

스프링 부트 3 백엔드 개발자 되기 (2판)

JPA+OAuth2+JWT+AWS와 배우는 스프링 부트 3
Java 백엔드 입문자를 위한 풀 패키지

06 블로그 기획하고 API 만들기

프로젝트 소개

난이도	★★☆☆
이름	블로그 API 만들기
프로젝트 폴더	Springboot-developer/chapter5
미션	블로그 글 CRUD API 구현
기능	<ul style="list-style-type: none">• 블로그 글 작성• 블로그 글 조회 (단건 조회/전체 목록 조회)• 블로그 글 삭제• 블로그 글 수정
활용 기술	<ul style="list-style-type: none">• 스프링 부트• 스프링 데이터 JPA• 롬복(Lombok)• H2

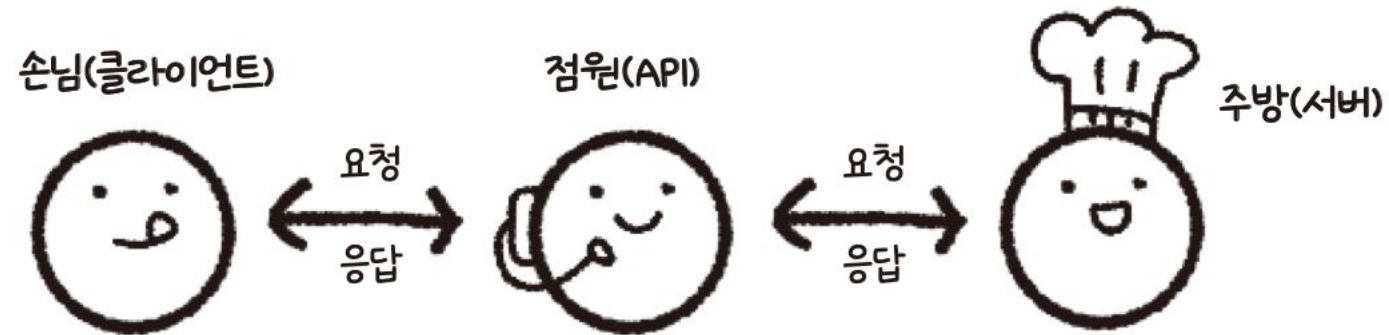
프로젝트 소개(cont.)



6.1 사전 지식 : API와 REST API

- API

- 손님과 주방을 연결해 주는 점원의 역할




6.1 사전 지식 : API와 REST API(cont.)

- REST API

- REST = Representational State Transfer
- 자원을 이름으로 구분하여 상태를 주고받는 API 방식
- API 설계 방식의 일종


- REST API 사용 방법

- URL에는 동사를 쓰지 말고, 자원을 표시해야 함
 - /students/1 → 올바른 사용
 - /get-student?student_id=1 → 올바르지 않은 사용
- 동사는 HTTP 메서드로 표현해야 함

설명	적합한 HTTP 메서드와 URL
id가 1인 블로그 글을 조회하는 API	GET /articles/1
블로그 글을 추가하는 API	POST /articles 
블로그 글을 수정하는 API	PUT /articles/1
블로그 글을 삭제하는 API	DELETE /articles/1

6.2 블로그 개발을 위한 엔티티 구성하기

- 프로젝트 준비하기
 - 기존 실습 파일 모두 삭제
 - application.yml 복사
 - build.gradle 새로 작성



삭제할 파일		
· Member	· data.sql	· MemberService
· MemberRepository	· TestControllerTest	
· TestController	· JUnitCycleTest	
· TestService	· JUnitTest	

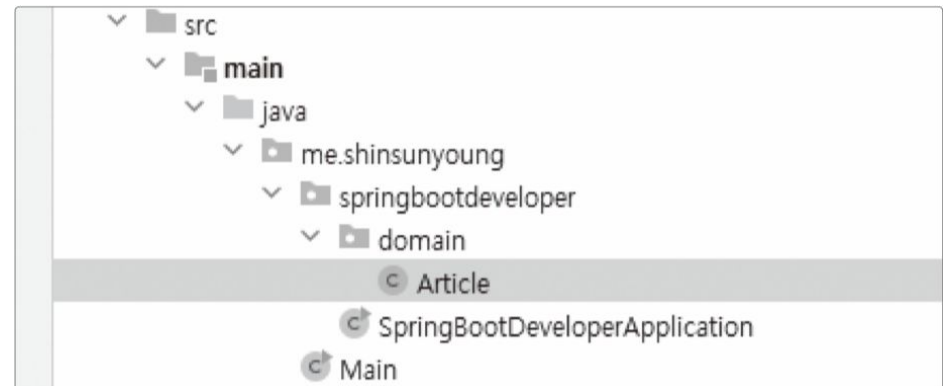
```
dependencies {  
    implementation 'org.springframework.boot:spring-boot-starter-web'  
    implementation 'org.springframework.boot:spring-boot-starter-data-jpa'  
    runtimeOnly 'com.h2database:h2'  
    compileOnly 'org.projectlombok:lombok'  
    annotationProcessor 'org.projectlombok:lombok'  
    testImplementation 'org.springframework.boot:spring-boot-starter-test'  
}
```

build.gradle

6.2 블로그 개발을 위한 엔티티 구성하기

- 엔티티 구성하기
 - domain/Article.java 파일 생성
 - 자세한 내용은 132쪽부터 참고

컬럼명	자료형	null 허용	키	설명
id	BIGINT	N	기본키	일련번호, 기본키
title	VARCHAR(255)	N		게시물의 제목
content	VARCHAR(255)	N		내용

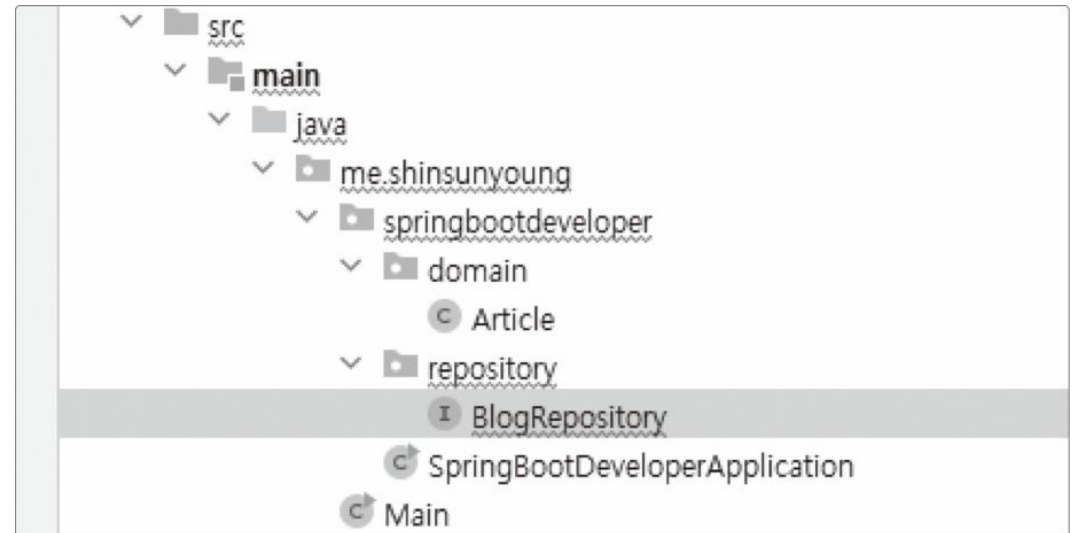


6.2 블로그 개발을 위한 엔티티 구성하기

- 리포지터리 만들기
 - repository/BlogRepository.java 파일 생성
 - 자세한 내용은 135쪽부터 참고

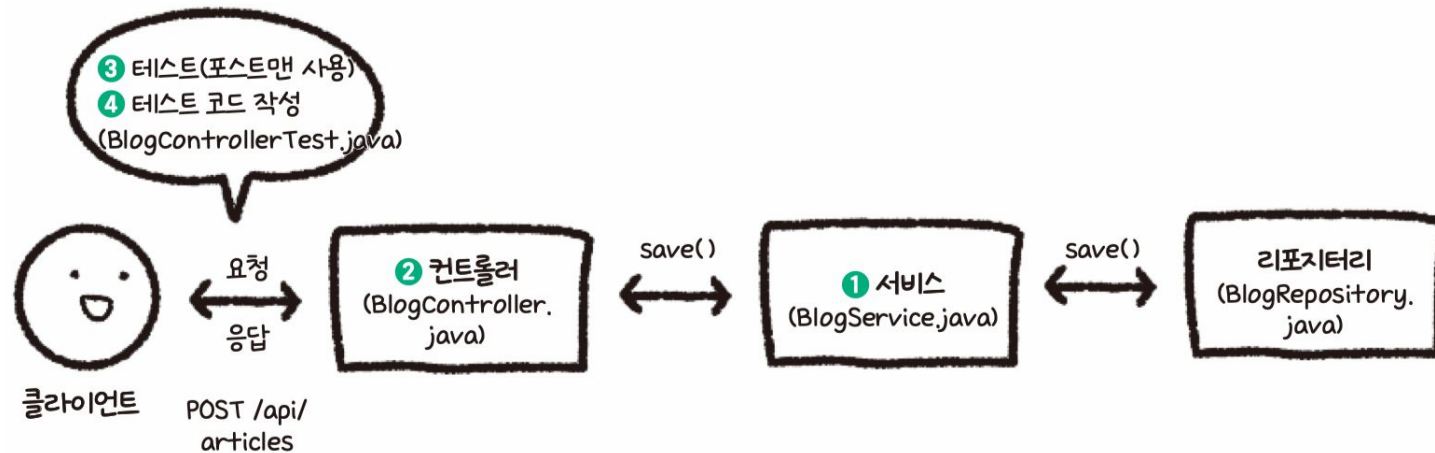
BlogRepository.java

```
public interface BlogRepository extends JpaRepository<Article, Long> {  
}
```



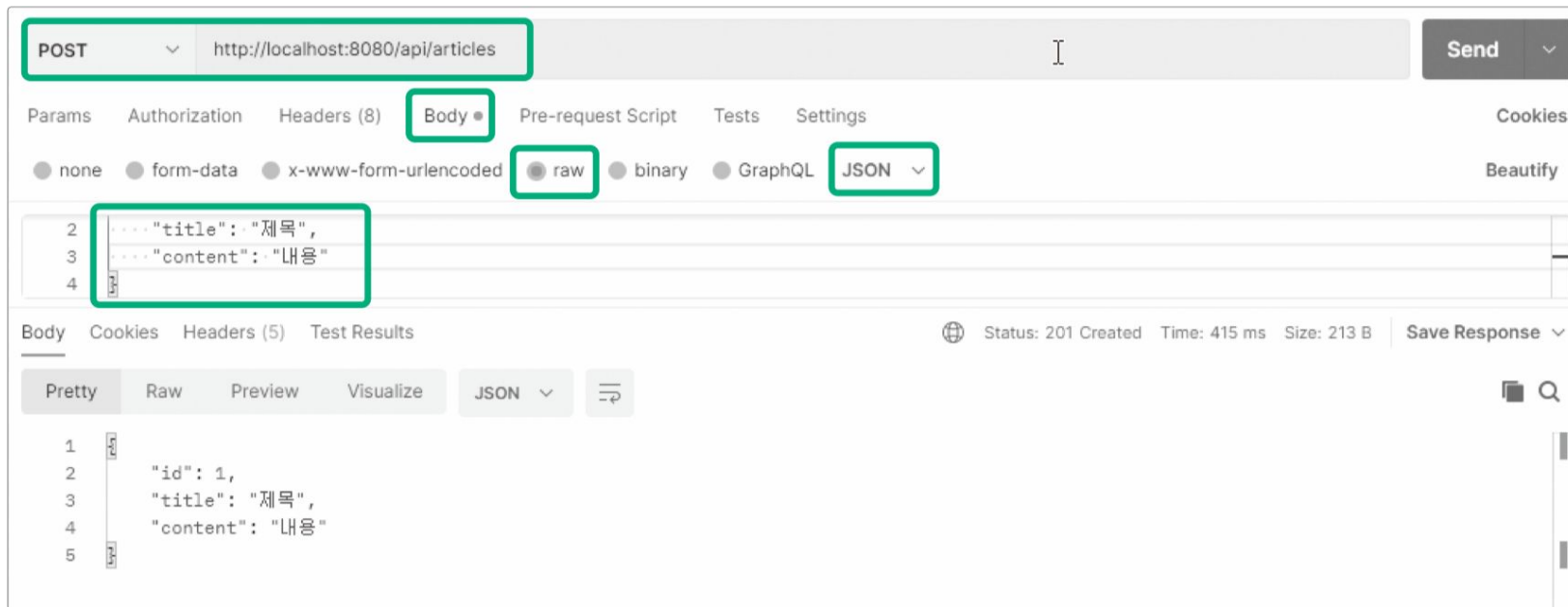
6.3 블로그 글 작성을 위한 API 구현하기

- 구현 및 실습 과정
 - 자세한 내용은 135쪽부터 참고
 - 서비스 클래스 메서드 구현
 - 컨트롤러 클래스 메서드 구현
 - API 테스트



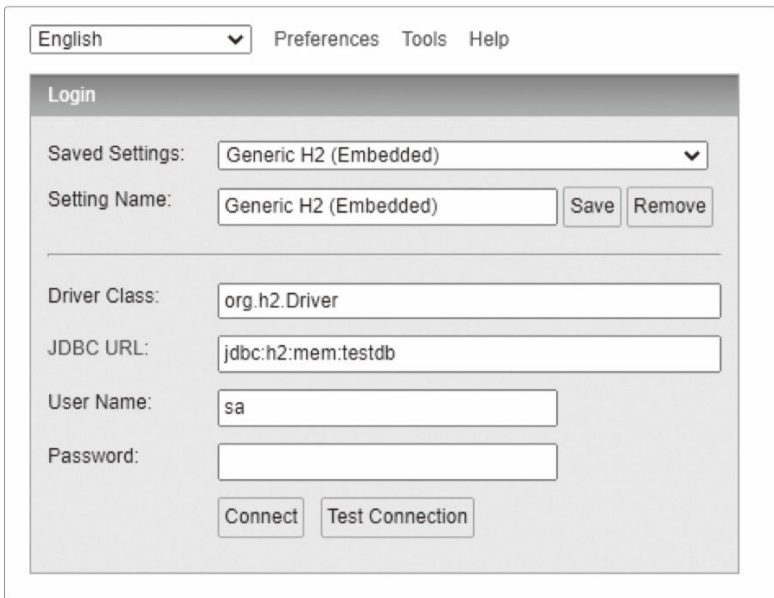
6.3 블로그 글 작성을 위한 API 구현하기 (cont.)

- 코드 작성 후 포스트맨에서 POST 요청
 - /api/articles



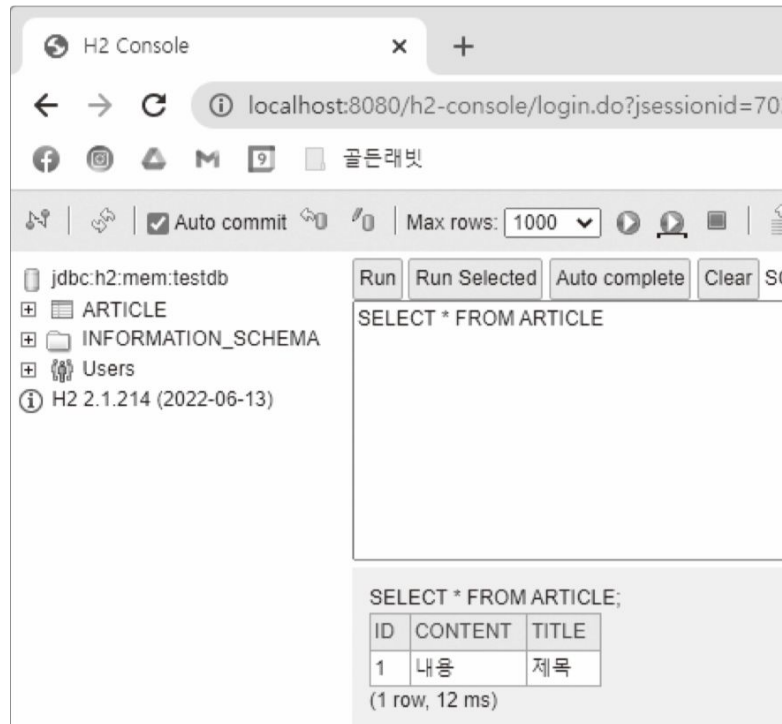
6.3 블로그 글 작성을 위한 API 구현하기 (cont.)

- H2 데이터베이스 콘솔에서 확인
 - 로그인 시 입력값 주의



The screenshot shows the H2 Console Login dialog box. It has a title bar with 'English', 'Preferences', 'Tools', and 'Help'. The 'Login' section contains the following fields and buttons:

- Saved Settings:** A dropdown menu showing 'Generic H2 (Embedded)'.
- Setting Name:** A text field containing 'Generic H2 (Embedded)', with 'Save' and 'Remove' buttons to its right.
- Driver Class:** A text field containing 'org.h2.Driver'.
- JDBC URL:** A text field containing 'jdbc:h2:mem:testdb'.
- User Name:** A text field containing 'sa'.
- Password:** An empty text field.
- Buttons:** 'Connect' and 'Test Connection' buttons at the bottom.



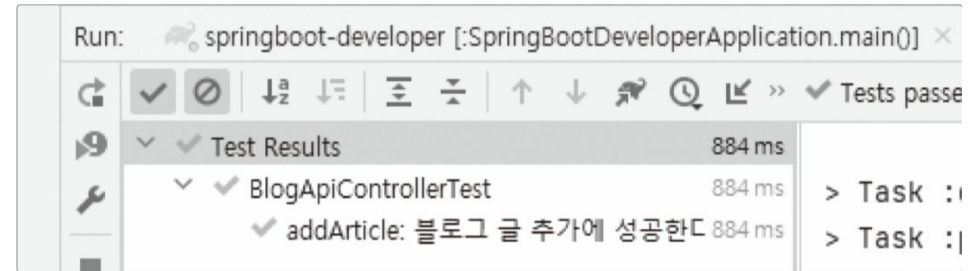
The screenshot shows the H2 Console web interface in a browser. The address bar shows 'localhost:8080/h2-console/login.do?jsessionid=703'. The left sidebar shows the database structure: 'jdbc:h2:mem:testdb', 'ARTICLE', 'INFORMATION_SCHEMA', 'Users', and 'H2 2.1.214 (2022-06-13)'. The main area shows the query 'SELECT * FROM ARTICLE' and its result:

ID	CONTENT	TITLE
1	내용	제목

(1 row, 12 ms)

6.3 블로그 글 작성을 위한 API 구현하기 (cont.)

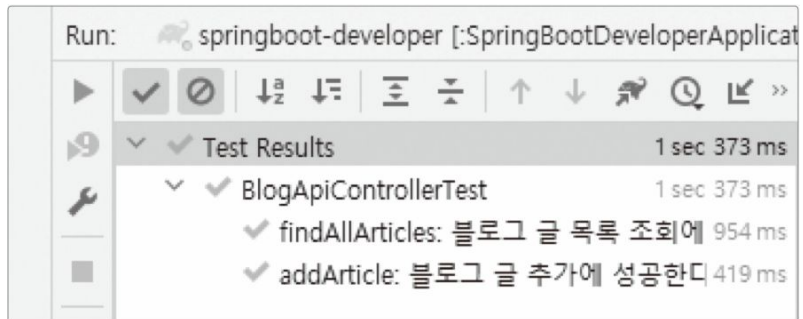
- 테스트 코드 작성하기
 - 142쪽부터 참고
 - BlogApiControllerTest.java
 - 자주 사용하는 테스트 메서드 확인



코드	설명
<code>assertThat(articles.size()).isEqualTo(1);</code>	블로그 글 크기가 1이어야 합니다.
<code>assertThat(articles.size()).isGreaterThan(2);</code>	블로그 글 크기가 2보다 커야 합니다.
<code>assertThat(articles.size()).isLessThan(5);</code>	블로그 글 크기가 5보다 작아야 합니다.
<code>assertThat(articles.size()).isZero();</code>	블로그 글 크기가 0이어야 합니다.
<code>assertThat(article.title()).isEqualTo("제목");</code>	블로그 글의 title값이 "제목"이어야 합니다.
<code>assertThat(article.title()).isNotEmpty();</code>	블로그 글의 title값이 비어 있지 않아야 합니다.
<code>assertThat(article.title()).contains("제");</code>	블로그 글의 title값이 "제"를 포함해야 합니다.

6.4 블로그 글 목록 조회를 위한 API 구현하기

- 실습 목록
 - 146쪽부터 참고
 - BlogService.java
 - ArticleResponse.java
 - BlogApiController.java
 - data.sql
 - 포스트맨 GET 요청 보내기
 - /api/articles/
 - 테스트 코드 작성하기
 - BlogApiControllerTest.java



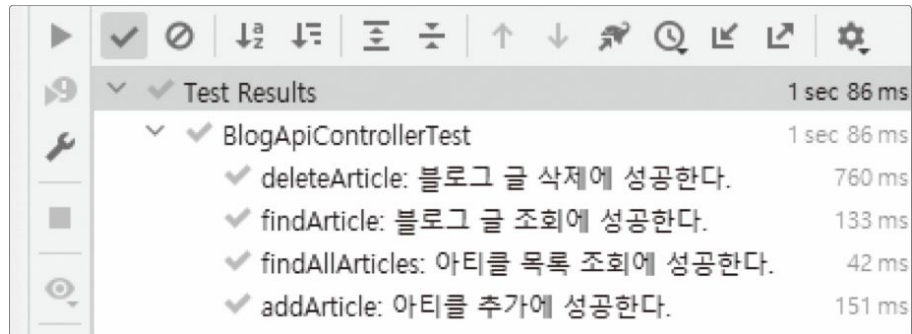
6.5 블로그 글 조회 API 구현하기

- 실습 목록

- 151쪽부터 참고
- BlogService.java
- BlogApiController.java
- 테스트 코드 작성하기
 - BlogApiControllerTest.java

6.6 블로그 글 삭제 API 구현하기

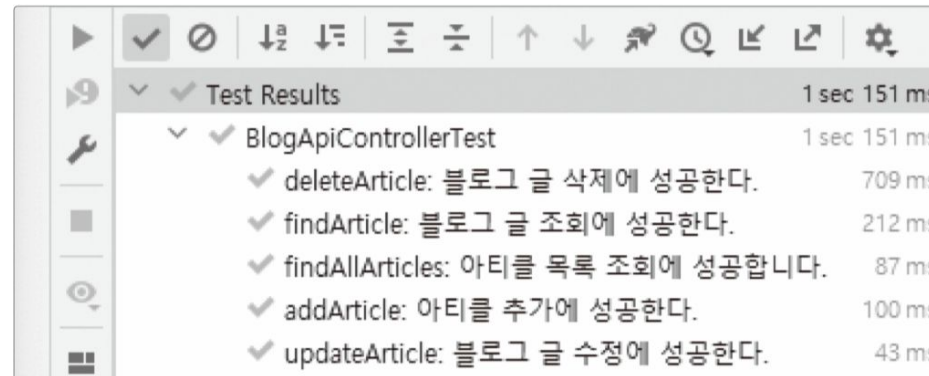
- 실습 목록
 - 154쪽부터 참고
 - BlogService.java
 - BlogApiController.java
 - 포스트맨 DELETE 요청 → GET 요청 보내기
 - [DELETE] /api/articles/1
 - [GET] /api/articles/
 - 테스트 코드 작성하기
 - BlogApiControllerTest.java



Test Results		1 sec 86 ms
BlogApiControllerTest		1 sec 86 ms
✓ deleteArticle: 블로그 글 삭제에 성공한다.		760 ms
✓ findArticle: 블로그 글 조회에 성공한다.		133 ms
✓ findAllArticles: 아티클 목록 조회에 성공한다.		42 ms
✓ addArticle: 아티클 추가에 성공한다.		151 ms

6.6 블로그 글 수정 API 구현하기

- 실습 목록
 - 157쪽부터 참고
 - Article.java
 - UpdateArticleRequest.java
 - BlogService.java
 - BlogApiController.java
 - 포스트맨 PUT 요청 → GET 요청 보내기
 - [PUT] /api/articles/1
 - [GET] /api/articles/
 - 테스트 코드 작성하기
 - BlogApiControllerTest.java



Test Results		1 sec 151 ms
✓	BlogApiControllerTest	1 sec 151 ms
✓	deleteArticle: 블로그 글 삭제에 성공한다.	709 ms
✓	findArticle: 블로그 글 조회에 성공한다.	212 ms
✓	findAllArticles: 아티클 목록 조회에 성공합니다.	87 ms
✓	addArticle: 아티클 추가에 성공한다.	100 ms
✓	updateArticle: 블로그 글 수정에 성공한다.	43 ms