

<https://pkg.go.dev/github.com/gogf/gf/v2/os/gtime>

1. TimestampTimestampMilliTimestampMicroTimestampNano
2. DateDatetime
3. SetTimeZone
- 4.

1

```
package main

import (
    "fmt"
    "github.com/gogf/gf/v2/os/gtime"
)

func main() {
    fmt.Println("Date      :", gtime.Date())
    fmt.Println("Datetime   :", gtime.Datetime())
    fmt.Println("Second      :", gtime.Timestamp())
    fmt.Println("Millisecond:", gtime.TimestampMilli())
    fmt.Println("Microsecond:", gtime.TimestampMicro())
    fmt.Println("Nanosecond :", gtime.TimestampNano())
}
```

```
Date      : 2018-07-22
Datetime   : 2018-07-22 11:52:22
Second     : 1532231542
Millisecond: 1532231542688
Microsecond: 1532231542688688
Nanosecond : 1532231542688690259
```

2StrToTime

NewStrToTimegtime.Time

```
2017-12-14 04:51:34 +0805 LMT
2017-12-14 04:51:34 +0805 LMT
2006-01-02T15:04:05Z07:00
2014-01-17T01:19:15+08:00
2018-02-09T20:46:17.897Z
2018-02-09 20:46:17.897
2018-02-09T20:46:17Z
2018-02-09 20:46:17
2018/10/31 - 16:38:46
2018-02-09
2018.02.09

01-Nov-2018 11:50:28
01/Nov/2018 11:50:28
01.Nov.2018 11:50:28
01.Nov.2018:11:50:28
'-''/''.'
```

Content Menu

- [1](#)
- [2StrToTime](#)

```

package main

import (
    "fmt"
    "github.com/gogf/gf/v2/os/glog"
    "github.com/gogf/gf/v2/os/gtime"
    "time"
)

func main() {
    array := []string{
        "2017-12-14 04:51:34 +0805 LMT",
        "2006-01-02T15:04:05Z07:00",
        "2014-01-17T01:19:15+08:00",
        "2018-02-09T20:46:17.897Z",
        "2018-02-09 20:46:17.897",
        "2018-02-09T20:46:17Z",
        "2018-02-09 20:46:17",
        "2018.02.09 20:46:17",
        "2018-02-09",
        "2017/12/14 04:51:34 +0805 LMT",
        "2018/02/09 12:00:15",
        "01/Nov/2018:13:28:13 +0800",
        "01-Nov-2018 11:50:28 +0805 LMT",
        "01-Nov-2018T15:04:05Z07:00",
        "01-Nov-2018T01:19:15+08:00",
        "01-Nov-2018 11:50:28 +0805 LMT",
        "01/Nov/2018 11:50:28",
        "01/Nov/2018:11:50:28",
        "01.Nov.2018:11:50:28",
        "01/Nov/2018",
    }
    cstLocal, _ := time.LoadLocation("Asia/Shanghai")
    for _, s := range array {
        if t, err := gtime.StrToTime(s); err == nil {
            fmt.Println(s)
            fmt.Println(t.UTC().String())
            fmt.Println(t.In(cstLocal).String())
        } else {
            glog.Error(s, err)
        }
        fmt.Println()
    }
}

```

StrToTimegtime.TimeUTCST()

```

2017-12-14 04:51:34 +0805 LMT
2017-12-13 20:46:34
2017-12-14 04:46:34 +0800 CST

2006-01-02T15:04:05Z07:00
2006-01-02 22:04:05
2006-01-03 06:04:05 +0800 CST

2014-01-17T01:19:15+08:00
2014-01-16 17:19:15
2014-01-17 01:19:15 +0800 CST

2018-02-09T20:46:17.897Z
2018-02-09 20:46:17
2018-02-10 04:46:17.897 +0800 CST

2018-02-09 20:46:17.897
2018-02-09 12:46:17
2018-02-09 20:46:17.897 +0800 CST

2018-02-09T20:46:17Z

```

2018-02-09 20:46:17
2018-02-10 04:46:17 +0800 CST

2018-02-09 20:46:17
2018-02-09 12:46:17
2018-02-09 20:46:17 +0800 CST

2018.02.09 20:46:17
2018-02-09 12:46:17
2018-02-09 20:46:17 +0800 CST

2018-02-09
2018-02-08 16:00:00
2018-02-09 00:00:00 +0800 CST

2017/12/14 04:51:34 +0805 LMT
2017-12-13 20:46:34
2017-12-14 04:46:34 +0800 CST

2018/02/09 12:00:15
2018-02-09 04:00:15
2018-02-09 12:00:15 +0800 CST

01/Nov/2018:13:28:13 +0800
2018-11-01 05:28:13
2018-11-01 13:28:13 +0800 CST

01-Nov-2018 11:50:28 +0805 LMT
2018-11-01 03:45:28
2018-11-01 11:45:28 +0800 CST

01-Nov-2018T15:04:05Z07:00
2018-11-01 22:04:05
2018-11-02 06:04:05 +0800 CST

01-Nov-2018T01:19:15+08:00
2018-10-31 17:19:15
2018-11-01 01:19:15 +0800 CST

01-Nov-2018 11:50:28 +0805 LMT
2018-11-01 03:45:28
2018-11-01 11:45:28 +0800 CST

01/Nov/2018 11:50:28
2018-11-01 03:50:28
2018-11-01 11:50:28 +0800 CST

01/Nov/2018:11:50:28
2018-11-01 03:50:28
2018-11-01 11:50:28 +0800 CST

01.Nov.2018:11:50:28
2018-11-01 03:50:28
2018-11-01 11:50:28 +0800 CST

01/Nov/2018
2018-10-31 16:00:00
2018-11-01 00:00:00 +0800 CST