

(查看hibernate源码:

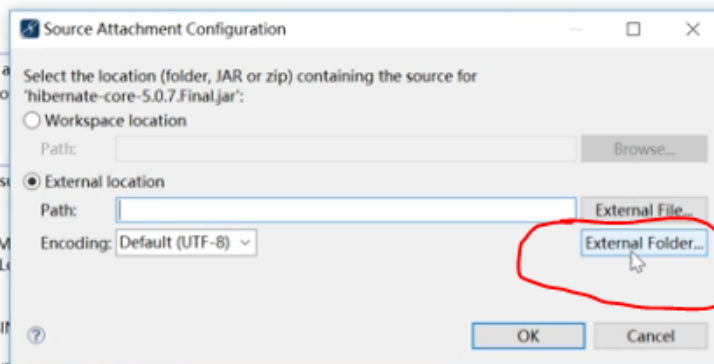
Class File Editor

Source not found

The JAR file hibernate-core-5.0.7.Final.jar has no source a
You can attach the source by clicking Attach Source below

Attach Source...

```
// Compiled from Configuration.java (version 1.6 : 50.0, s  
public class org.hibernate.cfg.Configuration {  
  
// Field descriptor #141 Lorg/hibernate/internal/CoreM  
private static final org.hibernate.internal.CoreMessageL  
  
// Field descriptor #143 Ljava/lang/String;  
public static final java.lang.String ARTEFACT_PROCESSING  
  
// Field descriptor #147 Lorg/hibernate/boot/registry/BootstrapServiceRegistry;  
private final org.hibernate.boot.registry.BootstrapServiceRegistry bootstrapServiceRegistry;  
  
// Field descriptor #149 Lorg/hibernate/boot/MetadataSources;  
private final org.hibernate.boot.MetadataSources metadataSources;  
  
// Field descriptor #151 Lorg/hibernate/boot/model/naming/ImplicitNamingStrategy;  
private org.hibernate.boot.model.naming.ImplicitNamingStrategy implicitNamingStrategy;  
  
// Field descriptor #153 Lorg/hibernate/boot/model/naming/PhysicalNamingStrategy;  
private org.hibernate.boot.model.naming.PhysicalNamingStrategy physicalNamingStrategy;
```



然后找到下载的hibernate解压包，进入project然后点击OK即可)

hibernate的核心API一共有6个。

①Configuration类：对hibernate进行配置，以及对它进行启动。在hibernate启动过程中，configuration类的实例首先定位映射文件的位置，读取这些配置，然后创建一个sessionfactory对象，虽然configuration类在hibernate项目中只是一个很小的角色，到那时它是hibernate启动时的第一个对象。configuration的作用启动hibernate程序，加载hibernate.cfg.xml配置文件。

注意事项：

a，通常情况下，configuration对象只会创建一个对象，即configuration是单例的。

②SessionFactory接口：（工厂模式）

SessionFactory接口的作用：加载连接数据库、扩展参数、映射信息，通过这些映射信息，帮助我们创建Session对象。

注意事项：

a，通常一个项目只需要创建一个SessionFactory即可。

③Session接口：

Session接口的作用：用于操作对象，从而操作数据库。

session的常用方法：

save ()：保存一个对象；

update ()：更新一个对象；

`delete ()` : 删除一个对象;

`get ()` : 查询一个对象。

注意事项:

a, `session`是一个非线程安全的对象, 可能会产生线程并发的问题, 在项目中需要创建多个`session`对象, 即一个线程就创建一个`session`对象。

④`Transaction`接口:

`Transaction`接口的作用: 用于执行事务的操作。主要是`commit`和`rollback`。

`Transaction`的方法:

a, `commit ()` : 提交事务;

b, `rollback ()` : 回滚事务;

⑤`Query`接口: 用于执行HQL查询

⑥`Criteria`接口: 用于执行基于对象的查询 (QBC查询)