-

```
package main

import (
        "context"
        "fmt"
        "github.com/gogf/gf/v2/os/gctx"
        "github.com/gogf/gf/v2/os/gtime"
        "github.com/gogf/gf/v2/os/gtimer"
        "time"
)

func main() {
        var (
                ctx = gctx.New()
                now = time.Now()
        )
        gtimer.AddTimes(ctx, time.Second, 10, func(ctx context.Context) {
                fmt.Println(gtime.Now(), time.Duration(time.Now().
UnixNano()-now.UnixNano()))
                now = time.Now()
        })

        select {}
}
```

:

```
2021-05-27 13:28:19 1.004516s
2021-05-27 13:28:20 997.262ms
2021-05-27 13:28:21 999.972ms
2021-05-27 13:28:22 1.00112s
2021-05-27 13:28:23 998.773ms
2021-05-27 13:28:24 999.957ms
2021-05-27 13:28:25 1.002468s
2021-05-27 13:28:26 997.468ms
2021-05-27 13:28:27 999.981ms
2021-05-27 13:28:28 1.002504s
```

Content Menu

-
-
-
- SetTimeoutSetInterval
- Exit

```
package main

import (
        "context"
        "github.com/gogf/gf/v2/os/gctx"
        "github.com/gogf/gf/v2/os/glog"
        "github.com/gogf/gf/v2/os/gtimer"
        "time"
)

func main() {
        var (
                ctx      = gctx.New()
                interval = time.Second
        )

        gtimer.AddSingleton(ctx, interval, func(ctx context.Context) {
                glog.Print(ctx, "doing")
                time.Sleep(5 * time.Second)
        })

        select {}
}
```

:

```
2021-11-14 11:50:42.192 {189cwi9mo40cfp73guzhugo100tnuedg} doing
2021-11-14 11:50:48.190 {189cwi9mo40cfp73guzhugo100tnuedg} doing
2021-11-14 11:50:54.192 {189cwi9mo40cfp73guzhugo100tnuedg} doing
2021-11-14 11:51:00.189 {189cwi9mo40cfp73guzhugo100tnuedg} doing
...
```

DelayAdd*

```
package main

import (
        "context"
        "fmt"
        "github.com/gogf/gf/v2/os/gctx"
        "github.com/gogf/gf/v2/os/gtime"
        "github.com/gogf/gf/v2/os/gtimer"
        "time"
)

func main() {
        var (
                ctx      = gctx.New()
                delay    = time.Second
                interval = time.Second
        )
        fmt.Println("Start:", gtime.Now())
        gtimer.DelayAdd(
                ctx,
                delay,
                interval,
                func(ctx context.Context) {
                        fmt.Println("Running:", gtime.Now())
                },
        )
        select {}
}
```

```
Start: 2021-05-27 13:26:02
Running: 2021-05-27 13:26:04
Running: 2021-05-27 13:26:05
Running: 2021-05-27 13:26:06
Running: 2021-05-27 13:26:07
Running: 2021-05-27 13:26:08
Running: 2021-05-27 13:26:09
Running: 2021-05-27 13:26:10
Running: 2021-05-27 13:26:11
...
```

SetTimeoutSetInterval

JavascriptSetTimeoutSetTimeoutSetIterval

```
package main

import (
        "context"
        "fmt"
        "github.com/gogf/gf/v2/os/gctx"
        "github.com/gogf/gf/v2/os/gtime"
        "github.com/gogf/gf/v2/os/gtimer"
        "time"
)

func main() {
        var (
                ctx      = gctx.New()
                timeout  = time.Second
                interval = time.Second
        )
        gtimer.SetTimeout(ctx, timeout, func(ctx context.Context) {
                fmt.Println("SetTimeout:", gtime.Now())
        })
        gtimer.SetInterval(ctx, interval, func(ctx context.Context) {
                fmt.Println("SetInterval:", gtime.Now())
        })
        select {}
}
```

```
SetInterval: 2021-05-27 13:20:50
SetTimeout: 2021-05-27 13:20:50
SetInterval: 2021-05-27 13:20:51
SetInterval: 2021-05-27 13:20:52
SetInterval: 2021-05-27 13:20:53
SetInterval: 2021-05-27 13:20:54
SetInterval: 2021-05-27 13:20:55
SetInterval: 2021-05-27 13:20:56
SetInterval: 2021-05-27 13:20:57
SetInterval: 2021-05-27 13:20:58
...
```

Exit

Exit

```
package main

import (
        "context"
        "fmt"
        "github.com/gogf/gf/v2/os/gctx"
        "github.com/gogf/gf/v2/os/gtime"
        "github.com/gogf/gf/v2/os/gtimer"
        "time"
)

func main() {
        var (
                ctx = gctx.New()
        )
        gtimer.SetInterval(ctx, time.Second, func(ctx context.Context) {
                fmt.Println("exit:", gtime.Now())
                gtimer.Exit()
        })
        select {}
}
```

```
exit: 2021-05-27 13:31:24
```