

gi18n ()

GFI18Ngi18n

```
import "github.com/gogf/gf/i18n/gi18n"
```

<https://godoc.org/github.com/gogf/gf/i18n/gi18n>

gi18nxml/ini/yaml/toml/jsontoml

gi18nil18nSetPathi18n

i18nen.toml/ja.toml/zh-CN.tomlen/editor.toml/en/user.tomlzh-CN/editor.toml/zh-CN/user.toml

i18n

```
i18n
  en.toml
  ja.toml
  ru.toml
  zh-CN.toml
  zh-TW.toml
i18n-dir
  en
    hello.toml
    world.toml
  ja
    hello.yaml
    world.yaml
  ru
    hello.ini
    world.ini
  zh-CN
    hello.json
    world.json
  zh-TW
    hello.xml
    world.xml
i18n-file
  en.toml
  ja.yaml
  ru.ini
  zh-CN.json
  zh-TW.xml
```

gi18ngresil18nil18n

g.I18n

g.I18n().T("#{hello} {#world}")

gi18ngi18n.New()i18n

```
i18n := gi18n.New()
i18n.T("#{hello} {#world}")
```

Content Menu

- -
 -
 -
-
- - T
 - Tf
 - Tfi
- I18N

T

TTranslateT

T/en/ja/ru/zh-CN/zh-TWManager

```
// T translates <content> with configured language and returns the translated
content.
// The parameter <language> specifies custom translation language ignoring
configured language.
func T(content string, language ...string) string
```

TT("hello")T("world")

T{#} 1. - ja.toml toml hello = "" world = "" - ru.toml toml hello = "" world = ""
- zh-CN.toml toml hello = "" world = "" 1. “go package main

```
import (
    "fmt"

    "github.com/gogf/gf/i18n/gil8n"
)

func main() {
    i18n := gil8n.New()
    i18n.SetLanguage("en")
    fmt.Println(i18n.Translate(`hello`))
    fmt.Println(i18n.Translate(`GF says: {#hello}{#world}!`))

    i18n.SetLanguage("ja")
    fmt.Println(i18n.Translate(`hello`))
    fmt.Println(i18n.Translate(`GF says: {#hello}{#world}!`))

    i18n.SetLanguage("ru")
    fmt.Println(i18n.Translate(`hello`))
    fmt.Println(i18n.Translate(`GF says: {#hello}{#world}!`))

    fmt.Println(i18n.Translate(`hello`, "zh-CN"))
    fmt.Println(i18n.Translate(`GF says: {#hello}{#world}!`, "zh-CN"))
}

```html
Hello
GF says: HelloWorld!

GF says: !

GF says: !

GF says: !
```
```

Tf

TfTranslateFormatfmtSprintf

```
// Tf translates, formats and returns the <format> with configured language
// and given <values>.
func Tf(format string, values ...interface{}) string
```

1.

- en.toml toml OrderPaid = "You have successfully complete order #%d payment, paid amount: %0.2f."
- zh-CN.toml toml OrderPaid = " #d %.2f"

2.

```

package main

import (
    "fmt"

    "github.com/gogf/gf/il8n/gil8n"
)

func main() {
    var (
        orderId      = 865271654
        orderAmount   = 99.8
    )

    il8n := gil8n.New()
    t.SetLanguage("en")
    fmt.Println(il8n.Tf(`#{OrderPaid}`, orderId, orderAmount))

    t.SetLanguage("zh-CN")
    fmt.Println(il8n.Tf(`#{OrderPaid}`, orderId, orderAmount))
}

You have successfully complete order #865271654 payment, paid amount:
99.80.
#865271654 99.80

```

il8n

Tf1

Tf1TranslateFormatLangfmtSprintf

```

// Tf1 translates, formats and returns the <format> with configured language
// and given <values>. The parameter <language> specifies custom translation
// language ignoring
// configured language. If <language> is given empty string, it uses the
// default configured
// language for the translation.
func Tf1(language string, format string, values ...interface{}) string

```

1.

- en.toml toml OrderPaid = "You have successfully complete order #%d payment, paid amount: %0.2f."
- zh-CN.toml toml OrderPaid = " #d %.2f"

2.

```

package main

import (
    "fmt"
    "github.com/gogf/gf/frame/g"
)

func main() {
    var (
        orderId      = 865271654
        orderAmount   = 99.8
    )
    fmt.Println(g.I18n().Tf1(`en`, `#{OrderPaid}`, orderId, orderAmount))
    fmt.Println(g.I18n().Tf1(`zh-CN`, `#{OrderPaid}`, orderId, orderAmount))
}

```

You have successfully complete order #865271654 payment, paid amount:
99.80.
#865271654 99.80

i18n

I18N

gil8nGF/gil8n

I18nLanguage

```
package main

import (
    "github.com/gogf/gf/frame/g"
    "github.com/gogf/gf/net/ghttp"
)

func main() {
    s := g.Server()
    s.BindHandler("/", func(r *ghttp.Request) {
        r.Response.WriteTplContent(`GF says: {#hello}{#world}!`, g.Map{
            "I18nLanguage": r.Get("lang", "zh-CN"),
        })
    })
    s.SetPort(8199)
    s.Run()
}
```

1. <http://127.0.0.1:8199>

GF says: !

2. <http://127.0.0.1:8199/?lang=ja>

GF says: !