

-

```
{{ }} gview.SetDelimiters
```

```
.
```

```
$
```

```
$var
```

**go**

```
{{"string"}} // string
{{`raw string`}} // string
{{'c'}} // byte
{{print nil}} // nil
```

**pipeline**

```
{{. | FuncA | FuncB | FuncC}}
```

pipeline:

- false0
- nilinterface
- 0array, slice, map, string

pipeline

```
gf<no value>
```

**if ... else ... end**

```
{{if pipeline}}...{{end}}
```

ifpipelinefalse

```
{{if .condition}}
...
{{else}}
    {{if .condition2}}
    ...
    {{end}}
{{end}}
```

else if

```
{{if .condition}}
...
{{else if .condition2}}
...
{{else}}
...
{{end}}
```

**range ... end**

```
{{range pipeline}} {{.}} {{end}}
```

pipelineslice, map, channel

**range.** `{{$.Session.Name}}`

Orange.

map:

```
{{range $key, $value := .MapContent}}
    {{$key}}:{{$value}}
{{end}}
```

slice:

## Content Menu

- [if ... else ... end](#)
- [range ... end](#)
- [with ... end](#)
- [define](#)
- [template](#)
- [include](#)
-

```

{{range $index, $elem := .SliceContent}}
  {{range $key, $value := $elem}}
    {{$key}}:{{$value}}
  {{end}}
{{end}}

```

## with ... end

```

{{with pipeline}}...{{end}}

withpipeline

{{with .Field.NestField.SubField}}
  {{.Var}}
{{end}}

```

## define

```

define()(template)

{{define "loop"}}
  <li>{{$.Name}}</li>
{{end}}

loop template

<ul>
  {{range .Items}}
    {{template "loop" .}}
  {{end}}
</ul>

```

*definetemplate/ParseFiles*

## template

```

{{template "" pipeline}}

pipeline

templatetemplate

```

*templatedefine/ParseFiles*

## include

### gf

```

{{include "()" pipeline}}

include()

{{include "()" .}}

templateincludetempalte

```

```

{/{*
comment content
support new line
*/}}

```