

parameterType指定接收参数的数据类型

1、基本数据类型：string、int、double等等

```
<select id="findUbyID" parameterType="int" resultType="users">
    select * from users where u_id=#{id}
</select>
```

2、接收pojo对象类型参数，如保存，修改时用的对象类型

```
<!--
需求：保存用户数据
参数：users对象
返回值：保存当前用户数据生成的id值
-->
<insert id="addUsers" parameterType="users" useGeneratedKeys="true" keyProperty="id">
    insert into users values(#{u_id},#{uname},#{upwd})
</insert>
```

```
//生产sqlSession对话对象
SqlSession openSession = sqlSessionFactory.openSession();
Users u = new Users();
u.setUname("李9857");
u.setUpwd("12939");
openSession.insert("Test.addUsers",u);
openSession.commit();
openSession.close();
```

3、接收map类型参数：

```
<!--多条件查询 接收map类型参数-->
<select id="findUbynamewhiz" parameterType="map" resultType="users">
    select * from users where uname=#{uname} and upwd=#{upwd}
</select>
```

```
/**
 * 传递map参数进行多条件查询接口方法
 */
@Override
public List<Users> findUbynamewhiz(Map<String, Object> m) {
    return openSession.selectList(UserDao.names+"findUbynamewhiz",m);
}
```

```
Map<String,Object> m = new HashMap<String,Object>();
m.put("uname","张三");
```

```

m.put("upwd", "123");
List<Users> findall = u.findUbynamewhiz(m);
for (Users users : findall) {
    System.out.println(users);
}

```

4、接收包装类型参数:

<!--多条件查询 接收包装类型参数-->

```

<select id="findUbybz" parameterType="cn.rong.zx.QuVo" resultType="users">
    select * from users where uname=#{u.uname} and upwd=#{u.upwd}
</select>

```

```

/**
 * 包装类
 */
public class QuVo {
    // 数据变量
    private Users u;
    public Users getU() {
        return u;
    }
    public void setU(Users u) {
        this.u = u;
    }
}

/**
 * 传递包装类参数进行多条件查询
 */
@Override
public List<Users> findUbybz(QuVo v) {
    return openSession.selectList(UserDao.names+"findUbybz",v);
}

```

```

/**
 * 多条件 包装类传参
 */
@Test
public void test03() {
    QuVo q = new QuVo();
}

```

```

Users s = new Users();
s.setUname("张三");
s.setUpwd("123");
q.setU(s);
List<Users> findall = u.findUbybz(q);
for (Users users : findall) {
    System.out.println(users);
}
}

```

5、数组类型参数：

6、List集合类型参数：

resultType返回数据类型

1、返回pojo对象类型数据，如查所有用户信息

```

<!--
    需求：查询所有
    返回值类型：集合类型list<Users>
    resultType:返回值类型    使用核心配置的别名
-->
<select id="findAll" resultType="users">
    select * from users
</select>

```

```

//执行所有查询    selectList
List<Users> selectList = openSession.selectList("Test.findAll");

```

2、返回int, long类型数据，如查总记录数

```

<select id="findUserWithQueryVoCount" parameterType="cn.rong.zx.QuVo" resultType="int">
    select COUNT(1) from users where uname=#{u.uname}
</select>

```

3、返回map类型数据 返回的属性名与数据库列名不一致时使用

```

<!--返回map类型数据-->
<resultMap type="user" id="userResultMap">
    <result column="_username" property="username"/>
    <result column="_adress" property="address"/>
</resultMap>

<select id="findUserWithResultMap" resultMap="userResultMap">
    select username _username,address _adress from user
</select>

```

