Hibernate.cfg.xml:

<!DOCTYPE hibernate-configuration PUBLIC

"-//Hibernate/Hibernate Configuration DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

<session-factory >

<!-- 数据库配置 -->

<property name="hibernate.connection.driver\_class">com.mysql.jdbc.Driver</property>

<property name="hibernate.connection.url">jdbc:mysql://localhost:3306/hibernateinheritmapping?serverTimezone=UTC</property>

<property name="hibernate.connection.username">root</property>

<property name="hibernate.connection.password">root</property>

<property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>

<!-- 配置自动建表 -->

<property name="hibernate.hbm2ddl.auto">update</property>

<!-- 加载所有映射 -->

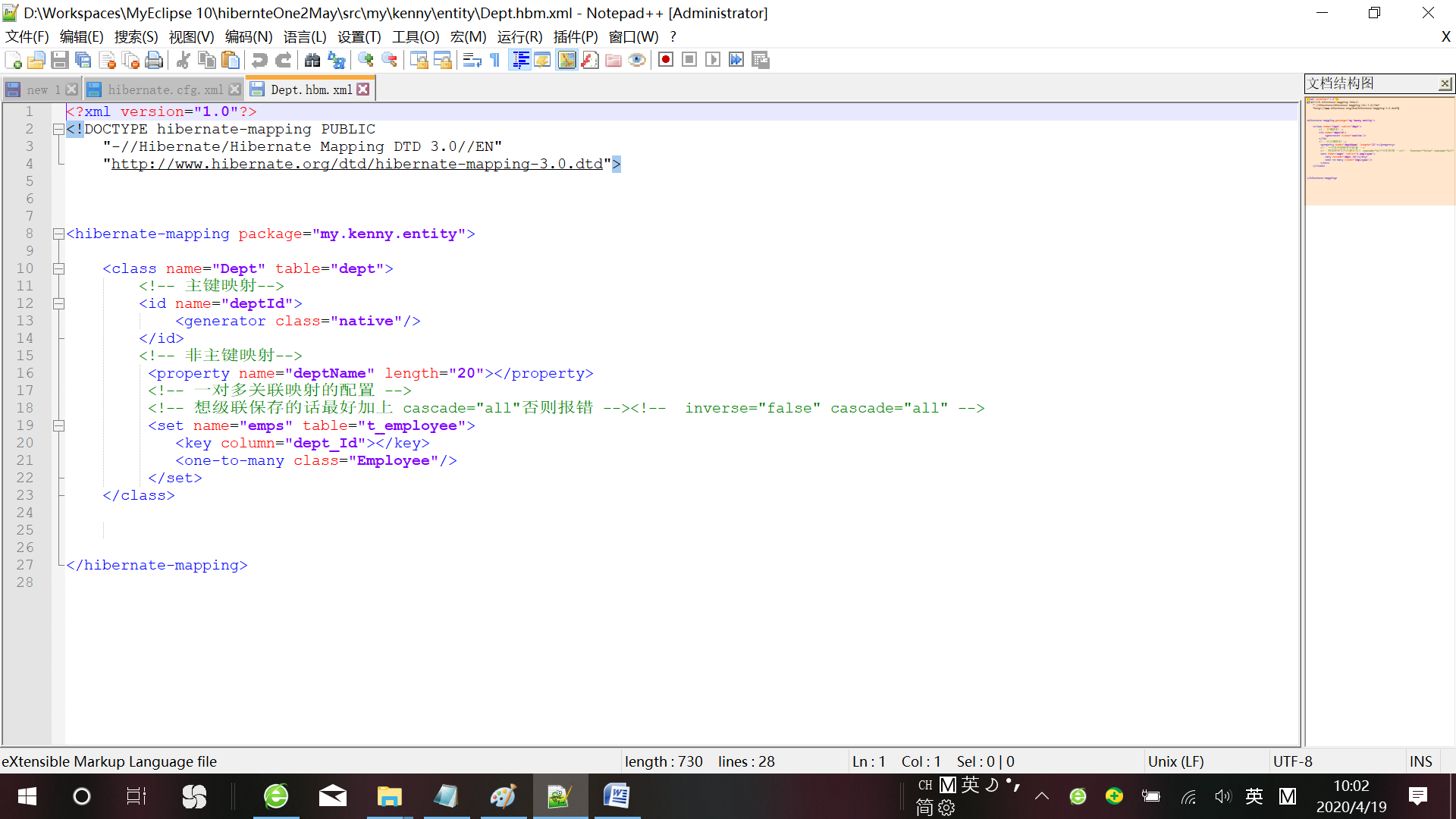
<mapping resource="my/kenny/entity/Cat.hbm.xml"/>

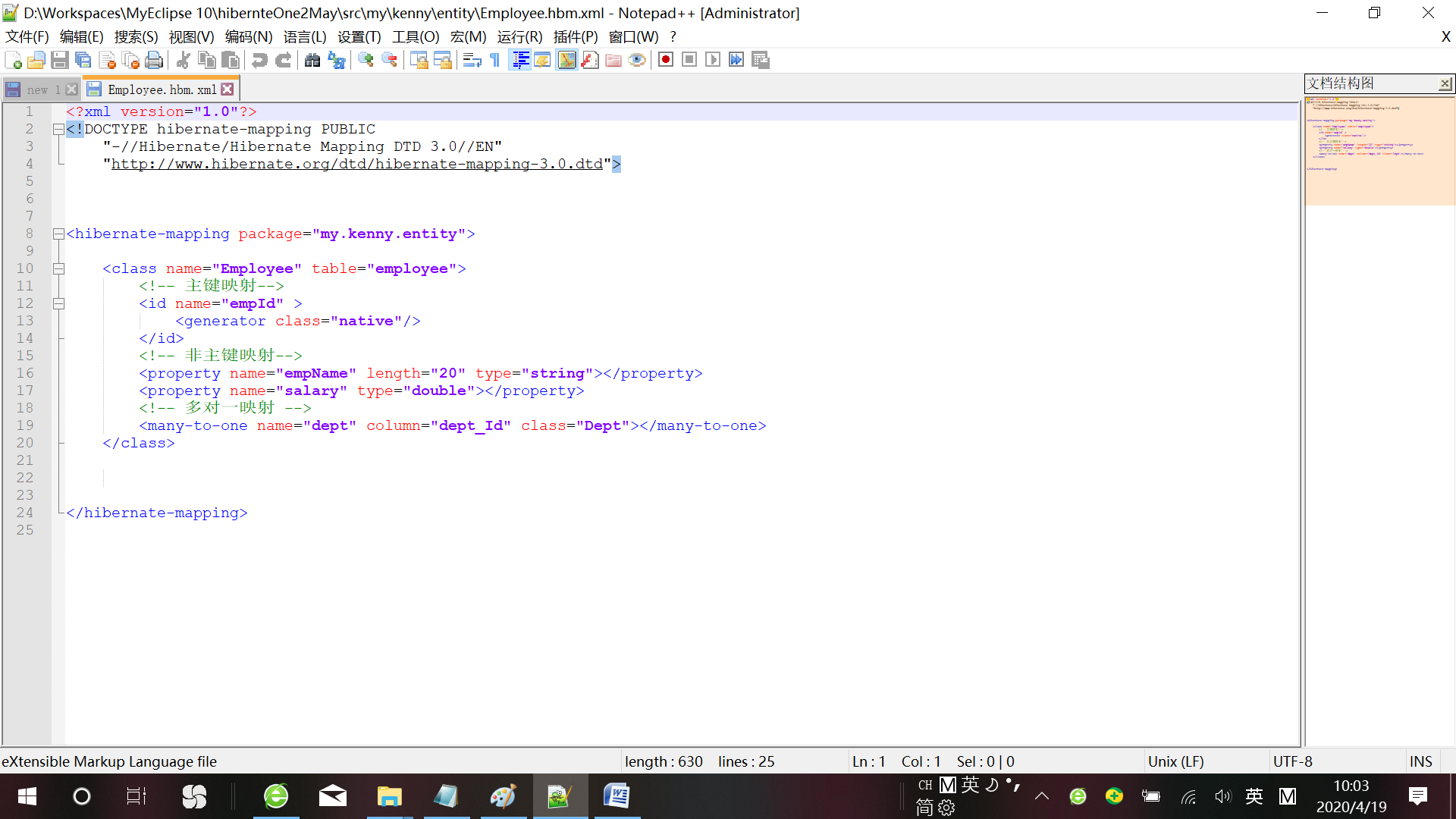
</session-factory>

</hibernate-configuration>

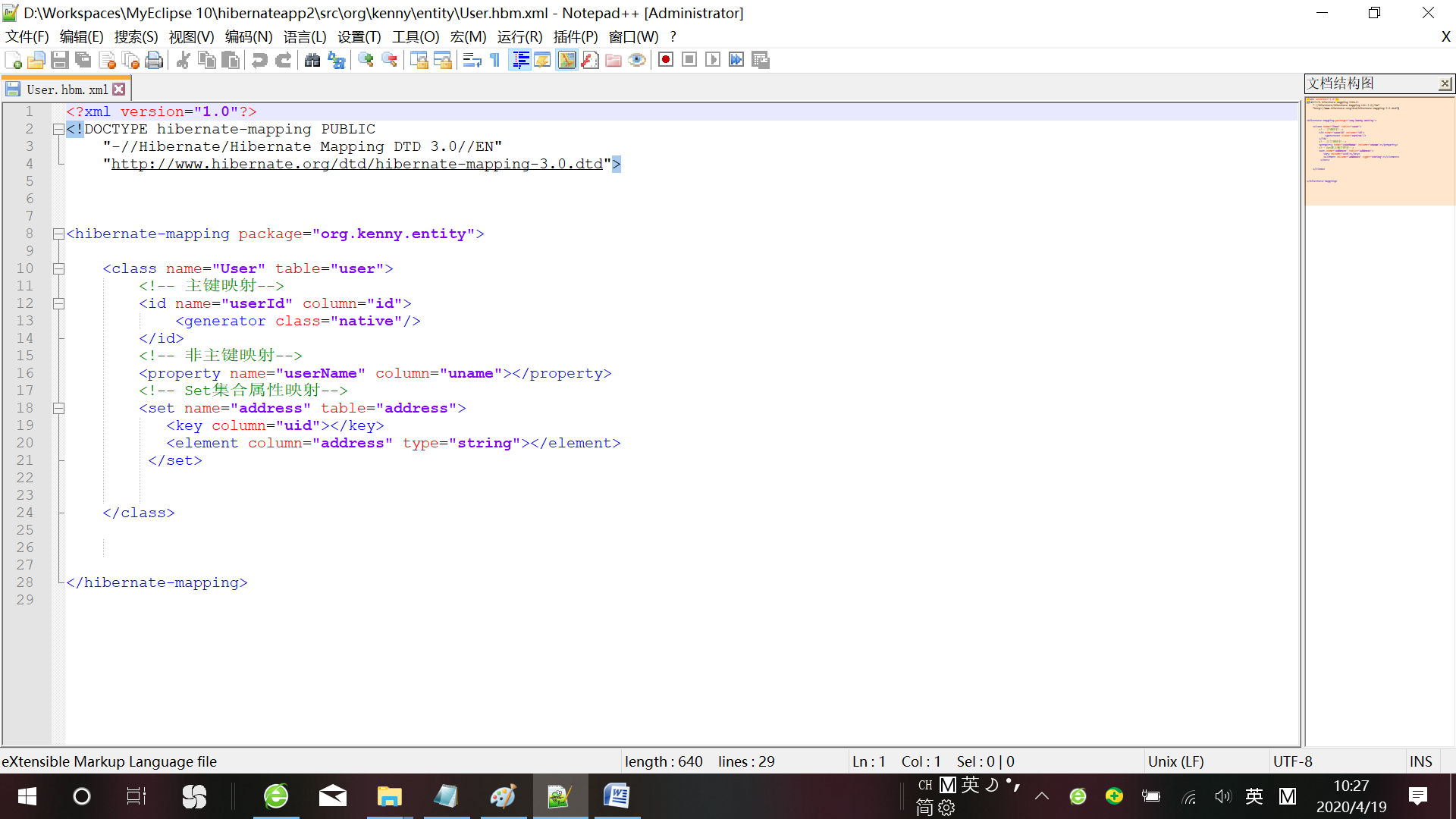
映射文件是个难点：这里以部门与员工转为案例

1. 一对多多对一

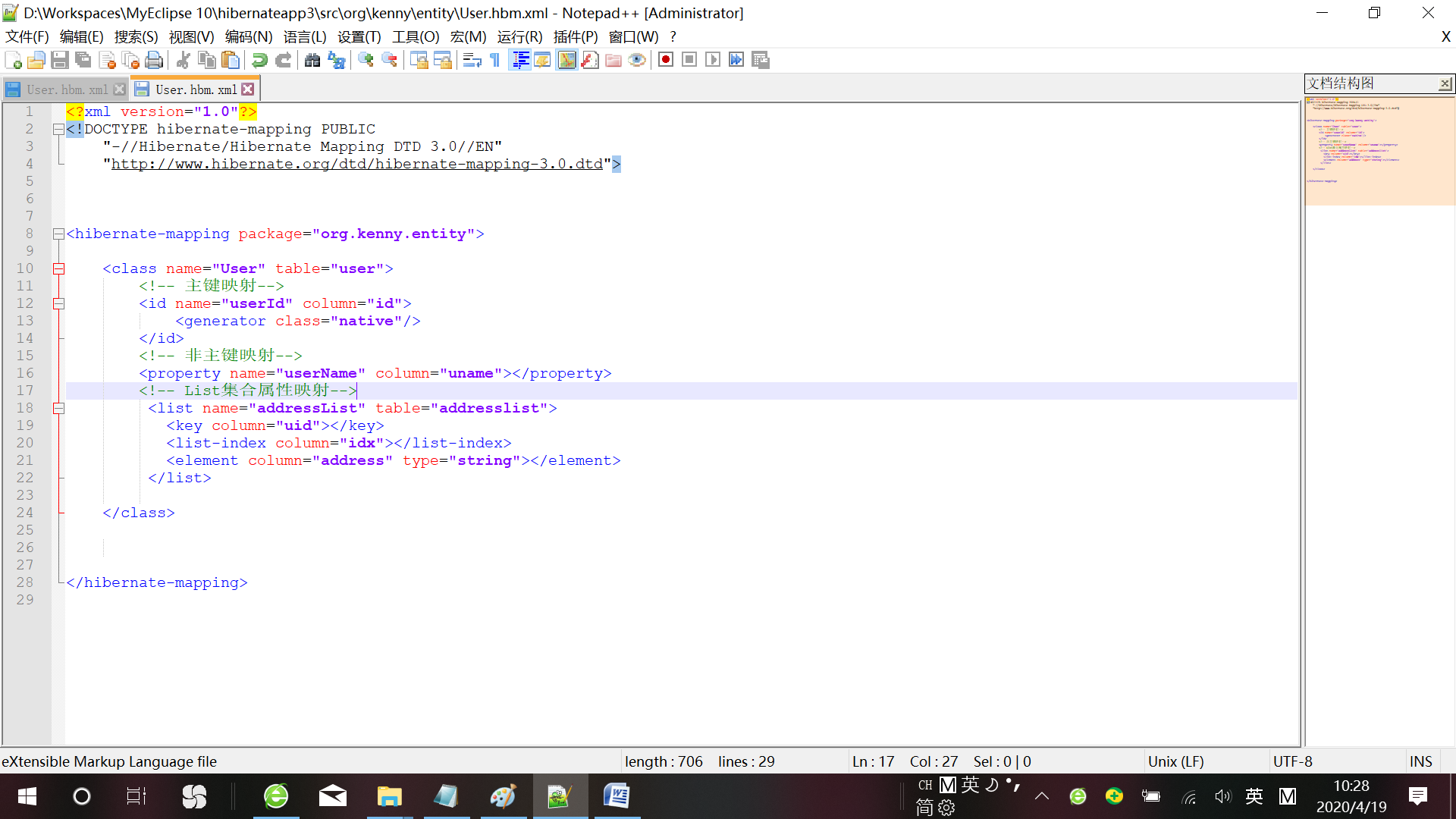




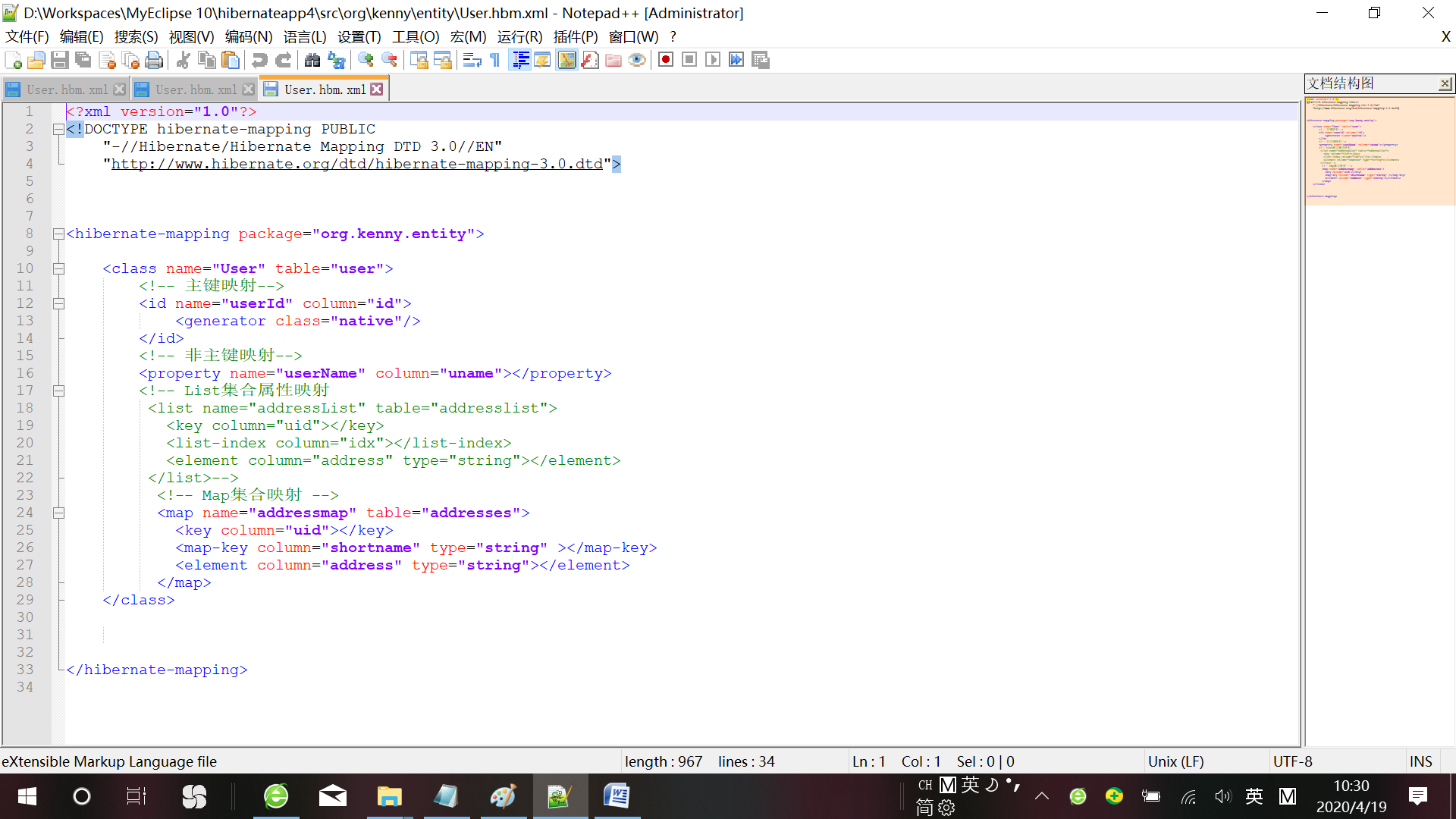
一对多中结合的映射（1）set



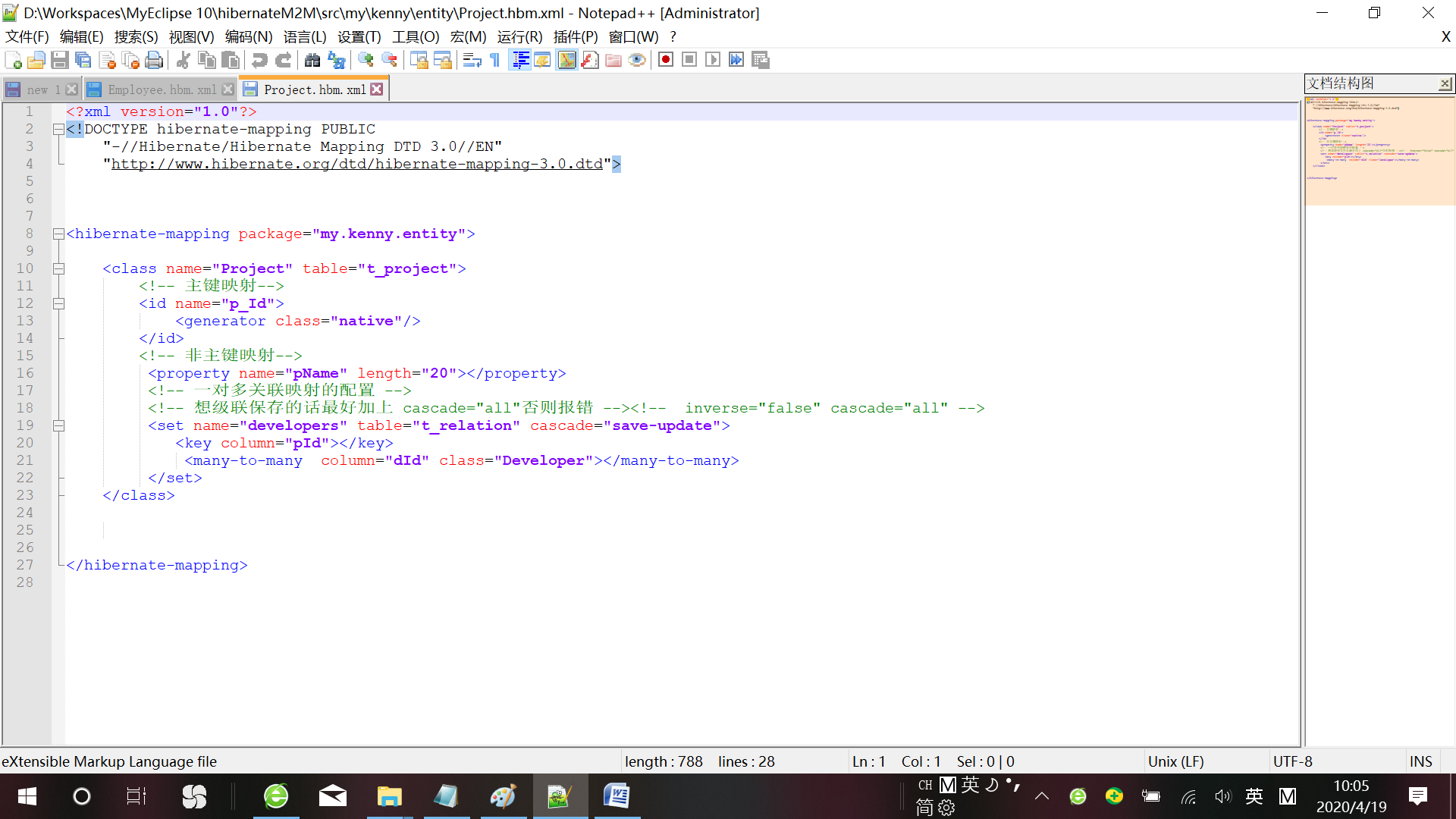
一对多中结合的映射（2）list集合

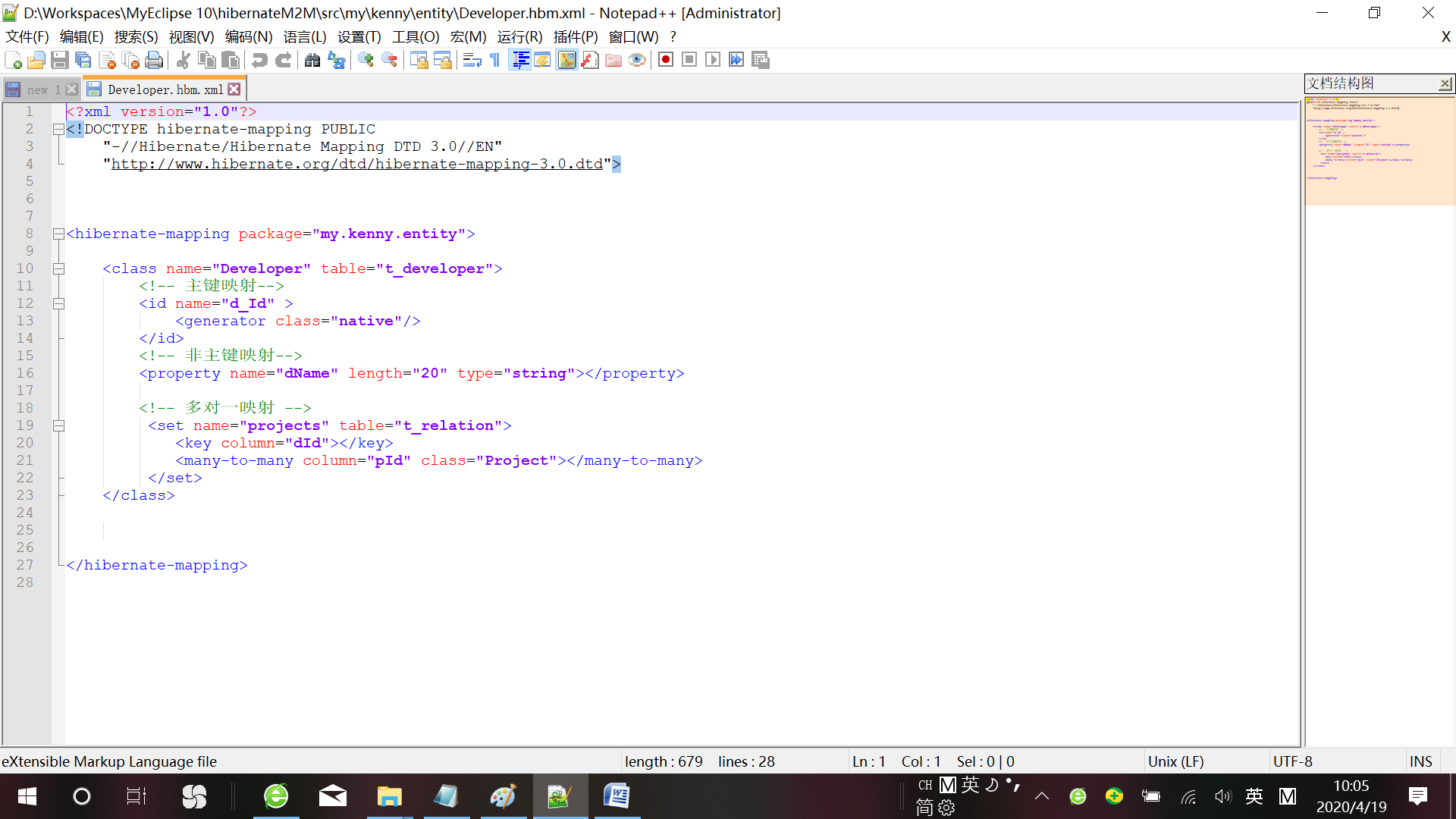


一对多中结合的映射（3）map集合

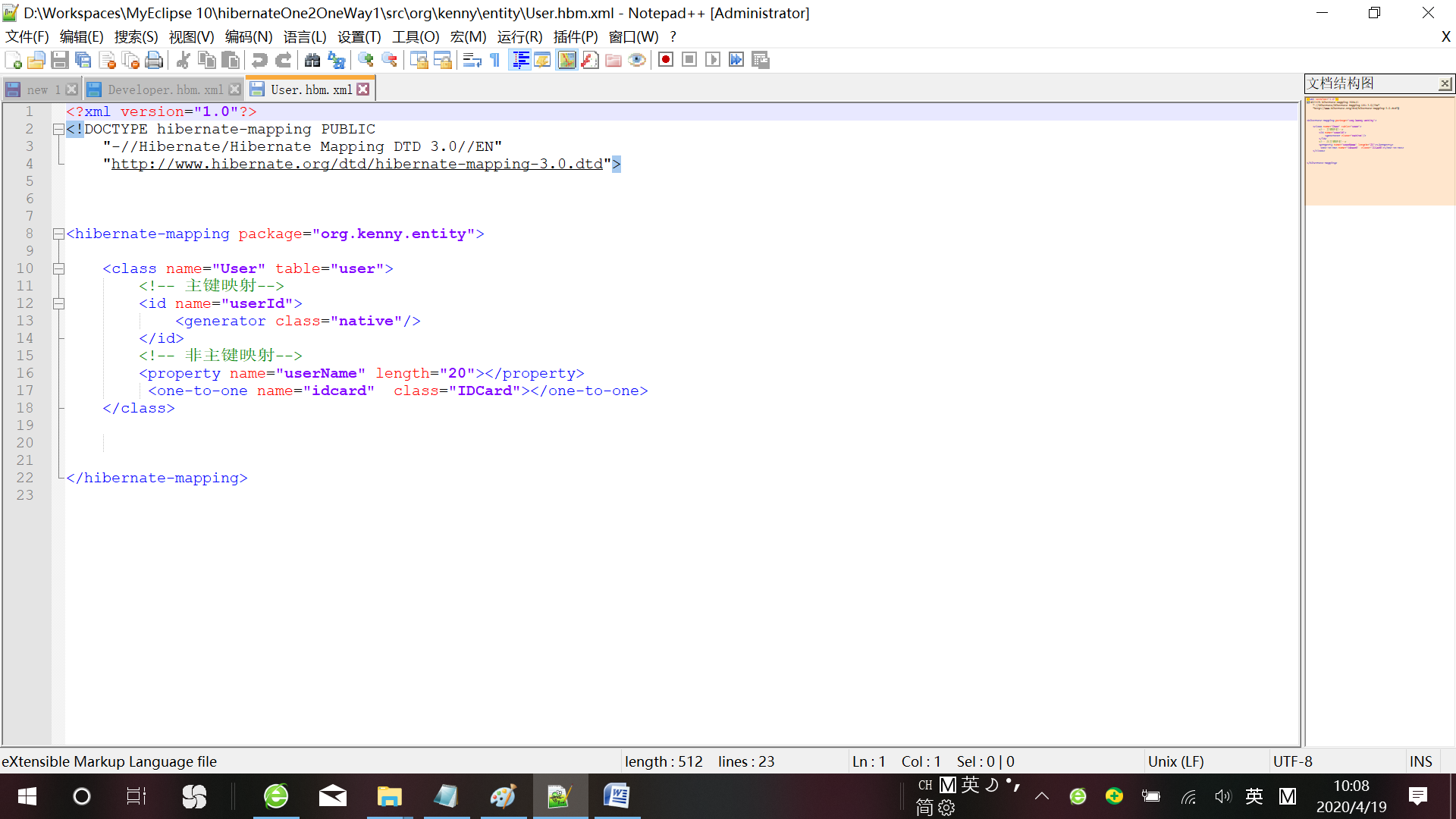


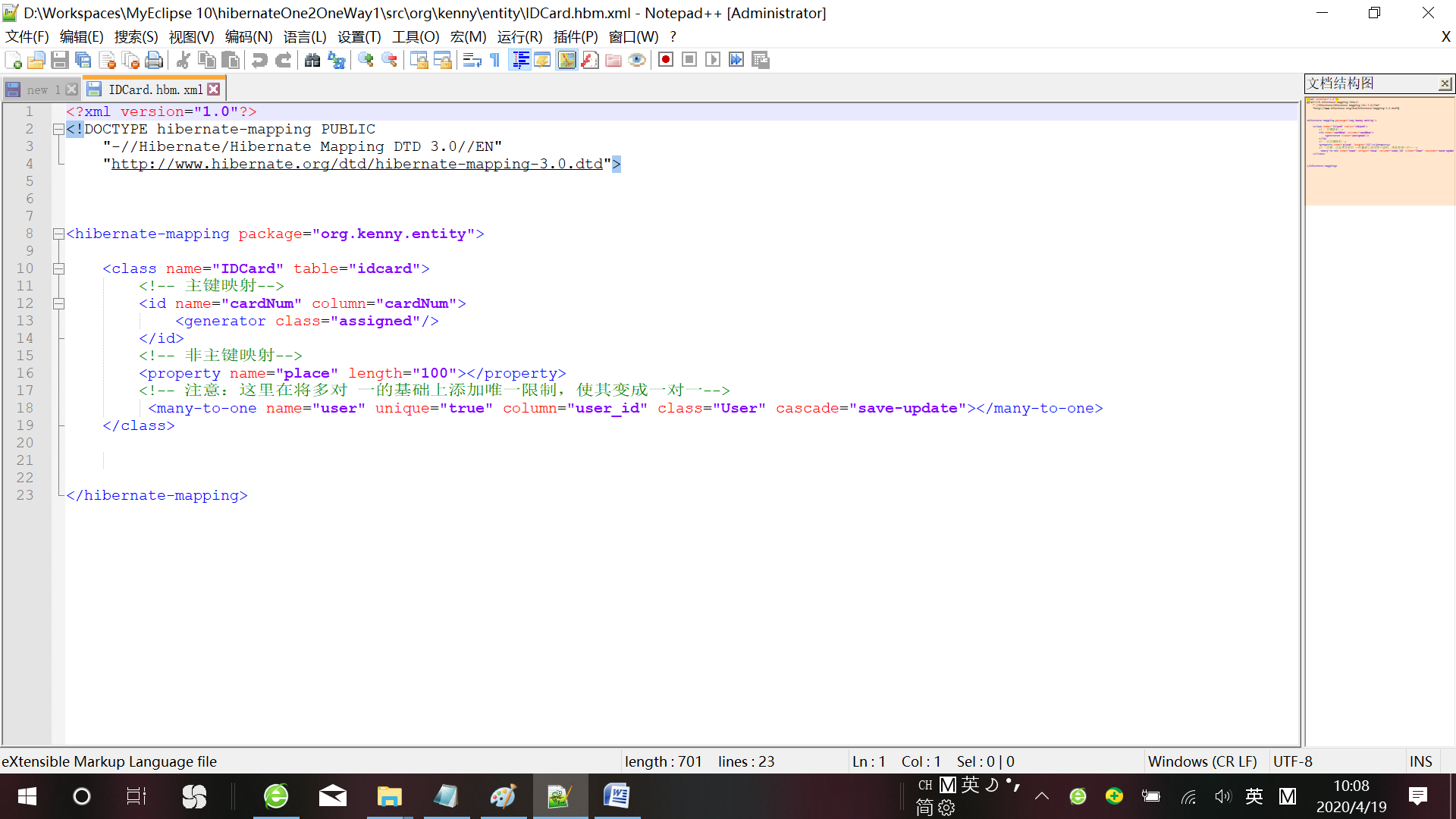
1. 多对多：案例：项目与开发人员



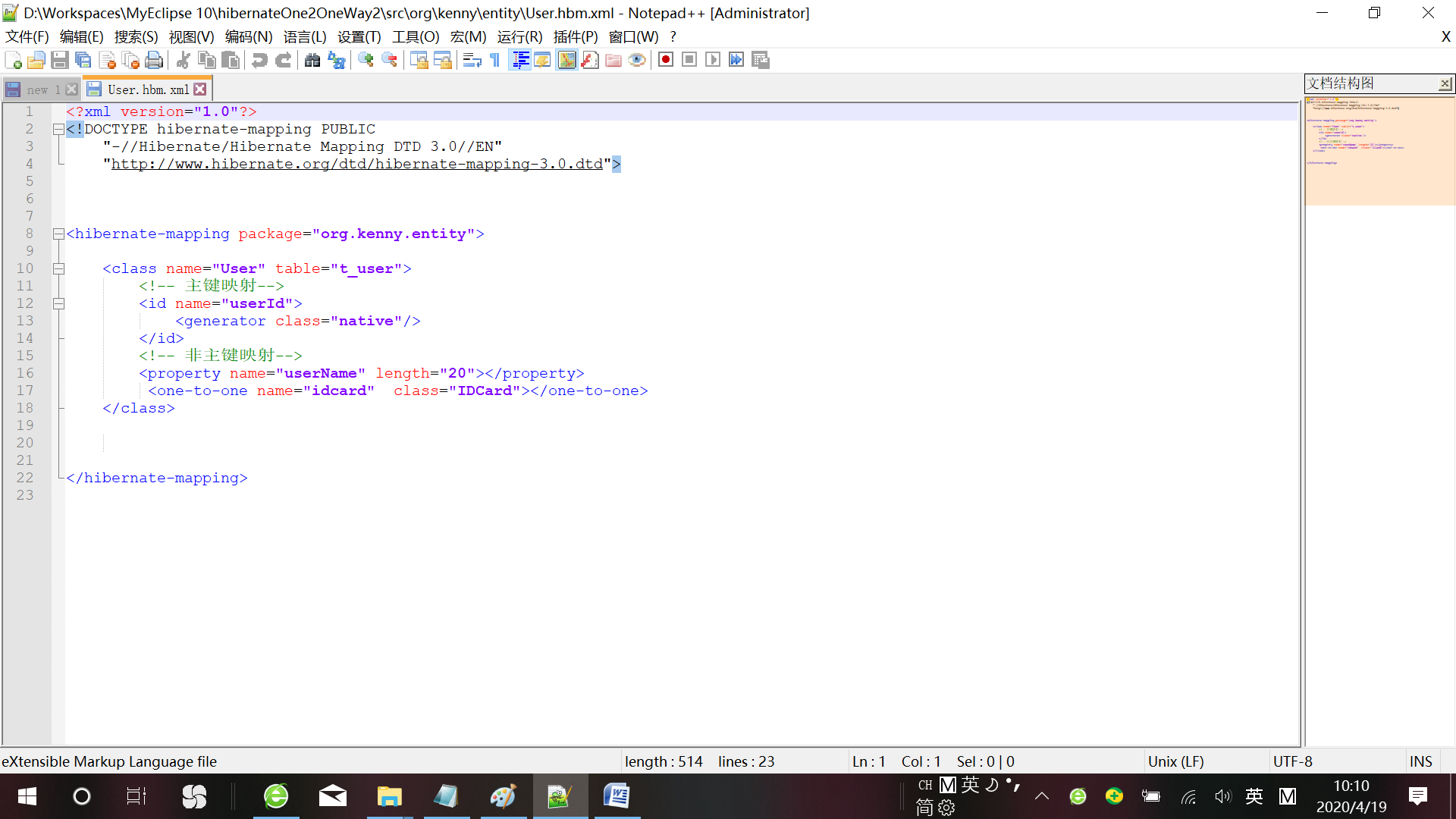


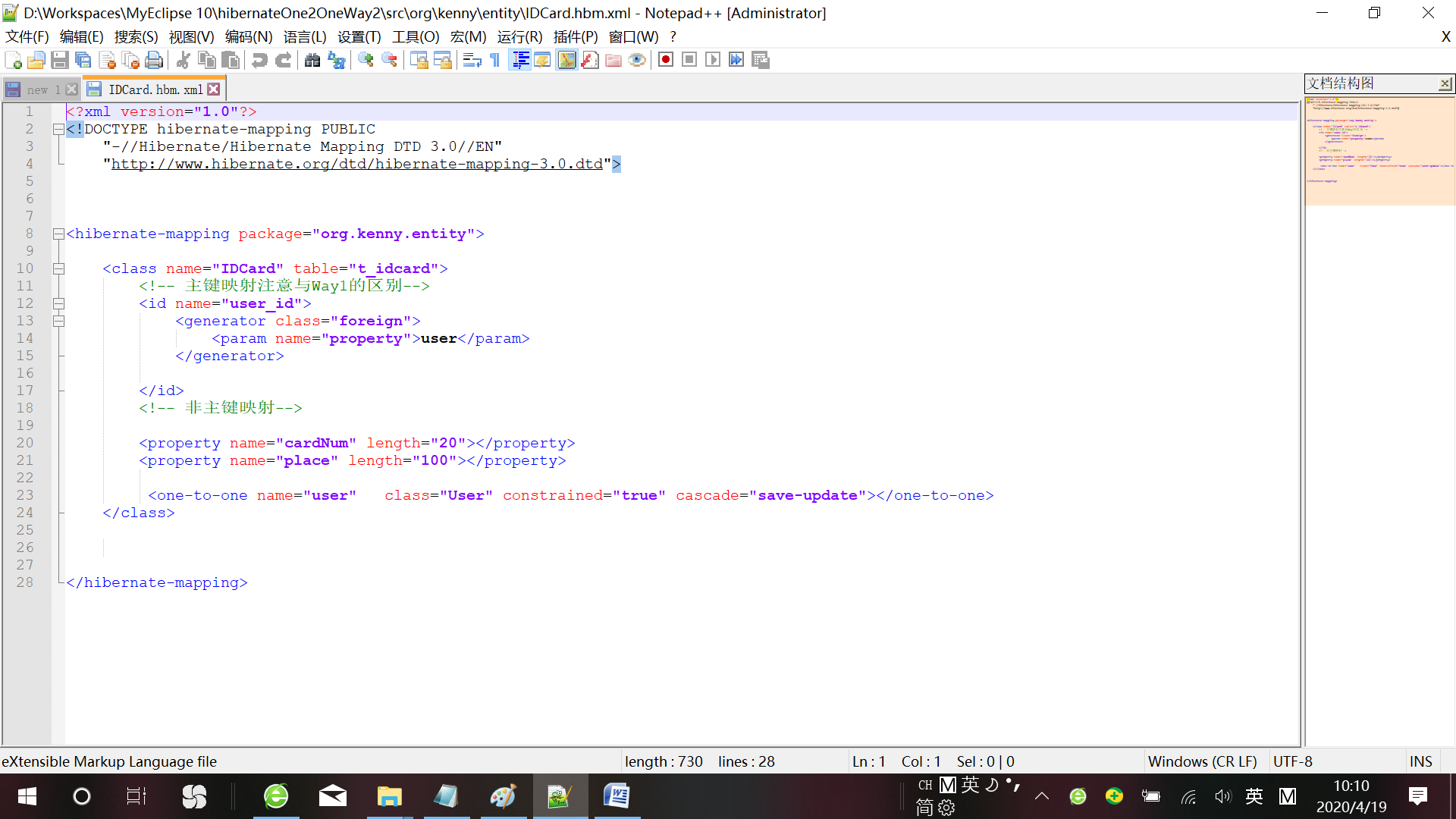
1. 一对一（有两种方式）方式一：案例：用户和身份证



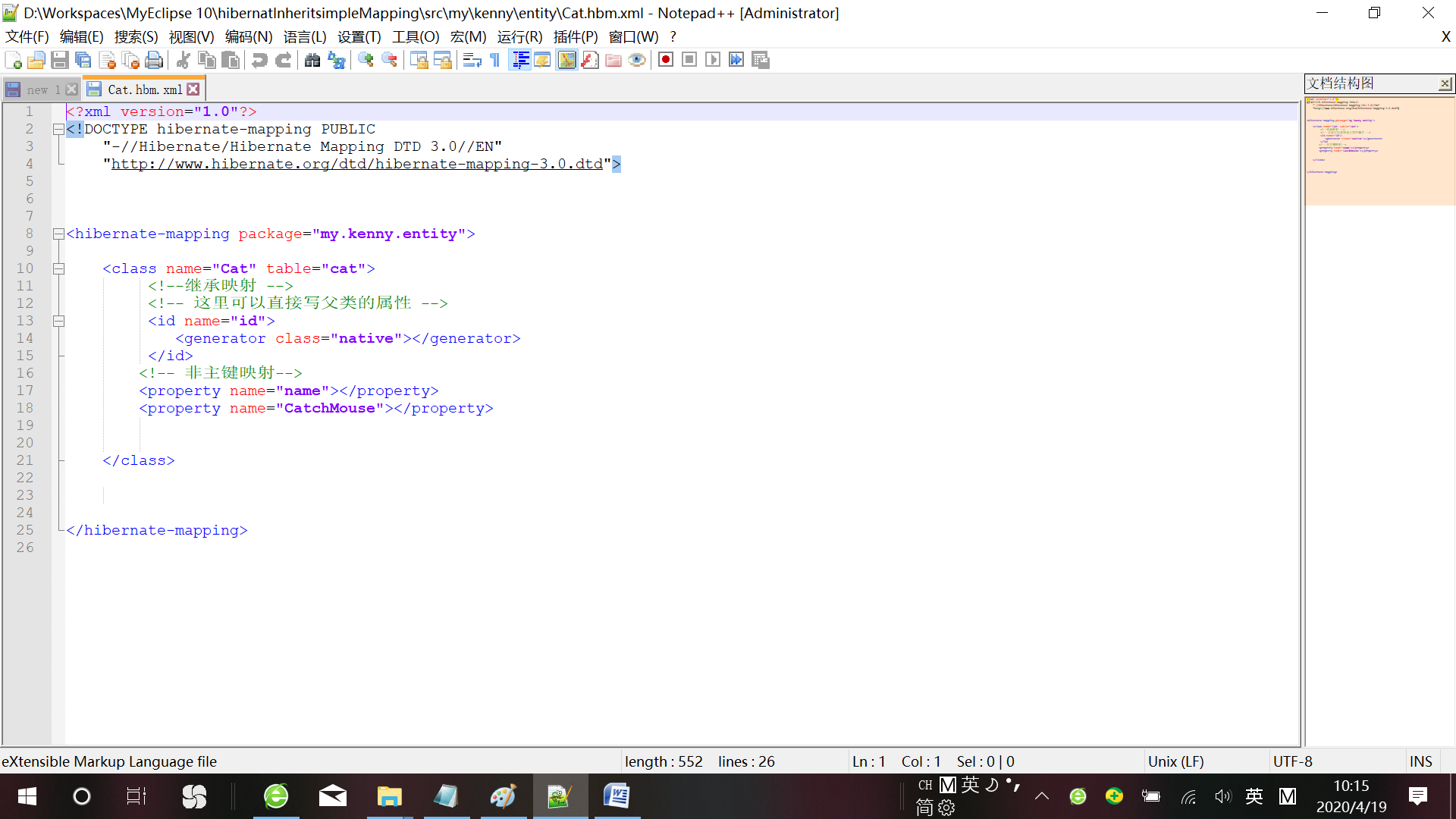


一对一方式（2）

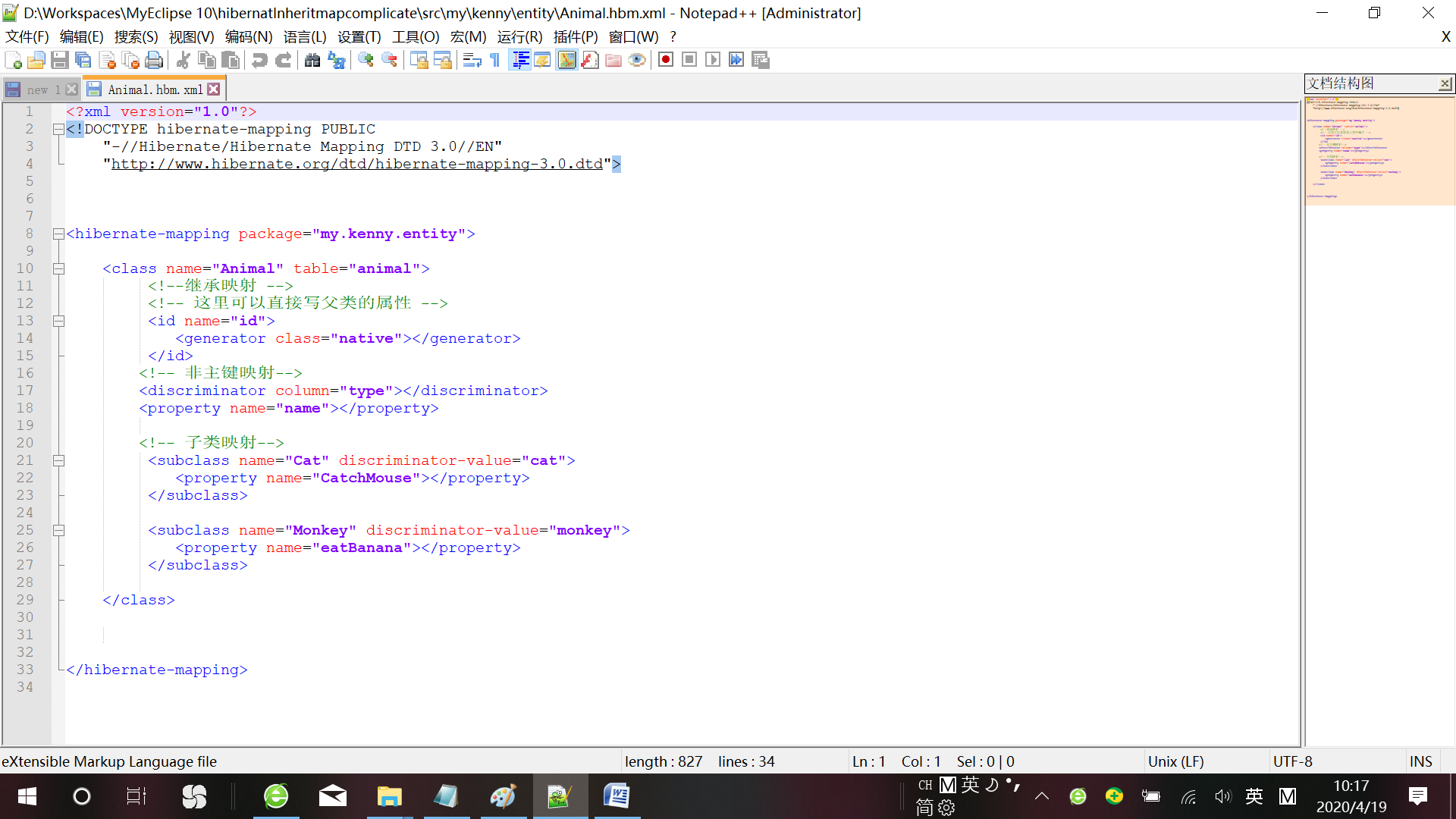




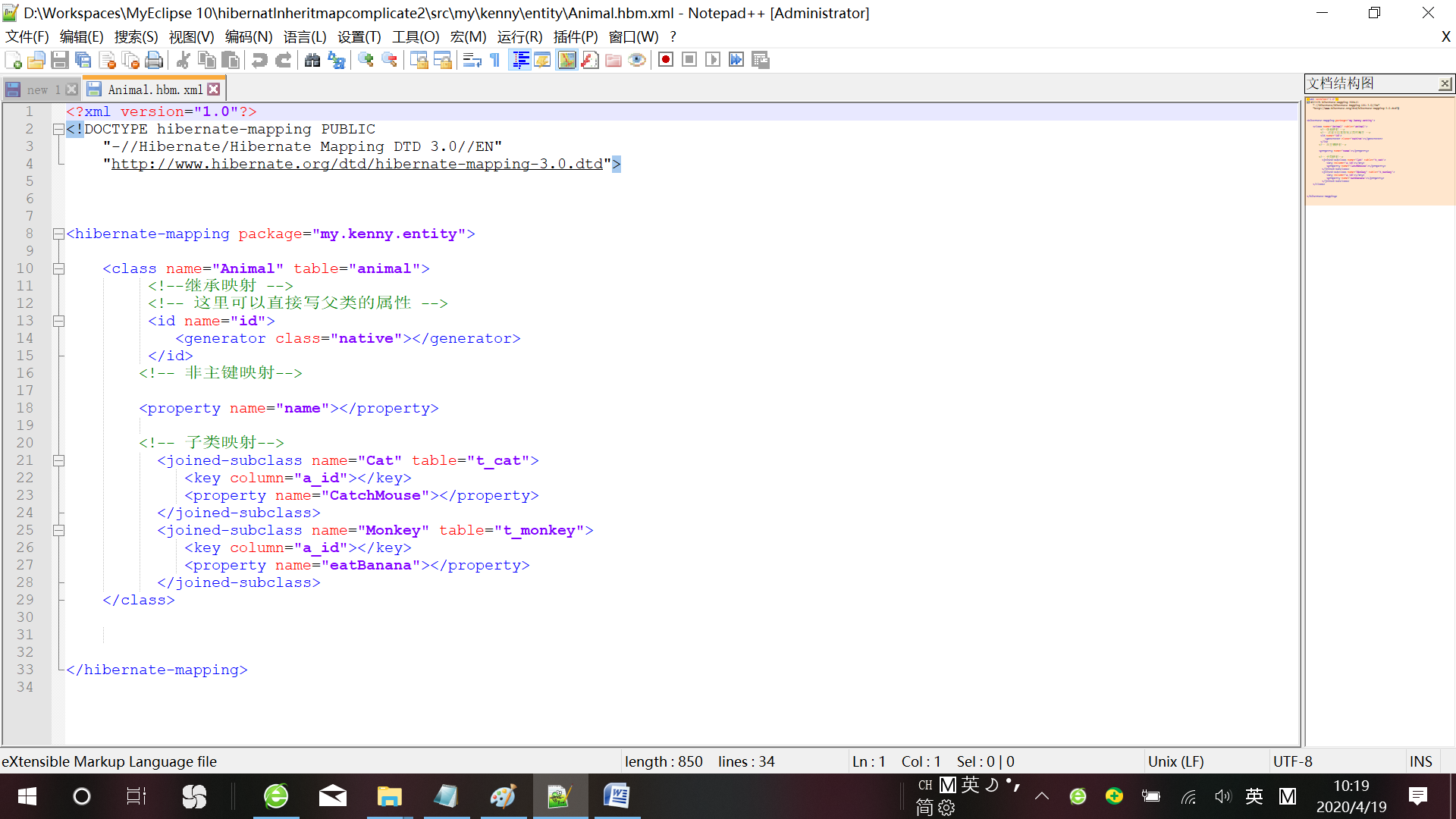
1. 简单继承映射：案例：动物与猫（父类不要映射）



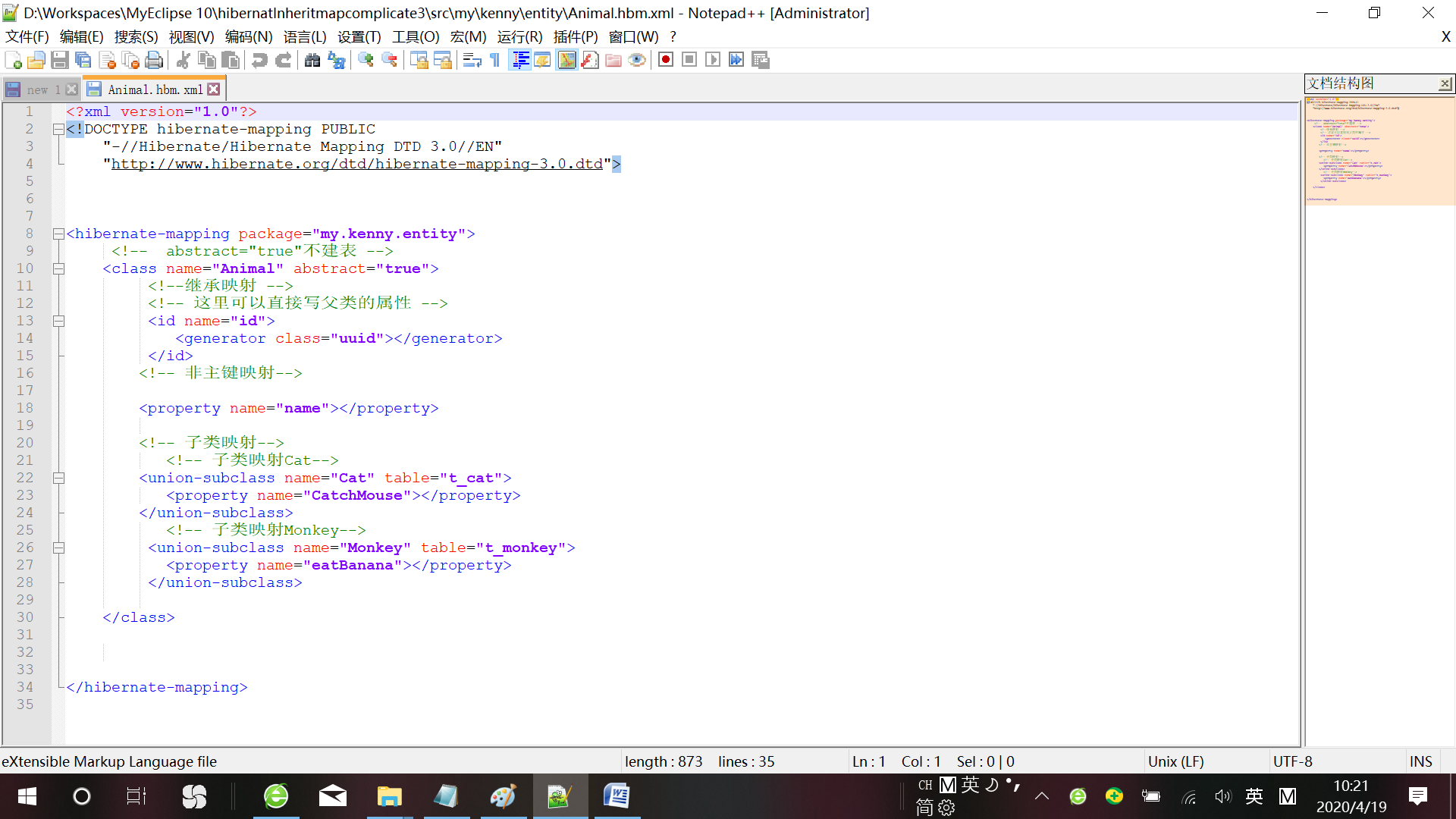
1. 复杂继承映射5.1.三个类只要一个配置文件和一个表，不太好



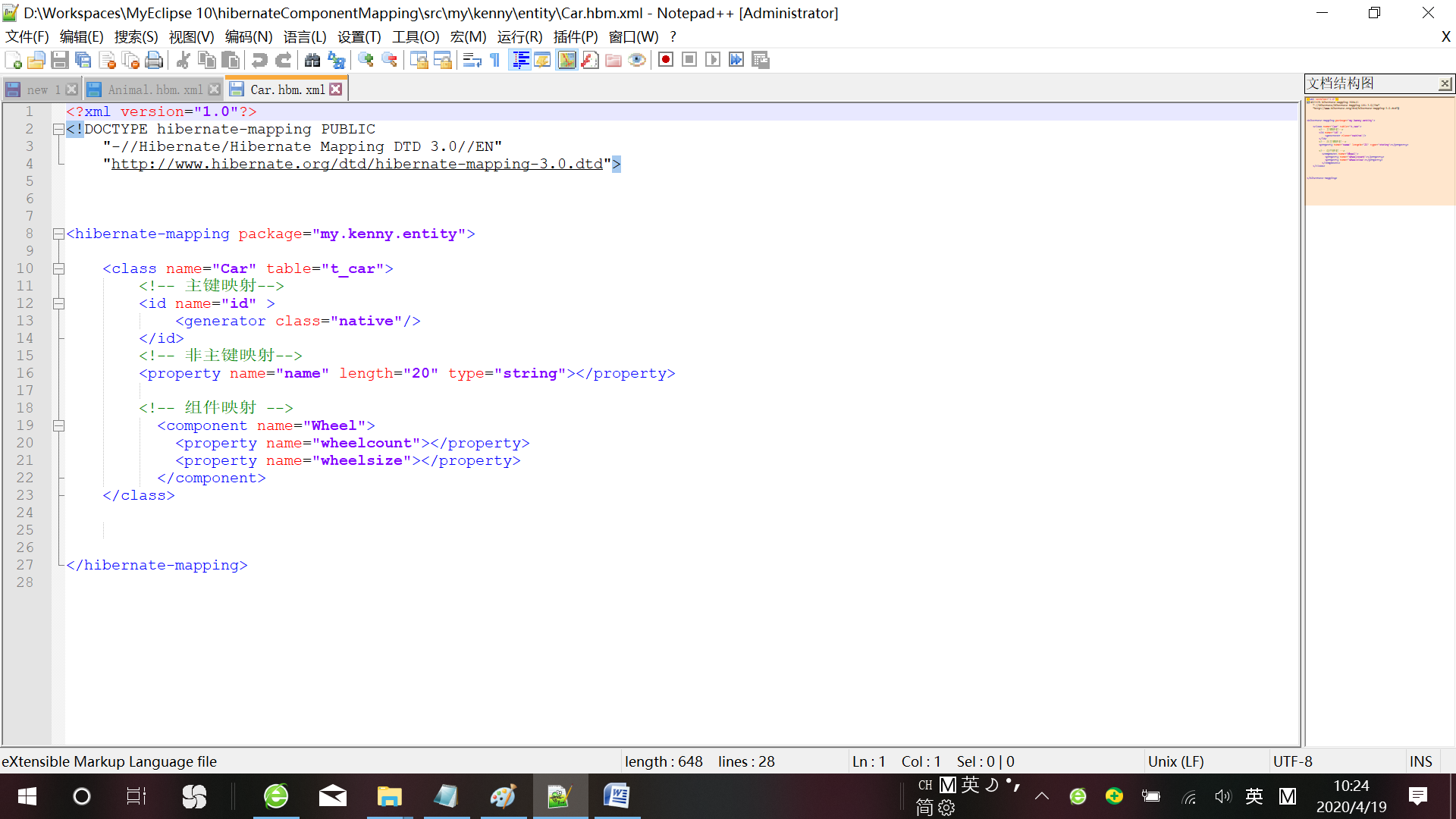
5.2三个类一个配置文件三张表



5.3三个类，一个配置文件，两张表，**这是推荐的用法**



1. 组件映射，以汽车和轮子为案例，在汽车配置文件在配置轮子



1. hibernate HQL相关映射配置

hibernate.cfg.xml:

<!DOCTYPE hibernate-configuration PUBLIC

"-//Hibernate/Hibernate Configuration DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

<session-factory >

<!-- 数据库配置 -->

<property name="hibernate.connection.driver\_class">com.mysql.jdbc.Driver</property>

<property name="hibernate.connection.url">jdbc:mysql://localhost:3306/empresa?serverTimezone=UTC</property>

<property name="hibernate.connection.username">root</property>

<property name="hibernate.connection.password">root</property>

<property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>

<!-- 配置自动建表 -->

<property name="hibernate.hbm2ddl.auto">update</property>

<!-- 连接池配置 -->

<!-- 先配置 启动管理类-->

<property name="hibernate.connection.provider\_class">org.hibernate.connection.C3P0ConnectionProvider</property>

<!-- 配置连接池参数信息-->

<property name="hibernate.c3p0.min\_size">2</property>

<property name="hibernate.c3p0.max\_size">6</property>

<property name="hibernate.c3p0.timeout">5000</property>

<property name="hibernate.c3p0.max\_statements">10</property>

<property name="hibernate.c3p0.acquire\_increment">2</property>

<property name="hibernate.c3p0.idle\_test\_period">3000</property>

<property name="hibernate.show\_sql">true</property>

<!-- 二级缓存配置 -->

<!--开启二级缓存 -->

<property name="hibernate.cache.use\_second\_level\_cache">true</property>

<!--开启查询缓存 -->

<property name="hibernate.cache.use\_query\_cache">true</property>

<!--指定使用哪一个二级缓存框架 -->

<property name="hibernate.cache.provider\_class">org.hibernate.cache.HashtableCacheProvider</property>

<!-- 加载所有映射 -->

<mapping resource="my/kenny/entity/Employee.hbm.xml"/>

<mapping resource="my/kenny/entity/Dept.hbm.xml"/>

<!--指定哪一些类要放入二级缓存 -->

<class-cache usage="read-write" class="my.kenny.entity.Employee"/>

<class-cache usage="read-write" class="my.kenny.entity.Dept"/>

<!--集合缓存 -->

<collection-cache usage="read-write" collection="my.kenny.entity.Dept.emps"/>

</session-factory>

</hibernate-configuration>

