

1604C054 – Hybrid Mobile Programming

Angular Binding and Angular Directive Part 1

WEEK 03

Informatics Engineering
Universitas Surabaya



Outline

1

Definition

2

Interpolation
Binding

3

Event Binding

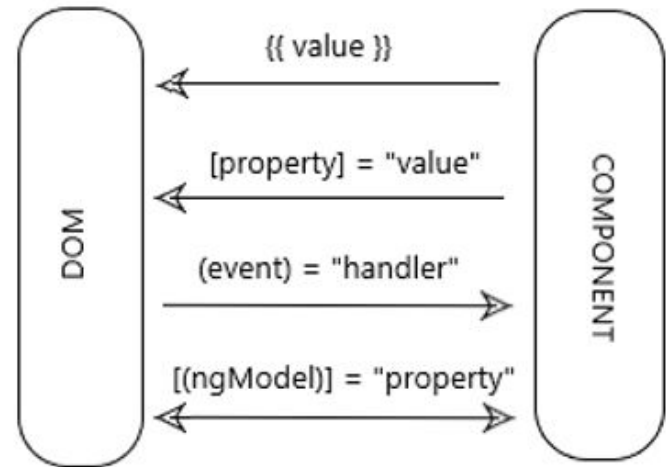
4

Property Binding

What is Binding?

Binding in the context of web development and frameworks like Angular refers to the process of establishing a connection or relationship between the data in your application (often stored in a component or model) and the user interface (UI) elements in your application's templates or views.

The primary goal of binding is to ensure that changes in one part of your application are reflected in another automatically, creating dynamic and responsive user interfaces.



What is Binding? (2)

Binding allows you to:

1. **Display Data:** You can display data from your application's data model (often JavaScript objects or variables) to your HTML templates. For example, you can show the user's name or a product's price in a web page.
2. **Capture User Input:** You can capture user input, such as form submissions, button clicks, or text input, and use that input to update your application's data model or trigger actions in response to user interactions.
3. **Keep UI in Sync:** When your data changes, either due to user input or other factors, the bound UI elements automatically update to reflect those changes. Conversely, when a user interacts with the UI, the data model is updated accordingly.

Interpolation Binding

The simplest and most commonly used forms of data binding in Angular. It allows you to display the value of a component property directly in your template.

Interpolation isn't limited to simple properties. You can also use expressions, such as mathematical **calculations** or **method** calls, inside the curly braces.

Syntax: The basic syntax for interpolation is

`{{ expression }}`

where `expression` is a reference to a property or an expression in your component.

Interpolation Binding (2)

Example 1 : We will modify List page. We add caption that include today date.

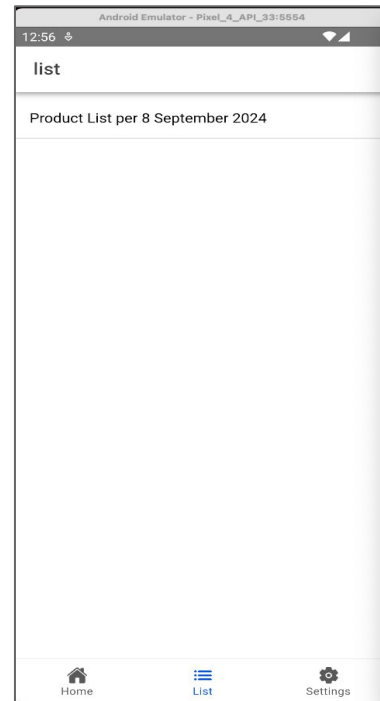
in **list.page.html** add:

```
<ion-list lines="full">
  <ion-item>
    <ion-text>
      Product List per {{today}}
    </ion-text>
  </ion-item>
</ion-list>
```

in **list.page.ts** add:

```
export class ListPage implements OnInit {
  today:string='8 September 2024'

  constructor() { }
```



Interpolation Binding (3)

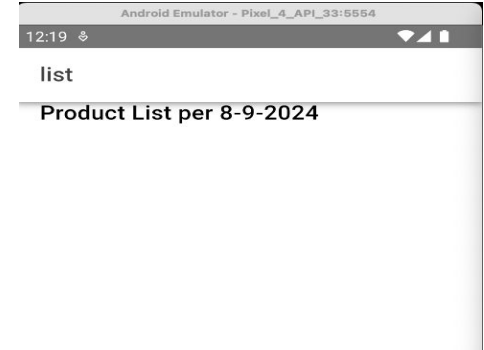
Example 2 : We will try to call method/function

in **list.page.html** add:

```
Product List per {{today_ind()}}
```

in **list.page.ts** add:

```
today_ind():string {  
    const currentDate = new Date();  
    // Get the current day (1-31)  
    const d = currentDate.getDate();  
    // Get the current month (0-11, where 0 is January and 11 is December)  
    const m = currentDate.getMonth() + 1; // Adding 1 to convert to 1-12 range  
    // Get the current year (four-digit year)  
    const y = currentDate.getFullYear();  
    return d+'-'+m+'-'+y;  
}
```



Exercise 1 (max 15 minutes)

- Can you change the month number with month name ?
- Can you do it without if or switch ? (hint:use array)
- Can you add day name before the date? (hint: `getDay()`)

Expected Result



Event Binding

Event binding in Angular allows you to listen for and respond to events (such as mouse clicks, key presses, or custom events) that occur in your application's user interface. You can use event binding to trigger functions or methods in your component class when specific events happen.

Event binding in Angular uses parentheses () to bind to events in your HTML templates. The syntax is as follows:

```
<element (event)="expression"></element>
```

Example:

```
<button (click)="onButtonClick()">Click Me</button>
```

Event Binding (2)

Example: We will add button to go to previous date listing. When user click the button, it will change the current date with yesterday date. Because we need to access **currentDate** in other function, we have to modify the **currentDate** variable in **today_ind()** function as class property.

```
export class ListPage implements OnInit {  
  currentDate = new Date();
```

Class property need to access with "this" keyword.

```
today_ind():string {  
  const arrayOfMonths = ["Januari", "Februari", "Maret", "April", "Mei", "Juni",  
    "Juli", "Agustus", "September", "Oktober", "November", "Desember"]  
  
  const d = this.currentDate.getDate();  
  const m = this.currentDate.getMonth();  
  const y = this.currentDate.getFullYear();  
  return d+' '+arrayOfMonths[m]+' '+y;  
}
```

Event Binding (3)

Add a button with event binding that call **goYesterday()** function in page.html

```
<ion-button size="small" (click)="goYesterday()">
  <ion-icon name="chevron-back-outline"></ion-icon>
</ion-button>
```

Add a function in page.ts :

```
goYesterday() {
  this.currentDate.setDate(this.currentDate.getDate() - 1);
}
```

list

Product List per 8 September 2024 <

list

Product List per 7 September 2024 <

Event Binding (3)

other event which can use in binding:

(mousedown) and **(mouseup)**: These events capture mouse button presses and releases.

(mousemove): Captures mouse movement within an element.

(keydown) and **(keyup)**: These events capture keyboard key presses and releases.

(focus) and **(blur)**: These events are triggered when an element gains or loses focus, typically used with form elements like input fields.

(submit): Captures form submission events.

(change): Triggered when the value of a form element (e.g., input, select) changes.

(input): Emits when the user interacts with an input element, such as typing or pasting text.

(mouseenter) and **(mouseleave)**: These events trigger when the mouse enters or leaves an element.

(dragstart), **(drag)**, and **(dragend)**: These events are related to drag-and-drop operations.

(scroll): Captures scrolling events on scrollable elements.




(resize): Triggered when the browser window is resized.

(load) and **(unload)**: These events are related to the loading and unloading of resources, such as images and pages.

Exercise 2 (max 15 minutes)

- Can you Add Tomorrow Button?
 - Tomorrow button will add 1 day after current day.
- Can you add Reset Button?
 - Reset button will reset into current day

Expected Result

list	
Product List per 8 September 2024	  

Property Binding

Property binding is a fundamental concept in Angular that allows you to associate (or bind) a property of an HTML element in your template with a property of a component. This binding establishes a connection between the component's property and the DOM element's property, enabling dynamic updates from the component to the DOM.

syntax

```
[binding_target] ="binding_expression"
```

example

```
<img [src]="imageUrl" alt="Image" />
```

- [src] is the binding target, representing the src property of the element.
- "imageUrl" is the binding expression, referring to a property named imageUrl in your component.

Property Binding (2)

Example : We only have list of product data for 5 days before. Therefore, we will disabled Yesterday button after it clicked 5 times.

Add in yesterday button

```
<ion-button (click)="goYesterday()" [disabled]="is5daysago"
```

Add in page.ts

```
is5daysago=false  
numberclicked=0  
-----  
goYesterday()  
{  
  this.currentDate.setDate(this.currentDate.getDate() - 1);  
  this.numberclicked++;  
  if(this.numberclicked==5) this.is5daysago=true  
}
```

Exercise 3 (max 10 minutes)

- Can you add disabled property on tomorrow button when h+5?
- Can you add disabled is False when reset button is clicked?
 - disabled = False
 - reset numberclicked = 0

Expected Result



Binding with Object

Example : Create object with name of variable is **product**. The key of products are productName, productDate and productPrice

Create interface Product in list.page.ts (on the top, after import)

```
interface Product {  
  productName: string,  
  productDate: Date,  
  productPrice: number,  
}
```

Create variable product with type Product in list.page.ts :

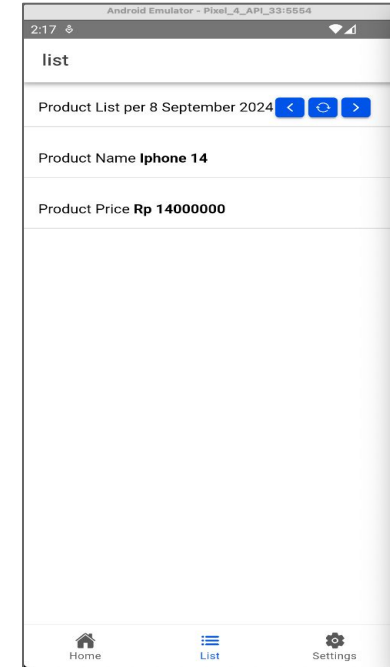
```
product:Product = {  
  productName: 'Iphone 14',  
  productDate: new Date(),  
  productPrice: 14000000,  
}
```

Binding with Object (2)

Example : Create object with name of variable is **product**. The key of products are productName, productDate and productPrice

Add interpolation binding of **product** in list.page.html

```
<ion-list lines="full">
  <ion-item>
    <ion-text>
      Product Name <b>{{product.productName}}</b>
    </ion-text>
  </ion-item>
</ion-list>
<ion-list lines="full">
  <ion-item>
    <ion-text>
      Product Price <b>Rp {{product.productPrice}}</b>
    </ion-text>
  </ion-item>
</ion-list>
```



Exercise 4 (max 30 minutes)

- Can you add quantity with increase and decrease button?
- Please add total amount with calculation **totalAmount = productPrice * quantity**
- Minimum quantity is 0 and maximum quantity is 10

Expected Result

list
Product List per 8 September 2024 < ↺ >
Product Name Iphone 14
Product Price Rp 14000000
Quantity <input type="button" value="-"/> 4 <input type="button" value="+"/>
Total Amount Rp 56000000

Exercise Deadline

- Please save your exercise today
- Exercise will be uploaded in ULS in the end of the class
- Upload your src folder
- Format: class_nrp_week3.zip (ex: KPA_111222_week3.zip)

PROJECT REPORT

Project Report

- Deadline project report in week 5
- Upload your src folder in ULS
- Create the report in PDF file including:
 - Name of team, member and NRP
 - Screenshot every page you have done
 - Give the explanation below screenshot

Thanks.