

gjson

1

```
func main() {
    j := gjson.New(nil)
    j.Set("name", "John")
    j.Set("score", 99.5)
    fmt.Printf(
        "Name: %s, Score: %v\n",
        j.Get("name").String(),
        j.Get("score").Float32(),
    )
    fmt.Println(j.MustToJsonString())

    // Output:
    // Name: John, Score: 99.5
    // {"name":"John","score":99.5}
}
```

2

```
func main() {
    j := gjson.New(nil)
    for i := 0; i < 5; i++ {
        j.Set(fmt.Sprintf(`%d.id`, i), i)
        j.Set(fmt.Sprintf(`%d.name`, i), fmt.Sprintf(`student-%d`,
i))
    }
    fmt.Println(j.MustToJsonString())

    // Output:
    // [{"id":0,"name":"student-0"}, {"id":1,"name":"student-1"}, {"id":
2,"name":"student-2"}, {"id":3,"name":"student-3"}, {"id":4,"name":"student-
4"}]
}
```

Content Menu

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

```

func main() {
    data := `
    {
      "users" : {
        "count" : 2,
        "list" : [
          {"name" : "Ming", "score" : 60},
          {"name" : "John", "score" : 59}
        ]
      }
    }
  `
    if j, err := gjson.DecodeToJson(data); err != nil {
        panic(err)
    } else {
        j.Set("users.list.1.score", 100)
        fmt.Println("John Score:", j.Get("users.list.1.score").
Float32())
        fmt.Println(j.MustToJsonString())
    }
    // Output:
    // John Score: 100
    // {"users":{"count":2,"list":[{"name":"Ming","score":60},{"name":"
John","score":100}]}}
}

```

JSONgjsonSet/nilgjson/

gjson

1

```

func main() {
    j := gjson.New(nil)
    j.Set("name", "John")
    j.Set("score", 99.5)
    fmt.Printf(
        "Name: %s, Score: %v\n",
        j.Get("name").String(),
        j.Get("score").Float32(),
    )
    fmt.Println(j.MustToJsonString())

    // Output:
    // Name: John, Score: 99.5
    // {"name":"John","score":99.5}
}

```

2

Content Menu

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

```

func main() {
    j := gjson.New(nil)
    for i := 0; i < 5; i++ {
        j.Set(fmt.Sprintf(`%d.id`, i), i)
        j.Set(fmt.Sprintf(`%d.name`, i), fmt.Sprintf(`student-%d`,
i))
    }
    fmt.Println(j.MustToJsonString())

    // Output:
    // [{"id":0,"name":"student-0"}, {"id":1,"name":"student-1"}, {"id":
2,"name":"student-2"}, {"id":3,"name":"student-3"}, {"id":4,"name":"student-
4"}]
}

```

```

func main() {
    data :=
        `{
    "users" : {
        "count" : 2,
        "list" : [
            {"name" : "Ming", "score" : 60},
            {"name" : "John", "score" : 59}
        ]
    }
}`
    if j, err := gjson.DecodeToJson(data); err != nil {
        panic(err)
    } else {
        j.Set("users.list.1.score", 100)
        fmt.Println("John Score:", j.Get("users.list.1.score").
Float32())
        fmt.Println(j.MustToJsonString())
    }
    // Output:
    // John Score: 100
    // {"users":{"count":2,"list":[{"name":"Ming","score":60}, {"name":"
John","score":100}]}}
}

```

JSOnGjsonSet/nilgjson/