

Hibernate Ve Maven Ile Veritabanındaki Tablolara Karşılık Gelen Java Sınıflarının Yaratılması

Hibernate ve Maven ile **reverse engineering** yaparak veritabanındaki tablolara karşılık gelecek **Java sınıfları** (POJO-Plain Old Java Object), **Dao** sınıfları, **hbm.xml** dosyaları yaratabiliriz. Bu sayede çok fazla sayıda veritabanı tablolarını **manuel** olarak yaratmaktan **kurtulmuş oluruz**. Bu işlemin yapılmasına reverse engineering(**tersinir mühendislik**) denir.

Önceki **makalede** (/tutorial/Hibernate-Ile-Veritabanındaki-Tablolara-Karsilik-Gelecek-Java-Siniflarinin-POJO-Otomatik-Olusturulmasi) bu işlemlerin **Eclipse** ile yapılması anlatılmıştır. Bu makalemizde ise maven kullanarak aynı işlemleri yapacağız.

Gerekenler:

hibernate.properties dosyası
model_reveng.xml dosyası
hibernate3-maven plugin
mysql-connector.jar

Klasör Yapısı

```
ProjeAdi
  src
    main
      java
      resources
        META-INF
          hibernate.properties
        reveng
          model_reveng.xml
    test
  pom.xml
```

hibernate.properties dosyası

src/main/resources/ klasörüne **META-INF** isminde bir klasör eklenir. Bu klasör içerisinde ise **hibernate.properties** isimli bir dosya yaratılır. Bu dosya **veritabanına bağlanmak** için gerekli ayarları içerecektir. Örnek bir hibernate.properties dosya içeriği şu şekildedir:

```
1 hibernate.connection.username=root
2 hibernate.connection.password=12
3 hibernate.connection.url=jdbc:mysql://127.0.0.1:3306/yourDB
4 hibernate.connection.driver_class=com.mysql.jdbc.Driver
5 hibernate.dialect=org.hibernate.dialect.MySQL5InnoDBDialect
```

model_reveng.xml dosyası

src/main/resources/ klasörüne **reveng** isminde bir klasör eklenir. Bu klasör içerisinde ise **model_reveng.xml** isimli bir dosya yaratılır. Bu dosya reverse engineering yapılması için, gerekli ayarları içerir. Örnek bir model_reveng.xml dosyası şu şekildedir:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE hibernate-reverse-engineering SYSTEM
3     "http://hibernate.sourceforge.net/hibernate-reverse-engineering-3.0.dtd">
4 <hibernate-reverse-engineering>
5     <table-filter match-name=".*" package="your.package.here"/>
6 </hibernate-reverse-engineering>
```

package kısmında oluşturulacak Java dosyalarının paketi belirtilir.

Maven Hibernate Plugin

```
1 <plugin>
2   <groupId>org.codehaus.mojo</groupId>
3   <artifactId>hibernate3-maven-plugin</artifactId>
4   <version>2.2</version>
5   <configuration>
6     <components>
7       <component>
8         <name>hbm2java</name>
9         <outputDirectory>src/main/java</outputDirectory>
10        <implementation>jdbcconfiguration</implementation>
11      </component>
12    </components>
13    <componentProperties>
14      <!--Reverse Eng using hbm2java: ejb3 give us annotated classes -->
15      <revengfile>/src/main/resources/reveng/model_reveng.xml</revengfile>
16      <propertyfile>/src/main/resources/META-INF/hibernate.properties</propertyfile>
17      <jdk5>true</jdk5>
18      <ejb3>true</ejb3>
19      <format>true</format>
20    </componentProperties>
21  </configuration>
22  <dependencies>
23    <dependency>
24      <groupId>mysql</groupId>
25      <artifactId>mysql-connector-java</artifactId>
26      <version>5.1.6</version>
27    </dependency>
28    <dependency>
29      <groupId>cglib</groupId>
30      <artifactId>cglib-nodep</artifactId>
31      <version>2.1_3</version>
32    </dependency>
33  </dependencies>
34 </plugin>
```

componentProperties kısmı en önemli konfigürasyonların olduğu kısımdır. **ejb3 true** denilerek **annotasyon** özellikli Java sınıflarının oluşturulması sağlanır. Ayrıca **jdk5** ile de JDK 5 yapısına uygun sınıfların üretilmesi sağlanır. Ayrıca dikkat ederseniz **propertyfile** ve **revengfile** ayarlarının aldıkları değerler önceki adımlarda belirtilen **adreslerdir**. Bu adresler **explicitly** olarak belirtilmiştir.

Goals

Hibernate maven plugin'in bazı goal(hedef) seçenekleri şunlardır:

mvn hibernate3:hbm2cfgxml: Bir tane **hibernate.cfg.xml** dosyası üretir. Bu dosya içerisine, reverse engineering ile üretilen sınıfların **map'leri** ve **hibernate.properties** dosyasındaki ayarlar eklenir.

mvn hibernate3:hbm2dao: bir dizi DAO sınıfları üretir.

mvn hibernate3:hbm2java: Bu komut ile veritabanındaki tablolara karşılık gelen Java sınıfları oluşturulur. Bu sınıflar oluşturulurken componentProperties kısmında bulunan **ejb3** ve **jdk5** değerlerine göre **annotasyon** tabanlı veya **JDK 5** sentakslı sınıflar üretilir.

Mysql Connector Dependency

```
1 <dependency>
2   <groupId>mysql</groupId>
3   <artifactId>mysql-connector-java</artifactId>
4   <version>5.1.6</version>
5 </dependency>
```