

代替mybatis-config.xml，该方法返回SqlSessionFactory，我们就可以获取SqlSession了

```
1  SqlSessionFactory mybatisConfig(String driver, String url, String username,
2      String password) throws SQLException {
3      // 获取DataSource, DataSource接口有多个实现类，我们用的是mybatis给我们提供的PooledDataSource
4      PooledDataSource pooledDataSource = new PooledDataSource(driver, url,
5          username, password);
6      // 若需要配置pooledDataSource, 则可以用他的set方法, 如
7      pooledDataSource.setLoginTimeout(6000);
8      DataSource dataSource = pooledDataSource;
9
10     // 配置事务管理, 这里我们使用JDBC的事务
11     TransactionFactory trcFactory = new JdbcTransactionFactory();
12     // 配置Environment对象, "development"是我们给起的名字
13     Environment env = new Environment("development", trcFactory, dataSource);
14     // 创建Configuration对象
15     Configuration config = new Configuration(env);
16     //<settings></settings>中的内容在此处配置
17     config.setLazyLoadingEnabled(true);
18
19     // 起别名
20     TypeAliasRegistry aliases = config.getTypeAliasRegistry();
21     aliases.registerAlias("bean", TestBean.class);
22
23     // 映射接口, 若有xml文件, 则xml的文件应和接口文件同名
24     config.addMapper(DaoMapper.class);
25
26     SqlSessionFactory sqlSessionFactory =
27         new SqlSessionFactoryBuilder().build(config);
28     return sqlSessionFactory;
29 }
```

解析1：配置DataSource

```
1
2  PooledDataSource pooledDataSource = new PooledDataSource(driver, url,username, password);
3
4  对应
5
6  <dataSource type="POOLED">
7      <property name="driver" value="{driver}" />
8      <property name="url" value="{url}" />
9      <property name="username" value="{username}" />
10     <property name="password" value="{password}" />
11 </dataSource>
```

解析2：配置JDBC事务

```
1  TransactionFactory trcFactory = new JdbcTransactionFactory();
2
3  对应
4
5  <transactionManager type="JDBC" />
```

解析3：配置Environment

```
1  Environment env = new Environment("development", trcFactory, dataSource);
2
3  对应
4
5  <environment id="development">
6      <transactionManager type="JDBC" />
7      <dataSource type="POOLED"></dataSource>
8  </environment>
```

解析4：装入Configuration

```
1  Configuration config = new Configuration(env);
2  对应
3  <configuration>
4      <environments default="development">
5          <environment id="development">
6              </environment>
7          </environments>
8      </configuration>
9
10  -----
11  config.setLazyLoadingEnabled(true);
12  对应
13  <settings>
14      <setting name="lazyLoadingEnabled" value="true"/>
15  </settings>
```

解析5：配置别名

```
1  TypeAliasRegistry aliases = config.getTypeAliasRegistry();
2  aliases.registerAlias("bean", TestBean.class);
3
4  对应
5
6  <typeAliases>
7      <typeAlias alias="bean" type="com.amiu.TestBean"/>
8  </typeAliases>
```

解析6：添加映射XML，xml名称应和DaoMapper.class并且在同一个包下

```
1  config.addMapper(DaoMapper.class);
2
3  对应
4
5  <mappers>
6      <mapper resource="com/amiu/mybatis/DaoMapper.xml" />
7  </mappers>
```

xml和java文件应同名且在同一个包下

amiu

DaoMapper.java

DaoMapper.xml

对应的mybatisConfig.xml

```
1  <?xml version="1.0" encoding="UTF-8" ?>
2  <!DOCTYPE configuration
3      PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
4      "http://mybatis.org/dtd/mybatis-3-config.dtd">
5
6  <configuration>
7      <!-- 导入配置文件 -->
8      <properties resource="com/amiu/mybatis/dbConfig.properties" />
9      <environments default="development">
10         <environment id="development">
11             <!-- 使用jdbc事务管理,事务控制由mybatis管理-->
12             <transactionManager type="JDBC" />
13             <!-- 数据库连接池,由mybatis管理-->
14             <dataSource type="POOLED">
15                 <property name="driver" value="{driver}" />
16                 <property name="url" value="{url}" />
17                 <property name="username" value="{username}" />
18                 <property name="password" value="{password}" />
19             </dataSource>
20         </environment>
21     </environments>
22     <!-- 延迟加载的全局开关 -->
23     <settings>
24         <setting name="lazyLoadingEnabled" value="true"/>
25     </settings>
26     <!-- 配置别名 -->
27     <typeAliases>
28         <typeAlias alias="bean" type="com.amiu.TestBean"/>
29     </typeAliases>
30     <!-- 配置mapper -->
31     <mappers>
32         <mapper resource="com/amiu/mybatis/DaoMapper.xml" />
33     </mappers>
34 </configuration>
```