1. Configuration的手动配置

Configuration 类包含了对一个 SqlSessionFactory 实例你可能关心的所有内容。在检查配置时，Configuration 类很有用，它允许你查找和操纵 SQL 映射（但当应用开始接收请求时不推荐使用）。你之前学习过的所有配置开关都存在于 Configuration 类，只不过它们是以 Java API 形式暴露的。以下是一个简单的示例，演示如何手动配置 Configuration 实例，然后将它传递给 build() 方法来创建 SqlSessionFactory。

DataSource dataSource = BaseDataTest.createBlogDataSource();  
TransactionFactory transactionFactory = new JdbcTransactionFactory();  
  
Environment environment = new Environment("development", transactionFactory, dataSource);  
  
Configuration configuration = new Configuration(environment);  
configuration.setLazyLoadingEnabled(true);  
configuration.setEnhancementEnabled(true);  
configuration.getTypeAliasRegistry().registerAlias(Blog.class);  
configuration.getTypeAliasRegistry().registerAlias(Post.class);  
configuration.getTypeAliasRegistry().registerAlias(Author.class);  
configuration.addMapper(BoundBlogMapper.class);  
configuration.addMapper(BoundAuthorMapper.class);  
  
SqlSessionFactoryBuilder builder = new SqlSessionFactoryBuilder();  
SqlSessionFactory factory = builder.build(configuration);

1. Congratulation的作用

为创建SqlSessionFactory的builder()传递配置参数

1. Builder()参数

五种读取不同配置信息的builder

SqlSessionFactory build(InputStream inputStream)  
SqlSessionFactory build(InputStream inputStream, String environment)  
SqlSessionFactory build(InputStream inputStream, Properties properties)  
SqlSessionFactory build(InputStream inputStream, String env, Properties props)  
SqlSessionFactory build(Configuration config)