

[CGA-JAVA-WFDA] Web Front-end Development Angular 2.1

[Dashboard](#) / [My courses](#) / [CGA-JAVA-WFDA-2.1](#) / [4. Angular Component & Template](#) / [\[Bài đọc\] Pipe trong Angular](#)


[Bài đọc] Pipe trong Angular

Giới thiệu về pipe

Pipe được sử dụng để chuyển đổi dữ liệu mà bạn hiển thị lên template cho người dùng có thể hiểu được.

Ví dụ là định dạng kiểu hiển thị datetime, viết hoa chữ cái, hiển thị tên thành phố, định dạng lại số hay đơn vị tiền, ...

Danh sách các pipe mà Angular cung cấp mặc định:

 AsyncPipe

 CurrencyPipe

 DatePipe

 DecimalPipe

 I18nPluralPipe

 I18nSelectPipe

 JsonPipe

 KeyValuePipe

 LowerCasePipe

 PercentPipe

 SlicePipe

 TitleCasePipe

 UpperCasePipe

- [AsyncPipe](#)
- [DecimalPipe](#)
- [JsonPipe](#)
- [PercentPipe](#)
- [UpperCasePipe](#)
- [CurrencyPipe](#)
- [I18nPluralPipe](#)
- [KeyValuePipe](#)
- [SlicePipe](#)
- [DatePipe](#)
- [I18nSelectPipe](#)
- [LowerCasePipe](#)
- [TitleCasePipe](#)

Sử dụng pipe

Ví dụ dưới đây sẽ sử dụng pipe để biến đổi ngày sinh nhật từ dữ liệu thô sang định dạng dễ đọc hơn

```
import { Component } from '@angular/core';
```

```
@Component({
  selector: 'app-hero-birthday',
  template: `<p>The hero's birthday is {{ birthday | date }}</p>`
})
export class HeroBirthdayComponent {
  birthday = new Date(1988, 3, 15);
}
```

```
//Fri Apr 15 1988 00:00:00 GMT+0700 (Indochina Time) -> April 15, 1988
```

Bên trong biểu thức nội suy `{{}}` chúng ta đã biến đổi ngày sinh nhật bằng biểu thức pipe `|` từ định dạng của js `Fri Apr 15 1988 00:00:00 GMT+0700 (Indochina Time)` sang `April 15, 1988`, dễ đọc cho người dùng

Ở ví dụ trên chúng ta mới chỉ sử dụng 1 pipe, ví dụ dưới đây chúng ta sẽ sử dụng 2 pipe `DatePipe` và `UpperCasePipe` liên tiếp nhau trong đó `DatePipe` sẽ là dữ liệu đầu vào của `UpperCasePipe`

```
The chained hero's birthday is
{{ birthday | date | uppercase}}
```

```
// APR 15, 1988
```

Tham số trong pipe

Để có thể bổ sung tham số trong pipe, chúng ta sử dụng dấu hai chấm : sau tên pipe đó (ví dụ currency:'EUR')

Và tương tự như vậy, ở ví dụ trên ta sẽ thêm tham số là định dạng của DatePipe

```
<p>The hero's birthday is {{ birthday | date:"MM/dd/yy" }} </p>
```

Tham số có thể được custom lại thông qua các thao tác thay đổi sự kiện (nhưng vẫn phải đúng với cú pháp của pipe đó)

Sử dụng tiếp với ví dụ trên, chúng ta sẽ không truyền tham số định dạng trực tiếp đằng sau datepipe mà sẽ ràng buộc thông qua một biến là format và chúng ta sẽ thay đổi format đó thông qua một button bên dưới

```
template: `
<p>The hero's birthday is {{ birthday | date:format }}</p>
<button (click)="toggleFormat()">Toggle Format</button>
```

Sự kiện click sẽ thay đổi format của datepipe giữa kiểu shortDate và fullDate

```
export class HeroBirthday2Component {
  birthday = new Date(1988, 3, 15); // April 15, 1988
  toggle = true; // start with true == shortDate

  get format() { return this.toggle ? 'shortDate' : 'fullDate'; }
  toggleFormat() { this.toggle = !this.toggle; }
}
```

The hero's birthday is 4/15/1988

Toggle Format

Last modified: Wednesday, 24 February 2021, 10:33 PM

Previous Activity

Jump to...

Next Activity

CHƯƠNG TRÌNH

- Career
- Premium
- Accelerator

TÀI NGUYÊN

- Blog
- Tạp chí Lập trình
- AgileBreakfast

Follow Us

