# 1.创建实体类

**public** **class** User {

**private** String id;

**private** String name;

**private** **int** age;

**public** String getId() {

**return** id;

}

**public** **void** setId(String id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **int** getAge() {

**return** age;

}

**public** **void** setAge(**int** age) {

**this**.age = age;

}

@Override

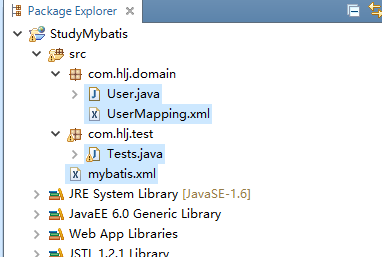
**public** String toString() {

**return** "User [id=" + id + ", name=" + name + ", age=" + age + "]";

}

}

# 2.编写连接数据库配置文件 mybatis.xml



<?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>

<environments default=*"development"*>

<environment id=*"development"*>

<transactionManager type=*"JDBC"* />

<!-- 配置数据库连接信息 -->

<dataSource type=*"POOLED"*>

<property name=*"driver"* value=*"com.mysql.jdbc.Driver"* />

<property name=*"url"* value=*"jdbc:mysql://localhost:3306/mybatis"* />

<property name=*"username"* value=*"root"* />

<property name=*"password"* value=*"root"* />

</dataSource>

</environment>

</environments>

<mappers>

<mapper resource=*"com/hlj/domain/userMapping.xml"*/>

</mappers>

</configuration>

# 3.编写映射文件，添加sql语句

<?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=*"com.hlj.domain.UserMapping"*>

<!-- 在select标签中编写查询的SQL语句， 设置select标签的id属性为getUser，id属性值必须是唯一的，不能够重复

使用parameterType属性指明查询时使用的参数类型，resultType属性指明查询返回的结果集类型

resultType="com.hlj.domain.User"就表示将查询结果封装成一个User类的对象返回

User类就是users表所对应的实体类

-->

<!--

根据id查询得到一个user对象

-->

<select id=*"getUser"* parameterType=*"String"*

resultType=*"com.hlj.domain.User"*>

select \* from user where id=#{id}

</select>

</mapper>

# 4.进行测试

**public** **class** Tests {

**public** **static** **void** main(String[] args) {

Tests tests = **new** Tests();

tests.test();

}

**public** **void** test(){

String resource = "mybatis.xml";

//使用类加载器加载mybatis的配置文件（它也加载关联的映射文件）

InputStream is = Tests.**class**.getClassLoader().getResourceAsStream(resource);

//构建sqlSession的工厂

SqlSessionFactory sessionFactory = **new** SqlSessionFactoryBuilder().build(is);

SqlSession session = sessionFactory.openSession();

/\*\*

\* 映射sql的标识字符串，

\* com.hlj.domain.UserMapping是userMapper.xml文件中mapper标签的namespace属性的值，

\* getUser是select标签的id属性值，通过select标签的id属性值就可以找到要执行的SQL

\*/

String statement = "com.hlj.domain.UserMapping.getUser";//映射sql的标识字符串

//执行查询返回一个唯一user对象的sql

User user = session.selectOne(statement, "1123");

System.*out*.println(user.toString());

}