OPENGIS

Giới thiệu
Angular Data Binding

Đỗ Thành Long

dtlong@opengis.vn

OPENGIS

1. Mục tiêu

Angular là nền tảng chính của Ionic, giúp tổ chức ứng dụng hiệu quả

- Hiểu Data Binding trong Angular là gì.
- Có mấy kiểu
- Dùng như thế nào
- Có lợi ích gì

OPENGIS

2. Data Binding là gì?

Angular là nền tảng chính của Ionic, giúp tổ chức ứng dụng hiệu quả

- Databinding là kỹ thuật nơi dữ liệu được đồng bộ giữa component và tầng view (template file html). Ví dụ khi người dùng cập nhật data ở tầng view thì Angular cũng cập nhật giá trị đó ở component.

- Data binding trong Angular có thể chia ra làm 2 nhóm. Đó là one way binding (binding 1 chiều) và two way binding (binding 2 chiều).

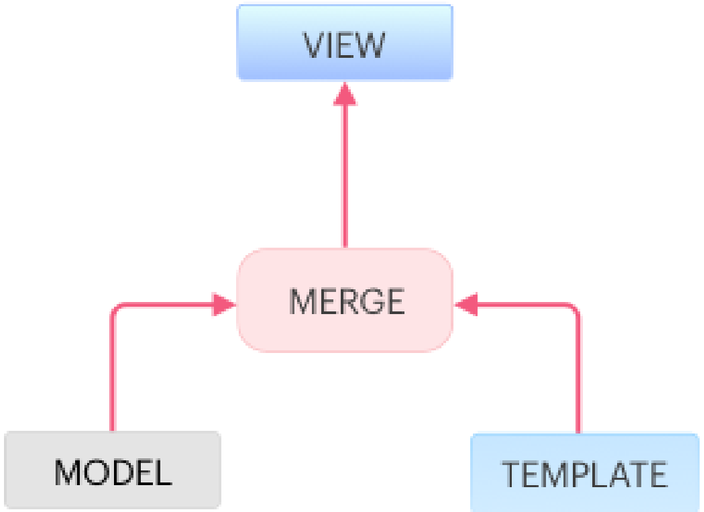
https://levunguyen.com/laptrinhweb/2021/06/05/su-dung-data-binding-trong-angular/

OPENGIS

2.1. One-Way Binding

- Dữ liệu chỉ được truyền một chiều từ Component đến Template hoặc từ Template đến Component.
- Các cú pháp binding như {{ data }}, [property], (event) thường được sử dụng cho One-Way Binding.
- Chúng ta sử dụng để hiển thị giá trị từ component sang view.

2.1. One-Way Binding



One-Way Data Binding

https://dzone.com/articles/explaining-data-binding-and-server-rendering-in-an-2

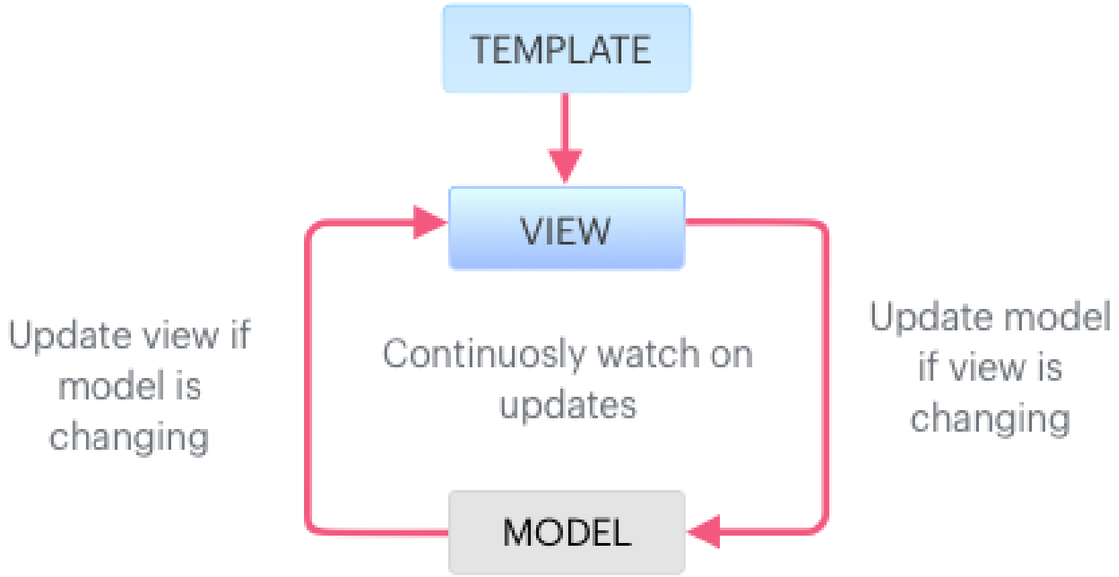
OPENGIS

2.2. Two-Way Binding

- Dữ liệu có thể được truyền cả hai chiều giữa Component và Template, cho phép cập nhật dữ liệu ở cả hai phía mà không cần phải làm nhiều công việc.
- [(ngModel)] là cú pháp thông dụng cho Two-Way Binding.
- Event Binding: Cho phép gắn các sự kiện (events) từ Template và gửi chúng đến Component để xử lý khi người dùng tương tác với ứng dụng.

OPENGIS

2.2. Two-Way Binding

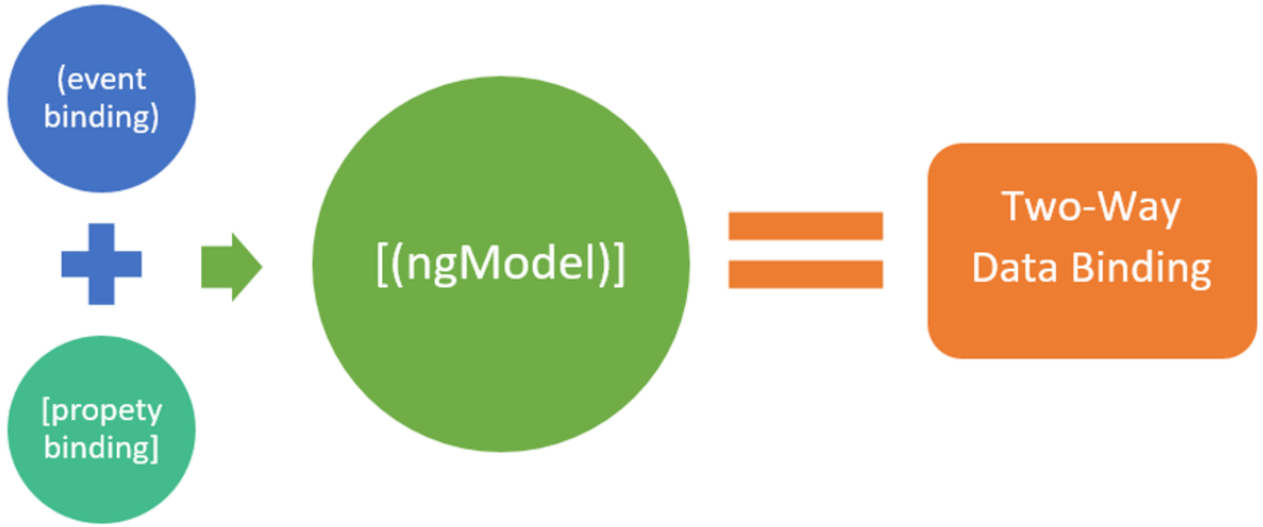


Two-Way Data Binding

https://dzone.com/articles/explaining-data-binding-and-server-rendering-in-an-2

OPENGIS

2.2. Two-Way Binding



OPENGIS

3. Một số kiểu binding

- Interpolation,
- Class and style binding,
- Event binding,
- Property binding

OPENGIS

3.1. Data Binding - Interpolation

- Interpolation là cách hiển thị dữ liệu từ Component lên Template bằng cú pháp {{ expression }}.
- Dùng để hiển thị giá trị biến hoặc kết quả của biểu thức.

OPENGIS

3.1. Data Binding - Interpolation

- **TypeScript (Component):**

```typescript
export class ExampleComponent {
  title: string = 'Chào mừng đến với Angular';
}
```

- **HTML (Template):**

```html
<h1>{{ title }}</h1>
```

- **Kết quả:** Hiển thị "Chào mừng đến với Angular" trên giao diện.

OPENGIS

3.2. Data Binding - Property Binding

- Property Binding dùng để gán giá trị từ Component vào thuộc tính của thẻ HTML bằng cú pháp [property]="value".

- Thay đổi thuộc tính động dựa trên dữ liệu.

OPENGIS

3.2. Data Binding - Property Binding

- **TypeScript (Component):**

```typescript
export class ExampleComponent {
  imageUrl: string = 'https://example.com/image.jpg';
}
```

- **HTML (Template):**

```html
<img [src]="imageUrl" alt="Hình ảnh ví dụ">
```

- **Kết quả:** Hiển thị hình ảnh từ URL được gán.

OPENGIS

3.3. Data Binding - Event Binding

- Event Binding cho phép gọi hàm trong Component khi có sự kiện từ giao diện, dùng cú pháp (event)="handler()".
- Thường dùng để xử lý tương tác người dùng.

OPENGIS

3.3. Data Binding - Event Binding

- **TypeScript (Component):**

```typescript
export class ExampleComponent {
  logMessage() {
    console.log('Nút đã được nhấn!');
  }
}
```

- **HTML (Template):**

```html
<button (click)="logMessage()">Nhấn tôi</button>
```

- **Kết quả:** In ra console "Nút đã được nhấn!" khi nhấn nút.

OPENGIS

3.4. Sử dụng class binding

- Bản chất là **Property binding**
- Sử dụng [class] binding để thêm và xóa name class từ một thành phần.

OPENGIS

3.4. Sử dụng class binding

BINDING TYPE	SYNTAX	INPUT TYPE	EXAMPLE INPUT VALUES
Single class binding	[class.sale]="onSale"	boolean , undefined , null	true, false
Multi-class binding	[class]="classExpression"	string	"my-class-1 my-class-2 my-class-3"

OPENGIS

3.5. Sử dụng style binding

- Bản chất là **Property binding**
- Sử dụng [style] binding để style đối tượng giao diện

OPENGIS

3.5. Sử dụng style binding

Để tạo một single style binding, chúng ta sử dụng prefix **style** theo sau là một dấu chấm và tên của style.

```
[style.width]="width"
```

- Ví dụ viết một style với kiểu dash-case:

```
<nav [style.background-color]="expression">test01</nav>
```

- Ví dụ viết một style với kiểu camelCase:

```
<nav [style.backgroundColor]="expression">test02</nav>
```

https://viblo.asia/p/su-dung-data-binding-trong-angular-13-RQqKL3Oz57z

OPENGIS

3.5. Sử dụng style binding

BINDING TYPE	SYNTAX	INPUT TYPE	EXAMPLE INPUT VALUES
Single style binding	[style.width]="width"	string , undefined , null	"100px"
Single style binding with units	[style.width.px]="width"	number , undefined , null	100
Multi-style binding	[style]="styleExpression"	string	"width: 100px; height: 100px"

https://viblo.asia/p/su-dung-data-binding-trong-angular-13-RQqKL3Oz57z

OPENGIS

3.6. Data Binding - Two-way Binding

- **Two-way Binding** đồng bộ dữ liệu hai chiều giữa Component và Template, dùng [(ngModel)].
- Yêu cầu import FormsModule trong app.module.ts.

OPENGIS

3.6. Data Binding - Two-way Binding

- **TypeScript (Component):**

```typescript
export class ExampleComponent {
  email: string = '';
}
```

- **HTML (Template):**

```html
<input [(ngModel)]="email" placeholder="Nhập email">
<p>Email: {{ email }}</p>
```

- **Kết quả:** Nhập email vào input, giá trị hiển thị ngay bên dưới.

OPENGIS

3.7. Điều kiện - Sử dụng ngIf

- **ngIf** hiển thị hoặc ẩn một phần tử dựa trên điều kiện, dùng cú pháp *ngIf="condition".
- Hỗ trợ điều kiện ngược với !condition.

OPENGIS

3.7. Điều kiện - Sử dụng ngIf

- **TypeScript (Component):**

```typescript
export class ExampleComponent {
  isVisible: boolean = true;
}
```

- **HTML (Template):**

```html
<div *ngIf="isVisible">
  <p>Phần này hiển thị khi isVisible là true.</p>
</div>
```

- **Kết quả:** Hiển thị đoạn văn khi `isVisible` là `true`.

OPENGIS

3.8. Vòng lặp - Sử dụng ngFor

- **ngFor** lặp qua một mảng và hiển thị các phần tử, dùng cú pháp *ngFor="let item of items".
- Thường dùng để tạo danh sách động.

OPENGIS

3.8. Vòng lặp - Sử dụng ngFor

- **TypeScript (Component):**

```typescript
export class ExampleComponent {
  fruits: string[] = ['Táo', 'Chuối', 'Cam'];
}
```

- **HTML (Template):**

```html
<ul>
  <li *ngFor="let fruit of fruits">{{ fruit }}</li>
</ul>
```

- **Kết quả:** Hiển thị danh sách: Táo, Chuối, Cam.

OPENGIS

3.9. Hàm trong Angular

- **Hàm** được định nghĩa trong Component và gọi từ Template để xử lý logic.
-
- Có thể kết hợp với Event Binding.

OPENGIS

3.9. Hàm trong Angular

- **TypeScript (Component):**

```typescript
export class ExampleComponent {
  counter: number = 0;
  increaseCounter() {
    this.counter++;
  }
}
```

- **HTML (Template):**

```html
<button (click)="increaseCounter()">Tăng</button>
<p>Giá trị: {{ counter }}</p>
```

OPENGIS

3.10. Kết hợp ngIf và ngFor

- Kết hợp **ngIf** và **ngFor** để kiểm tra điều kiện trước khi lặp mảng.
- Giúp tránh lỗi khi mảng rỗng.

OPENGIS

3.10. Kết hợp ngIf và ngFor

- **TypeScript (Component):**

```typescript
export class ExampleComponent {
  products: string[] = ['Sách', 'Bút'];
}
```

- **HTML (Template):**

```html
<div *ngIf="products.length > 0">
  <ul>
    <li *ngFor="let product of products">{{ product }}</li>
  </ul>
</div>
<p *ngIf="products.length === 0">Không có sản phẩm.</p>
```

OPENGIS

THANK YOU