

フィルター

サンプルコード : [OriginalFilter.vue](#)

フィルターとは

テキストをフォーマットするツール。表示したい要素を全て大文字にしたり小文字にしたりなど行う時に使用する

computed とは何が違うのか

- 同じようなことを`computed`を使って実装できるが、データごとに実装が必要となるため、冗長になる

参考 : [フィルター](#)

```
<h1>{{ upperCaseTitle }}</h1>
<p>{{ upperCaseMsg }}</p>
```

```
title: string = "hello vue.js";
msg: string = "the progressive javascript framework";
//computed
get upperCaseTitle(): string {
  return this.title.toUpperCase();
}
get upperCaseMsg(): string {
  return this.msg.toUpperCase();
}
```

同じようなロジックの処理を 2 回書いている

ローカルフィルターを設定

```
<h3>{{ title | upperCase }}</h3>
<p>{{ msg | upperCase }}</p>
```

```
@Component({
  filters: {
    upperCase(val: string): string | null {
      if (!val) {
        return null;
      }
      return val.toUpperCase();
    },
  },
})
```

グローバルフィルターで設定

他ファイル(コンポーネント)でも使用する場合はグローバルのところで設定する 設定場所は`main.ts`

```
import Vue from "vue";
import App from "./App.vue";

Vue.config.productionTip = false;
Vue.filter("upperCase", function (val: string): string {
  return val.toUpperCase();
});
new Vue({
  render: (h) => h(App),
}).$mount("#app");
```