

# gtest ()

gtest

```
import "github.com/gogf/gf/test/gtest"
```

<https://godoc.org/github.com/gogf/gf/test/gtest>

```
func C(t *testing.T, f func(t *T))
func Assert(value, expect interface{})
func AssertEQ(value, expect interface{})
func AssertGE(value, expect interface{})
func AssertGT(value, expect interface{})
func AssertIN(value, expect interface{})
func AssertLE(value, expect interface{})
func AssertLT(value, expect interface{})
func AssertNE(value, expect interface{})
func AssertNI(value, expect interface{})
func Error(message ...interface{})
func Fatal(message ...interface{})
```

1. CCaseCC
2. AssertAssertEQ
3. AssertGE
4. AssertINAssertNIslicemap

```
_testGo/Test*
```

gstr

```
package gstr_test
```

```
import (
    "github.com/gogf/gf/test/gtest"
    "github.com/gogf/gf/text/gstr"
    "testing"
)
```

```
func Test_Trim(t *testing.T) {
    gtest.C(t, func(t *T) {
        t.Assert(gstr.Trim(" 123456\n "), "123456")
        t.Assert(gstr.Trim("#123456#;", "#;"), "123456")
    })
}
```

```
package gstr_test
```

```
import (
    . "github.com/gogf/gf/test/gtest"
    "github.com/gogf/gf/text/gstr"
    "testing"
)
```

```
func Test_Trim(t *testing.T) {
    C(t, func() {
        Assert(gstr.Trim(" 123456\n "), "123456")
        Assert(gstr.Trim("#123456#;", "#;"), "123456")
    })
}
```

CC PASS

```
=== RUN    Test_Trim
[ASSERT] EXPECT 123456#; == 123456
1. /Users/john/Workspace/Go/GOPATH/src/github.com/gogf/gf/text/gstr/gstr_z_unit_trim_test.go:20
2. /Users/john/Workspace/Go/GOPATH/src/github.com/gogf/gf/text/gstr/gstr_z_unit_trim_test.go:18
--- FAIL: Test_Trim (0.00s)
FAIL
```