# 环境

<dependency>

<groupId>org.mybatis</groupId>

<artifactId>mybatis</artifactId>

<version>3.3.0</version>

</dependency>

如果需要日志则:

<dependency>

<groupId>log4j</groupId>

<artifactId>log4j</artifactId>

<version>1.2.17</version>

</dependency>

log4j.properties:

|  |
| --- |
| ##define an appender named console  log4j.appender.console=org.apache.log4j.ConsoleAppender  #The Target value is System.out or System.err  log4j.appender.console.Target=System.out  #set the layout type of the apperder  log4j.appender.console.layout=org.apache.log4j.PatternLayout  #set the layout format pattern  log4j.appender.console.layout.ConversionPattern = [%-5p] %m%n  #这里xxx表示的是mapper的namespace的值  log4j.logger.xxx = debug,console #还可以为trace,console |

# 配置文件

1.主配置文件:

|  |
| --- |
| <?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>  <!--配置运行环境-->  <environments default="mysqlEM">  <environment id="mysqlEM">  <transactionManager type="JDBC"></transactionManager>  <dataSource type="POOLED">  <property name="driver" value="com.mysql.jdbc.Driver"></property>  <property name="url" value="jdbc:mysql://47.96.27.61:3306/test"></property>  <property name="username" value="lw"></property>  <property name="password" value="linwei"></property>  </dataSource>  </environment>  </environments>  <!--配置映射文件-->  <mappers>  <mapper resource="dao/mapper.xml"></mapper>  <!--<mapper resource="dao/mapper2.xml"></mapper>-->  </mappers>  </configuration> |

2.映射配置文件:

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" ?>  <!DOCTYPE mapper  PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"  "http://mybatis.org/dtd/mybatis-3-mapper.dtd">  <mapper namespace="dao.IStudentDao">  <insert id="insertStudent" parameterType="entity.Student">  INSERT INTO student(name,age,score) VALUES (#{name},#{age},#{score})  </insert>  <insert id="insertStudentCacheId" parameterType="entity.Student">  INSERT INTO student(name,age,score) VALUES (#{name},#{age},#{score})  <selectKey resultType="int" keyProperty="id" order="AFTER">  SELECT @@IDENTITY  </selectKey>  </insert>  <delete id="deleteStudentById" parameterType="entity.Student">  DELETE FROM student WHERE id=#{xxx} <!--这里的#{}仅仅十个占位符,里面放什么都行,和上面的不一样-->  </delete>  <update id="updateStudent" parameterType="entity.Student">  UPDATE student SET name=#{name},age=#{age},score=#{score} WHERE id=#{id}  </update>  <select id="selectAllStudents" resultType="entity.Student">  SELECT id,name,age,score FROM student  </select>  <select id="selectStudentById" resultType="entity.Student">  SELECT id,name,age,score FROM student WHERE id=#{jjj}  </select>  <!--SELECT id,name,age,score FROM student WHERE name LIKE concat('%',#{xxx})-->  <select id="selectStudentsByName" resultType="entity.Student">  SELECT id,name,age,score FROM student WHERE name LIKE '%' #{xxx} '%'  </select>  </mapper> |

# 工具类

|  |
| --- |
| package utils;  import org.apache.ibatis.io.Resources;  import org.apache.ibatis.session.SqlSession;  import org.apache.ibatis.session.SqlSessionFactory;  import org.apache.ibatis.session.SqlSessionFactoryBuilder;  import java.io.IOException;  import java.io.InputStream;  public class MybatisUtils {  private static SqlSessionFactory sqlSessionFactory;  public static SqlSession getSqlSession(){  try{  InputStream is = Resources.getResourceAsStream("mybatis.xml");  if(sqlSessionFactory == null){  sqlSessionFactory = new SqlSessionFactoryBuilder().build(is);  }  return sqlSessionFactory.openSession();  }catch (IOException e){  e.printStackTrace();  }  return null;  }  } |