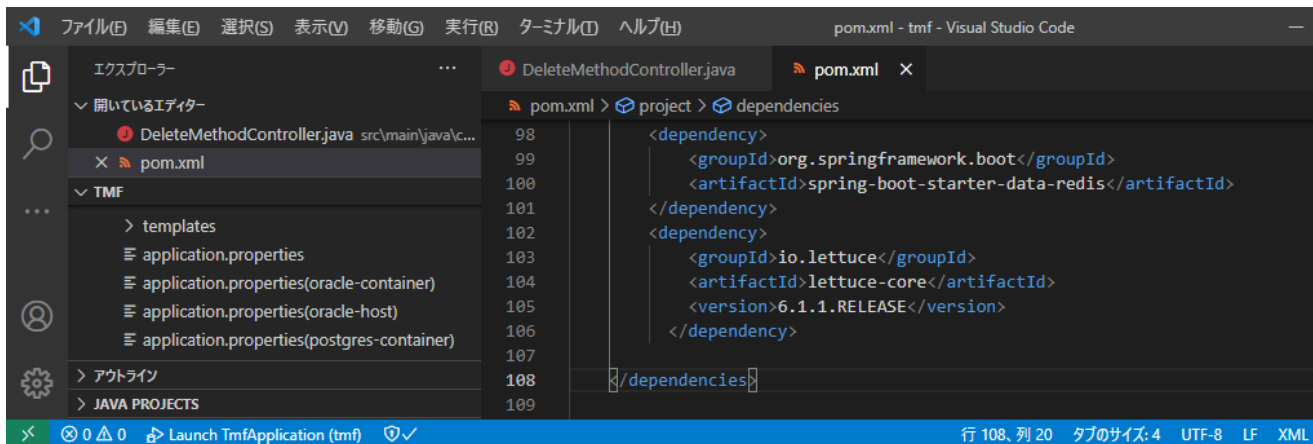


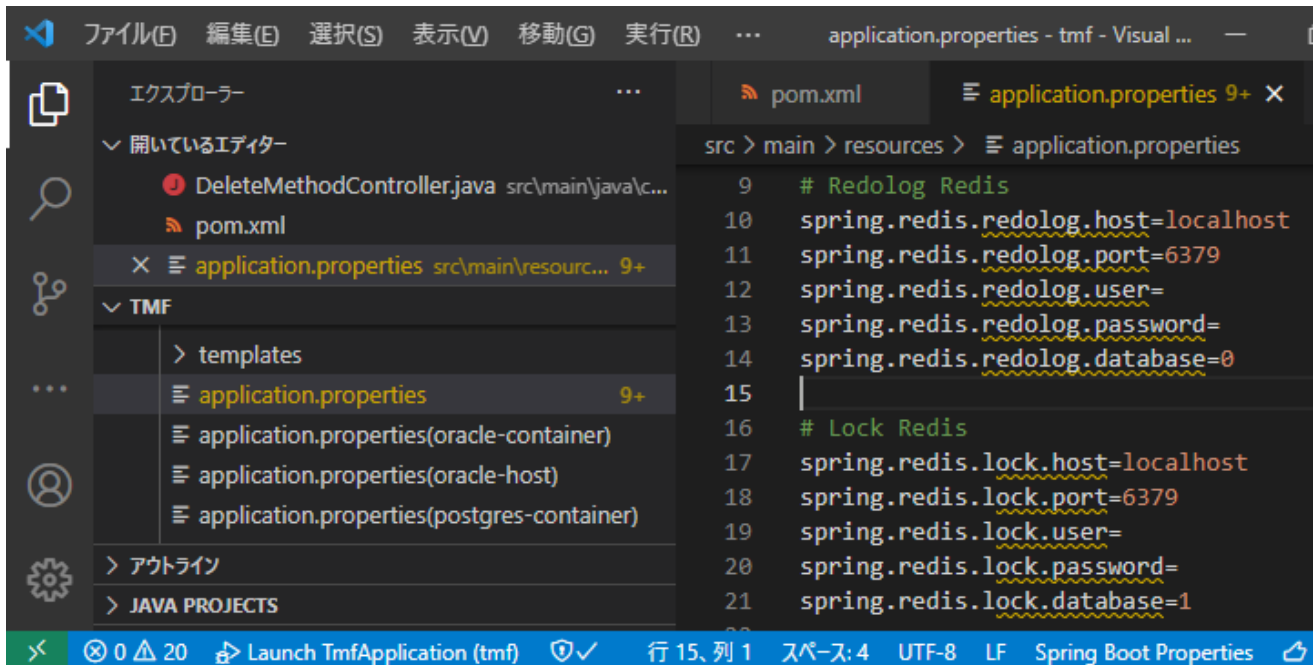
# ホストのJavaからコンテナのRedisへの接続

## 1. Redisの設定

- a. 「pom.xml」に「<dependency>〜</dependency>」を追記。



- b. application.propertiesのを以下のように修正。



- c. ソースを以下のように修正

```
src > main > java > com > example > tmf > RedologConnectionFactory.java > user
1 package com.example.tmf;
2
3 import org.springframework.beans.factory.annotation.Value;
4 import org.springframework.context.annotation.Bean;
5 import org.springframework.context.annotation.Configuration;
6 import org.springframework.data.redis.connection.RedisStandaloneConfiguration;
7 import org.springframework.data.redis.connection.lettuce.LettuceConnectionFactory;
8
9
10 //=====
11 // Redolog用 ConnectionFactory クラス
12 //=====
13 @Configuration
14 public class RedologConnectionFactory{
15
16     // application.propertiesの値を設定
17     @Value("${spring.redis.redolog.host}") String host;
18     @Value("${spring.redis.redolog.port}") int port;
19     @Value("${spring.redis.redolog.user}") String user;
20     @Value("${spring.redis.redolog.password}") String password;
21     @Value("${spring.redis.redolog.database}") int database;
22
23     //=====
24     // Redolog用 ConnectionFactory を返す
25     //=====
26     @Bean
27     @Primary
28     public LettuceConnectionFactory redologLettuceConnectionFactory(){
29
30         // lettuce(native Redis Client) Instance(比(hostName, portNo)を設定)
31         RedisStandaloneConfiguration redisStandaloneConfiguration = new RedisStandaloneConfiguration(host, port);
32         // userNameを設定
33         redisStandaloneConfiguration.setUsername(user);
34         // passwordを設定
35         redisStandaloneConfiguration.setPassword(password);
36         // databaseを設定
37         redisStandaloneConfiguration.setDatabase(database);
38
39         return new LettuceConnectionFactory(redisStandaloneConfiguration);
40     }
41 }
```

```
src > main > java > com > example > tmf > RedologRedisTemplate.java > RedologRedisTemplate > getRedologR
1 package com.example.tmf;
2
3 import org.springframework.beans.factory.annotation.Autowired;
4 import org.springframework.beans.factory.annotation.Qualifier;
5 import org.springframework.context.annotation.Bean;
6 import org.springframework.context.annotation.Configuration;
7 import org.springframework.data.redis.connection.lettuce.LettuceConnectionFactory;
8 import org.springframework.data.redis.core.RedisTemplate;
9 import org.springframework.data.redis.serializer.JdkSerializationRedisSerializer;
10 import org.springframework.data.redis.serializer.StringRedisSerializer;
11
12 //=====
13 // Redolog用 RedisTemplate クラス
14 // @Configuration必須。
15 //=====
16 @Configuration
17 public class RedologRedisTemplate{
18
19     @Autowired
20     @Qualifier("redologLettuceConnectionFactory")
21     private LettuceConnectionFactory lettuceConnectionFactory;
22
23     //=====
24     // Redolog用 RedisTemplate。@Bean必須。
25     //=====
26     @Bean
27     public RedisTemplate<String, Redolog> getRedologRedisTemplate(){
28
29         RedisTemplate<String, Redolog> redisTemplate = new RedisTemplate<>();
30         redisTemplate.setConnectionFactory(lettuceConnectionFactory);
31         redisTemplate.setKeySerializer(new StringRedisSerializer());
32         redisTemplate.setValueSerializer(new JdkSerializationRedisSerializer());
33         redisTemplate.setHashKeySerializer(redisTemplate.getKeySerializer());
34         redisTemplate.setHashValueSerializer(redisTemplate.getValueSerializer());
35         return redisTemplate;
36     }
37 }
```

