

디플로이먼트, 노드포트, 서비스
mysql 연동

backend-user : external mysql service yaml 작성 및 적용

```
PS C:\server\service> ipconfig
```

Windows IP 구성

...

무선 LAN 어댑터 로컬 영역 연결* 3:

미디어 상태 : 미디어 연결 끊김
연결별 DNS 접미사. . . . :

무선 LAN 어댑터 로컬 영역 연결* 4:

미디어 상태 : 미디어 연결 끊김
연결별 DNS 접미사. . . . :

무선 LAN 어댑터 Wi-Fi:

연결별 DNS 접미사. . . . :
IPv4 주소 : **192.168.0.135**
서브넷 마스크 : 255.255.255.0
기본 게이트웨이 : 192.168.0.1

내 PC IP 확인

```
apiVersion: v1
kind: Service
metadata:
  name: k8s-external-user-mysql-service
spec:
  ports:
    - port: 3306
```

```
---
apiVersion: v1
kind: Endpoints
metadata:
  name: k8s-external-user-mysql-service
subsets:
  - addresses:
    - ip: 192.168.0.135
    ports:
    - port: 13306
```

내 PC IP

입력을 통해 생성

파일을 통해 생성

서식을 통해 생성

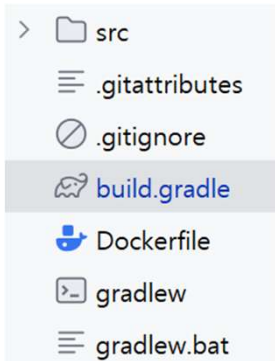
현재 선택된 네임스페이스에 생성할 리소스를 명시하는 YAML

```
1 apiVersion: v1
2 kind: Service
3 metadata:
4   name: k8s-external-user-mysql-service
5 spec:
6   ports:
7     - port: 3306
8
9 ---
10
11 apiVersion: v1
12 kind: Endpoints
13 metadata:
14   name: k8s-external-user-mysql-service
15 subsets:
16   - addresses:
17     - ip: 192.168.0.135
18     ports:
19       - port: 13306
```

업로드

Cancel

backend-user : build.gradle 의존성 추가



```
dependencies {  
    implementation 'org.springframework.boot:spring-boot-starter-web'  
    implementation 'org.springframework.cloud:spring-cloud-starter-openfeign'  
  
    implementation 'org.springframework.boot:spring-boot-starter-data-jpa'  
    runtimeOnly 'com.mysql:mysql-connector-j:8.4.0'  
  
    compileOnly 'org.projectlombok:lombok'  
    annotationProcessor 'org.projectlombok:lombok'  
    testImplementation 'org.springframework.boot:spring-boot-starter-test'  
    testRuntimeOnly 'org.junit.platform:junit-platform-launcher'  
}
```

backend-user : application-dev.yml DB 접속 정보 입력

```
server:
  port: 8080

spring:
  datasource:
    url: jdbc:mysql://k8s-external-user-mysql-
service:3306/user?serverTimezone=UTC&useSSL=true&autoReconnect=true&useUnicode=true&characterEncoding=utf-8
    username: user
    password: 1234
    driver-class-name: com.mysql.cj.jdbc.Driver
  hikari:
    connection-test-query: SELECT 1 # HikariCP 유효성 검사 추가
    validation-timeout: 5000
  jpa:
    hibernate:
      ddl-auto: create # 오직 테스트 환경에서만
      generate-ddl: true # 오직 테스트 환경에서만 (spring.jpa.generate-ddl)
    show-sql: true
    open-in-view: false
```

application-dev.yml

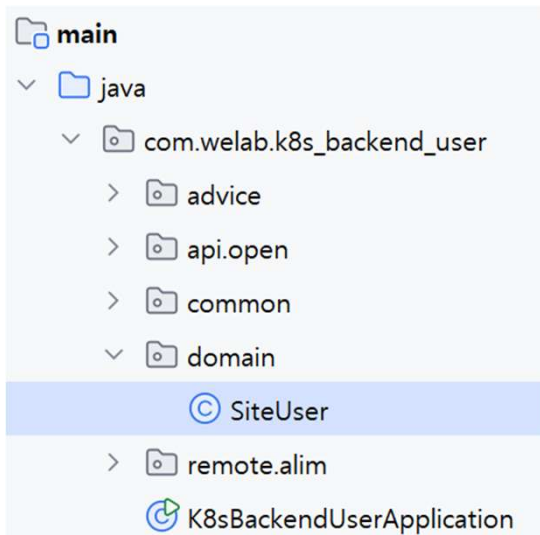
backend-user : application-local.yml DB 접속 정보 입력

```
server:
  port: 8080

spring:
  datasource:
    url: jdbc:mysql://localhost:13306/user?serverTimezone=UTC&useSSL=true&autoReconnect=true&useUnicode=true&characterEncoding=utf-8
    username: user
    password: 1234
    driver-class-name: com.mysql.cj.jdbc.Driver
  hikari:
    connection-test-query: SELECT 1 # HikariCP 유효성 검사 추가
    validation-timeout: 5000
  jpa:
    hibernate:
      ddl-auto: create # 오직 테스트 환경에서만
      generate-ddl: true # 오직 테스트 환경에서만 (spring.jpa.generate-ddl)
    show-sql: true
    open-in-view: false
```

application-dev.yml

backend-user : SiteUser Entity 추가



```
@Slf4j
@Entity
@Table(name = "site_user")
public class SiteUser {
    @Id
    @Column(name = "id")
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Getter
    @Setter
    private Long id;

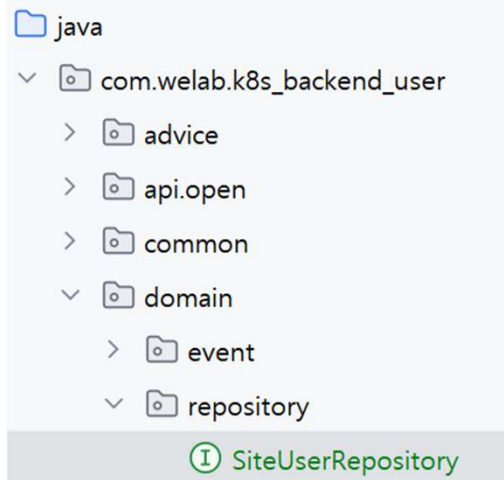
    @Column(name = "user_id", unique = true, nullable = false)
    @Getter @Setter
    private String userId;

    @Column(name = "password", nullable = false)
    @Getter @Setter
    private String password;

    @Column(name = "phone_number", nullable = false)
    @Setter @Getter
    private String phoneNumber;

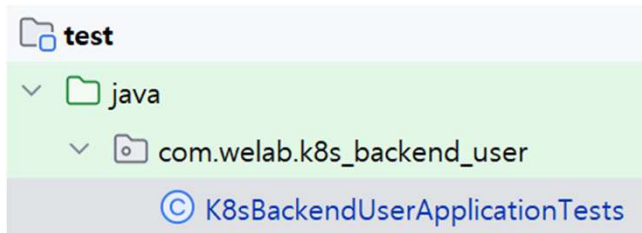
    @Column(name = "deleted", nullable = false)
    @Getter @Setter
    private Boolean deleted = false;
}
```

backend-user : SiteUserRepository 추가



```
@Repository
public interface SiteUserRepository extends JpaRepository<SiteUser, Long> {
    SiteUser findById(String userId);
}
```

backend-user : SpringBootTest contextLoads 주석처리



```
//@ActiveProfiles("test")
@SpringBootTest
class K8sBackendUserApplicationTests {

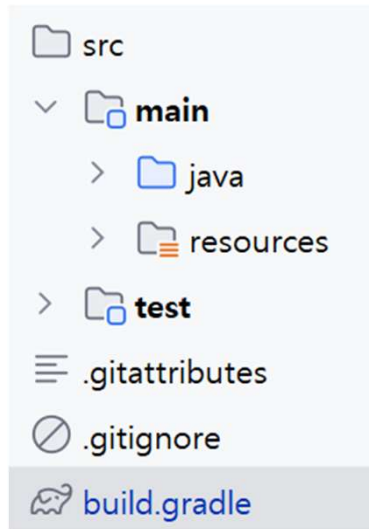
    // @Test
    // void contextLoads() {
    // }

}
```

test시, jdbc 연결 시도 실패로 인한 에러가 발생하여 빌드가 진행이 안됨

-> test 비활성화 必

backend-user : 앱 버전 업그레이드



```
plugins {  
    id 'java'  
    id 'org.springframework.boot' version '3.5.0'  
    id 'io.spring.dependency-management' version '1.1.7'  
}  
  
group = 'com.welab'  
version = '0.0.3'  
  
...
```

backend-user : 코드 commit & push

Commit & Push to Github

backend-user : Jenkins Pipeline 빌드

Status

Changes

▶ 파라미터와 함께 빌드

구성

Pipeline 삭제

Move

Full Stage View

Stages

Rename

Pipeline Syntax

Pipeline kube-backend-user

매개변수가 필요한 빌드입니다.

TAG

origin/main

☐ RELEASE

▶ 매개변수가 필요한 빌드입니다.

Cancel

Declarative: Checkout SCM	Declarative: Tool Install	Set Version	Build & Test Application	Build Docker Image	Push Docker Image
4s	367ms	1min 14s	49s	7s	11s
4s	367ms	1min 14s	49s	7s	11s

backend-user : Deployment 이미지 버전 업데이트

Deployments

이름	이미지	레이블	파드	생성 시간 ↑
k8s-backend-alim-deployment	solarhc/k8s-backend-alim:0.0.1	-	1 / 1	16 minutes ago
k8s-backend-user-deployment	solarhc/k8s-backend-user:0.0.1	-	1 / 1	2 hours ago
	solarhc/k8s-api-gateway:0.0.3	-	1 / 1	2 hours ago

97 spec:
98 replicas: 1
99 selector:
100 matchLabels:
101 app: k8s-backend-user
102 template:
103 metadata:
104 creationTimestamp: null
105 labels:
106 app: k8s-backend-user
107 spec:
108 containers:
109 - name: k8s-backend-user
110 image: solarhc/k8s-backend-user: 0.0.3
111 ports:

이 액션은 다음 커맨드와 동일합니다. `kubectl apply -f <spec.yaml>`

Update

Cancel

Scale

Edit

Restart

Delete