

lonely\_fireworks的专栏

趁着还年轻，记忆还好，写点东西吧。

目录视图 摘要视图 RSS 订阅

个人资料



LeeFranker

访问： 879137次  
积分： 3033  
等级： BLOG > 5  
排名： 第12144名  
  
原创： 33篇 转载： 18篇  
译文： 0篇 评论： 89条

文章搜索

文章分类

- Android (30)
- 摄像 (1)
- Java (10)
- 软件错误 (1)
- c/c++ (2)
- Android UI (3)
- MAC (2)
- IOS (0)
- eclipse (4)
- coco2dx (1)

文章存档

- 2014年07月 (1)
- 2014年06月 (10)
- 2014年05月 (4)
- 2014年03月 (1)
- 2014年01月 (4)

展开

阅读排行

- JAVA字符串格式化-String.format()的使用 (591389)
- Android之监测手机网络状态 (30366)
- GridView 横向滚动 (27675)
- Android开发随笔之Scroll (27149)
- BigDecimal除法注意事项 (20297)

JAVA字符串格式化-String.format()的使用

标签： java date string 语言 测试 null

2012-09-10 11:01 591541人阅读 评论

分类： Java ( 9 )

版权声明：本文为博主原创文章，未经博主允许不得转载。

常规类型的格式化

String类的format()方法用于创建格式化的字符串以及连接多个字符串对象。熟悉C语言的同学应该记得C语言的printf()方法，两者有类似之处。format()方法有两种重载形式。

format(String format, Object... args) 新字符串使用本地语言环境，制定字符串格式和参数生成格式化的新字符串。

format(Locale locale, String format, Object... args) 使用指定的语言环境，制定字符串格式和参数生成格式化的字符串。

显示不同转换符实现不同数据类型到字符串的转换，如图所示。

转 换 符	说 明	示 例
%s	字符串类型	"mingrisoft"
%c	字符类型	'm'
%b	布尔类型	true
%d	整数类型（十进制）	99
%x	整数类型（十六进制）	FF
%o	整数类型（八进制）	77
%f	浮点类型	99.99
%a	十六进制浮点类型	FF.35AE
%e	指数类型	9.38e+5
%g	通用浮点类型（和e类型中较短的）	
%h	散列码	
%%	百分比类型	%

Android开发之Service优

(15332)

ListView中item删除动画

(15327)

Android 获取View在屏幕

(14357)

解决windows下eclipse中

(13656)

关于android分辨率兼容

(11918)

评论排行

JAVA字符串格式化-String

(45)

Android开发随笔之Scroll

(12)

解决windows下eclipse中

(7)

GridView 横向滚动

(7)

ListView中item删除动画

(3)

关于android分辨率兼容

(2)

APK放到data/data/下面

(2)

Android之ActivityNotFou

(1)

Android捕获全局异常信

(1)

Android之监测手机网络

(1)

推荐文章

\* 【观点】第二十三期：程序员应该如何积累财富？

\* Android检查更新下载安装

\* 动手打造史上最简单的Recycleview 侧滑菜单

\* TCP网络通讯如何解决分包粘包问题

\* SDCC 2017之大数据技术实战线上峰会

\* 快速集成一个视频直播功能

最新评论

Android 获取View在屏幕中的位置

Rgen\_xiao: 我是直接使用view.getx与view.gety。但没具体测试这两个坐标指向的具体位置。

JAVA字符串格式化-String.format

LuoCliper: mark~~~~

JAVA字符串格式化-String.format

一口大黑锅: 感谢

JAVA字符串格式化-String.format

Errol\_King: fab!

JAVA字符串格式化-String.format

lopez66: 好文章，得留名！！

JAVA字符串格式化-String.format

郁闷的中交兵: 收藏了谢谢分享

java-覆盖equals时规则不容忽视

我怀念的requesthouse: equals中的判断为null 是多余的哈，因为obj instanceof TestEquals ...

JAVA字符串格式化-String.format

Foolish\_nick: 谢谢

APK放到data/data/下面提示解析

hello\_1s: 还是没懂诶，我也遇到了相同的问题，求帮忙

JAVA字符串格式化-String.format

b20111102: 收藏

%n 换行符

%tx 日期与时间类型 ( x代表不同的日期与时间转换符)

测试用例

[java]

01. public static void main(String[] args) {

02. String str=null;

03. str=String.format("Hi,%s", "王力");

04. System.out.println(str);

05. str=String.format("Hi,%s:%s.%s", "王南","王力","王张");

06. System.out.println(str);

07. System.out.printf("字母a的大写是: %c %n", 'A');

08. System.out.printf("3>7的结果是: %b %n", 3>7);

09. System.out.printf("100的一半是: %d %n", 100/2);

10. System.out.printf("100的16进制数是: %x %n", 100);

11. System.out.printf("100的8进制数是: %o %n", 100);

12. System.out.printf("50元的书打8.5折扣是: %f 元%n", 50\*0.85);

13. System.out.printf("上面价格的16进制数是: %a %n", 50\*0.85);

14. System.out.printf("上面价格的指数表示: %e %n", 50\*0.85);

15. System.out.printf("上面价格的指数和浮点数结果的长度较短的是: %g %n", 50\*0.85);

16. System.out.printf("上面的折扣是%d%% %n", 85);

17. System.out.printf("字母A的散列码是: %h %n", 'A');

18. }

输出结果

[plain]

01. Hi,王力

02. Hi,王南:王力.王张

03. 字母a的大写是: A

04. 3>7的结果是: false

05. 100的一半是: 50

06. 100的16进制数是: 64

07. 100的8进制数是: 144

08. 50元的书打8.5折扣是: 42.500000 元

09. 上面价格的16进制数是: 0x1.54p5

10. 上面价格的指数表示: 4.250000e+01

11. 上面价格的指数和浮点数结果的长度较短的是: 42.5000

12. 上面的折扣是85%

13. 字母A的散列码是: 41

搭配转换符的标志，如图所示。

标志	说明	示例	结果
+	为正数或者负数添加符号	("%+d",15)	+15
-	左对齐	("%-5d",15)	15
0	数字前面补0	("%04d", 99)	0099
空格	在整数之前添加指定数量的空格	("% 4d", 99)	99
,	以“,”对数字分组	("%,f", 9999.99)	9,999.990000
(	使用括号包含负数	("%(f", -99.99)	(99.990000)
#	如果是浮点数则包含小数点，如果是16进制或8进制则添加0x或0	("%#x", 99)	0x63
		("%#o", 99)	0143

<	格式化前一个转换符所描述的参数	("%"和%"<3.2f", 99.450000和99.45)	99.45
\$	被格式化的参数索引	("%"1\$d,%"2\$s", 99,"abc")	99,abc

测试用例

```
[java]
01. public static void main(String[] args) {
02.     String str=null;
03.     //$使用
04.     str=String.format("格式参数$的使用: %1$d,%2$s", 99,"abc");
05.     System.out.println(str);
06.     //$+使用
07.     System.out.printf("显示正负数的符号: %+d与%d%n", 99,-99);
08.     //$0使用
09.     System.out.printf("最牛的编号是: %03d%n", 7);
10.     //$空格使用
11.     System.out.printf("Tab键的效果是: % 8d%n", 7);
12.     //$使用
13.     System.out.printf("整数分组的效果是: %,d%n", 9989997);
14.     //$空格和小数点后面个数
15.     System.out.printf("一本书的价格是: % 50.5f元%n", 49.8);
16. }
```

输出结果

```
[plain]
01. 格式参数$的使用: 99,abc
02. 显示正负数的符号: +99与-99
03. 最牛的编号是: 007
04. Tab键的效果是:      7
05. 整数分组的效果是: 9,989,997
06. 一本书的价格是:           49.800000元
```

日期和事件字符串格式化

在程序界面中经常需要显示时间和日期，但是其显示的 格式经常不尽人意，需要编写大量的代码经过各种算法才得到理想的日期与时间格式。字符串格式中还有%tx转换符没有详细介绍，它是专门用来格式化日期和时 间的。%tx转换符中的x代表另外的处理日期和时间格式的转换符，它们的组合能够将日期和时间格式化成多种格式。

常见日期和时间组合的格式，如图所示。

转 换 符	说 明	示 例
c	包括全部日期和时间信息	星期六 十月 27 14:21:20 CST 2007
F	“年-月-日”格式	2007-10-27
D	“月/日/年”格式	10/27/07
r	“HH:MM:SS PM”格式（ 12时制 ）	02:25:51 下午
T	“HH:MM:SS”格式（ 24时制 ）	14:28:16
R	“HH:MM”格式（ 24时制 ）	14:28

测试用例

[java]

```

01. public static void main(String[] args) {
02.     Date date=new Date();
03.     //c的使用
04.     System.out.printf("全部日期和时间信息: %tc%n",date);
05.     //f的使用
06.     System.out.printf("年-月-日格式: %tF%n",date);
07.     //d的使用
08.     System.out.printf("月/日/年格式: %tD%n",date);
09.     //r的使用
10.     System.out.printf("HH:MM:SS PM格式 (12时制): %tr%n",date);
11.     //t的使用
12.     System.out.printf("HH:MM:SS格式 (24时制): %tT%n",date);
13.     //R的使用
14.     System.out.printf("HH:MM格式 (24时制): %tR",date);
15. }

```

输出结果

[plain]

```

01. 全部日期和时间信息: 星期一 九月 10 10:43:36 CST 2012
02. 年-月-日格式: 2012-09-10
03. 月/日/年格式: 09/10/12
04. HH:MM:SS PM格式 (12时制): 10:43:36 上午
05. HH:MM:SS格式 (24时制): 10:43:36
06. HH:MM格式 (24时制): 10:43

```

定义日期格式的转换符可以使日期通过指定的转换符生成新字符串。这些日期转换符如图所示。

[java]

```

01. public static void main(String[] args) {
02.     Date date=new Date();
03.     //b的使用, 月份简称
04.     String str=String.format(Locale.US, "英文月份简称: %tb",date);
05.     System.out.println(str);
06.     System.out.printf("本地月份简称: %tb%n",date);
07.     //B的使用, 月份全称
08.     str=String.format(Locale.US, "英文月份全称: %tB",date);
09.     System.out.println(str);
10.     System.out.printf("本地月份全称: %tB%n",date);
11.     //a的使用, 星期简称
12.     str=String.format(Locale.US, "英文星期的简称: %ta",date);
13.     System.out.println(str);
14.     //A的使用, 星期全称
15.     System.out.printf("本地星期的简称: %tA%n",date);
16.     //C的使用, 年前两位
17.     System.out.printf("年的前两位数字 (不足两位前面补0): %tC%n",date);
18.     //y的使用, 年后两位
19.     System.out.printf("年的后两位数字 (不足两位前面补0): %ty%n",date);
20.     //j的使用, 一年的天数
21.     System.out.printf("一年中的天数 (即年的第几天): %tj%n",date);
22.     //m的使用, 月份
23.     System.out.printf("两位数字的月份 (不足两位前面补0): %tm%n",date);
24.     //d的使用, 日 (二位, 不够补零)
25.     System.out.printf("两位数字的日 (不足两位前面补0): %td%n",date);
26.     //e的使用, 日 (一位不补零)
27.     System.out.printf("月份的日 (前面不补0): %te",date);
28. }

```

输出结果

[plain]

```

01. 英文月份简称: Sep
02. 本地月份简称: 九月
03. 英文月份全称: September
04. 本地月份全称: 九月
05. 英文星期的简称: Mon
06. 本地星期的简称: 星期一
07. 年的前两位数字 (不足两位前面补0): 20
08. 年的后两位数字 (不足两位前面补0): 12
09. 一年中的天数 (即年的第几天): 254
10. 两位数字的月份 (不足两位前面补0): 09

```

关闭

- 11. 两位数字的日（不足两位前面补0）： 10
- 12. 月份的日（前面不补0）： 10

和日期格式转换符相比，时间格式的转换符要更多、更精确。它可以将时间格式化成时、分、秒甚至时毫秒等单位。格式化时间字符串的转换符如图所示。

转 换 符	说 明	示 例
H	2位数字24时制的小时（不足2位前面补0）	15
I	2位数字12时制的小时（不足2位前面补0）	03
k	2位数字24时制的小时（前面不补0）	15
l	2位数字12时制的小时（前面不补0）	3
M	2位数字的分钟（不足2位前面补0）	03
S	2位数字的秒（不足2位前面补0）	09
L	3位数字的毫秒（不足3位前面补0）	015
N	9位数字的毫秒数（不足9位前面补0）	562000000
p	小写字母的上午或下午标记	中：下午 英：pm
z	相对于GMT的RFC822时区的偏移量	+0800
Z	时区缩写字符串	CST
s	1970-1-1 00:00:00 到现在所经过的秒数	1193468128
Q	1970-1-1 00:00:00 到现在所经过的毫秒数	1193468128984

测试代码

```
[java]
01. public static void main(String[] args) {
02.     Date date = new Date();
03.     //H的使用
04.     System.out.printf("2位数字24时制的小时（不足2位前面补0）:%tH%n", date);
05.     //I的使用
06.     System.out.printf("2位数字12时制的小时（不足2位前面补0）:%tI%n", date);
07.     //k的使用
08.     System.out.printf("2位数字24时制的小时（前面不补0）:%tk%n", date);
09.     //l的使用
10.     System.out.printf("2位数字12时制的小时（前面不补0）:%tl%n", date);
11.     //M的使用
12.     System.out.printf("2位数字的分钟（不足2位前面补0）:%tM%n", date);
13.     //S的使用
14.     System.out.printf("2位数字的秒（不足2位前面补0）:%tS%n", date);
15.     //L的使用
16.     System.out.printf("3位数字的毫秒（不足3位前面补0）:%tL%n", date);
17.     //N的使用
18.     System.out.printf("9位数字的毫秒数（不足9位前面补0）:%tN%n", date);
19.     //p的使用
```

```
20.     String str = String.format(Locale.US, "小写字母的上午或下午标记(英): %tp", date);
21.     System.out.println(str);
22.     System.out.printf("小写字母的上午或下午标记(中): %tp%n", date);
23.     //z的使用
24.     System.out.printf("相对于GMT的RFC822时区的偏移量:%tz%n", date);
25.     //Z的使用
26.     System.out.printf("时区缩写字符串:%tZ%n", date);
27.     //s的使用
28.     System.out.printf("1970-1-1 00:00:00 到现在所经过的秒数: %ts%n", date);
29.     //Q的使用
30.     System.out.printf("1970-1-1 00:00:00 到现在所经过的毫秒数: %tQ%n", date);
31. }
```

输出结果

```
[plain]
01. 2位数字24时制的小时(不足2位前面补0):11
02. 2位数字12时制的小时(不足2位前面补0):11
03. 2位数字24时制的小时(前面不补0):11
04. 2位数字12时制的小时(前面不补0):11
05. 2位数字的分钟(不足2位前面补0):03
06. 2位数字的秒(不足2位前面补0):52
07. 3位数字的毫秒(不足3位前面补0):773
08. 9位数字的毫秒数(不足9位前面补0):773000000
09. 小写字母的上午或下午标记(英): am
10. 小写字母的上午或下午标记(中): 上午
11. 相对于GMT的RFC822时区的偏移量:+0800
12. 时区缩写字符串:CST
13. 1970-1-1 00:00:00 到现在所经过的秒数: 1347246232
14. 1970-1-1 00:00:00 到现在所经过的毫秒数: 1347246232773
```

顶 踩  
137 6

上一篇 Android 3.0系统上 解密GMail动画  
下一篇 Android LOG标签自定义

相关文章推荐

- String.format()方法使用说明
- Python全栈工程师特训班--韦玮
- CString类Format ( ) 的用法
- Blink在阿里集团的应用实践--陈守元
- string.format()函数的用法
- Vue2.x知识点面面通
- String.Format用法
- 大型Web构架设计案例解析
- 发现和记录--String.Format 将指定的 String 中的...
- 机器学习案例实战--欺诈检测
- JAVA字符串占位符替换
- Android开发实战30分钟集成第三方SDK
- Java使用占位符拼接字符串
- 我的Java开发学习之旅----->Java 格式化类(java....
- java字符串格式化String.format()
- JAVA字符串格式化-String.format()的使用

查看评论

44楼 [LuoCliper](#) 2017-09-02 18:11发表



mark~~~~

43楼 [一口大黑锅](#) 2017-08-28 10:51发表



感谢

42楼 [Errol\\_King](#) 2017-08-18 14:36发表



fab!

41楼 [lopez66](#) 2017-07-20 12:04发表



好文章，得留名！！

40楼 [郁闷的中交兵](#) 2017-05-08 10:35发表



收藏了谢谢分享

39楼 [Foolish\\_nick](#) 2017-03-09 15:43发表



谢谢

38楼 [b20111102](#) 2017-02-22 14:22发表



收藏

37楼 [尼玛旦](#) 2017-01-13 22:46发表



感谢整理分享

36楼 [Phoebe星丫头](#) 2017-01-03 21:04发表



学习了，还是很全的

35楼 [feiyuDragon](#) 2016-12-02 18:58发表



文章是不错，但是不够全，找不到字符串前补齐的

34楼 [SEU\\_Calvin](#) 2016-10-25 14:16发表



学习了

33楼 [SEU\\_Calvin](#) 2016-10-25 14:16发表



不错

32楼 [yishuyuaxue](#) 2016-10-08 17:27发表



感谢分享

31楼 [qq\\_16436021](#) 2016-08-19 17:22发表



屌

30楼 [crane-yuan](#) 2016-08-09 16:59发表



转载了，欢迎访问我的博客：<a href="http://blog.csdn.net/u014717036" target="\_blank">Java开发猫</a>

29楼 [梦Dreamer](#) 2016-07-26 11:50发表



真棒，不错的文章

28楼 [没水没电没空调](#) 2016-07-21 19:59发表



写的很详细，对我有很大帮助，不错的文章

27楼 [G987507](#) 2016-06-20 14:06发表



好帖，收藏+转载了。

26楼 [bjjq1](#) 2016-06-14 14:34发表



很详细，感谢分享

25楼 Junzz 2016-06-03 13:53发表



赞，好详细呀！！

24楼 SafeWinter 2016-05-26 12:29发表



学习了，总结得很好，谢谢！

23楼 Chitty\_Tina 2016-05-26 11:05发表



棒棒哒~ 赞赞赞~

22楼 daiyhong 2016-03-15 09:51发表



楼主，这一项有问题：“+ 为正数或者负数添加符号 ("%d",15) +15”，你应该写成，为正数添加符号，因为不行的，而且负数自带符号，不需要再另行添加

21楼 我还是太连清啊 2016-01-06 21:19发表



很详细，博主辛苦了

20楼 haonanhaomafan 2015-11-27 16:25发表



666

19楼 a417074865 2015-11-04 11:04发表



好贴，转载了，我会标明出处的哈。

Re: a417074865 2015-11-04 11:13发表



楼主我能不能转载

18楼 麦小麦ing 2015-10-21 18:17发表



String类型前补0怎么补？？？

17楼 wlccomeon 2015-10-04 10:37发表



晕圈了~

16楼 sunny-sky 2015-09-15 11:13发表



赞一个

15楼 j\_a\_n\_v\_i 2015-09-11 17:42发表

janvi good

14楼 laonew4 2015-09-10 09:09发表



谢谢分享

13楼 米玖 2015-08-31 08:53发表



不错的文章

12楼 mumuqingwei 2015-08-12 10:47发表



很不错，收了。

11楼 neodi 2015-05-05 16:07发表



没有找到点赞按钮 打字感谢吧

10楼 baidu\_26974557 2015-03-29 17:06发表



非常感谢，对我帮助很大，写的很好

9楼 ZtianyAndroid 2014-09-04 11:29发表



很好



8楼 ONLY24G 2014-08-28 14:58发表



写的非常的好，简单明了一看就懂，感谢楼主分享啊

7楼 JWS012 2014-08-05 17:28发表



不错，写的很详细，一下子就看明白了！

6楼 Romen 2014-08-05 14:08发表



感谢分享

5楼 水煮时代 2014-07-29 10:57发表



透彻 简单明了

4楼 ouyango321 2014-07-25 14:37发表



不错

3楼 BlueBerryVanilla 2014-07-07 14:51发表



总结的很全面

2楼 dy00544 2014-06-11 09:44发表



不错，写的很详细  
c++也基本差不多。

1楼 goumenzi13 2014-03-31 11:04发表



不错的文章，找半天了！

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

\* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

公司简介 | 招贤纳士 | 广告服务 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-660-0108 | 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 |

江苏乐知网络技术有限公司

京 ICP 证 09002463 号 | Copyright © 1999-2017, CSDN.NET, All Rights Reserved