Go Strings

The Go string is a sequence of variable-width characters.

Go strings and text files occupy less memory or disk space.

Since, UTF-8 is the standard, Go doesn't need to encode and decode strings.

➔ Go Strings are value type, the initial value of string is empty "" by default.
➔ Strings are immutable, it means if you create a string you cannot modify the content of string.

String Example:-

```go
package main

import (
    "fmt"
    "reflect"
)

func main() {
    var x string = "Hello Gautam"
    fmt.Println(x)
    fmt.Println(reflect.TypeOf(x))
}
```

Output :-

```
Hello Gautam
string
```

Functions Of string :-

1. len():- This is a inbuilt function of string which returns the length of string.

```go
package main

import "fmt"

func main() {
    var str string
    str = "I love my country India"
    fmt.Println(str)
    fmt.Println(len(str))
}
```

2. ToUpper():- This function is used to convert the string into Upper alphabet character.

```go
package main

import (
    "fmt"
    "strings"
)

func main() {
    str := "India"
    fmt.Println(strings.ToUpper(str))
}
```

3. ToLower():- This function is used to convert the string into Lower alphabet character.

```go
package main
```

```
import (
    "fmt"
    "strings"
)

func main() {
    str := "INDIA"
    fmt.Println(strings.ToLower(str))
}
```

4. HasPrefix() :- This function is used to check if an substring is available at the starting of the original string or not.

```
package main

import (
    "fmt"
    "strings"
)

func main() {
    s := "INDIA"
    fmt.Println(strings.HasPrefix(s, "IN"))
    fmt.Println(strings.HasPrefix(s, "DI"))

}
```

5. HasSuffix() :- This function is used to check if an substring is available on the last of original string or not.

```
package main

import (
    "fmt"
    "strings"
)
```

```go
func main() {
    s := "INDIA"
    fmt.Println(strings.HasSuffix(s, "IA"))
    fmt.Println(strings.HasSuffix(s, "IND"))
    fmt.Println(strings.HasSuffix(s, "DIA"))
}
```

6. Join() :- This function is used to join two or more string.

```go
package main

import (
    "fmt"
    "strings"
)

func main() {
    var arr = []string{"a", "b", "c", "d"}
    fmt.Println(strings.Join(arr, "*"))
}
```

7. Repeat() :- This function is used to repeat the string number of times.

```go
package main

import (
    "fmt"
    "strings"
)

func main() {
    var str = "Gautam "
    fmt.Println(strings.Repeat(str, 5))
}
```

8. Contains() :- This function is used to Check if the original string contains a substring or not.

```go
package main

import (
    "fmt"
    "strings"
)

func main() {
    str := "Hi...there"
    fmt.Println(strings.Contains(str, "th"))
    fmt.Println(strings.Contains(str, "abc"))
}
```

9. Index() :- This function is used to get the index of a substring.

```go
package main

import (
    "fmt"
    "strings"
)

func main() {
    str := "Hi...there"
    fmt.Println(strings.Index(str, "he"))
}
```

Count():- This function is used to count a number of character appeared in the string.

```go
package main

import (
    "fmt"
    "strings"
```

```go
)

func main() {
    str := "Hello"
    fmt.Println(strings.Replace(str, "l", "y", 2))
}
```

NOTE :- There are lot more Functions available in "strings" package