



URL Parsing



+

URL Parsing #1

Data string url bisa dikonversi kedalam bentuk `url.URL`.

Dengan menggunakan tipe tersebut akan ada banyak informasi yang bisa kita manfaatkan, diantaranya adalah jenis protokol yang digunakan, path yang diakses, query, dan lainnya.

Contohnya seperti pada gambar di sebelah kanan.

Function `url.Parse` digunakan untuk parsing string ke bentuk url. Mengembalikan 2 data, variabel objek bertipe `url.URL` dan error (jika ada).

Melalui variabel objek tersebut pengaksesan informasi url akan menjadi lebih mudah, contohnya seperti nama host bisa didapatkan lewat `u.Host`, protokol lewat `u.Scheme`, dan lainnya.

Selain itu, query yang ada pada url akan otomatis diparsing juga, menjadi bentuk `map[string][]string`, dengan key adalah nama elemen query, dan value array string yang berisikan value elemen query.

```
1 package main
2
3 import (
4     "fmt"
5     "net/url"
6 )
7
8 func main() {
9     var urlString = "http://developer.com:80/hello?name=airell&age=23"
10    var u, e = url.Parse(urlString)
11    if e != nil {
12        fmt.Println(e.Error())
13        return
14    }
15
16    fmt.Printf("url: %s\n", urlString)
17
18    fmt.Printf("protocol: %s\n", u.Scheme)
19    fmt.Printf("host: %s\n", u.Host)
20    fmt.Printf("path: %s\n", u.Path)
21
22    var name = u.Query()["name"][0]
23    var age = u.Query()["age"][0]
24    fmt.Printf("name: %s, age: %s\n", name, age)
25 }
26
```



URL Parsing- Sesi 8

URL Parsing #2

Ketika kita jalankan, maka hasilnya akan seperti pada gambar di bawah.

```
url: http://developer.com:80/hello?name=airell&age=23
protocol: http
host: developer.com:80
path: /hello
name: airell, age: 23
```