## 格式化工具类 dateformat

静态字段：

static int LONG 2017年7月25日 下午07时23分07秒 （时间日期均为long）

Constant for long style pattern.

static int MEDIUM 2017-7-25 19:20:13（日期时间的输出格式均为medium）

Constant for medium style pattern.

static int SHORT 17-7-25 下午7:28 （日期时间的输出均为short）

Constant for short style pattern.

实例化：不提供构造函数，通过提供静态方法获取

日期时间的示例化：传入日期时间获得日期时间

public final static DateFormat getDateTimeInstance()

public final static DateFormat getDateTimeInstance(int dateStyle, int timeStyle)

public final static DateFormat getDateTimeInstance(int dateStyle, int timeStyle, Locale aLocale)

DateFormat dateTimeInstance = DateFormat.getDateTimeInstance();

System.out.println(dateTimeInstance.format(date));

输出：2017-7-25 19:20:13 即默认时间日期均为 medium

DateFormat I2 = DateFormat.getDateTimeInstance(DateFormat.LONG, DateFormat.LONG);

System.out.println(I2.format(date));

输出：2017年7月25日 下午07时23分07秒

DateFormat d3 = DateFormat.getDateTimeInstance(DateFormat.SHORT, DateFormat.MEDIUM);

System.out.println(d3.format(date));

输出：17-7-25 19:25:45

DateFormat d4 = DateFormat.getDateTimeInstance(DateFormat.MEDIUM, DateFormat.SHORT);

System.out.println(d4.format(date));

输出：2017-7-25 下午7:27

时间的实例化：传入日期时间值只输出时间 和日期时间的实例化类似

**public** **final** **static** DateFormat getTimeInstance()

{

**return** *get*(***DEFAULT***, 0, 1, Locale.*getDefault*(Locale.Category.***FORMAT***));

}

例如：

DateFormat timeInstance = DateFormat.*getTimeInstance*();

Date date = **new** Date(System.*currentTimeMillis*());

System.***out***.println(timeInstance.format(date));

输出：18:55:31

日期的实例化：传入日期时间值输出日期 和日期时间的实例化类似

public final static DateFormat getDateInstance()

{

return get(0, DEFAULT, 2, Locale.getDefault(Locale.Category.FORMAT));

}

例如：

DateFormat dateInstance = DateFormat.getDateInstance();

System.out.println(dateInstance.format(date));

输出：2017-7-25

方法：

Format（date）；

Parse（string datetime）；替代new date（string datetime）

Parse 方法的使用：

Time 字符串为任意模式的时间都可以（short long medium）

dateTimeInstance.parse(time) 默认解析为date short模式的时间，

即输出此格式：Fri Jan 13 10:00:00 CST 2017