

第 5 章

过 滤 器



视频讲解

从 Vue 3.0 开始,过滤器已删除,不再支持。建议用计算属性或方法代替过滤器,而不是使用过滤器。因此,本章将简单介绍 Vue 原有的内置过滤器,而不对过滤器做过多深入的介绍。

在第 3 章中提到过,Vue 2.x 允许在表达式后面添加可选的过滤器,以管道符“|”表示。事实上,过滤器的本质是一个函数,Vue 2.x 支持在任何出现表达式的地方添加过滤器。过滤器接收管道符前面的值作为初始值,返回值为经过处理后的输出值。同时,过滤器也能接收额外的参数,始终以表达式的值作为第一个参数,其他参数跟在过滤器名称后面,参数之间以空格分隔。多个过滤器也可以进行串联,上一个过滤器的输出结果可以作为下一个过滤器的输入。

5.1 内置过滤器

5.1.1 字母过滤器

Vue 2.x 内置了三种字母过滤器 `capitalize`、`uppercase`、`lowercase`,用于处理英文字符。

- (1) `capitalize` 过滤器用于将表达式中的首字母转换为大写。
- (2) `uppercase` 过滤器用于将表达式中的所有字母转换为大写。
- (3) `lowercase` 过滤器用于将表达式中的所有字母转换为小写。

三种过滤器的用法如例 5-1 所示。

【例 5-1】 字母过滤器的使用

```
{{ 'abcd ' | capitalize }}  
{{ 'abcd ' | uppercase }}  
{{ 'ABCD ' | lowercase }}
```

上面三行代码将分别输出 'Abcd'、'ABCD' 和 'abcd'。

5.1.2 json 过滤器

Vue 2.x 的 `json` 过滤器可以将表达式的值转换为 JSON 字符串,输出的结果等于表达式经过 `JSON.stringify()` 处理后的结果。`json` 过滤器可接收一个类型为 `Number` 的参数,用于决定转换后的 JSON 字符串的缩进距离,默认的缩进距离为 2。使用方式如例 5-2 所示。

【例 5-2】 json 过滤器的使用

```
{{ student | json 4 }}
```

5.1.3 限制过滤器

Vue 2.x 内置了三个限制过滤器：limitBy、filterBy、orderBy，用于处理并返回过滤后的数组。这三个过滤器所处理的表达式的值必须是数组。

limitBy 过滤器的作用是限制数组为前 N 个元素，通过传入的第一个参数指定 N。第二个参数可选，用于指定开始的偏移量，默认为 0 不偏移。

filterBy 过滤器的第一个参数可以是字符串或者函数。如果第一个参数是字符串，那么将在每个数组元素中搜索它，返回包含该字符串的元素组成的数组。如果 filterBy 的第一个参数是函数，则过滤器将根据函数的返回结果进行过滤。此时 filterBy 过滤器将调用 JavaScript 组中内置的函数 filter 对数组进行处理，待过滤数组中的每个元素都作为参数输入并传入 filterBy 中的函数，只有函数返回结果为 true 的数组元素才符合条件并将其存入一个新的数组，最终 filterBy 的返回结果即为这个新的数组。

在对象中，过滤器将在所有属性中搜索。可以通过指定搜索字段的方式缩小搜索范围。在多个字段中搜索时，字段与字段之间通过空格分隔，或者将搜索字段存放在一个数组中，如例 5-3 所示。

【例 5-3】 limitBy 与 filterBy 过滤器的使用

```
<div v-for="item in items | limitBy 10 5">
</div>
<div v-for="student in students | filterBy 'sky' in 'name' 'nickname'"></div>
```

此外，还可以使用动态参数作为搜索目标或搜索字段，结合 v-model 实现输入提示效果。通过将输入值作为检索字段实现输入提示，如例 5-4 所示。

【例 5-4】 动态提示

```
<template>
  <div class="app">
    <input v-model="input" />
    <li v-for="user in users | filterBy input in 'name'">
      { { user.name } }
    </li>
  </div>
</template>

<script>
import { reactive, toRefs } from "vue";
export default {
  setup() {
    const data = reactive({
      input: "",
      users: [{ name: "SKY" }, { name: "GRASS" }, { name: "NANA" }],
    });
  }
};
```

```
});

return {
  ...toRefs(data),
};
},
};
};
</script>
```

orderBy 过滤器的作用是返回排序后的数组。第一个参数可以是字符串、数组或者函数；第二个参数 order 可选，order < 0 为降序，其他为升序，默认为升序排列。

若输入参数为字符串，则可同时传入多个字符串作为排序键，字符串之间以空格分隔，或者存入一个数组中。此时将按照传入键名的顺序或数组的顺序进行排序，如例 5-5 所示。

【例 5-5】 orderBy 过滤器的使用

```
<li v-for="user in users | orderBy 'lastName ' firstName ' age '"></li>
<li v-for="user in users | orderBy sortList"></li>
```

5.1.4 currency 过滤器

currency 过滤器的作用是将数字转换为货币形式输出。第一个参数为 String 类型的货币符号，默认为美元符号 \$；第二个参数为 Number 类型，表示保留的小数位，默认为 2，如例 5-6 所示。

【例 5-6】 currency 过滤器的使用

```
<div>{{ amount | currency }}</div>
<div>{{ amount | currency '¥' }}</div>
<div>{{ amount | currency '$ '3 }}</div>
```

当 amount 为 12345 时，上面的代码将分别显示 \$12,345.00、¥12,345.00 和 \$12,345.000。

5.1.5 debounce 过滤器

debounce 过滤器的作用是延迟执行时间。接收的表达式值必须为函数，一般与 v-on 等指令结合使用。debounce 接收一个可选的参数作为延迟时间，单位为 ms，默认为 300ms。经过 debounce 包装的处理器在调用之后将至少延迟设定的时间再执行。如果在延迟结束前再次调用，则延迟时长将重置为设定的时间。使用方法如例 5-7 所示，实现了对按键事件处理的延迟。

【例 5-7】 debounce 过滤器的使用

```
<input @keyup="onKeyUp | debounce 500">
```

5.2 本章小结

本章介绍了 Vue 原有的内置过滤器：字母过滤器、json 过滤器、限制过滤器、currency 过滤器和 debounce 过滤器。它们在处理表达式时很便利，但是这种方式打破了大括号内表达式“只是 JavaScript”的原则。因此从 Vue 3.0 开始，建议用计算属性或方法替换过滤器。

5.3 练习题

一、填空题

1. `<div>{{ amount | currency '$ '3}}</div>` 代码显示的结果是_____。
2. debounce 过滤器的作用是_____。
3. Vue 2.x 内置的三个限制过滤器是_____、_____、_____。
4. currency 过滤器的作用是_____。
5. Vue 2.x 内置的三个过滤器 capitalize、uppercase、lowercase 用于处理_____。

二、单选题

1. (`{{ 'bcd' | capitalize }}`)、(`{{ 'abcd' | uppercase }}`)、(`{{ 'ABCD' | lowercase }}`) 三个字母过滤器的结果是()。
A. 'Abcd'、'ABCD'、'abcd' B. 'Abcd'、'ABCD'、'aBCD'
C. 'ABCD'、'Abcd'、'abcd' D. 以上都不对
2. (`{{ student | json }}`) 的结果是()。
A. "student" B. 'student' C. {student} D. '{student}'
3. 经过 debounce 过滤器包装的处理器在调用之后将至少延迟设定的时间再执行。如果在延迟结束前再次调用，则()。
A. 立即执行
B. 将在上次延长结束后再延长第二次设定的时间
C. 在上次延长结束后立即执行
D. 延迟时长将重置为设定的时间
4. Vue 2.x 内置的三个限制过滤器处理的表达式的值是()。
A. 数组 B. 函数 C. 数组和函数 D. 以上都不是
5. 下列()过滤器可以返回排序后的数组。
A. limitBy B. filterBy C. orderBy D. debounce

三、判断题

1. Vue 2.x 的内置过滤器不可以进行串联，上一个过滤器的输出结果不可以作为下一个过滤器的输入。 ()
2. orderBy 过滤器的作用是返回排序后的数组，因此其第一个参数只能传入数组。 ()
3. debounce 过滤器接收的表达式必须为函数。 ()
4. limitBy 过滤器的作用是限制数组为前 N 个元素，第一个参数用于指定开始的偏移量，第二个参数指定 N。 ()

5. 当 filterBy 的第一个参数是函数时,最终 filterBy 的返回结果是函数返回结果为 true 的数组元素合并组成的新数组。 ()

四、问答题

1. 说明三种字母过滤器的作用。
2. 如何使用限制过滤器限制数组为第 3 个元素开始的 10 个元素?
3. currency 过滤器的第一个参数采取默认形式,想要修改第二个参数(也就是小数位)时,该如何表示?

五、动手做

1. 使用计算属性,实现三种字母过滤器的效果。
2. 使用计算属性,在第 3 章“动手做”制作的贴吧的所有帖子标题与内容中搜索包含 sky 的部分,并返回包含 sky 的元素组成的数组。