# date

显示或设置系统时间与日期

## 概要

date [OPTION]... [+FORMAT]  
date [-u|--utc|--universal] [MMDDhhmm[[CC]YY][.ss]]

## 主要用途

* 转换时间到选定的格式，默认为当前。
* 设置系统时间。

## 参数

format：输出的时间格式。

format可用的转义序列如下：  
  
%% 百分号  
%a 当地缩写的工作日名称（例如，Sun）  
%A 当地完整的工作日名称（例如，Sunday）  
%b 当地缩写的月份名称（例如，Jan）  
%B 当地完整的月份名称（例如，January）  
%c 当地的日期和时间（例如，Thu Mar 3 23:05:25 2005）  
%C 世纪，和%Y类似，但是省略后两位（例如，20）  
%d 一月中的一天（例如，01）  
%D 日期，等价于%m/%d/%y  
%e 一月中的一天，格式使用空格填充，等价于%\_d  
%F 完整的日期；等价于%+4Y-%m-%d  
%g ISO标准计数周的年份的最后两位数字  
%G ISO标准计数周的年份，通常只对%V有用  
%h 等价于%b  
%H 小时，范围（00..23）  
%I 小时，范围（00..23）  
%j 一年中的一天，范围（001..366）  
%k 小时，使用空格填充，范围（0..23），等价于%\_H  
%l 小时，使用空格填充，范围（1..12），等价于%\_I  
%m 月，范围（01..12）  
%M 分钟，范围（00..59）  
%n 换行符  
%N 纳秒，范围（000000000..000000000）  
%p 用于表示当地的AM或PM，如果未知则为空白  
%P 类似于%p，但用小写表示  
%q 季度，范围（1..4）  
%r 当地以12小时表示的时钟时间（例如，11:11:04 PM）  
%R 24小时每分钟；等价于%H:%M  
%s 自协调世界时1970年01月01日00时00分以来的秒数  
%S 秒数，范围（00..60）  
%t 水平制表符  
%T 时间；等价于%H:%M:%S  
%u 一周中的一天（1..7），1代表星期一  
%U 一年中的第几周，周日作为一周的起始（00..53）  
%V ISO标准计数周，该方法将周一作为一周的起始（01..53）  
%w 一周中的一天（0..6），0代表星期天  
%W 一年中的第几周，周一作为一周的起始（00..53）  
%x 当地的日期表示（例如，12/31/99）  
%X 当地的时间表示（例如，23:13:48）  
%y 年份后两位数字，范围（00..99）  
%Y 年份  
%z +hhmm格式的数值化时区格式（例如，-0400）  
%:z +hh:mm格式的数值化时区格式（例如，-04:00）  
%::z +hh:mm:ss格式的数值化时区格式（例如，-04:00:00）  
%:::z 数值化时区格式，相比上一个格式增加':'以显示必要的精度（例如，-04，+05:30）  
%Z 时区缩写（如EDT）  
  
默认情况下，日期用零填充数字字段；以下可选的符号可以跟在'%'后面:  
  
- (连字符) 不要填充相应的字段。  
\_ (下划线) 使用空格填充相应的字段。  
0 (数字0) 使用数字0填充相应的字段。  
+ 用数字0填充，未来年份大于4位数字则在前面加上'+'号。  
^ 允许的情况下使用大写。  
# 允许的情况下将默认的大写转换为小写，默认的小写转换为大写。  
  
在任何标志之后都有一个可选的字段宽度，如小数；然后是一个可选的修饰符，在可用的情况下，使用E来使用当地语言环境的替代表示，  
使用O来使用当地语言环境的替代数字符号。

## 选项

长选项与短选项等价  
  
-d, --date=STRING 解析字符串并按照指定格式输出，字符串不能是'now'。  
--debug 注释已解析的日期，并将有疑问的用法发送到标准错误。  
-f, --file=DATEFILE 类似于--date; 一次从DATEFILE处理一行。  
-I[FMT], --iso-8601[=FMT] 按照ISO 8601格式输出，FMT可以为'date'(默认)，'hours'，'minutes'，'seconds'，'ns'。例如：2006-08-14T02:34:56-06:00  
-R, --rfc-email 按照RFC 5322格式输出，例如: Mon, 14 Aug 2006 02:34:56 -0600  
--rfc-3339=FMT 按照RFC 3339格式输出，FMT可以为'date', 'seconds','ns'中的一个，例如：2006-08-14 02:34:56-06:00  
-r, --reference=FILE 显示文件的上次修改时间。  
-s, --set=STRING 根据字符串设置系统时间。  
-u, --utc, --universal 显示或设置世界协调时(UTC)。  
--help 显示帮助信息并退出。  
--version 显示版本信息并退出。

## 返回值

返回状态为成功除非给出了非法选项或非法参数。

## 例子

# 格式化输出：  
date +"%Y-%m-%d"  
2009-12-07  
  
# 输出昨天日期：  
date -d "1 day ago" +"%Y-%m-%d"  
2012-11-19  
  
# 2秒后输出：  
date -d "2 second" +"%Y-%m-%d %H:%M.%S"  
2012-11-20 14:21.31  
  
# 传说中的 1234567890 秒：  
date -d "1970-01-01 1234567890 seconds" +"%Y-%m-%d %H:%M:%S"  
# 或者  
date -d@1234567890 +"%F %T"  
# 输出结果  
2009-02-13 23:02:30  
  
# 时间格式转换：  
date -d "2009-12-12" +"%Y/%m/%d %H:%M.%S"  
# 输出结果  
2009/12/12 00:00.00  
  
# apache格式转换：  
date -d "Dec 5, 2009 12:00:37 AM" +"%Y-%m-%d %H:%M.%S"  
# 输出结果  
2009-12-05 00:00.37  
  
# 格式转换后时间游走：  
date -d "Dec 5, 2009 12:00:37 AM 2 year ago" +"%Y-%m-%d %H:%M.%S"  
# 输出结果  
2007-12-05 00:00.37  
  
# 时间加减操作：  
date +%Y%m%d # 显示年月日  
date -d "+1 day" +%Y%m%d # 显示前一天的日期  
date -d "-1 day" +%Y%m%d # 显示后一天的日期  
date -d "-1 month" +%Y%m%d # 显示上一月的日期  
date -d "+1 month" +%Y%m%d # 显示下一月的日期  
date -d "-1 year" +%Y%m%d # 显示前一年的日期  
date -d "+1 year" +%Y%m%d # 显示下一年的日期  
  
# 设定时间：  
date -s # 设置当前时间，只有root权限才能设置，其他只能查看  
date -s 20120523 # 设置成20120523，这样会把具体时间设置成00:00:00  
date -s 01:01:01 # 设置具体时间，不会对日期做更改  
date -s "01:01:01 2012-05-23" # 这样可以设置全部时间  
date -s "01:01:01 20120523" # 这样可以设置全部时间  
date -s "2012-05-23 01:01:01" # 这样可以设置全部时间  
date -s "20120523 01:01:01" # 这样可以设置全部时间  
  
# 有时需要检查一组命令花费的时间：  
start=$(date +%s)  
nmap wangchujiang.com &> /dev/null  
end=$(date +%s)  
difference=$(( end - start ))  
# 显示执行时间  
echo $difference seconds.  
  
# 当你考虑输出带有时间的字符串时，例如（Current time: 2019/05/19）：  
# 通常使用的方法：  
echo "Current time: $(date +"%Y/%m/%d")"  
# 另一种方法：  
suffix='Current time:'  
# 注意如果换成单引号就不能替换变量了。  
date +"${suffix} %Y/%m/%d"

### 注意

1. 该命令是GNU coreutils包中的命令，相关的帮助信息请查看man -s 1 date或info coreutils 'date invocation'。