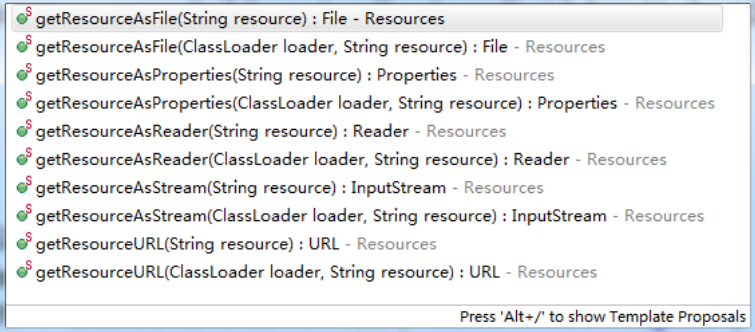
# API详解

Dao中需要***通过SqlSession对象来操作DB***。而SqlSession对象的创建，需要SqlSessionFactory（工厂对象）。***SqlSessionFactory对象***需要通过其构建器对象***SqlSessionFactoryBuilder的build()***方法，在加载了主配置文件的输入流对象后创建。



###### Resources类

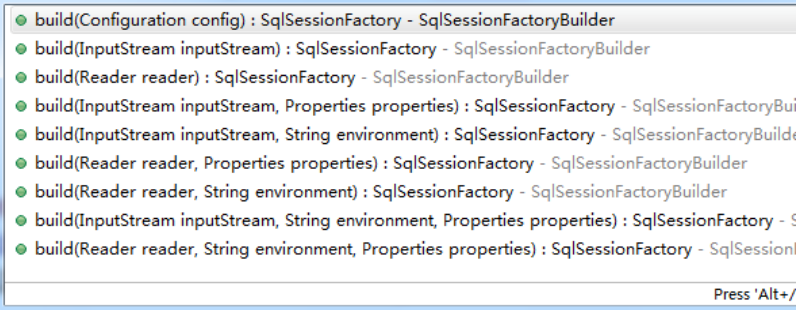
Resources类，用于读取资源文件。其有很多方法通过加载并解析资源文件，返回不同类型的IO流对象。



###### SqlSessionFactoryBuilder类

SqlSessionFactory的创建，需要使用SqlSessionFactoryBuilder对象在创建完工厂对象后，就完成了其历史使命，即可被销毁。所以，一般会将SqlSessionFactoryBuilder对象创建为一个方法内的局部对象，方法结束，对象销毁。

其被重载的build()方法较多：



###### SqlSessionFactory接口

SqlSessionFactory接口对象是一个重量级对象（系统开销大的对象），是线程安全的，所以一个应用只需一个该对象即可。创建SqlSession需要使用SqlSessionFactory接口的openSession()方法。

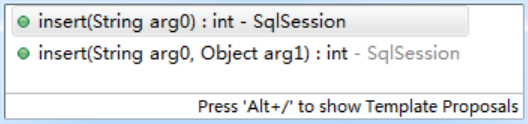
* openSession(true)：创建一个有自动提交功能的SqlSession
* openSession(false)：创建一个非自动提交功能的SqlSession，，需手动提交
* openSession()：同openSession(false)

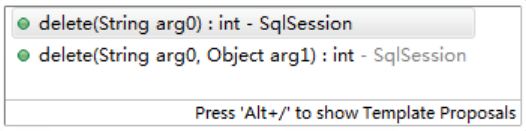
###### SqlSession接口

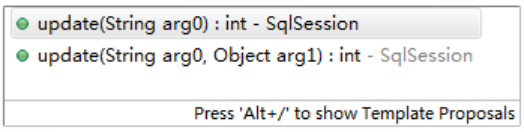
SqlSession接口对象用于执行持久化操作。一个SqlSession对应着一次数据库会话，一次会话以SqlSession对象的创建开始，以SqlSession对象的关闭结束。

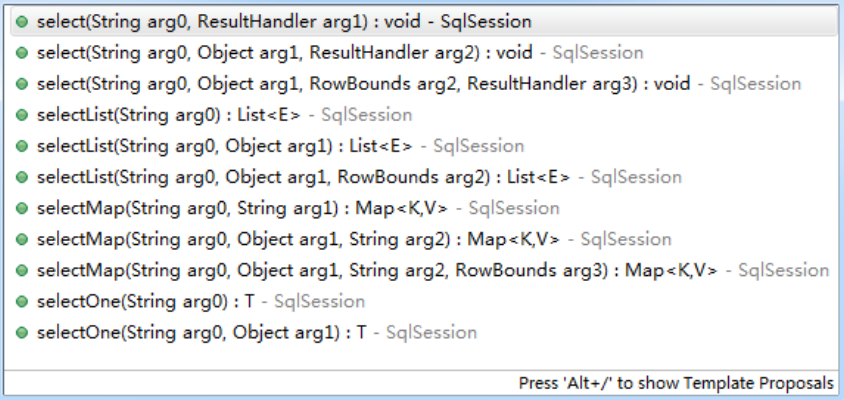
SqlSession接口对象是线程不安全的，所以每次数据库会话结束前，需要马上调用其close()方法，将其关闭。再次需要会话，再次创建。而在关闭时会判断当前的SqlSession是否被提交：若没有被提交，则会执行回滚后关闭；若已被提交，则直接将SqlSession关闭。所以，SqlSession无需手工回滚。

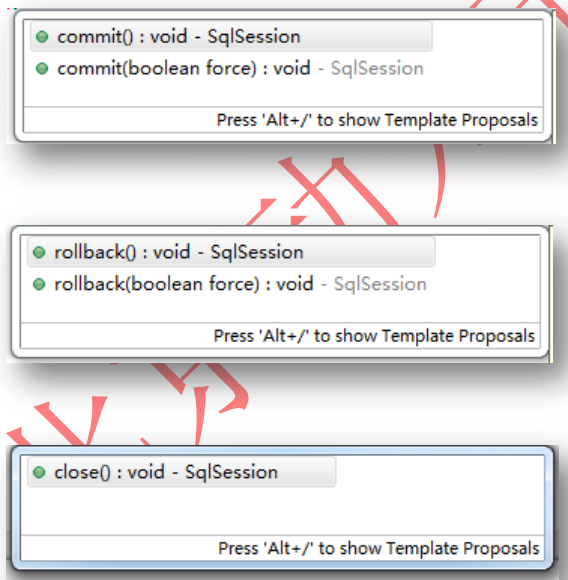
SqlSession接口常用的方法有：











###### 源码分析