



ANGULAR COMPLETE BEGINNER TO ADVANCE NOTES

Topic: Components, Variables, Interpolation & Dynamic UI



PART 1: INTRODUCTION TO ANGULAR COMPONENT

1 What is a Component?

A Component is the **basic building block of Angular**.

Every Angular application is made of multiple components.

A component contains:

1. HTML → Structure (View)
2. TS → Logic
3. CSS → Styling

Example:

header.component.ts
header.component.html
header.component.css

2 How Component Works Internally

From Paint Diagram Concept:

Component Class (TS)



Properties / Variables



Interpolation {{ }}



HTML Template



Browser UI

Angular automatically connects TypeScript and HTML.

We don't manually manipulate DOM like JavaScript.



PART 2: TYPESCRIPT VARIABLES INSIDE COMPONENT

③ Creating Variables (Properties) in Component

Inside:

```
export class HeaderComponent {  
  
    title = "My First Variable";  
  
    anotherTitle: string = "Ecom AI Mart";  
  
}
```

These are called:

Class Properties

Because they belong to the class.

4 Two Ways to Create Variables

1. Without Data Type (Type Inference)

```
title = "My First Variable";
```

TypeScript automatically detects type → string

2. With Data Type (Strongly Typed)

```
anotherTitle: string = "Ecom AI Mart";
```

Syntax:

```
variableName : dataType = value
```

Examples:

```
count: number = 10;  
isActive: boolean = true;  
price: number = 999;
```



PART 3: STRONGLY TYPED VS LOOSELY TYPED

◆ JavaScript (Loosely Typed)

```
let title = "Hello";  
title = 100;  
title = true;
```

Allowed ✅ (no restriction)

◆ TypeScript (Strongly Typed)

```
let title: string = "Hello";
```

```
title = 100; // ✅ Error  
title = true; // ✅ Error
```

Only string allowed.

This gives:

- ✓ Safety
 - ✓ Error detection early
 - ✓ Better code quality
-



PART 4: INTERPOLATION (MOST IMPORTANT)

5 What is Interpolation?

Interpolation means:

Displaying TypeScript variable inside HTML using {{ }}

Syntax:

```
{{ variableName }}
```

6 Example

TypeScript:

```
title = "My First Variable";
```

HTML:

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

Output:

My First Variable

7 Flow of Interpolation (From Paint File Concept)

TS Property → {{ }} → HTML → Browser

Angular automatically:

- Reads variable
- Injects value
- Updates UI

No DOM manipulation required.



PART 5: MAKING UI DYNAMIC

8 Hardcoded UI (Wrong Approach)

```
<a>Products</a>
```

If product name changes → You must change HTML.

9 Dynamic UI (Correct Angular Approach)

TypeScript:

```
productName: string = "Products";
```

HTML:

```
<a>{{ productName }}</a>
```

If you change:

```
productName = "All Products";
```

UI automatically updates.

This is called:

Dynamic Data Binding



PART 6: ARRAYS IN COMPONENT

10 Creating Array

```
links: string[] = ["Home", "Profile", "Products", "Contact"];
```

11 Access Array in HTML

```
<a>{{ links[0] }}</a>
<a>{{ links[1] }}</a>
<a>{{ links[2] }}</a>
<a>{{ links[3] }}</a>
```

Output:

Home
Profile

Products
Contact

- ◆ **If You Change TS**

```
links: string[] = ["Home", "My Profile", "All Products", "Contact"];
```

UI updates automatically.



PART 7: METHODS IN COMPONENT

12 Creating Function

```
changeTitle() {  
    this.anotherTitle = "Updated Title";  
}
```

Important:

We use:

```
this.variableName
```

Because we are inside class.

13 Strong Type Checking in Methods

If:

```
anotherTitle: string = "Ecom AI Mart";
```

Then:

```
this.anotherTitle = 100; // ✗ Error
```

Because it is string type.



PART 8: CLASS & OOP CONCEPT

In Angular:

```
export class HeaderComponent {
```

This is a class.

Inside class:

- Variables → Properties
- Functions → Methods

Angular is based on:

OOP (Object Oriented Programming)



PART 9: TYPESCRIPT COMPILATION PROCESS

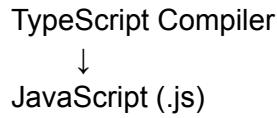
14 Does TS Convert to JS?

Yes

Internally:

TypeScript (.ts)





Why We Don't See JS File?

Because Angular CLI handles it internally.

When you run:

```
ng build
```

Production JavaScript files are generated.



PART 10: COMPLETE FLOW (BEGINNER TO ADVANCE SUMMARY)

Angular Component Architecture

Component Class (TS)

```
↓  
Properties / Methods
```

```
↓  
Interpolation {{ }}
```

```
↓  
HTML Template
```

```
↓  
Browser Rendering
```



WHAT WE LEARNED

- ✓ What is Component
 - ✓ Structure of Component
 - ✓ Variables in TypeScript
 - ✓ Strongly Typed Concept
 - ✓ Interpolation
 - ✓ Dynamic UI
 - ✓ Arrays in Component
 - ✓ Methods in Class
 - ✓ this keyword
 - ✓ TypeScript Compilation
-



WHAT COMES NEXT (ADVANCE)

Next Topics:

1. Types of Data Binding
 - Interpolation
 - Property Binding
 - Event Binding
 - Two-way Binding
 2. Component Communication
 3. Angular Lifecycle Hooks
 4. TypeScript Deep Dive
 5. Full Homepage Static Creation
 6. Dynamic Component Integration
-



FINAL UNDERSTANDING

Angular Power =

Component
+ TypeScript
+ Interpolation
+ Data Binding
+ OOP

If student understands this foundation clearly,
Angular becomes very easy.