

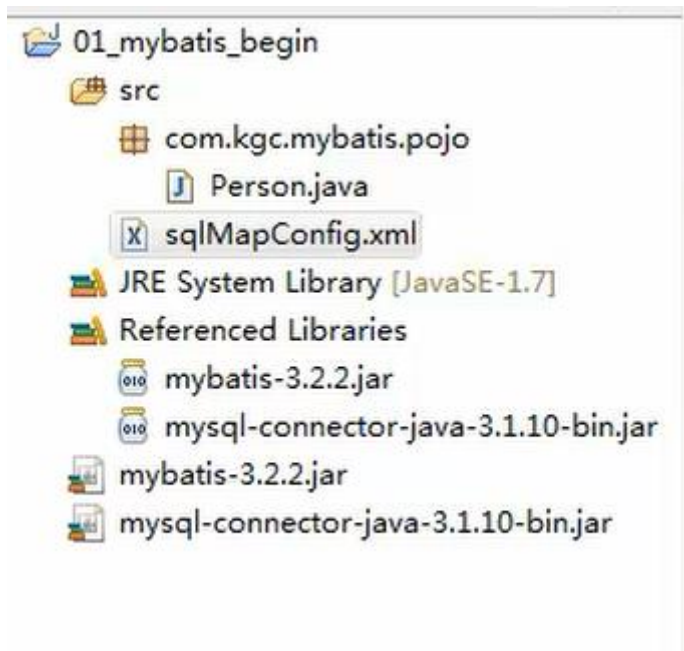
## JDBC

- ☐ 加载驱动
- ☐ 获取连接对象
- ☐ 创建处理命令对象
- ☐ 执行查询或更新
- ☐ 解析结果集
- ☐ 释放资源



## MyBatis的优势:

- 1、消除大量的JDBC冗余代码
- 2、低学习曲线
- 3、友好的spring集成支持
- 4、友好的第三方缓存类库集成支持
- 5、良好的性能



```
<configuration>
  <environments default="hello_mysql">
    <environment id="hello_mysql">
      <transactionManager type="jdbc"/>
      <dataSource type="pooled">
        <property name="driver" value="org.gjt.mm.mysql.Driver"/>
        <property name="url" value="jdbc:mysql://localhost:3306/mydb01"/>
        <property name="username" value="root"/>
        <property name="password" value="ok"/>
      </dataSource>
    </environment>
  </environments>

  <mappers>
    <mapper resource="com/kgc/mybatis/pojo/PersonMapper.xml"/>
  </mappers>
</configuration>
```

**insert**

```

@Test
public void addPersonTest(){
    SqlSession session = null ;
    try {
        //1.加载sqlMapConfig.xml
        InputStream is = Resources.getResourceAsStream("sqlMapConfig.xml");
        //2.解析sqlMapConfig.xml,创建SqlSessionFactory对象
        SqlSessionFactory factory = new SqlSessionFactoryBuilder().build(is);
        //3.通过factory创建session对象
        session = factory.openSession();
        Person person = new Person("tom",21,"China");
        //4.执行插入操作,返回影响行数
        int count = session.insert("com.kgc.mybatis.pojo.addPerson", person);

        System.out.println(count>0?"insert succ!" : "insert fail!");

        //5.提交或回滚事务
        session.commit();
    } catch (IOException e) {
        e.printStackTrace();
    }finally{
        //6.关闭session
        if(session!=null){
            session.close();
        }
    }
}

```

## select

```

<mapper namespace="com.kgc.mybatis.pojo">
    <insert id="addPerson" parameterType="com.kgc.mybatis.pojo.Person">
        insert into t_person values(null,#{pname},#{age},#{address})
    </insert>
    <select id="getPersonList" resultType="com.kgc.mybatis.pojo.Person">
        select * from t_person
    </select>
    <update id="updatePesron">
        update t_person set pname = #{pname} , age = #{age} , address = #{address} where pid = #{pid}
    </update>

```

```

@Test
public void getPersonListTest(){
    List<Person> personList = session.selectList("com.kgc.mybatis.pojo.getPersonList");
    for(Person person : personList){
        System.out.println(person);
    }
}

```

## del和update

```

@Test
public void updatePersonTest(){
    Person person = new Person(3,"lina",30,"KGC");
    int count = session.update("com.kgc.mybatis.pojo.updatePesron", person);
    System.out.println(count>0?"update succ!" : "update fail!");
}

```

```

@Test
public void delPersonTest(){
    int count = session.delete("com.kgc.mybatis.pojo.deletePerson", 2);
    System.out.println(count>0?"delete succ!" : "delete fail!");
}

```

```

<update id="updatePesron" parameterType="com.kgc.mybatis.pojo.Person">
    update t_person set pname = #{pname} , age = #{age} , address = #{address} where pid = #{pid}
</update>
<delete id="deletePerson" parameterType="int">
    delete from t_person where pid = #{pid}
</delete>

```

**environments 表示多个环境**

**主要放：数据库连接**

**sqlMapConfig.xml 配置文件中**

**类的配置文件中 (PersonMapper.xml) :**

