

()

: ADBD2020R_2020081117082420200817230456101973579317

: 2020-08-17 23: 04: 56

: _201703150110_

:
:

/

()

Springer

Taylor & Francis

)

CNKI

: 1900-01-01 2020-08-17

	7%	0%
引	7%	7%
单	2.1%	" "
[659]	[9360]	[196]
[1]	[182]	[659]
[1]	[477]	[659]
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
1	0	0



1.	_201703150110_	9360
	7% (659)	7% (659)
	(0)	
1	" "	2.1% 196
	- 2013-08-01	
2	-	1.9% 181
	https://www.etl.org.c - 2019	
3	153 2 (1)	1.4% 129
	- 2018-05-17	
4	1502697204_	1.3% 125
	- 2019-06-02	
5	Python	0.7% 67
	- 2020-07-31	
6		0.7% 64
	- 2009-04-15	

7	-	- 2009-04-15	0.7% 64
8	"	"	0.7% 64
	-	- 2009-03-27	
9			0.7% 63
	; -	- 2020-06-30	
10			0.6% 59
	; ; ; ; -	- 2014-10-25	
11	() -	- 2017-06-01	0.6% 59
12	-03312143-	- 2016-06-29	0.6% 57
	-	- 2016-06-29	
13	Struts, Spring, Ibatis		0.6% 53
	; -	- 2011-06-30	
14			0.6% 52
	-	http://www.xzbu.com/ - 2017	
15	7245948_	-	0.4% 38
	-	- 2019-06-08	
16			0.4% 37
	; ; -	- 2017-08-05	

3171 3171

2020 6

1			4000	
2				
3				
4				
5				
(1)	A4	3	2.5	
2	26			
(2)			0.5	1
		1	0.25	0.25
		0.25	0.25	
(3)			1.5	
(4)		0.5		
			0.25	
(5)				24
6				

	3
()	3
()	3
	3
()	3
1	4
2	4
3	5
4	5
5	6
6	6
7	7
()	7
	8
()	8
1	8
2	8
3	9
4	9
()	10
1	10
2	12
3	14
4	15
5	16
6	17
	19
()	19
()	19
	19
()	19
()	20
	20

()

4g 5g

()

spring+springmvc+mybatis
 eclipse+navicat+google,

css+html+jquery+ajax+bootstrap

mysql,

()

1

2

no
Yes
3

4

no
yes

5

6

7

yes
No

()

1 css, javascript, jquery, html css javascript jquery json

HTML
CSS
JAVASCRIPT
JQUERY

()

1 pvd id

name	pvd			
id	int	20		
name	varchar	200		
pvd	varchar	200		

id int 20
name varchar 200
pvd varchar 200

2

gname gpvd id

gname	gpvd			
id	int	20		
gname	varchar	220		
gpvd	varchar	220		

id int 20
gname varchar 220
gpvd varchar 220

3

jiage id name age
wh yx ns
di zhi dh

jiage	wh	id	name	age	ns
di zhi		dh			
id	int	50			
name	varchar	200			
age	varchar	100			
jiage	varchar	200			
wh	varchar	200			
yx	varchar	200			
ns	varchar	200			
di zhi	varchar	220			
dh	varchar	200			

id int 50
name varchar 200
age varchar 100
jiage varchar 200
wh varchar 200
yx varchar 200
ns varchar 200
di zhi varchar 220
dh varchar 200

4

Dname

ji age	ni aoshu	ing		
Id	int	200		
Dname	varchar	225		
ji age	int	225		
ni aoshu	varchar	225		
ing	varchar	225		

```

Id int 200
Dname varchar 225
ji age int 225
ni aoshu varchar 225
ing varchar 225

```

```

( )
1

```

```

    ajax url: http://localhost:8080/dd/chazhi @RequestMapping("/chazhi")
    public String cha(prop, HttpServletResponse hp) gson
success: function(data){ for(var i=0; i<data.length; i++){ $("#dname"+i).html(data[i].dname)
$("#ji age"+i).html(data[i].ji age) $("#ni aoshu"+i).html(data[i].ni aoshu) $("#ing"+i).attr("src", data[i].ing) }
2

```

```

    Controller @RequestMapping("/dengl u")
    public void chaxun(dd u, HttpServletResponse hp) throws IOException {
hp.setCharacterEncoding("utf-8");
hp.setContentType("text/html; charset=utf-8");
dd i = us.chaxun(u);
try {
Gson gson = new Gson();
hp.getWriter().print(gson.toJson(i));
} catch (Exception e) {
// TODO handle exception
}
}
    mapper <select id="chaxun" resultType="dd" parameterType="dd" ><!-- parameterType
String , resultType="dd" -->
select * from user where name=#{name} and pwd=#{pwd};
</select>

```

```

@RequestMapping("/update")
    public void updata(dd d, HttpServletResponse hp) {
hp.setCharacterEncoding("utf-8");
hp.setContentType("text/html; charset=utf-8");
int i = us.updata(d);
if (i != 0) {
try {
Gson gson = new Gson();
hp.getWriter().print(gson.toJson(" "));
} catch (Exception e) {
// TODO handle exception
}
}
    @RequestMapping("/dengl u")

```

```

public void chaxun(dd u, HttpServletResponse hp) throws IOException {
    hp.setCharacterEncoding("utf-8");
    hp.setContentType("text/html; charset=utf-8");
    ddi = us.chaxun(u);
    try {
        Gson gson = new Gson();
        hp.getWriter().print(gson.toJson(i));
    } catch (Exception e) {
        // TODO handle exception
    }
}
3

```

```

                                ID @RequestMapping("/idzhi")
public void idzhi(String uid, HttpServletResponse hp) {
    hp.setCharacterEncoding("utf-8");
    hp.setContentType("text/html; charset=utf-8");
    id = Integer.valueOf(uid);
    try {
        pro p = ps.idzhi(id);
        hp.getWriter().print(new Gson().toJson(""));
    } catch (Exception e) {
        // TODO handle exception
    }
    mapper.sql
        <select id="idzhi" resultType="pro" parameterType="pro"><!-- parameterType
            int String, resultType="dd" -->
    select * from pro where id=#{id};
</select>
4

```

```

    input
@RequestMapping("/sousuo")
public void sousuo(pro p, HttpServletResponse hp) {
    hp.setCharacterEncoding("utf-8");
    hp.setContentType("text/html; charset=utf-8");
    List<pro> list = ps.sou(p);
    try {
        hp.getWriter().print(new Gson().toJson(list));
    } catch (Exception e) {
        // TODO handle exception
    }
    sql
    <select id="sou" resultType="pro" parameterType="pro">
    select * from pro where dname like concat('%', #{dname}, '%')
</select>
5

```

```

                                @RequestMapping("/hui shou")
public void kui shou(hs h, HttpServletResponse hp) {

```

```

hp.setCharacterEncoding("utf-8");
hp.setContentType("text/html; charset=utf-8");
int i = hss.kuishou(h);
if (i != 0) {
try {
hp.getWriter().print(new Gson().toJson("")); } catch (Exception e) {
// TODO handle exception
}
}

```

java mapper sql

```

<insert id="kuishou" parameterType="hs">
insert into hs(name, age, jiage, wh, yx, ns, di zhi, dh)
value({name}, {age}, {jiage}, {wh}, {yx}, {ns}, {di zhi}, {dh})
</insert>
6

```

30%

```

@RequestMapping("/limi")
public void glyym(int star, int end, HttpServletResponse hp) throws IOException {
hp.setCharacterEncoding("utf-8");
hp.setContentType("text/html; charset=utf-8");
HashMap<String, Integer> hl = new HashMap<String, Integer>();
System.out.println(star + "star");
System.err.println(end + "end");
hl.put("a", star);
hl.put("b", end);
List<hs> list = hss.limi(hl);
for (hs pd : list) {
System.out.println(
pd.getId() + pd.getName() + pd.getAge() + pd.getJiage() + pd.getWh() + pd.getYx() + pd.getNs());
}
hp.getWriter().print(new Gson().toJson(list));
}

```

()

spring, springmvc, mybatis, html javascript, css, ps, jquery, ajax, mysql

()

()

spring, springmvc, mybatis, ajax, mysql, html javascript, css, ps, jquery,

windows unix mysql
spring IOC ACP Web MVC
ACP Spring CRM Spring DAO Spring Context Spring Web Spring Web MVC Spring Core mybatis Spring
sql sql DAO
html css javascript ps

()

- [1] Java Web [J]. 2017, 04 214-216
- [2] HTML JAVA WEB [J]. 2014, 11 140-141
- C
- [3] Java Web [J]. 2014, 14 7.
- [4] JADE Java Web [J].
- 2014, 02 63-66
- [5] JADE Java Web [J]. 2014, 03 26-29+45.
- [6] JavaWeb [J]. 2014, 03 69-72
- [7] Java Web [D]. 2014.

1.				
2.				
3.				
4.		30%	" "	
5.				sql

1	2			
id	int	20		
name	varchar	200		
pwd	varchar	200		

1	4.11	TB_TK		
	50.00%			
20140219130301_46-			-2014-02-19	
id	int	20		
testContent	varchar	200		
rightAns	varchar	200		

2	5-11	TB_TK		
	50.00%			
			--	-2014-10-21
id	int	20		
testContent	varchar	200		
rightAns	varchar	200		

- 1.
- 2.
- 3.
- 4.

5.

6.

7.



 amtc@cnki.net

 <http://check.cnki.net/>

 <http://e.weibo.com/u/3194559873/>