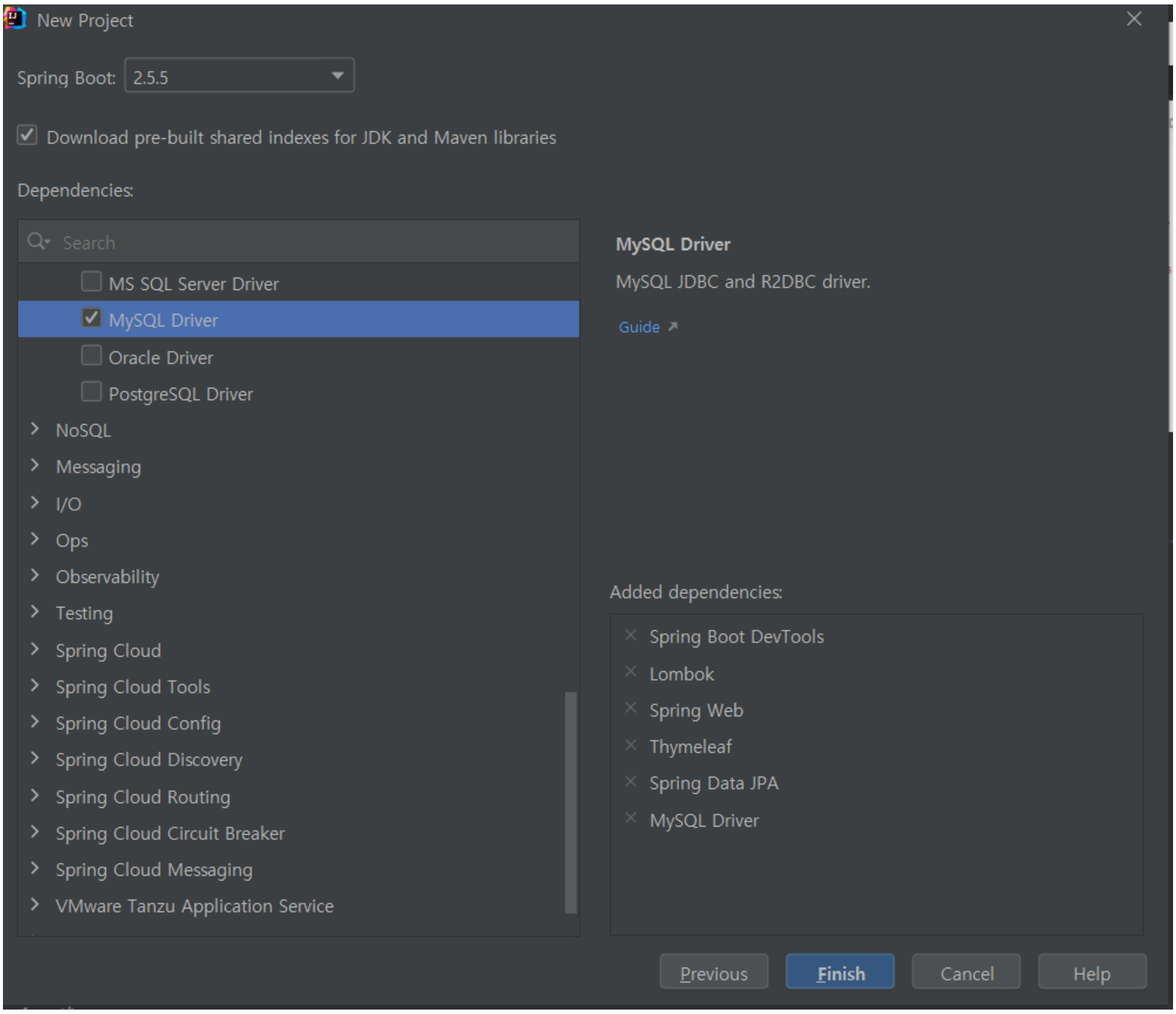
이렇게 뜬다.

# 새로운 프로젝트 만들기

## 프로젝트 sb 생성 & library 세팅

프로젝트 명 : sb

세팅



New Project

Spring Boot: 2.5.5

☑ Download pre-built shared indexes for JDK and Maven libraries

Dependencies:

**MySQL Driver**

MySQL JDBC and R2DBC driver.

Added dependencies:

- Spring Boot DevTools
- Lombok
- Spring Web
- Thymeleaf
- Spring Data JPA
- MySQL Driver

---

# 개발 시작

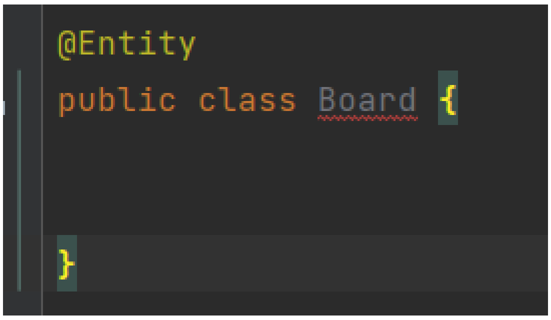
sb - entity 패키지 생성

---

P -> R -> G => 웹의 개발 구조 => PRG 구조

---

entity - Board.java 만들기



```java
@Entity
public class Board {


}
```

이렇게 에러가 난다

왜? Id 즉 primary key 가 없어서!

```java
@Entity
@Builder
@AllArgsConstructor
@NoArgsConstructor
@Getter
@ToString
public class Board {

    @Id // 이거 붙여줘야 오류가 위의 잡아짐
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long bno; // id 는 절때 수정하면 안됨

    private String title;

    private String content;

    private String writer;

    @CreationTimestamp // 자동으로 등록시간이 할당됨
    private LocalDateTime regDate;

    @UpdateTimestamp // 자동으로 수정 시간이 할당됨
    private LocalDateTime modDate;

    public void change(String title, String content) { // 하지만 제목 내용은 바뀔 수 있음.
        this.title = title;
        this.content = content;
    }

}
```

VO 는 완성

이제 dto 와 pageMaker 같은애들이 필요

---

sb - dto - BoardDTO 생성

```java
@Data
@Builder
@AllArgsConstructor
```

```
@NoArgsConstructor
public class BoardDTO {

    private Long bno;

    private String title;

    private String content;

    private String writer;

    private LocalDateTime regDate;
    private LocalDateTime modDate;
}
```

여기서는 어노테이션이 필요 없음

---

sb - config 페키지 생성

RootConfig.java 생성

```
@Configuration
public class RootConfig {
}
```

이렇게 해주기

그다음 ModelMapper build.gradle 해주기.

```
// https://mvnrepository.com/artifact/org.modelmapper/modelmapper
implementation group: 'org.modelmapper', name: 'modelmapper', version: '2.4.4'

```

---

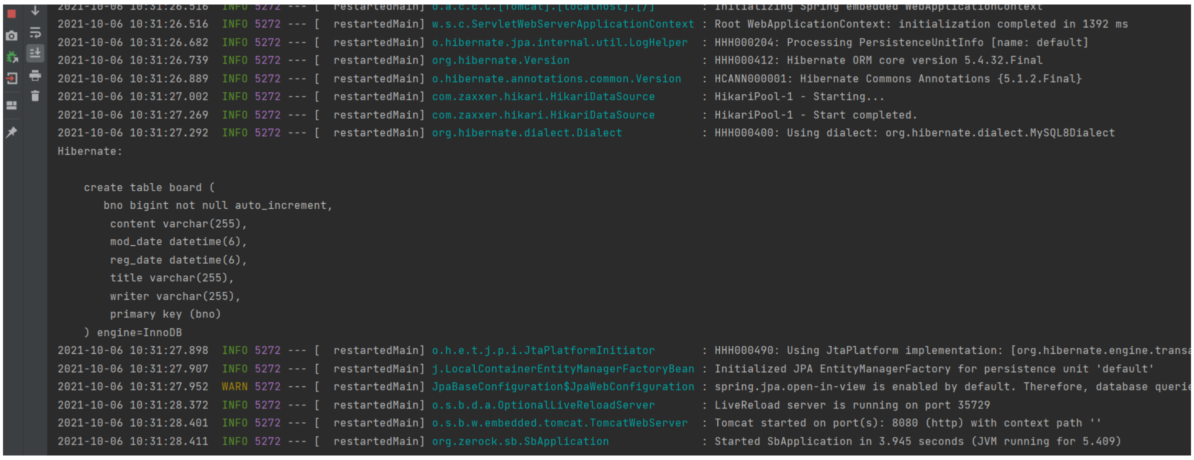
Build.gradle 세팅해주기

```
dependencies {
    implementation 'org.springframework.boot:spring-boot-starter-data-jpa'
    implementation 'org.springframework.boot:spring-boot-starter-thymeleaf'
    implementation 'org.springframework.boot:spring-boot-starter-web'
    compileOnly 'org.projectlombok:lombok'
    testCompileOnly 'org.projectlombok:lombok'
    developmentOnly 'org.springframework.boot:spring-boot-devtools'
    runtimeOnly 'mysql:mysql-connector-java'
    annotationProcessor 'org.projectlombok:lombok'
    testAnnotationProcessor 'org.projectlombok:lombok'

    testImplementation 'org.springframework.boot:spring-boot-starter-test'
    implementation group: 'org.modelmapper', name: 'modelmapper', version: '2.4.4'

    implementation group: 'nz.net.ultraq.thymeleaf', name: 'thymeleaf-layout-dialect', version: '2.5.3'
}
```

application.properties 세팅해주기

```
logging.level.org.springframework=INFO

spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.datasource.url=jdbc:mysql://localhost:3306/bootdb
spring.datasource.username=bootuser
spring.datasource.password=bootuser

spring.datasource.hikari.maximum-pool-size=3

spring.jpa.hibernate.ddl-auto=update
spring.jpa.properties.hibernate.format_sql=true
spring.jpa.show-sql=true
```

그다음



```
2021-10-06 10:31:26.516  INFO 5272 --- [  restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationContext
2021-10-06 10:31:26.516  INFO 5272 --- [  restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1392 ms
2021-10-06 10:31:26.682  INFO 5272 --- [  restartedMain] o.hibernate.jpa.internal.util.LogHelper  : HHH000204: Processing PersistenceUnitInfo [name: default]
2021-10-06 10:31:26.739  INFO 5272 --- [  restartedMain] org.hibernate.Version                    : HHH000412: Hibernate ORM core version 5.4.32.Final
2021-10-06 10:31:26.889  INFO 5272 --- [  restartedMain] o.hibernate.annotations.common.Version   : HCANN000001: Hibernate Commons Annotations {5.1.2.Final}
2021-10-06 10:31:27.002  INFO 5272 --- [  restartedMain] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Starting...
2021-10-06 10:31:27.269  INFO 5272 --- [  restartedMain] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Start completed.
2021-10-06 10:31:27.292  INFO 5272 --- [  restartedMain] org.hibernate.dialect.Dialect            : HHH000400: Using dialect: org.hibernate.dialect.MySQL8Dialect
Hibernate: 

    create table board (
       bno bigint not null auto_increment,
        content varchar(255),
        mod_date datetime(6),
        reg_date datetime(6),
        title varchar(255),
        writer varchar(255),
        primary key (bno)
    ) engine=InnoDB
2021-10-06 10:31:27.898  INFO 5272 --- [  restartedMain] o.h.e.t.j.p.i.JtaPlatformInitiator       : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transa
2021-10-06 10:31:27.907  INFO 5272 --- [  restartedMain] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2021-10-06 10:31:27.952  WARN 5272 --- [  restartedMain] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database querie
2021-10-06 10:31:28.372  INFO 5272 --- [  restartedMain] o.s.b.d.a.OptionalLiveReloadServer       : LiveReload server is running on port 35729
2021-10-06 10:31:28.401  INFO 5272 --- [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 8080 (http) with context path ''
2021-10-06 10:31:28.411  INFO 5272 --- [  restartedMain] org.zerock.sb.SbApplication              : Started SbApplication in 3.945 seconds (JVM running for 5.409)
```

이렇게 뜨면 성공

---

# 최신 버전의 세팅 - QueryDSL

## 세팅1

dsl - Domain specific Language

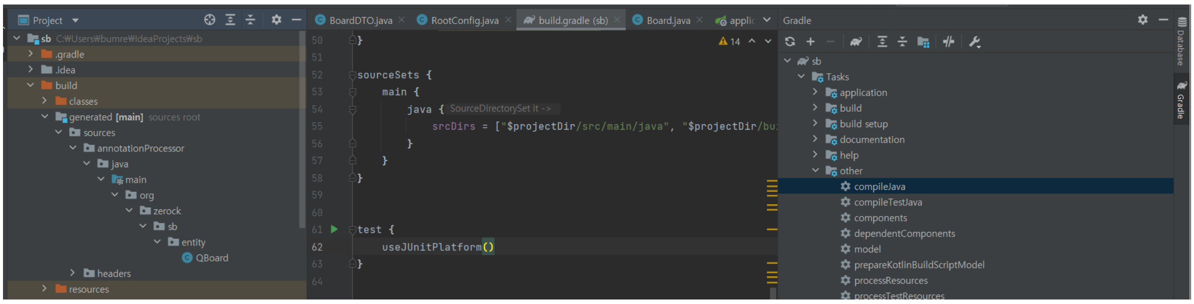
대한민국은 대부분이 queryDsl 을 사용한다.

build.gradle ㄱㄱ

```
buildscript {
    ext {
        queryDslVersion = "4.4.0"
    }
}
.
.
.

dependencies {
        .
        .
        .
        .

    // QueryDSL
    implementation "com.querydsl:querydsl-jpa:${queryDslVersion}"
    annotationProcessor(
            "javax.persistence:javax.persistence-api",
            "javax.annotation:javax.annotation-api",
            "com.querydsl:querydsl-apt:${queryDslVersion}:jpa")

}

 sourceSets {
    main {
        java {
            srcDirs = ["$projectDir/src/main/java",
"$projectDir/build/generated"]
        }
    }
}
.
.
.
```

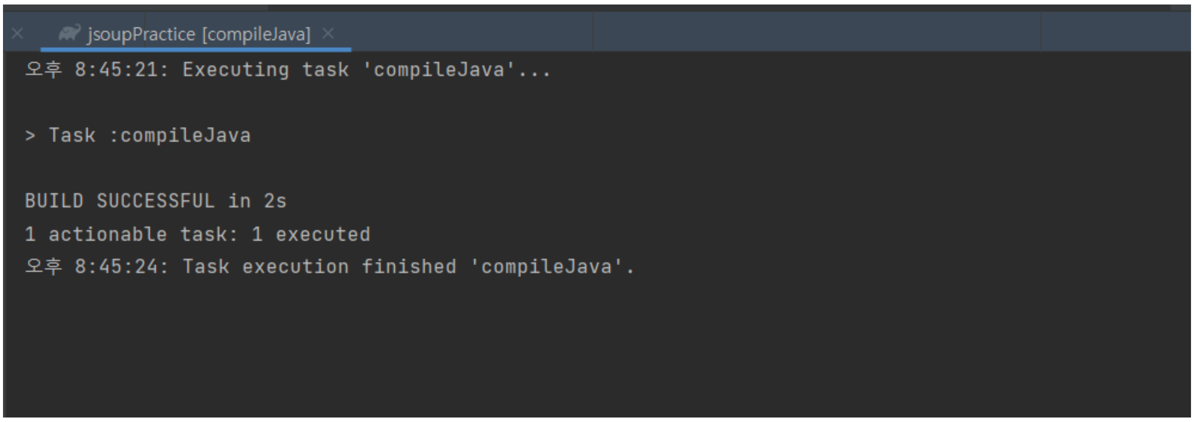
이렇게 세팅을 해주고

그다음



```
}

sourceSets {
    main {
        java { SourceDirectorySet it ->
            srcDirs = ["$projectDir/src/main/java", "$projectDir/bu
        }
    }
}


test {
    useJUnitPlatform()
}
```

맨 오른쪽 상단에 Gradle  들어가서 tasks - other - compilejava 실행.



```
오후 8:45:21: Executing task 'compileJava'...

> Task :compileJava

BUILD SUCCESSFUL in 2s
1 actionable task: 1 executed
오후 8:45:24: Task execution finished 'compileJava'.
```

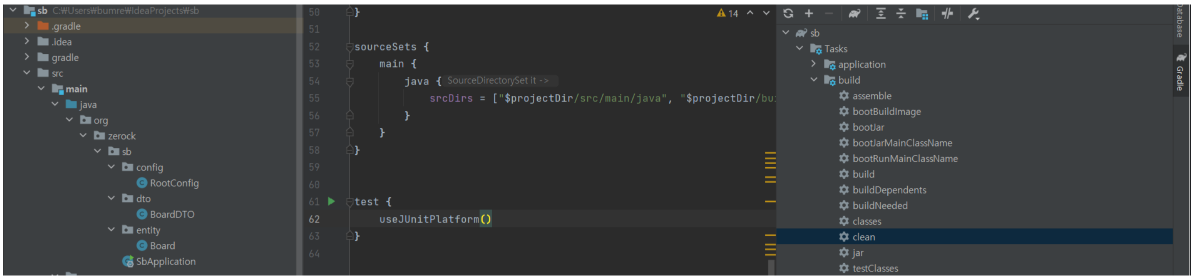
그다음 왼쪽에 build 밑에 쭉쭉 내려가다 보면 QBoard 가 생긴다. 그럼 성공.

★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★

10:56 - 10:59 분쯤에 강사님이 면접때 "queryDSL 을 사용해서 ~~~ 세팅하는게 어려웠습니다." 라는 이 면접 내용 이거 꼭 듣고 기록해두기!!!

★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★

clean 하면 build 파일이 사라진다.



```
}

sourceSets {
    main {
        java { SourceDirectorySet it ->
            srcDirs = ["$projectDir/src/main/java", "$projectDir/bu
        }
    }
}


test {
    useJUnitPlatform()
}

```

그다음 다시

other - compilejava 실행

그래서 위에처럼

build - generated - entity - QBoard 가 나오면 성공!!!!

---

## 세팅 2 - BoardRepository 생성 - JpaRepository 사용

sb - repository - BoardRepository 인터페이스 생성

그리고 JpaRepository extends 해주기

```
public interface BoardRepository extends JpaRepository<Board, Long> {
}
```

그다음

repository - search - BoardSearch 인터페이스 생성 (repository 패키지 안에 search 패키지를 만든다.)

---

네이밍 룰이 존재한다.

BoardSearch 인터페이스 무조건 이름 같고 뒤에 BoardSearch 클래스는 무조건 Impl 만 붙어야 한다.

---

BoardSearchImpl

그다음 빨간줄 뜨는데 construnt 어쩌구 해주고 그다음 지울거 지워서 밑에처럼 만들어줘야 한다.

그다음 마지막으로 BoardSearch implements 해주기

```
public class BoardSearchImpl extends QuerydslRepositorySupport implements
BoardSearch {

    public BoardSearchImpl() {
        super(Board.class);
    }
}
```

BoardRepository ㄱㄱ

```
public interface BoardRepository extends JpaRepository<Board, Long>, BoardSearch
{ // BoardSearch 추가
// extends 는 여러개 가능
}
```

---

## 흐름 정리1

세팅1 : Querydsl -> Q도메인 확인 (QBoard)

세팅2 : Querydsl 사용 -> 인터페이스 설정 -> 구현 -> 동작

**Querydsl 를 사용하기 위해서는 이렇게 두가지 세팅이 필요한것이다.**

여기 11시 이후부터 이부분 면접 얘기 찾을것 - 분쯤 또 면접 내용 꿀팁 나옴 꼭 듣자 제발 ㅠㅡㅠ

---

# Querydsl 동작하는지 확인 테스트 하기

BoardSearch ㄱㄱ

```
public interface BoardSearch {

    void search1(); // 추가
}
```

BoardSearchImpl ㄱㄱ

그다음 implement method 해주기.

```
@Log4j2
public class BoardSearchImpl extends QuerydslRepositorySupport implements
BoardSearch {

    public BoardSearchImpl() {
        super(Board.class);
    }

    @Override // 추가
    public void search1() {
        log.info("------------------search 1");
    }
}
```

test ㄱㄱ

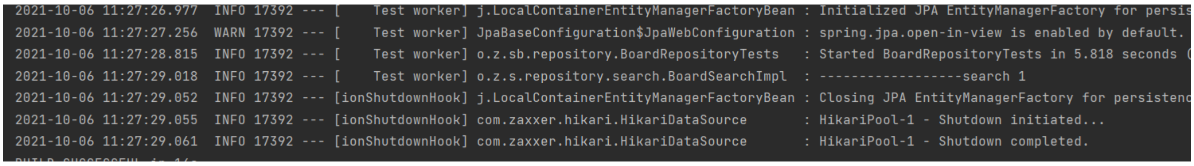
BoardRepositoryTests 생성

```
@SpringBootTest
@Log4j2
public class BoardRepositoryTests {

    @Autowired
    private BoardRepository boardRepository;

    @Test
    public void testSearch1() {

        boardRepository.search1();

    }
}
```

테스트 실행!!

그럼 밑에처럼 로그가 찍히는 것을 확인해줘야한다.

로그가 찍힘으로 repository가 잘 작동하는지 확인 할 수 있다.



```
2021-10-06 11:27:26.977  INFO 17392 --- [    Test worker] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persis
2021-10-06 11:27:27.256  WARN 17392 --- [    Test worker] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default.
2021-10-06 11:27:28.815  INFO 17392 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Started BoardRepositoryTests in 5.818 seconds (
2021-10-06 11:27:29.018  INFO 17392 --- [    Test worker] o.z.s.repository.search.BoardSearchImpl  : ------------------search 1
2021-10-06 11:27:29.052  INFO 17392 --- [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistenc
2021-10-06 11:27:29.055  INFO 17392 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Shutdown initiated...
2021-10-06 11:27:29.061  INFO 17392 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Shutdown completed.
```

# 동적 쿼리 만들기

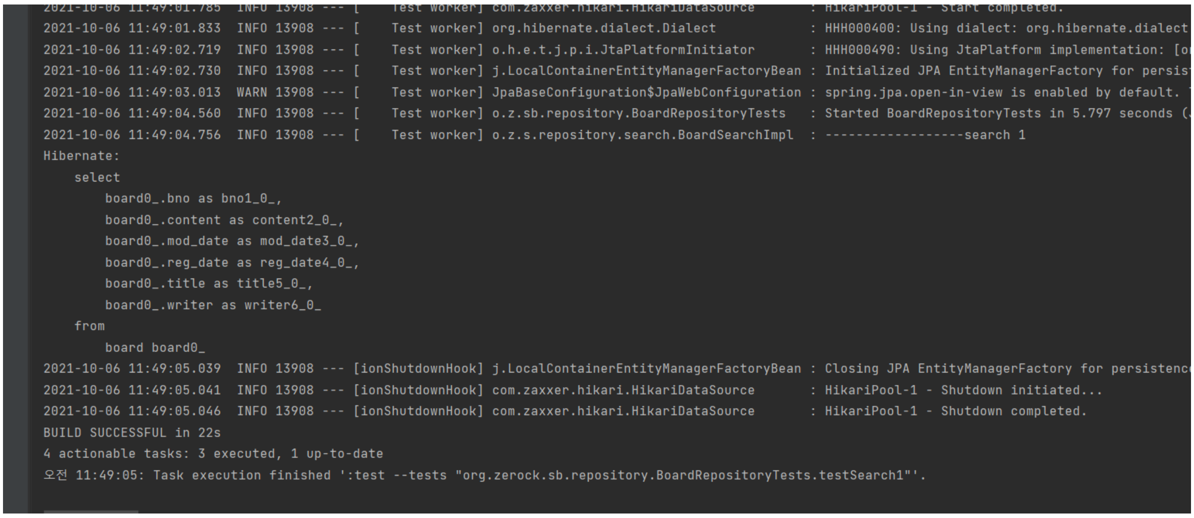
BoardSearchImpl ㄱㄱ

```
@Override
public void search1() {
    log.info("------------------search 1");
    QBoard board = QBoard.board; // 추가

    JPQLQuery<Board> jpqlQuery = from(board); // 추가

    jpqlQuery.fetch(); // select 문 에 사용 ★★★

}
```

그다음

test에 가서

BoardRepositoryTests 실행



```
2021-10-06 11:49:01.785  INFO 13908 --- [    Test worker] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Start completed.
2021-10-06 11:49:01.833  INFO 13908 --- [    Test worker] org.hibernate.dialect.Dialect            : HHH000400: Using dialect: org.hibernate.dialect
2021-10-06 11:49:02.719  INFO 13908 --- [    Test worker] o.h.e.t.j.p.i.JtaPlatformInitiator       : HHH000490: Using JtaPlatform implementation: [o
2021-10-06 11:49:02.730  INFO 13908 --- [    Test worker] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persis
2021-10-06 11:49:03.013  WARN 13908 --- [    Test worker] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default.
2021-10-06 11:49:04.560  INFO 13908 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Started BoardRepositoryTests in 5.797 seconds (
2021-10-06 11:49:04.756  INFO 13908 --- [    Test worker] o.z.s.repository.search.BoardSearchImpl  : ------------------search 1
Hibernate: 
    select
        board0_.bno as bno1_0_,
        board0_.content as content2_0_,
        board0_.mod_date as mod_date3_0_,
        board0_.reg_date as reg_date4_0_,
        board0_.title as title5_0_,
        board0_.writer as writer6_0_ 
    from
        board board0_
2021-10-06 11:49:05.039  INFO 13908 --- [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistenc
2021-10-06 11:49:05.041  INFO 13908 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Shutdown initiated...
2021-10-06 11:49:05.046  INFO 13908 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Shutdown completed.
BUILD SUCCESSFUL in 22s
4 actionable tasks: 3 executed, 1 up-to-date
오전 11:49:05: Task execution finished ':test --tests "org.zerock.sb.repository.BoardRepositoryTests.testSearch1"'.
```

성공

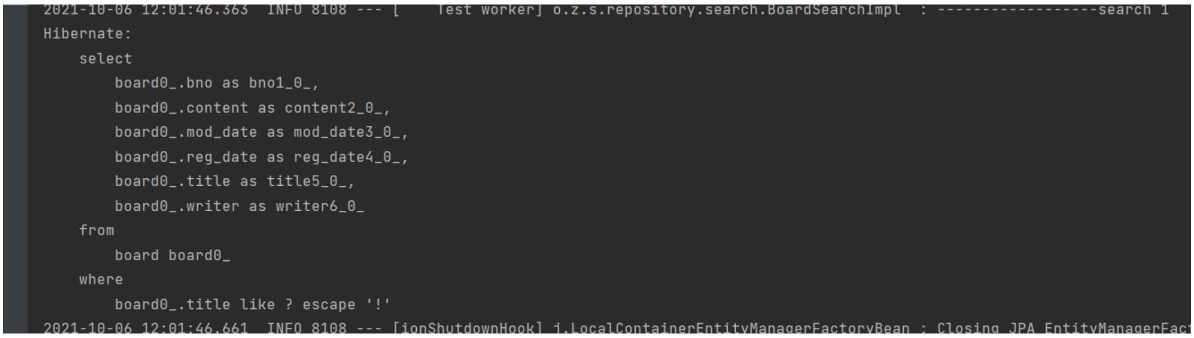
**이 fetch 라는 단어를 이용하면 실제로 실행이 된다고 한다.**

---

BoardSearchImpl ㄱㄱ

```
@Override
public void search1() {
    log.info("------------------search 1");

    QBoard board = QBoard.board;

    JPQLQuery<Board> jpqlQuery = from(board);

    char type = 'T';

    String keyword = "10";

    if (type == 'T') {
        jpqlQuery.where(board.title.contains(keyword));
    }

    jpqlQuery.fetch();
}
```

테스트 ㄱㄱ

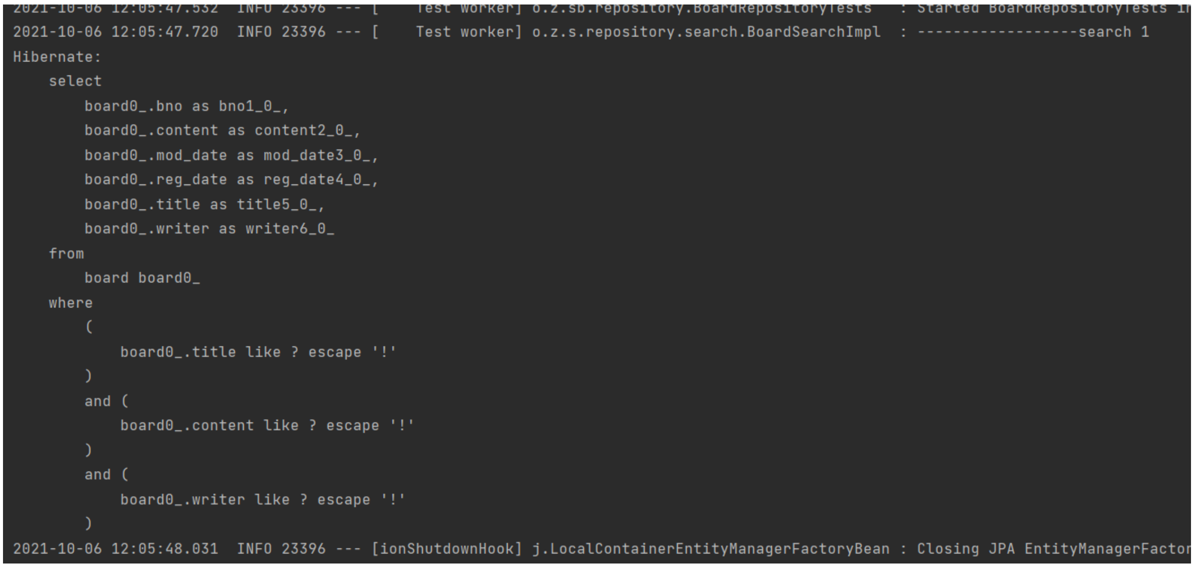


```
2021-10-06 12:01:46.363  INFO 8108 --- [    Test worker] o.z.s.repository.search.BoardSearchImpl  : ------------------search 1
Hibernate: 
    select
        board0_.bno as bno1_0_,
        board0_.content as content2_0_,
        board0_.mod_date as mod_date3_0_,
        board0_.reg_date as reg_date4_0_,
        board0_.title as title5_0_,
        board0_.writer as writer6_0_ 
    from
        board board0_ 
    where
        board0_.title like ? escape '!'
2021-10-06 12:01:46.661  INFO 8108 --- [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFact
```

BoardSearchImpl 유지

```
@Override
    public void search1() {
        log.info("------------------search 1");

        QBoard board = QBoard.board;

        JPQLQuery<Board> jpqlQuery = from(board);

        String keyword = "10";
        char[] typeArr = {'T', 'C', 'W'};

        for (char type : typeArr) {
            if (type == 'T') {
                jpqlQuery.where(board.title.contains(keyword));
            } else if (type == 'C') {
                jpqlQuery.where(board.content.contains(keyword));
            } else if (type == 'W') {
                jpqlQuery.where(board.writer.contains(keyword));
            }
        }
        jpqlQuery.fetch();
    }
```

테스트 ㄱㄱ



```
2021-10-06 12:05:47.532  INFO 23396 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Started BoardRepositoryTests in
2021-10-06 12:05:47.720  INFO 23396 --- [    Test worker] o.z.s.repository.search.BoardSearchImpl  : ------------------search 1
Hibernate: 
    select
        board0_.bno as bno1_0_,
        board0_.content as content2_0_,
        board0_.mod_date as mod_date3_0_,
        board0_.reg_date as reg_date4_0_,
        board0_.title as title5_0_,
        board0_.writer as writer6_0_ 
    from
        board board0_ 
    where
        (
            board0_.title like ? escape '!'
        ) 
        and (
            board0_.content like ? escape '!'
        ) 
        and (
            board0_.writer like ? escape '!'
        )
2021-10-06 12:05:48.031  INFO 23396 --- [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactor
```

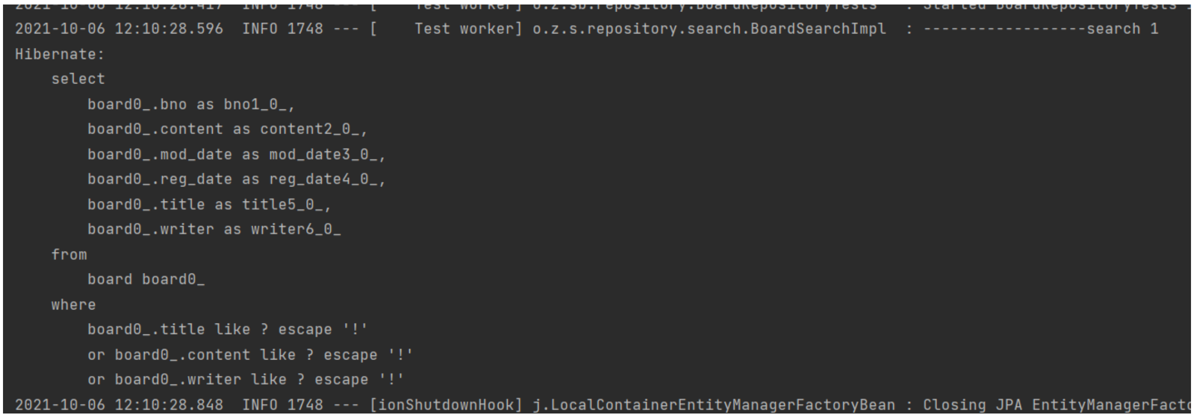
하지만 우리가 원하는것은 and 가 아니라 or 이다.

바꿔줘야 한다.

BoardSearchImpl 유지

```java
@Override
public void search1() {
    log.info("------------------search 1");

    QBoard board = QBoard.board;

    JPQLQuery<Board> jpqlQuery = from(board);

    String keyword = "10";
    char[] typeArr = {'T', 'C', 'W'};

    BooleanBuilder condition = new BooleanBuilder(); // 추가

    for (char type : typeArr) {
        if (type == 'T') {
            condition.or(board.title.contains(keyword)); // 수정
        } else if (type == 'C') {
            condition.or(board.content.contains(keyword)); // 수정
        } else if (type == 'W') {
            condition.or(board.writer.contains(keyword)); // 수정
        }
    }
    jpqlQuery.where(condition);
    jpqlQuery.fetch();
}
```

테스트 ㄱㄱ

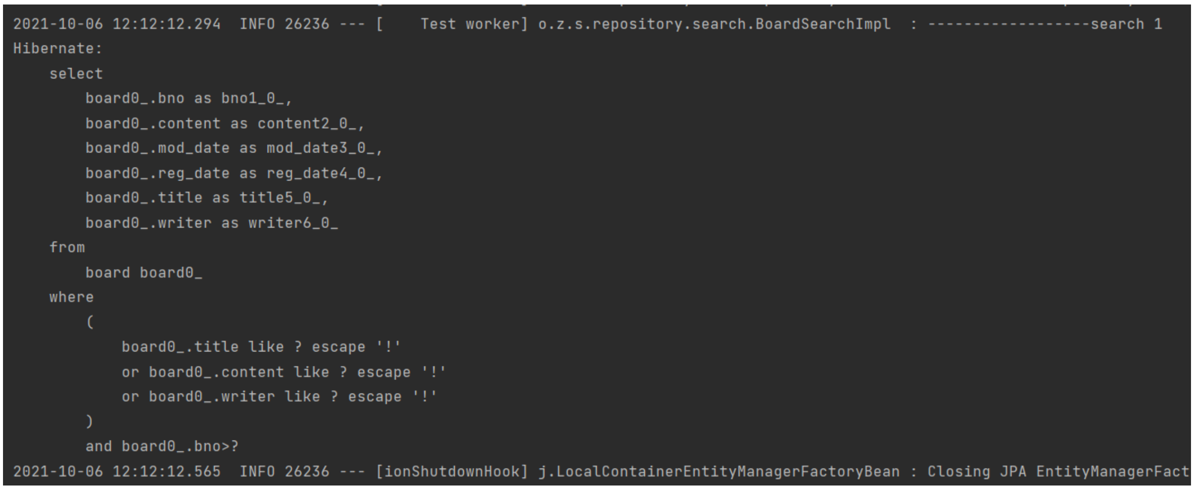


```
2021-10-06 12:10:28.417  INFO 1748 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Started BoardRepositoryTests
2021-10-06 12:10:28.596  INFO 1748 --- [    Test worker] o.z.s.repository.search.BoardSearchImpl  : ------------------search 1
Hibernate: 
    select
        board0_.bno as bno1_0_,
        board0_.content as content2_0_,
        board0_.mod_date as mod_date3_0_,
        board0_.reg_date as reg_date4_0_,
        board0_.title as title5_0_,
        board0_.writer as writer6_0_ 
    from
        board board0_ 
    where
        board0_.title like ? escape '!' 
        or board0_.content like ? escape '!' 
        or board0_.writer like ? escape '!'
2021-10-06 12:10:28.848  INFO 1748 --- [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFacto
```

BoardSearchImpl 유지

```
@Override
public void search1() {
    log.info("------------------search 1");

    QBoard board = QBoard.board;

    JPQLQuery<Board> jpqlQuery = from(board);

    String keyword = "10";
    char[] typeArr = {'T', 'C', 'W'};

    BooleanBuilder condition = new BooleanBuilder();

    for (char type : typeArr) {
        if (type == 'T') {
            condition.or(board.title.contains(keyword));
        } else if (type == 'C') {
            condition.or(board.content.contains(keyword));
        } else if (type == 'W') {
            condition.or(board.writer.contains(keyword));
        }
    }
    jpqlQuery.where(condition);
    jpqlQuery.where(board.bno.gt(0L)); // bno > 0 // 추가
    jpqlQuery.fetch();
}
```

테스트 ㄱㄱ



```
2021-10-06 12:12:12.294  INFO 26236 --- [    Test worker] o.z.s.repository.search.BoardSearchImpl  : ------------------search 1
Hibernate: 
    select
        board0_.bno as bno1_0_,
        board0_.content as content2_0_,
        board0_.mod_date as mod_date3_0_,
        board0_.reg_date as reg_date4_0_,
        board0_.title as title5_0_,
        board0_.writer as writer6_0_ 
    from
        board board0_ 
    where
        (
            board0_.title like ? escape '!' 
            or board0_.content like ? escape '!' 
            or board0_.writer like ? escape '!'
        ) 
        and board0_.bno>?
2021-10-06 12:12:12.565  INFO 26236 --- [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFact
```

하지만 위의 쿼리에는 문제가 하나 있다.

바로 페이징 처리와 order by 처리이다.

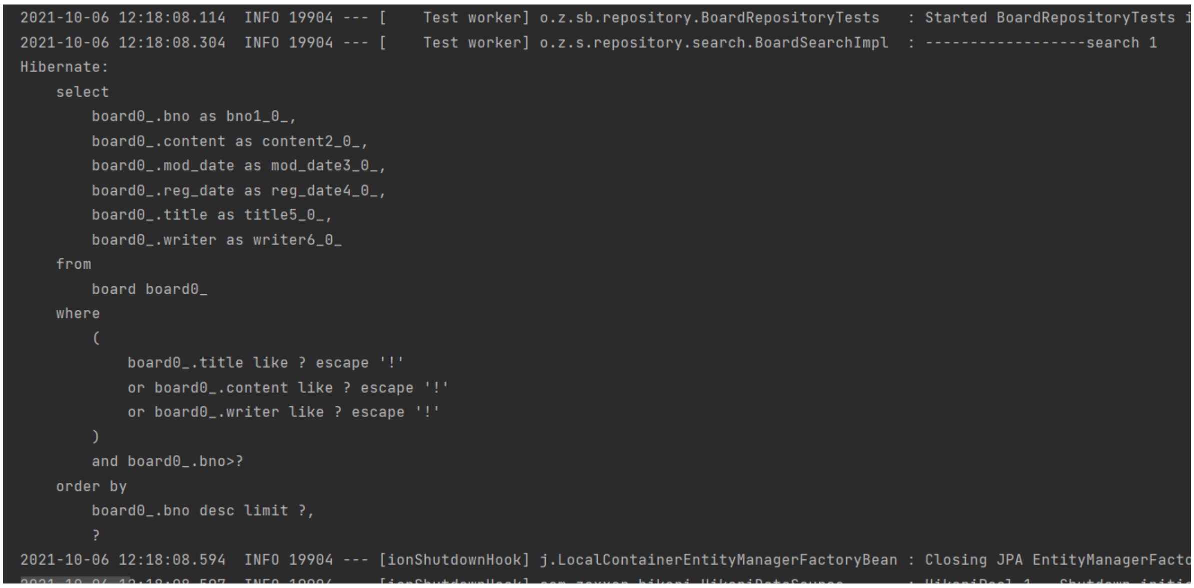
---

## Paging 처리 및 order by 적용하기

BoardSearchImpl ㄱㄱ

```java
@Override
public void search1() {
    log.info("------------------search 1");

    QBoard board = QBoard.board;

    JPQLQuery<Board> jpqlQuery = from(board);

    Pageable pageable = PageRequest.of(1, 10, Sort.by("bno").descending()); // 추가

    String keyword = "10";
    char[] typeArr = {'T', 'C', 'W'};

    BooleanBuilder condition = new BooleanBuilder();

    for (char type : typeArr) {
        if (type == 'T') {
            condition.or(board.title.contains(keyword));
        } else if (type == 'C') {
            condition.or(board.content.contains(keyword));
        } else if (type == 'W') {
            condition.or(board.writer.contains(keyword));
        }
    }
    jpqlQuery.where(condition);
    jpqlQuery.where(board.bno.gt(0L)); // bno > 0

    JPQLQuery<Board> pagingQuery =  this.getQuerydsl().applyPagination(pageable, jpqlQuery); // 추가
    pagingQuery.fetch(); // 수정
}
```

테스트 ㄱㄱ



```
2021-10-06 12:18:08.114  INFO 19904 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Started BoardRepositoryTests i
2021-10-06 12:18:08.304  INFO 19904 --- [    Test worker] o.z.s.repository.search.BoardSearchImpl  : ------------------search 1
Hibernate: 
    select
        board0_.bno as bno1_0_,
        board0_.content as content2_0_,
        board0_.mod_date as mod_date3_0_,
        board0_.reg_date as reg_date4_0_,
        board0_.title as title5_0_,
        board0_.writer as writer6_0_ 
    from
        board board0_ 
    where
        (
            board0_.title like ? escape '!' 
            or board0_.content like ? escape '!' 
            or board0_.writer like ? escape '!'
        ) 
        and board0_.bno>? 
    order by
        board0_.bno desc limit ?,
        ?
2021-10-06 12:18:08.594  INFO 19904 --- [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFacto
2021-10-06 12:18:08.597  INFO 19904 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Shutdown initia
```

order by 와 limit 이 붙는 것을 확인 할 수 있다.

---

하지만 또다시 문제가 하나 있다

페이징 처리 결과물이 있어야 한다.

코드를 슬쩍 바꿀것이다.

★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★

오늘것들은 그냥 다 외워야 한다. 오늘것들은 그냥 이렇게 사용을 한다!!!

무작정 외워야 한다!!!!!!!!!!!!!

★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★

## 서브 쿼리 만들기

BoardSearch ㄱㄱ

```
import org.springframework.data.domain.Pageable; // 이걸로 import 하기

public interface BoardSearch {

    void search1(char[] typeArr, String keyword, Pageable pageable);
}
```

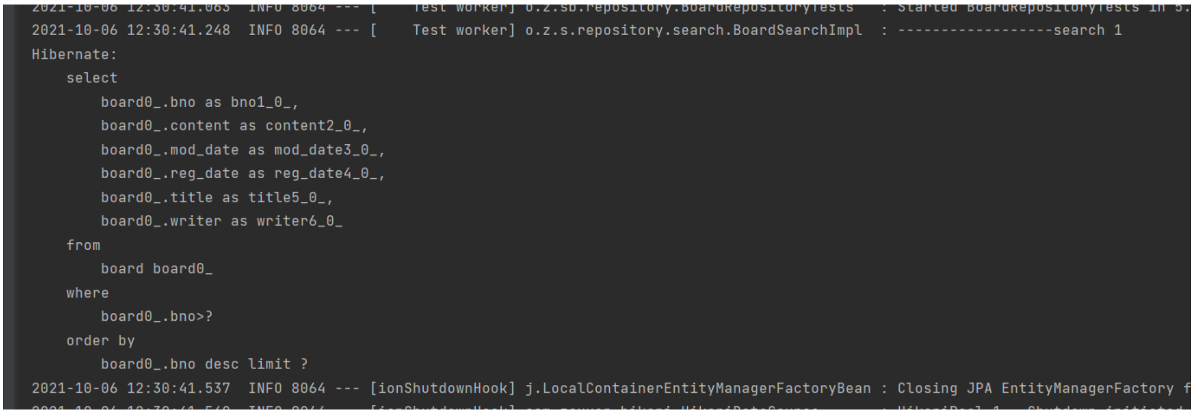
BoardSearchImpl ㄱㄱ

```java
@Override
public void search1(char[] typeArr, String keyword, Pageable pageable) {
    log.info("------------------search 1");

    QBoard board = QBoard.board;

    JPQLQuery<Board> jpqlQuery = from(board);

    // 이부분 삭제
    // Pageable pageable = PageRequest.of(1, 10, Sort.by("bno").descending());
    // String keyword = "10";
    // char[] typeArr = {'T', 'C', 'W'};

    // 검색 조건이 없다면
    if (typeArr != null && typeArr.length > 0) {

        BooleanBuilder condition = new BooleanBuilder();

        for (char type : typeArr) {
            if (type == 'T') {
                condition.or(board.title.contains(keyword));
            } else if (type == 'C') {
                condition.or(board.content.contains(keyword));
            } else if (type == 'W') {
                condition.or(board.writer.contains(keyword));
            }
        }
        jpqlQuery.where(condition);
    }


    jpqlQuery.where(board.bno.gt(0L)); // bno > 0

    JPQLQuery<Board> pagingQuery =  this.getQuerydsl().applyPagination(pageable,
jpqlQuery);

    pagingQuery.fetch();
}
```

테스트 ㄱㄱ

BoardRepositoryTests ㄱㄱ

```java
@SpringBootTest
@Log4j2
public class BoardRepositoryTests {

    @Autowired
    private BoardRepository boardRepository;

    @Test
    public void testSearch1() {

        char[] typeArr = null; // 추가
        String keyword = null; // 추가
        Pageable pageable = PageRequest.of(0,10, Sort.by("bno").descending());
// 추가

        boardRepository.search1(typeArr, keyword, pageable); // 추가
    }
}
```



```
2021-10-06 12:30:41.063  INFO 8064 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Started BoardRepositoryTests in 5.
2021-10-06 12:30:41.248  INFO 8064 --- [    Test worker] o.z.s.repository.search.BoardSearchImpl  : ------------------search 1
Hibernate: 
    select
        board0_.bno as bno1_0_,
        board0_.content as content2_0_,
        board0_.mod_date as mod_date3_0_,
        board0_.reg_date as reg_date4_0_,
        board0_.title as title5_0_,
        board0_.writer as writer6_0_ 
    from
        board board0_ 
    where
        board0_.bno>? 
    order by
        board0_.bno desc limit ?
2021-10-06 12:30:41.537  INFO 8064 --- [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory f
```

12: 30 면접 어쩌구

BoardSearch ㄱㄱ

```
public interface BoardSearch {

    Page<Board> search1(char[] typeArr, String keyword, Pageable pageable);
    // void -> Page<Board> 로 변환
}
```

BoardSearchImpl ㄱㄱ

```
@Override
public Page<Board> search1(char[] typeArr, String keyword, Pageable pageable) {
       // void -> Page<Board> 로 변환
    .
    .
    .
    return null;
}
```

그다음

마지막으로

```
@Override
public Page<Board> search1(char[] typeArr, String keyword, Pageable pageable) {
    log.info("-------------------search 1");

    QBoard board = QBoard.board;

    JPQLQuery<Board> jpqlQuery = from(board);

    // 이부분 삭제
    // Pageable pageable = PageRequest.of(1, 10, Sort.by("bno").descending());
    // String keyword = "10";
    // char[] typeArr = {'T', 'C', 'W'};

    // 검색 조건이 없다면
    if (typeArr != null && typeArr.length > 0) {

        BooleanBuilder condition = new BooleanBuilder();

        for (char type : typeArr) {
            if (type == 'T') {
                condition.or(board.title.contains(keyword));
            } else if (type == 'C') {
                condition.or(board.content.contains(keyword));
            } else if (type == 'W') {
                condition.or(board.writer.contains(keyword));
            }
        }
        jpqlQuery.where(condition);
    }

    jpqlQuery.where(board.bno.gt(0L)); // bno > 0

    JPQLQuery<Board> pagingQuery =  this.getQuerydsl().applyPagination(pageable, jpqlQuery);

    List<Board> boardList = pagingQuery.fetch(); // 추가 ★★★
    long totalCount = pagingQuery.fetchCount(); // 추가 ★★★

    return new PageImpl<>(boardList, pageable, totalCount); // 추가 ★★★
}
```

그다음

BoardRepositoryTests ㄱㄱ

```
@SpringBootTest
@Log4j2
public class BoardRepositoryTests {

    @Autowired
    private BoardRepository boardRepository;

    @Test
    public void testSearch1() {

        char[] typeArr = null;
        String keyword = null;
        Pageable pageable = PageRequest.of(0,10, Sort.by("bno").descending());
        Page<Board> result = boardRepository.search1(typeArr, keyword,
pageable);

        result.get().forEach(board -> {
            log.info(board);
        });
    }
}
```

그다음 db 에 데이터를 넣어야 한다.

MySQL db 연결시킨다음 insert 한 20~30 번 해주기

insert는 그냥 아무거나 적어서 똑같은거 20~30 번 해주기

bootuser

bootuser

bootdb

원하면 다른 db 사용 가능

---

# Model

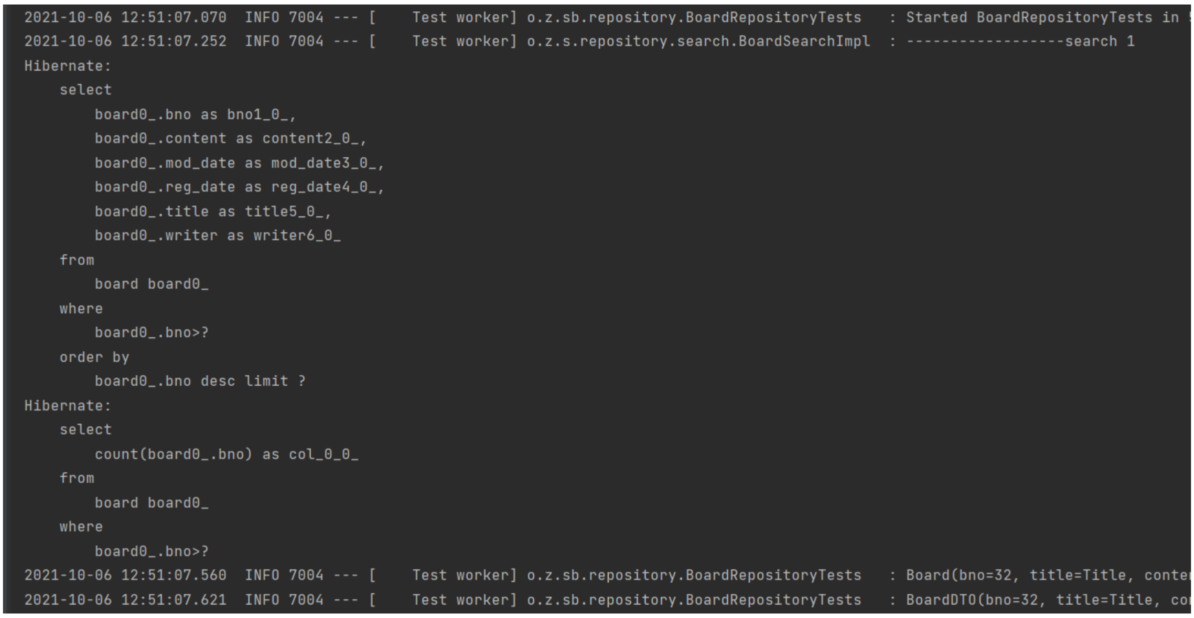
config - RootConfig.java ㄱㄱ

```
@Configuration
public class RootConfig {

    @Bean //추가
    public ModelMapper modelMapper() {
        ModelMapper modelMapper = new ModelMapper();
        modelMapper.getConfiguration()
                .setFieldMatchingEnabled(true)

.setFieldAccessLevel(org.modelmapper.config.Configuration.AccessLevel.PRIVATE)
                .setMatchingStrategy(MatchingStrategies.LOOSE);

        return modelMapper; //추가
    }
}
```

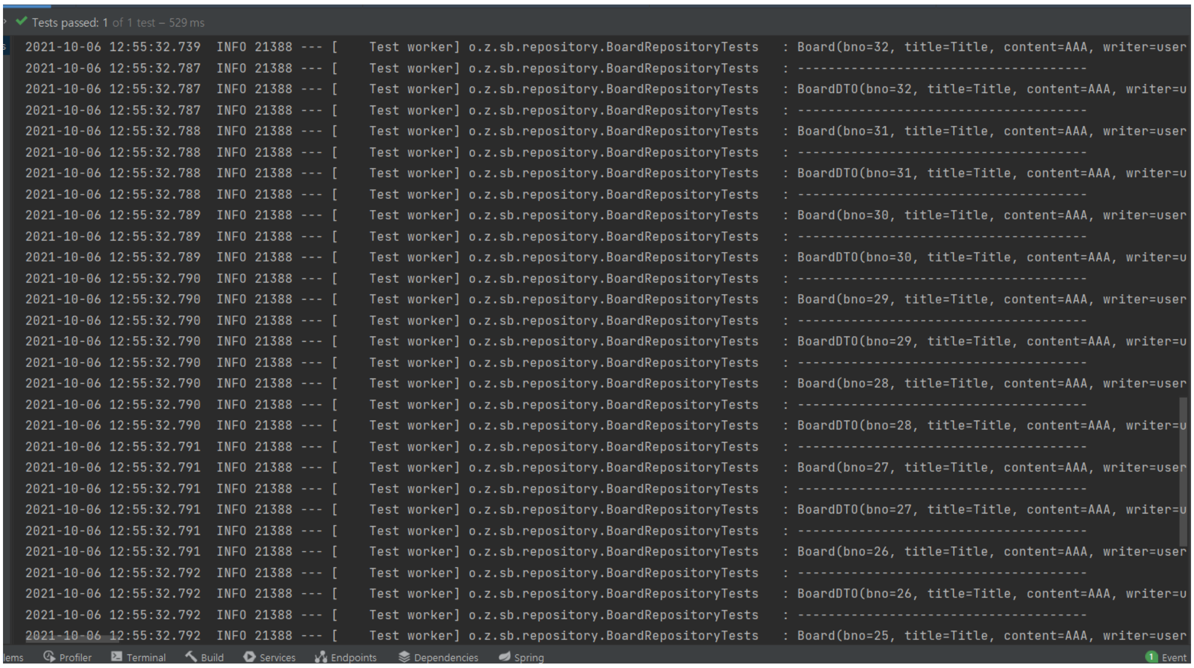
BoardRepositoryTests ㄱㄱ

```
@SpringBootTest
@Log4j2
public class BoardRepositoryTests {

    @Autowired
    private BoardRepository boardRepository;
    @Autowired
    private ModelMapper modelMapper; // 추가 ★★★

    @Test
    public void testSearch1() {

        char[] typeArr = null;
        String keyword = null;
        Pageable pageable = PageRequest.of(0,10, Sort.by("bno").descending());

        Page<Board> result = boardRepository.search1(typeArr, keyword,
pageable);

        result.get().forEach(board -> {
            log.info(board);

            BoardDTO boardDTO = modelMapper.map(board, BoardDTO.class); //추가

            log.info(boardDTO);
        });
    }
}
```

테스트 ㄱㄱ



```
2021-10-06 12:51:07.070  INFO 7004 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Started BoardRepositoryTests in
2021-10-06 12:51:07.252  INFO 7004 --- [    Test worker] o.z.s.repository.search.BoardSearchImpl  : ------------------search 1
Hibernate: 
    select
        board0_.bno as bno1_0_,
        board0_.content as content2_0_,
        board0_.mod_date as mod_date3_0_,
        board0_.reg_date as reg_date4_0_,
        board0_.title as title5_0_,
        board0_.writer as writer6_0_ 
    from
        board board0_ 
    where
        board0_.bno>? 
    order by
        board0_.bno desc limit ?
Hibernate: 
    select
        count(board0_.bno) as col_0_0_ 
    from
        board board0_ 
    where
        board0_.bno>?
2021-10-06 12:51:07.560  INFO 7004 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Board(bno=32, title=Title, conte
2021-10-06 12:51:07.621  INFO 7004 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : BoardDTO(bno=32, title=Title, co
```



```
2021-10-06 12:55:32.739  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Board(bno=32, title=Title, content=AAA, writer=user
2021-10-06 12:55:32.787  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.787  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : BoardDTO(bno=32, title=Title, content=AAA, writer=u
2021-10-06 12:55:32.787  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.788  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Board(bno=31, title=Title, content=AAA, writer=user
2021-10-06 12:55:32.788  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.788  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : BoardDTO(bno=31, title=Title, content=AAA, writer=u
2021-10-06 12:55:32.788  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.789  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Board(bno=30, title=Title, content=AAA, writer=user
2021-10-06 12:55:32.789  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.789  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : BoardDTO(bno=30, title=Title, content=AAA, writer=u
2021-10-06 12:55:32.790  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.790  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Board(bno=29, title=Title, content=AAA, writer=user
2021-10-06 12:55:32.790  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.790  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : BoardDTO(bno=29, title=Title, content=AAA, writer=u
2021-10-06 12:55:32.790  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.790  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Board(bno=28, title=Title, content=AAA, writer=user
2021-10-06 12:55:32.790  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.790  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : BoardDTO(bno=28, title=Title, content=AAA, writer=u
2021-10-06 12:55:32.791  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.791  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Board(bno=27, title=Title, content=AAA, writer=user
2021-10-06 12:55:32.791  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.791  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : BoardDTO(bno=27, title=Title, content=AAA, writer=u
2021-10-06 12:55:32.791  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.791  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Board(bno=26, title=Title, content=AAA, writer=user
2021-10-06 12:55:32.792  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.792  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : BoardDTO(bno=26, title=Title, content=AAA, writer=u
2021-10-06 12:55:32.792  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : --------------------------------------
2021-10-06 12:55:32.792  INFO 21388 --- [    Test worker] o.z.sb.repository.BoardRepositoryTests   : Board(bno=25, title=Title, content=AAA, writer=user
```

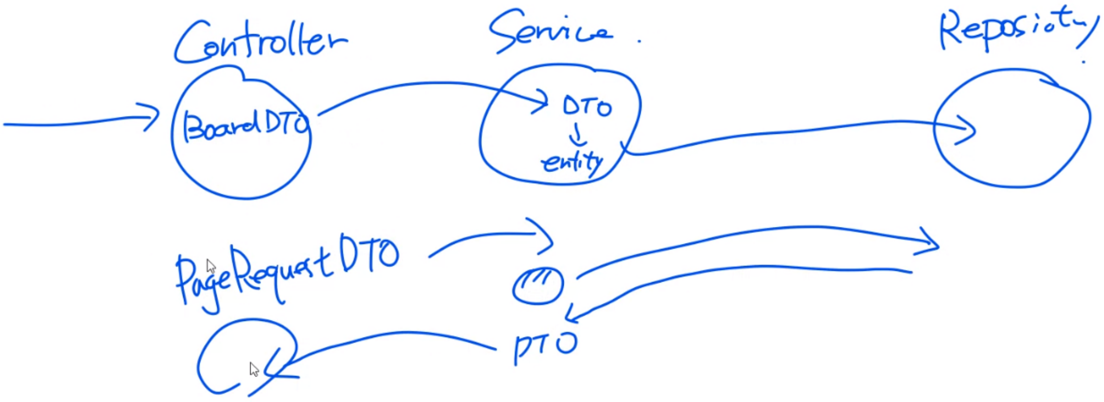
이렇게 테스타 결과가 나오면 성공

---

## 흐름 정리2

CRUD, 페이징

QueryDSL 검색, DTO 변환


Controller
Service .
Reposiotry
BoardDTO
DTO
entity
PageRequest DTO
pTO