**mybatis的config.xml配置**

**ps--》用于配置mybatis的环境**

**连接池，数据库驱动，数据库url，用户名，密码，sql操作日志，事务，类型别名，类型处理器，扫描的mapper地址。。。**

**-------------------------------基本结构----------------------------------------**

<?xml version="1.0" encoding="UTF-8" ?>

<!DOCTYPE configuration PUBLIC "-//mybatis.org//DTD Config 3.0//EN"

"http://mybatis.org/dtd/mybatis-3-config.dtd">

<configuration>

</configuration>

**-------------------------------重要配置---------------------------------------**

**数据库环境**

<environments default="development">

<environment id="development">

<!-- 事务管理器 -->

<transactionManager type="JDBC">

<property name="..." value="..."/>

</transactionManager>

<!-- 数据源-连接池 -->

<dataSource type="POOLED">

<property name="driver" value="${driver}"/>

<property name="url" value="${url}"/>

<property name="username" value="${username}"/>

<property name="password" value="${password}"/>

</dataSource>

</environment>

</environments>

<mappers>

<mapper resource="org/mybatis/example/Mapper.xml"/>

</mappers>

**-------------------------------扩展配置---------------------------------------**

**引入properties文件配置的数据（键值对）**

<properties resource="jdbc.properties" />

**别名**

**-->扫描类**

< typeAliases>

<typeAlias alias="Author" type="domain.blog.Author"/>

<typeAlias alias="Blog" type="domain.blog.Blog"/>

</typeAliases>

**-->扫描包（没有同名类时）**

<typeAliases>
<package name="domain.blog"/>
</typeAliases>

**ps-》会使用 Bean 的首字母小写的非限定类名来作为它的别名。 比如 domain.blog.Author 的别名为 author**

**内置别名**

\_byte byte

\_long long

string String

long Long

integer Integer

date Date

map Map

hashmap HashMap

list List

arraylist ArrayList

**类型处理器（实现java和数据库类型的转换-参数和结果集使用）**

BooleanTypeHandler java.lang.Boolean, boolean 数据库兼容的 BOOLEAN

ByteTypeHandler java.lang.Byte, byte 数据库兼容的 NUMERIC 或 BYTE

LongTypeHandler java.lang.Long, long 数据库兼容的 NUMERIC 或 LONG INTEGER

FloatTypeHandler java.lang.Float, float 数据库兼容的 NUMERIC 或 FLOAT

StringTypeHandler java.lang.String CHAR, VARCHAR

DateOnlyTypeHandler java.util.Date DATE

TimeOnlyTypeHandler java.util.Date TIME

SqlDateTypeHandler java.sql.Date DATE

**自定义类型处理器**

**a）**实现 org.apache.ibatis.type.TypeHandler 接口， 或继承一个很便利的类 org.apache.ibatis.type.BaseTypeHandler

b）重写方法

c）配置自己的类型处理器

<typeHandlers>

<typeHandler handler="org.mybatis.example.ExampleTypeHandler"/>

</typeHandlers>

d）在mapper的Xml里使用处理器进行转换

<result typeHandler="org.apache.ibatis.type.EnumTypeHandler"/>

#{roundingMode, typeHandler=org.apache.ibatis.type.EnumTypeHandler}

**日志配置（打印sql到控制台）---如果集成了spring配置方式不一样**

**ps-》（支持各种日志框架）**SLF4J、Apache Commons Logging、Log4j 2、Log4j、JDK logging具体选择哪个日志实现工具由MyBatis的内置日志工厂确定。它会使用最先找到的（按上文列举的顺序查找）

**指定Log4j进行日志输出**

<settings>

<setting name="logImpl" value="LOG4J"/>

</settings>

**Log4j的配置文件**

# 记录的级别（其他类错误才记录TRACE,DEBUG,info,warn,ERROR）

log4j.rootLogger=DEBUG, stdout

# 要记录的类（指定要记录的）

# TRACE跟踪所有的信息，DEBUG只记录sql语句

log4j.logger.org.mybatis.example（命令空间或包名）=TRACE

# 控制台输出.

log4j.appender.stdout=org.apache.log4j.ConsoleAppender

log4j.appender.stdout.layout=org.apache.log4j.PatternLayout

log4j.appender.stdout.layout.ConversionPattern=%5p [%t] - %m%n