

properties元素

properties是一个配置属性的元素，让用户可以在配置文件上下文中使用。mybatis提供了3种方式来配置

property子元素

```
1 <properties>
2   <property name="" value=""/>
3 </properties>
```

properties配置文件

```
1 <properties resource="jdbc.properties"/>
```

程序参数传递

其实这种方式一般会和第二种配合使用，作用对特殊配置进行覆盖或重写，以上面的jdbc.properties为例，在使用到数据库配置密

```
1 public static SqlSessionFactory getSqlSessionFactoryByXml() {
2     synchronized (Lock) {
3         if (null != sqlSessionFactory) {
4             return sqlSessionFactory;
5         }
6         String resource = "mybatis-config.xml";
7         InputStream inputStream;
8         InputStream is = null;
9         try {
10             // 加载数据库配置文件
11             is = Resources.getResourceAsStream("jdbc.properties");
12             Properties properties = new Properties();
13             properties.load(is);
14
15             // 获取加密信息
16             String userName = properties.getProperty("jdbc.username");
17             String pwd = properties.getProperty("jdbc.password");
18
19             // 解密用户名和密码，并重置属性
20             properties.setProperty("jdbc.username", CyperTool.decodeByBase64(userName));
21             properties.setProperty("jdbc.password", CyperTool.decodeByBase64(pwd));
22             // 读取mybatis配置文件
23             inputStream = Resources.getResourceAsStream(resource);
24             // 通过SqlSessionFactoryBuilder类的builder方法进行构建，并使用程序传递的方式覆盖原有属性
25             sqlSessionFactory = new SqlSessionFactoryBuilder().build(inputStream, properties);
26         } catch (IOException e) {
27             e.printStackTrace();
28             return null;
29         }
30         return sqlSessionFactory;
31     }
32 }
```

setting元素