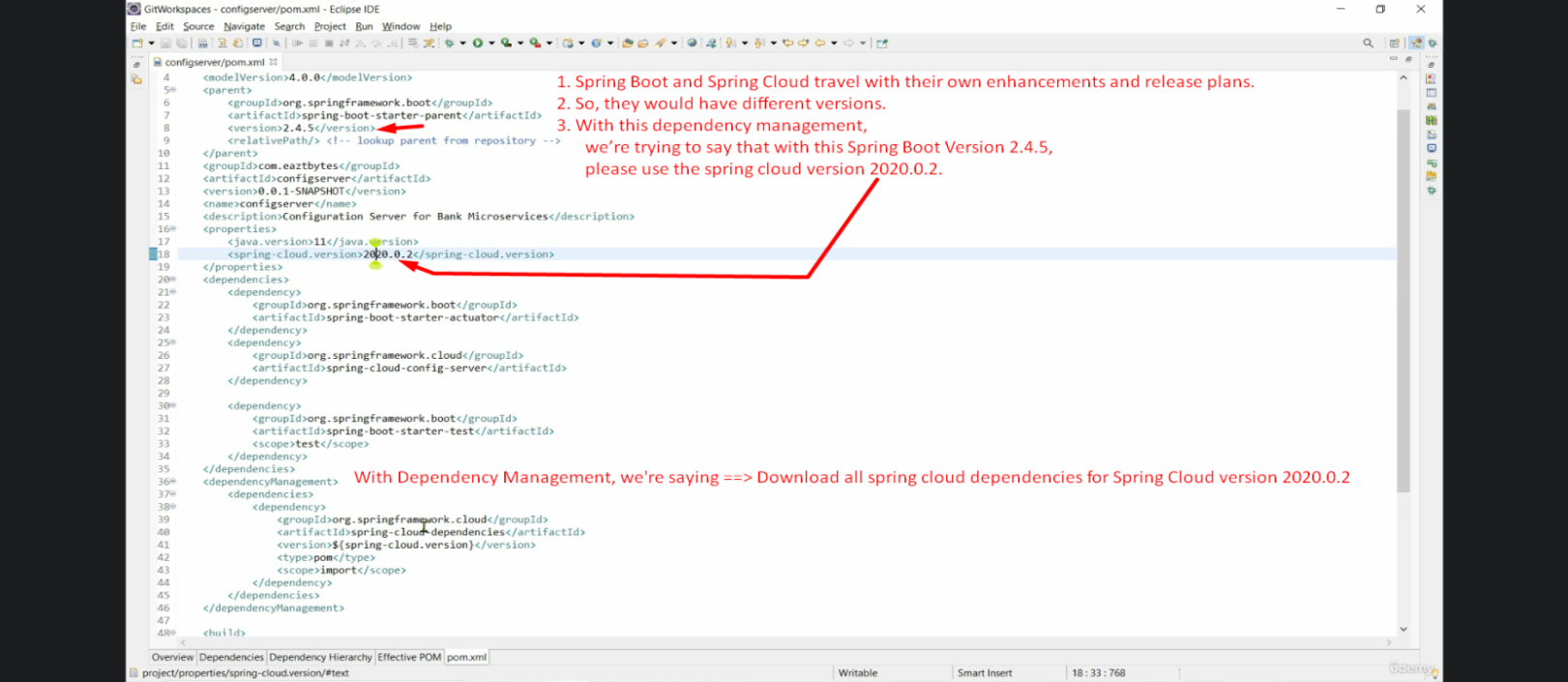
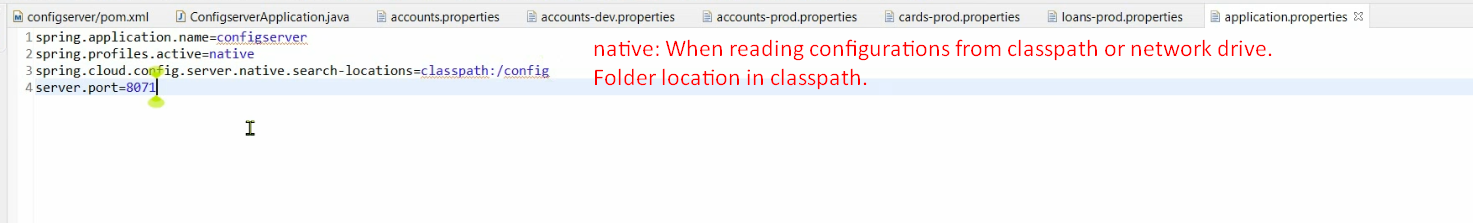
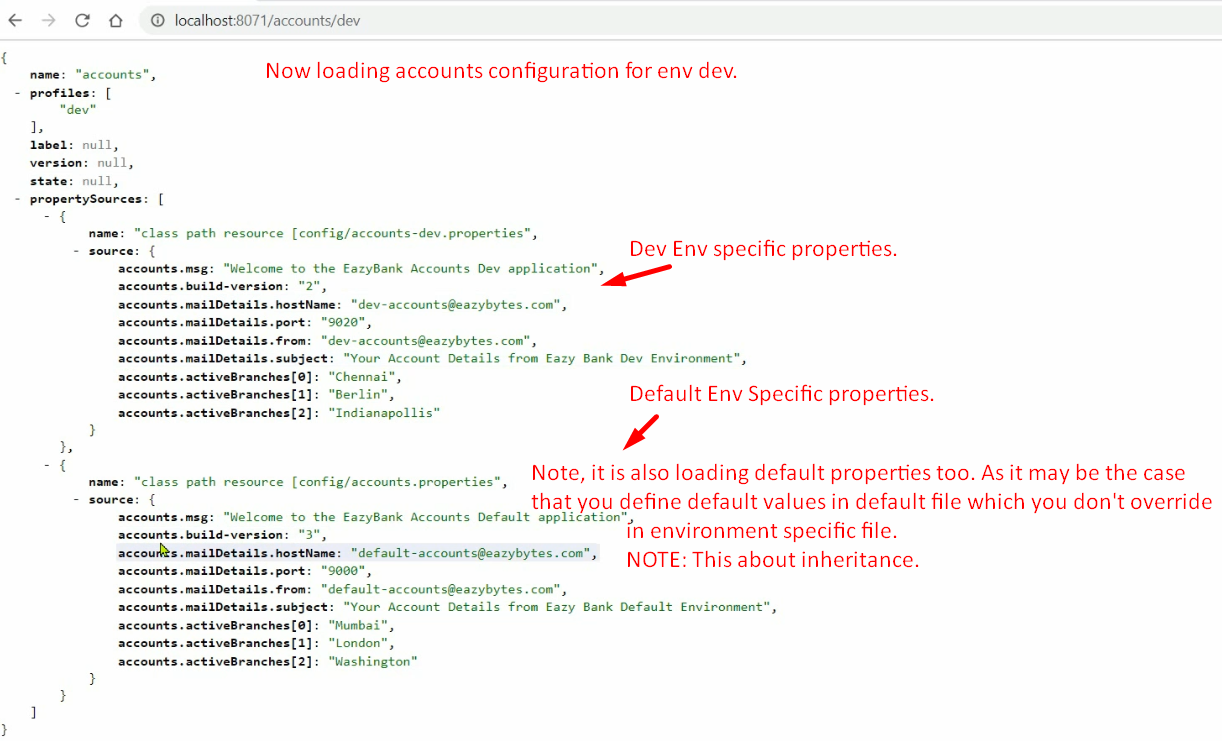
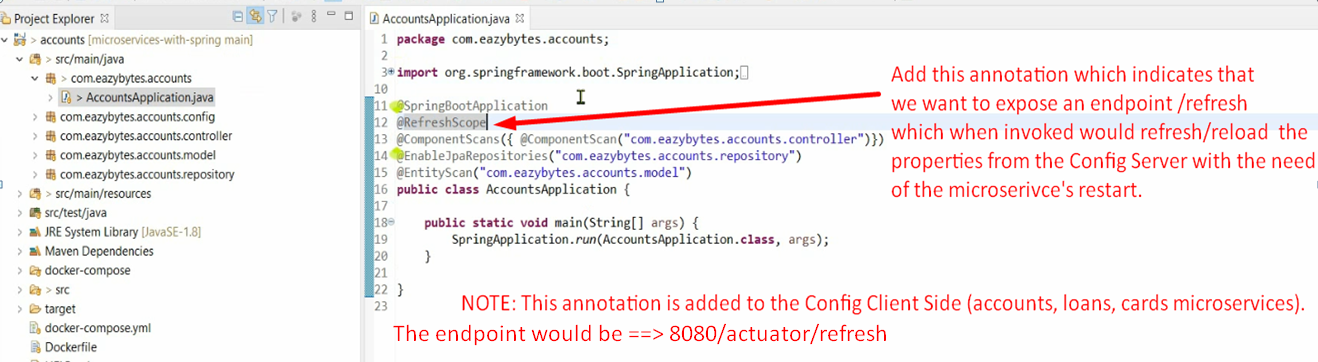
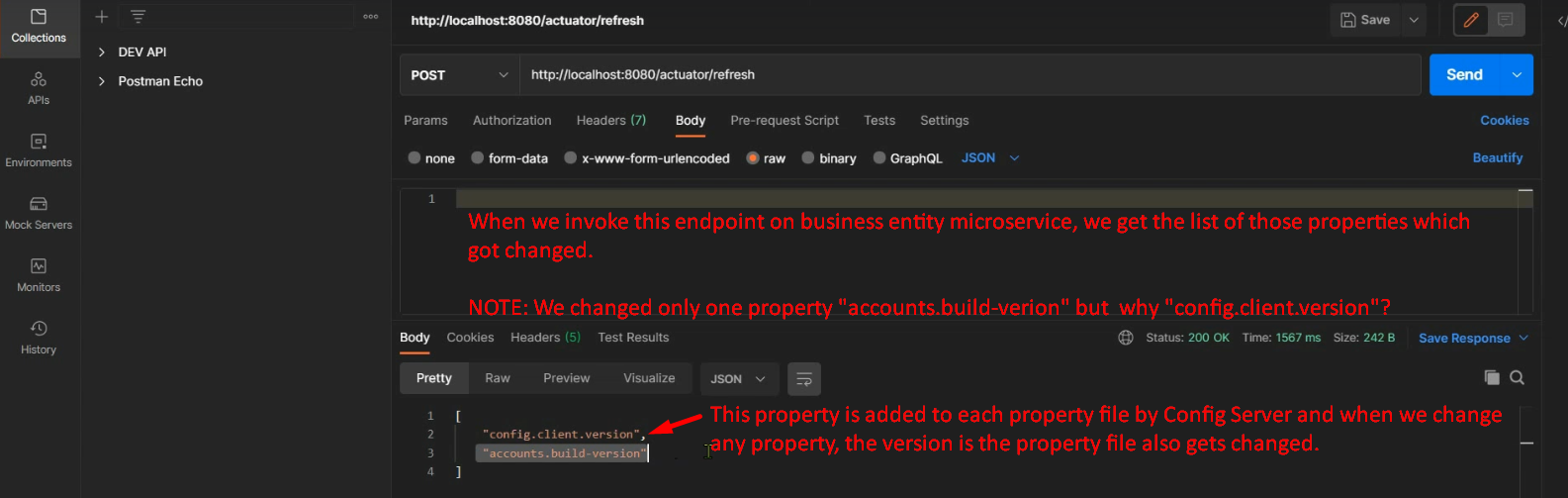
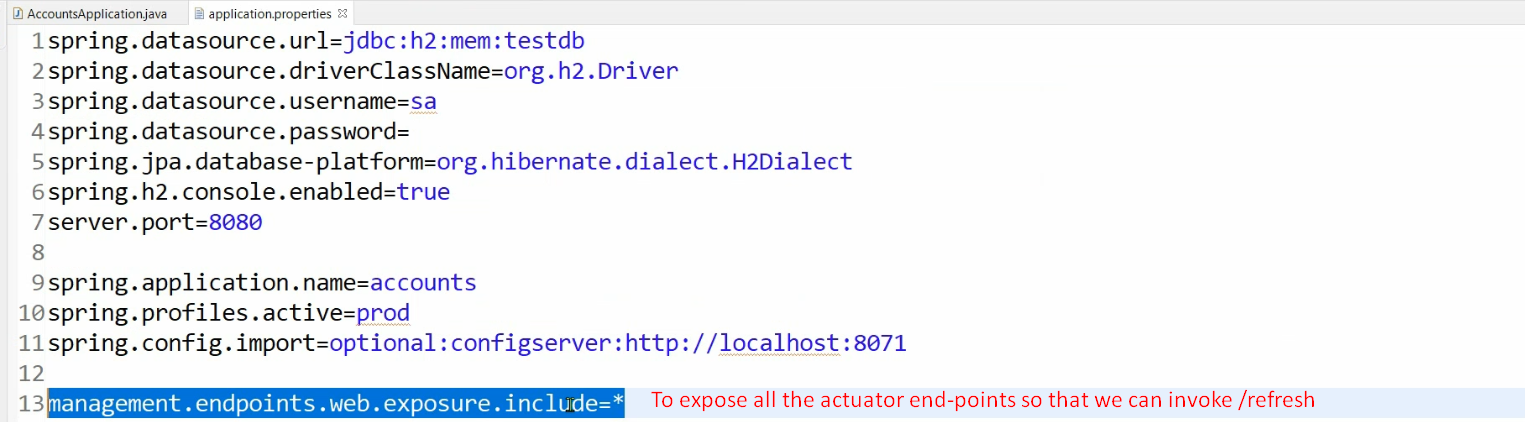
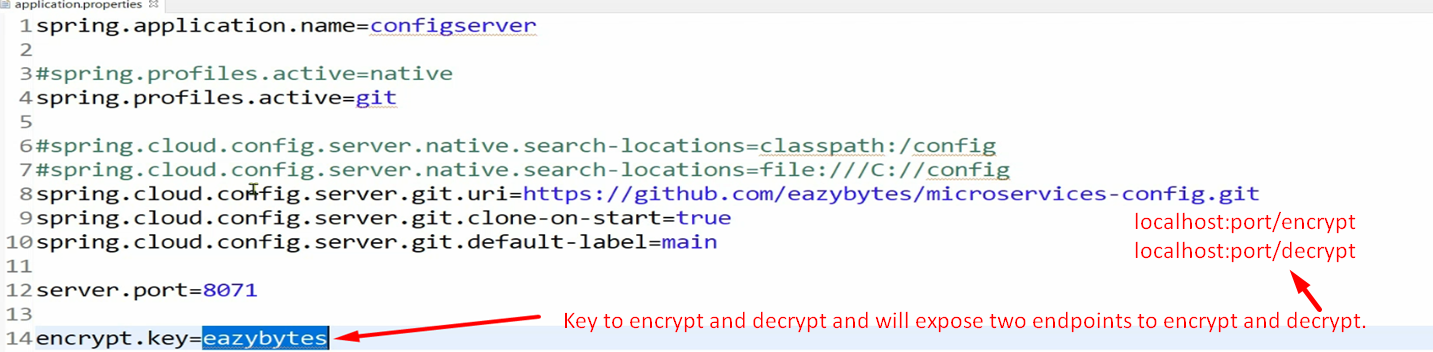
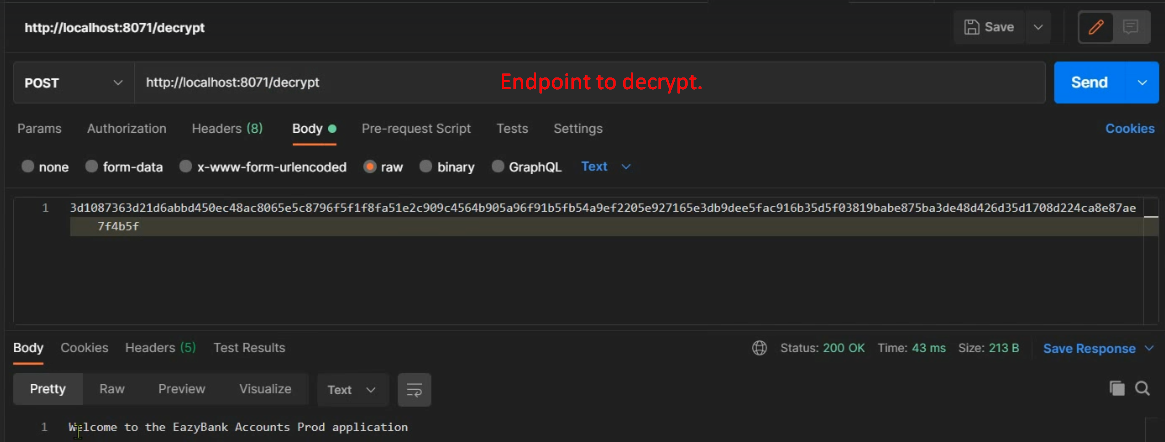
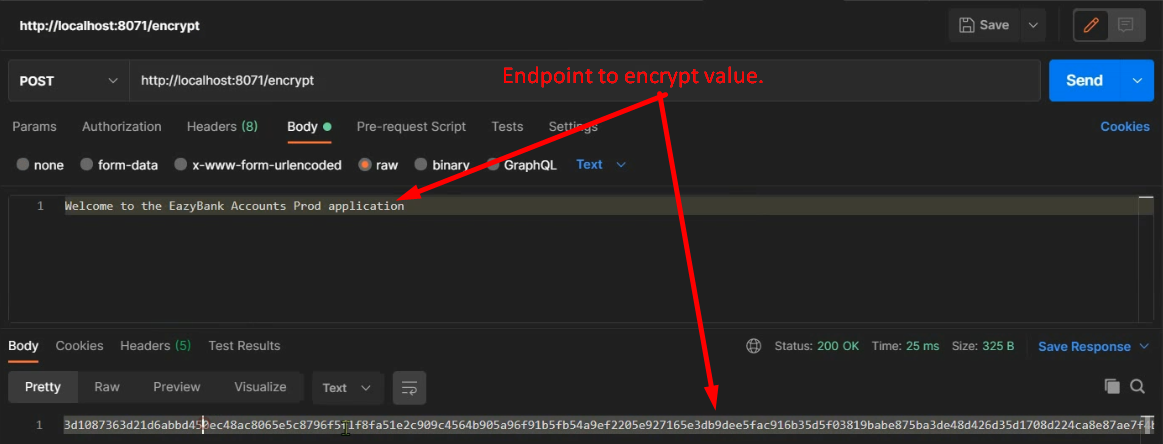
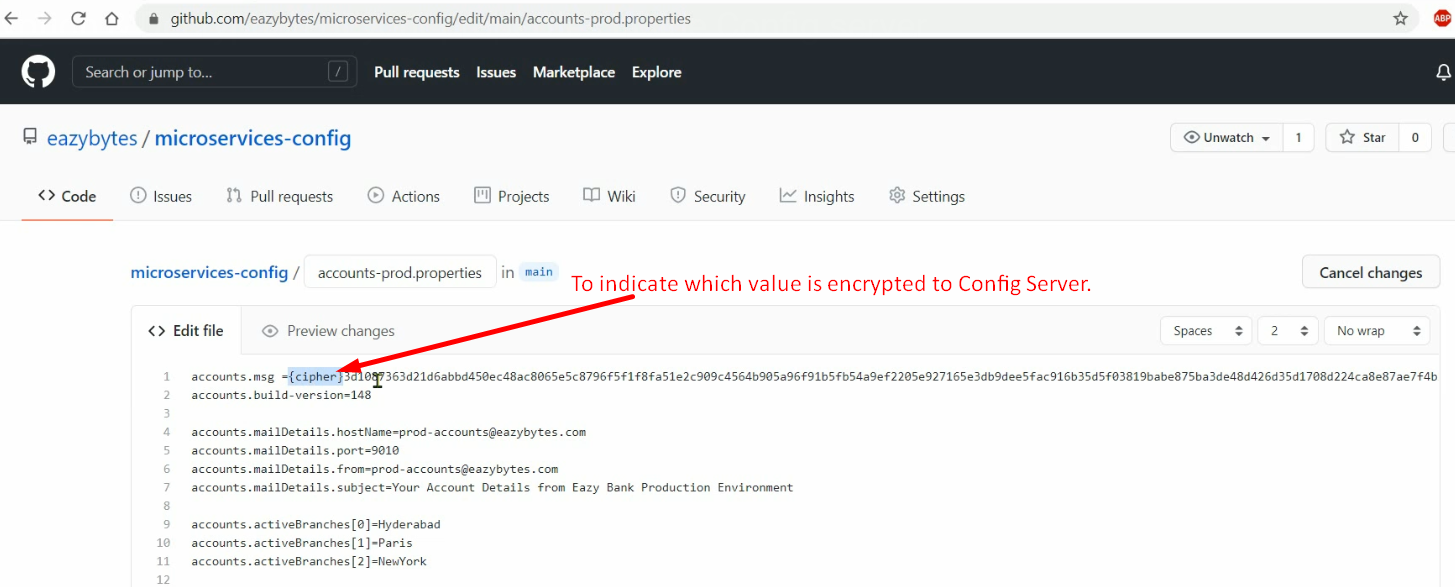
1. **Step 01**: Add the following Dependency:

</dependency>  
 <groupId>org.springframework.cloud</groupId>

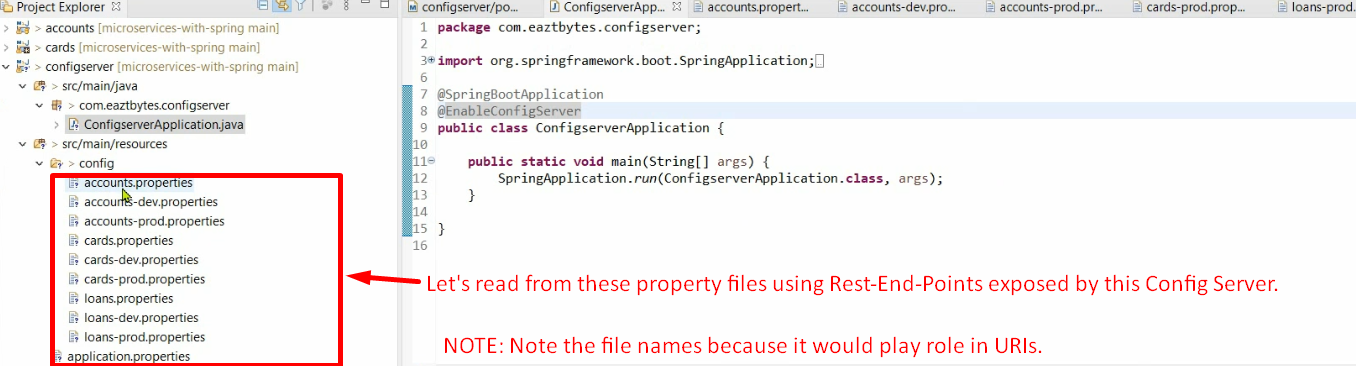
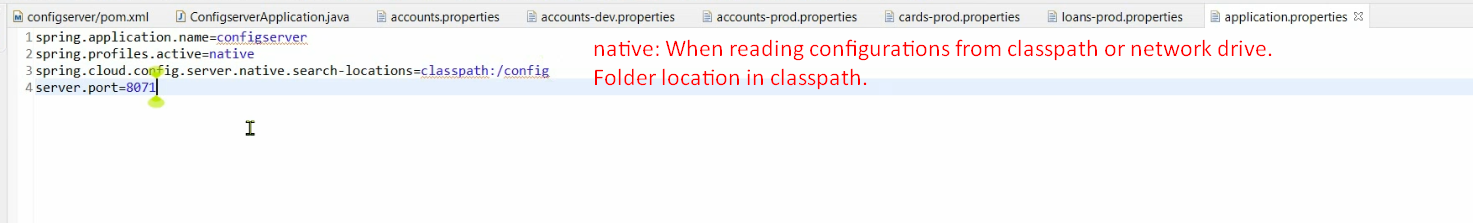
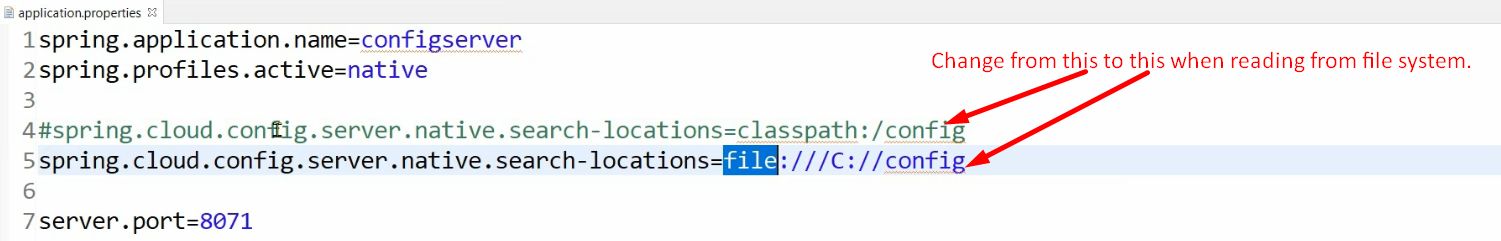
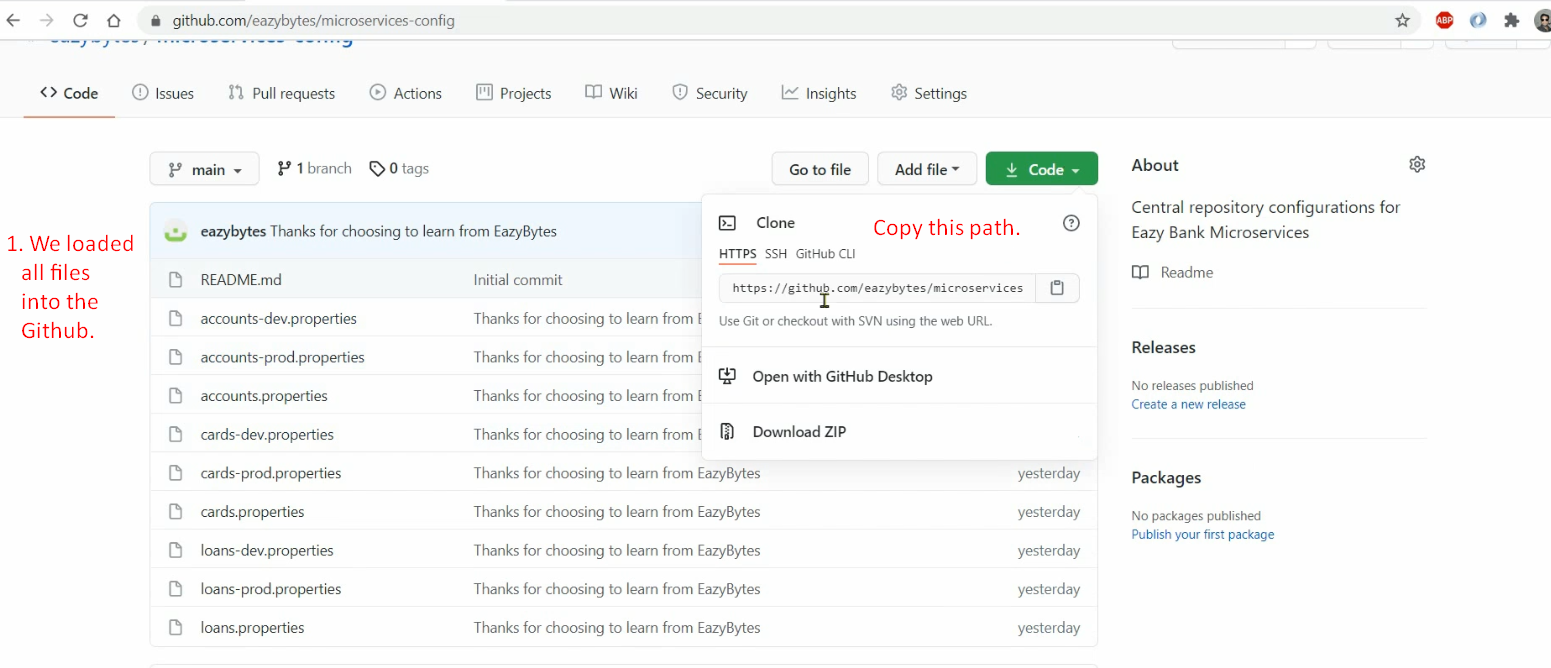
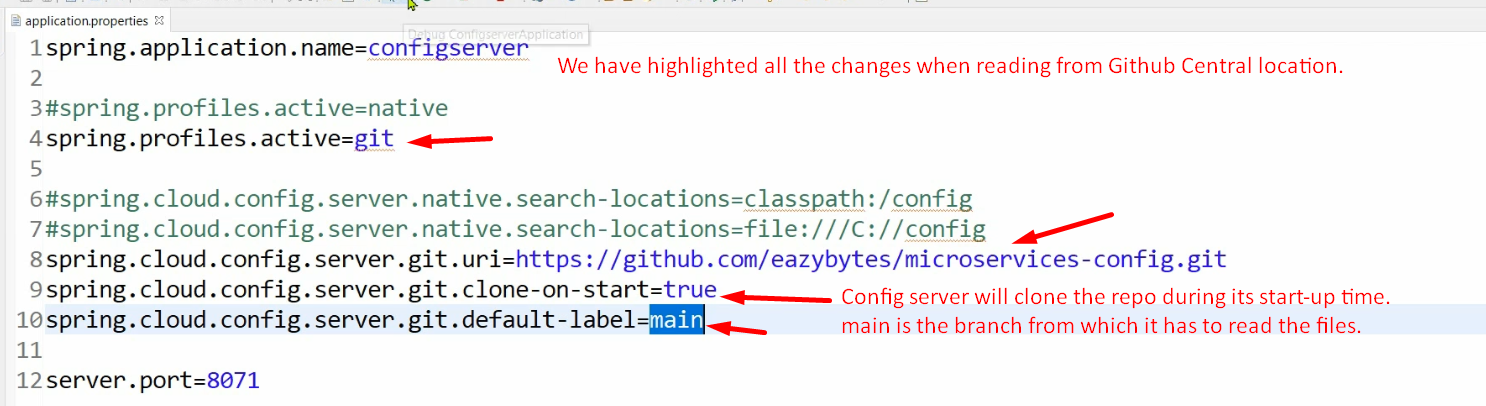
<artifactId>spring-cloud-starter-config</artifactId>

</dependency>  
**NOTE**: When you add the above dependency, the following dependency under <dependencyManagement> is added automatically.

* 1. 

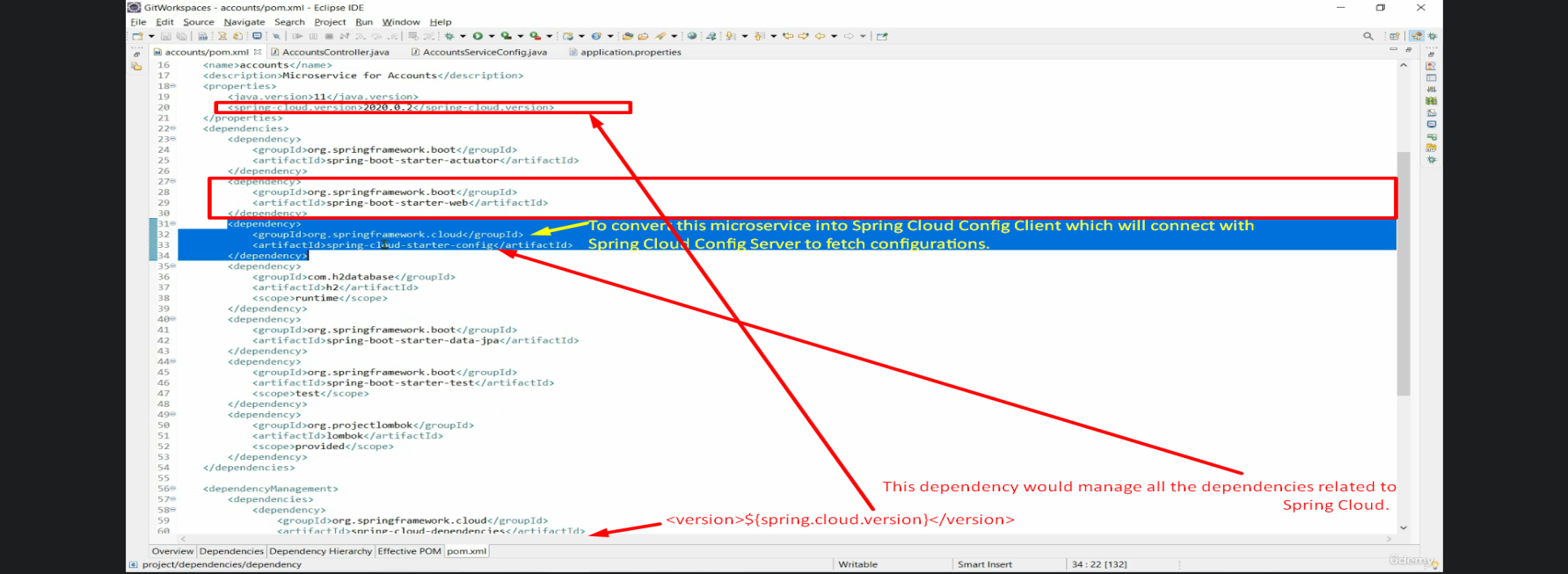
1. **Step 02**:
   1. **@EnableConfigServer:** Would convert your microservice into Spring Cloud Config Server. 
2. **Step 03**: From where it has to read the configurations.
   1.   
      **native**: When reading from local file system.
3. **Step 04**: Endpoint URL based on property file name  
   
4. @RefreshScope:
   1.   
      
5. Encrypt and Decrypt:
   1. 
   2. 

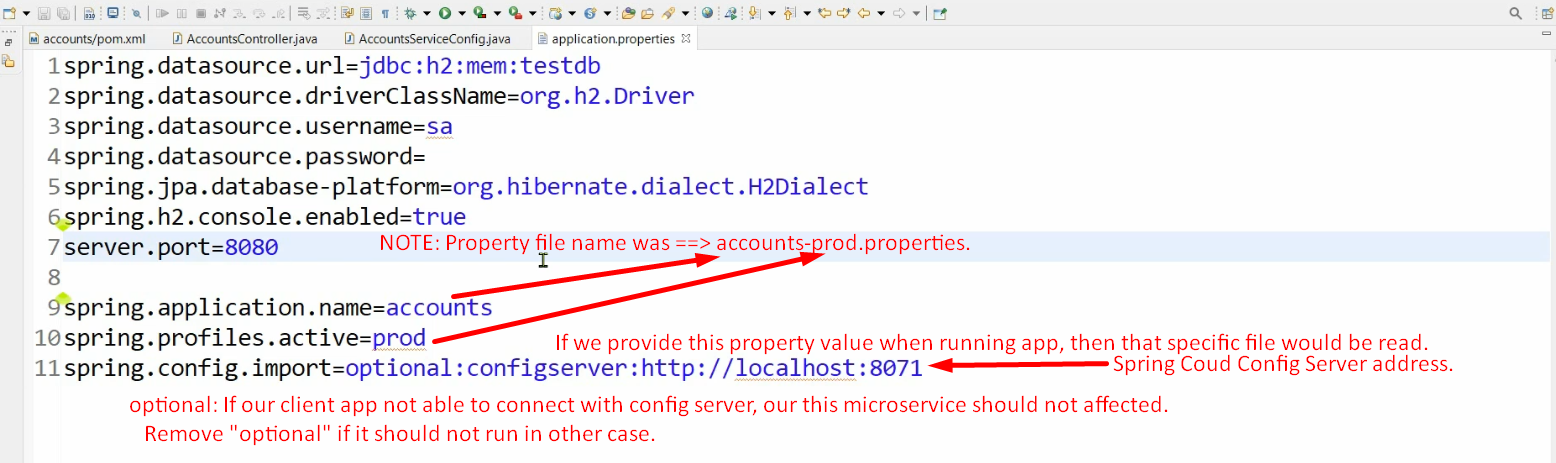
Reading files from different locations

1. **Reading file from classpath**:  
   
2. **Reading File from Local File System**:
   1. 
3. **Reading File from Github:**
   1. Pull all the files under some branch.   
      Don’t put under any folder  
      
   2. Make configurations in application.properties file.  
      

Now Let Create Client Server Reading Properties file from Spring Cloud Config Server.

1. Adding dependency.



1. Configuration in application.properties file  
   
2. First properties file from the Config Server would be fetched then its properties would be added to spring context.   
   Then a bean of the AccountSerivceConfig would be created and all the properties staring with accounts would be mapped to properties.  
   