**第三节 接口方式[**推荐**]/动态SQL语句**

2016年3月19日 星期六

08:31

MVC目录结构:

Src

-- com.shxt.servlet【控制层】

--com.shxt.service【业务逻辑层】

--com.shxt.model【实体Bean，用来承载数据】

--com.shxt.dao【data access object数据访问层，执行SQL】

**使用接口的规则**

计算机生成了可选文字:
@ UserMapper.xml 
lag4j.praperties 
f'http://mybatis.or /dtd/ batis-3-ma er.dtd"> 
4 
namespace 'com. shxt. mapper. UserMapper 
5' <mapper 
serMapperjava 
package com.shxt.mapper; 
6 
8 
9 
10 
11 
12 
13 
14 
15 
result Ty 
namespace—3 [3 
"com. shxt. model . User' 
<select id="Load" parameter Type="i 
select 
user id user 
account, 
password pwd, 
user name userName, 
status 
from 
sys user 
4 
5 
6 
8 
9 
10 
11 
12 
import 
public 
com.shxt . model. User; 
interfac UserMapper { 
ublic resutType 
ser 10 d( n eger shxt); 
resultType*i Nif@ 
Design Source 
My8atisTestjava 
EC]ßJfiü--€ 
26 
27 
28 
29 
30 
31 
32 
33 
34 
35 
36 
37 
38 
public void 
Sq1Session sq1Session 
null; 
try { 
sq1Session = MyBatisUti1s. etS [Session 
getMapper(UserMapper.c1ass) load(3); 
sq1Session 
User user 
System. out. print In (user 
} catch (Exception e) { 
e. printStackTrace(); 
MyBatisUti1s . cLoseSqLSession(sq1Session) ; 

**接口方式:处理多个参数的规则**

计算机生成了可选文字:
My8atisTestjava 
public void 
SqISession sqISession 
null; 
try { 
sqISession = myaatislJtiIs.getSqLSession(); 
sqISession . getmapper (Usermapper. class) . I i stayStatusNameß2 
catch (Exception e) { 
e. printStackTrace(); 
myaati sot i Is . c LoseSqLSess ion (sqISession ) ; 
@Test 
public void 
u] UserMapperjava 
public 
list8yStatusNameßI map); 
public listayStatusNameß2 (String u erName,String 
tus); 
lag4j.praperties 
UserMapper.xml 
< / select> 
(select id= "ListByStatusNamee2" 
select 
user id userld, 
account, 
password pwd, 
user name userName, 
status 
sys _ user 
where 
user name like 
< / select> 
result T 
on. shxt. model . User" 
*JNR, 
paraml, param2,param3k 

**动态SQL之Where语句**

计算机生成了可选文字:
WhereiäÄ - 
(select " 
p" 
sel ect 
(if name.'=nuLL and user name.' 
parameterType= " 
result Type= ' 'con. shxt. model . User" > 
user id userld, 
account, 
password pwd, 
user name userName, 
status 
sys _ user 
where 
an user name 1 
e conca 
user name 
if and status.' 
and 
< / select> 
WhereiäÄ - 
(select "ListByWhere32" 
select 
user id userld, 
account, 
password pwd, 
user name userName, 
status 
f rorrl 
s s user 
paramet erType= ' 'map " 
result Type= ' 'con. shxt. model . User" > 
and or 
(if test= "user and user 
user _ name like 
(if test= and status! 
or 

计算机生成了可选文字:
<select id="ListByWhere03" parameterType 
select 
user id userld, 
account, 
password pwd, 
user name userNam 
"map ' 
result Type 
"com. shxt. model . User 
status 
from 
sys user 
<trim prefix 
<if test 
and 
whereQW 
refixOverrides="AND IOR 
"where 
'user name!=nuLL and user name! — 
user name like concat(#{user name}, '%' ) 
<if test="status!=nuLL and status! 
and status=#{status} 
</select> 

更新

计算机生成了可选文字:
<update id— "update01 " parameterType 
update sys user 
<se 
< if 
user name— 
user ame 
<if test="account ! = nut L 
account=#{account} , 
<if 
<if 
</set> 
where 
user id=#{userld} 
</update> 
"com. shxt. model . User 
-->and 
Il or 
<update id—"update02" parameterType="com. shxt.modeL. User"> 
update sys user 
<trim prefix="set " suffixoverrides 
<if test="userName!=nuLL 
user name=#{userName}, 
<if test="account ! = nut L 
account=#{account} , 
<if 
<if 
where 
user id=#{userld} 
</update> 

添加操作

计算机生成了可选文字:
<insert id="add" parameterType 
<trim prefix="(" suffix 
test= userName. =nu 
user name, 
<if test="account ! — 
account , 
"com. shxt. model . User 
suffixOverrides 
<if 
password, 
<if 
status, 
</trim> 
<trlm pre va ues 
nut L 
su 
su 
es 
user ame. =nu 
#{userName}, 
<if test="account ! — 
<if 
<if 
#{status}, 
nut L 

批量处理

计算机生成了可选文字:
"com. shxt. model . User 
select 
user id userld, 
account, 
password pwd, 
user name userName, 
status 
from 
sys user 
where 
user id in 
<trim prefix="(" suffix 
<foreach collection 
#{shxt}, 
</foreach> 
</select> 
" suffixoverrides 
"List" item— "shxt" 
<insert parameterType 
insert into sys user 
(account, password, user name) 
VALUES 
for(int i=0; i<list.size 
(i) 
User shxt = list. 
ist" 
<foreach collection 
ist" item—"shxt" separator ' 
(#{shxt . account} , . pwd} , . userName}) 
k/foreach> 
<insert parameterType 
insert into sys user 
(account, password, user name) 
VALUES 
<trim suffixoverrides 
"List" 
<foreach item— "shxt" index ' 
(#{shxt . account} , . pwd} , . userName}) , 
</foreach> 