

gjsonJSONJsonJSON, XML, INI, YAML, TOML, PROPERTIESstructJson

NewLoad*<https://pkg.go.dev/github.com/gogf/gf/v2/encoding/gjson>

New

JSON

```
jsonContent := `{"name":"john", "score":"100"}`
j := gjson.New(jsonContent)
fmt.Println(j.Get("name"))
fmt.Println(j.Get("score"))
// Output:
// john
// 100
```

XML

```
jsonContent := `<?xml version="1.0" encoding="UTF-8"?><doc><name>john<
/name><score>100</score></doc>`
j := gjson.New(jsonContent)
// Note that there's root node in the XML content.
fmt.Println(j.Get("doc.name"))
fmt.Println(j.Get("doc.score"))
// Output:
// john
// 100
```

Strcut

```
type Me struct {
    Name string `json:"name"`
    Score int   `json:"score"`
}
me := Me{
    Name: "john",
    Score: 100,
}
j := gjson.New(me)
fmt.Println(j.Get("name"))
fmt.Println(j.Get("score"))
// Output:
// john
// 100
```

Struct

Content Menu

- [New](#)
 - [JSON](#)
 - [XML](#)
 - [Strcut](#)
 - [Struct](#)
- [Load*](#)
 - [Load](#)
 - [LoadContent](#)

```

type Me struct {
    Name string `tag:"name"`
    Score int `tag:"score"`
    Title string
}
me := Me{
    Name: "john",
    Score: 100,
    Title: "engineer",
}
// The parameter <tags> specifies custom priority tags for struct
// conversion to map,
// multiple tags joined with char ', '.
j := gjson.NewWithTag(me, "tag")
fmt.Println(j.Get("name"))
fmt.Println(j.Get("score"))
fmt.Println(j.Get("Title"))
// Output:
// john
// 100
// engineer

```

Load*

LoadLoadContentJsonJson

Load

1. JSON

```

jsonFilePath := gtest.DataPath("json", "data1.json")
j, _ := gjson.Load(jsonFilePath)
fmt.Println(j.Get("name"))
fmt.Println(j.Get("score"))

```

2. XML

```

jsonFilePath := gtest.DataPath("xml", "data1.xml")
j, _ := gjson.Load(jsonFilePath)
fmt.Println(j.Get("doc.name"))
fmt.Println(j.Get("doc.score"))

```

LoadContent

```

jsonContent := `{"name":"john", "score":"100"}`
j, _ := gjson.LoadContent(jsonContent)
fmt.Println(j.Get("name"))
fmt.Println(j.Get("score"))
// Output:
// john
// 100

```