java日期格式大全 format SimpleDateFormat

/\*\*

\* 字符串转换为java.util.Date<br>

\* 支持格式为 yyyy.MM.dd G 'at' hh:mm:ss z 如 '2002-1-1 AD at 22:10:59 PSD'<br>

\* yy/MM/dd HH:mm:ss 如 '2002/1/1 17:55:00'<br>

\* yy/MM/dd HH:mm:ss pm 如 '2002/1/1 17:55:00 pm'<br>

\* yy-MM-dd HH:mm:ss 如 '2002-1-1 17:55:00' <br>

\* yy-MM-dd HH:mm:ss am 如 '2002-1-1 17:55:00 am' <br>

\* @param time String 字符串<br>

\* @return Date 日期<br>

\*/

public static Date stringToDate(String time){

SimpleDateFormat formatter;

int tempPos=time.indexOf("AD") ;

time=time.trim() ;

formatter = new SimpleDateFormat ("yyyy.MM.dd G 'at' hh:mm:ss z");

if(tempPos>-1){

time=time.substring(0,tempPos)+

"公元"+time.substring(tempPos+"AD".length());//china

formatter = new SimpleDateFormat ("yyyy.MM.dd G 'at' hh:mm:ss z");

}

tempPos=time.indexOf("-");

if(tempPos>-1&&(time.indexOf(" ")<0)){

formatter = new SimpleDateFormat ("yyyyMMddHHmmssZ");

}

else if((time.indexOf("/")>-1) &&(time.indexOf(" ")>-1)){

formatter = new SimpleDateFormat ("yyyy/MM/dd HH:mm:ss");

}

else if((time.indexOf("-")>-1) &&(time.indexOf(" ")>-1)){

formatter = new SimpleDateFormat ("yyyy-MM-dd HH:mm:ss");

}

else if((time.indexOf("/")>-1) &&(time.indexOf("am")>-1) ||(time.indexOf("pm")>-1)){

formatter = new SimpleDateFormat ("yyyy-MM-dd KK:mm:ss a");

}

else if((time.indexOf("-")>-1) &&(time.indexOf("am")>-1) ||(time.indexOf("pm")>-1)){

formatter = new SimpleDateFormat ("yyyy-MM-dd KK:mm:ss a");

}

ParsePosition pos = new ParsePosition(0);

java.util.Date ctime = formatter.parse(time, pos);

return ctime;

}

/\*\*

\* 将java.util.Date 格式转换为字符串格式'yyyy-MM-dd HH:mm:ss'(24小时制)<br>

\* 如Sat May 11 17:24:21 CST 2002 to '2002-05-11 17:24:21'<br>

\* @param time Date 日期<br>

\* @return String 字符串<br>

\*/

public static String dateToString(Date time){

SimpleDateFormat formatter;

formatter = new SimpleDateFormat ("yyyy-MM-dd HH:mm:ss");

String ctime = formatter.format(time);

return ctime;

}

/\*\*

\* 将java.util.Date 格式转换为字符串格式'yyyy-MM-dd HH:mm:ss a'(12小时制)<br>

\* 如Sat May 11 17:23:22 CST 2002 to '2002-05-11 05:23:22 下午'<br>

\* @param time Date 日期<br>

\* @param x int 任意整数如：1<br>

\* @return String 字符串<br>

\*/

public static String dateToString(Date time,int x){

SimpleDateFormat formatter;

formatter = new SimpleDateFormat ("yyyy-MM-dd KK:mm:ss a");

String ctime = formatter.format(time);

return ctime;

}

/\*\*

\*取系统当前时间:返回只值为如下形式

\*2002-10-30 20:24:39

\* @return String

\*/

public static String Now(){

return dateToString(new Date());

}

/\*\*

\*取系统当前时间:返回只值为如下形式

\*2002-10-30 08:28:56 下午

\*@param hour 为任意整数

\*@return String

\*/

public static String Now(int hour){

return dateToString(new Date(),hour);

}

/\*\*

\*取系统当前时间:返回值为如下形式

\*2002-10-30

\*@return String

\*/

public static String getYYYY\_MM\_DD(){

return dateToString(new Date()).substring(0,10);

}

/\*\*

\*取系统给定时间:返回值为如下形式

\*2002-10-30

\*@return String

\*/

public static String getYYYY\_MM\_DD(String date){

return date.substring(0,10);

}

public static String getHour(){

SimpleDateFormat formatter;

formatter = new SimpleDateFormat ("H");

String ctime = formatter.format(new Date());

return ctime;

}

public static String getDay(){

SimpleDateFormat formatter;

formatter = new SimpleDateFormat ("d");

String ctime = formatter.format(new Date());

return ctime;

}

public static String getMonth(){

SimpleDateFormat formatter;

formatter = new SimpleDateFormat ("M");

String ctime = formatter.format(new Date());

return ctime;

}

public static String getYear(){

SimpleDateFormat formatter;

formatter = new SimpleDateFormat ("yyyy");

String ctime = formatter.format(new Date());

return ctime;

}

public static String getWeek(){

SimpleDateFormat formatter;

formatter = new SimpleDateFormat ("E");

String ctime = formatter.format(new Date());

return ctime;

}

在jsp页面中的日期格式和sqlserver中的日期格式不一样，怎样统一？

在页面上显示输出时，用下面的函数处理一下

public class DateUtil(){

public static String fmtShortEnu(Date myDate) {

SimpleDateFormat formatter = new SimpleDateFormat("yyyy/MM/dd");

String strDate = formatter.format(myDate);

return strDate;

}

}

new java.text.SimpleDateFormat("yyyy-MM-dd HH:mm:ss");

new java.text.SimpleDateFormat("yyyy-MM-dd")

建议还是把sqlserver的字段类型改成varchar的吧，用字符串处理可以完全按照自己的意愿处理，没有特殊的需求，不要使用date型

字串日期格式转换

用的API是SimpleDateFormat，它是属於java.text.SimpleDateFormat，所以请记得import进来！

用法：

SimpleDateFormat sdf=new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");

这一行最重要，它确立了转换的格式，yyyy是完整的西元年，MM是月份，dd是日期， 至於HH:mm:ss就不需要我再解释了吧！

ps:为什麽有的格式大写，有的格式小写，那是怕避免混淆，例如MM是月份，mm是分；HH是24小时制，而hh是12小时制

1.字串转日期：

　2002-10-8 15:30:22要把它转成日期，可以用

　Date date=sdf.parse("2002-10-8 15:30:22");

2.日期转字串

　假如把今天的日期转成字串可用

　String datestr=sdf.format(new Date());

　这个字串的内容便类似2002-10-08 14:55:38

透过这个API我们便可以随心所欲的将日期转成我们想要的字串格式，例如希望将日期输出成2002年10月08日，

我们可以这麽写：

SimpleDateFormat sdf=new SimpleDateFormat("yyyy年MM月dd日");

String datestr=sdf.format(new Date());

datestr便会依照我们设定的格式输出

//对日期格式的转换成（"yyyy-MM-dd"）格式的方法

public java.sql.Date Convert(String str)

{

java.text.SimpleDateFormat sdf = new java.text.SimpleDateFormat("yyyy-MM-dd");

try

{

java.util.Date d = sdf.parse(str);

java.sql.Date d1 = new java.sql.Date(d.getTime());

return d1;

}

catch(Exception ex)

{

ex.printStackTrace();

return null;

}

}

应用如下：

ctmt.setDate(7,this.Convert(info.getManBirth())); // @DATETIME

常用日期问题集锦

1、获取服务器端当前日期：

<%@ page import="java.util.Date"%>

<%

Date myDate = new Date();

%>

2、获取当前年、月、日：

<%@ page import="java.util.Date"%>

<%

Date myDate = new Date();

int thisYear = myDate.getYear() + 1900;//thisYear = 2003

int thisMonth = myDate.getMonth() + 1;//thisMonth = 5

int thisDate = myDate.getDate();//thisDate = 30

%>

3、按本地时区输出当前日期

<%@ page import="java.util.Date"%>

<%

Date myDate = new Date();

out.println(myDate.toLocaleString());

%>

输出结果为：

2003-5-30

4、获取数据库中字段名为”publish\_time“、类型为Datetime的值

<%@ page import="java.util.Date"%>

<%

...连接数据库...

ResultSet rs = ...

Date sDate = rs.getDate("publish\_time");

%>

1. 按照指定格式打印日期

**Java代码**

1. <%@ page **import**="java.util.Date"%>
2. <%@ page **import**="java.text.DateFormat"%>
3. <%
4. Date dNow = **new** Date();
6. SimpleDateFormat formatter = **new** SimpleDateFormat("E yyyy.MM.dd 'at' hh:mm:ss a zzz");
7. out.println("It is " + formatter.format(dNow));
8. %>
10. 输出的结果为：
11. It is 星期五 2003.05.30 at 11:30:46 上午 CST
12. (更为详尽的格式符号请参看SimpleDateFormat类)
14. 6、将字符串转换为日期
15. <%@ page **import**="java.util.Date"%>
16. <%@ page **import**="java.text.DateFormat"%>
17. <%
18. String input = "1222-11-11";
19. SimpleDateFormat formatter = **new** SimpleDateFormat("yyyy-MM-dd");
20. Date t = **null**;
21. **try**{
22. t = formatter.parse(input);
23. out.println(t);
24. }**catch**(ParseException e){
25. out.println("unparseable using " + formatter);
26. }
27. %>
28. 输出结果为：
29. Fri Nov 11 00:00:00 CST 1222
31. 7、计算日期之间的间隔
32. <%@ page **import**="java.util.Date"%>
33. <%@ page **import**="java.text.DateFormat"%>
34. <%
35. String input = "2003-05-01";
36. SimpleDateFormat formatter = **new** SimpleDateFormat("yyyy-MM-dd");
37. Date d1 = **null**;
38. **try**{
39. d1 = formatter.parse(input);
40. }**catch**(ParseException e){
41. out.println("unparseable using " + formatter);
42. }
44. Date d2 = **new** Date();
46. **long** diff = d2.getTime() - d1.getTime();
47. out.println("Difference is " + (diff/(1000\*60\*60\*24)) + " days.");
48. %>
49. 输出结果为：
50. Difference is 29 days.
52. 8、日期的加减运算
53. 方法：用Calendar类的add()方法
54. <%@ page **import**="java.util.\*"%>
55. <%@ page **import**="java.text.\*"%>
56. <%
57. Calendar now = Calendar.getInstance();
58. SimpleDateFormat formatter = **new** SimpleDateFormat("E yyyy.MM.dd 'at' hh:mm:ss a zzz");
59. out.println("It is now " + formatter.format(now.getTime()));
60. now.add(Calendar.DAY\_OF\_YEAR,-(365\*2));
61. out.println("<br>");
62. out.println("Two years ago was " + formatter.format(now.getTime()));
63. %>
64. 输出结果为：
65. It is now 星期五 2003.05.30 at 01:45:32 下午 CST
66. Two years ago was 星期三 2001.05.30 at 01:45:32 下午 CST
68. 9、比较日期
69. 方法：用equals()、before()、after()方法
70. <%@ page **import**="java.util.\*"%>
71. <%@ page **import**="java.text.\*"%>
72. <%
73. DateFormat df = **new** SimpleDateFormat("yyy-MM-dd");
74. Date d1 = df.parse("2000-01-01");
75. Date d2 = df.parse("1999-12-31");
77. String relation = **null**;
78. **if**(d1.equals(d2))
79. relation = "the same date as";
80. **else** **if**(d1.before(d2))
81. relation = "before";
82. **else**
83. relation = "after";
84. out.println(d1 +" is " + relation + ' ' + d2);
85. %>
86. 输出结果为：
87. Sat Jan 01 00:00:00 CST 2000 is after Fri Dec 31 00:00:00 CST 1999
89. 10、记录一件事所花费的时间
90. 方法：调用两次System.getTimeMillis()方法，求差值
91. <%@ page **import**="java.text.\*"%>
92. <%
93. **long** t0,t1;
94. t0 = System.currentTimeMillis();
95. out.println("Cyc starts at " + t0);
96. **int** k = 0;
97. **for**(**int** i =0;i<100000;i++){
98. k += i;
99. }
100. t1 = System.currentTimeMillis();
101. out.println("<br>");
102. out.println("Cyc ends at " + t1);
103. out.println("<br>");
104. out.println("This run took " + (t1-t0) + "ms.");
105. %>
107. 输出结果为：
108. Cyc starts at 1054275312432
109. Cyc ends at 1054275312442
110. This run took 10ms.
112. 其它：如何格式化小数
114. <%@ page **import**="java.text.\*"%>
115. <%
116. DecimalFormat df = **new** DecimalFormat(",###.00");
117. **double** aNumber = 33665448856.6568975;
118. String result = df.format(aNumber);
119. out.println(result);
120. %>
122. 输出结果为：
123. 33,665,448,856.66

126. ======================
128. 日期比较：
130. 在JAVA中日期的计算与比较可以使用Date和DateFormat来解决，下面是一段示例代码：
132. **import** java.text.\*;
133. **import** java.util.\*;
135. **public** **class** Test{
137. **public** **static** **void** main(String[] args){
138. **try**{
139. Date date=**new** Date();
140. DateFormat df=DateFormat.getDateTimeInstance();
141. String now=df.format(date);
142. System.out.println("现在时间:"+now);
144. System.out.println("现在时间是否在16:00之前:"+date.before(df.parse("2004-12-24 16:00:00")));
145. }
146. **catch**(ParseException e){System.out.print(e.getMessage());
147. }
148. }
149. }