

Angular Fundamentals

Getting Started with Angular



Jim Cooper

Software Engineer

@jimthecoop | jcoop.io

Version Check



This course was created by using:

- Angular 16



Version Check



This course is 100% applicable to:

- Angular 8-16



Recommended Prerequisites

HTML

JavaScript

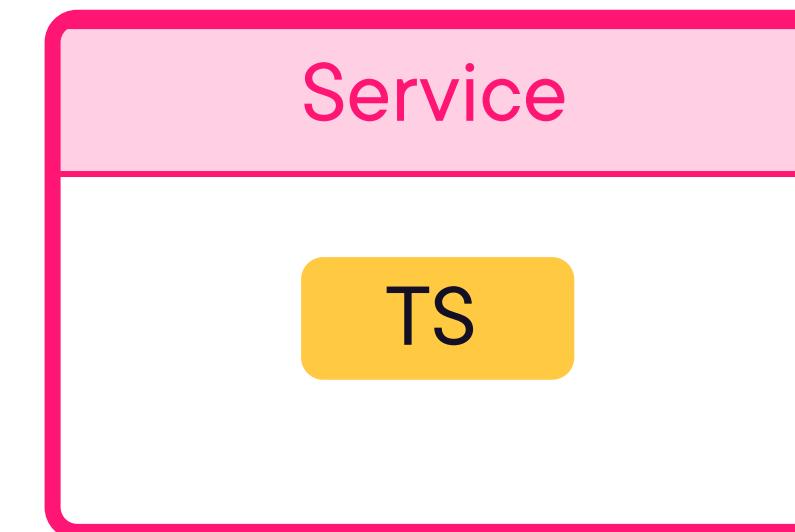
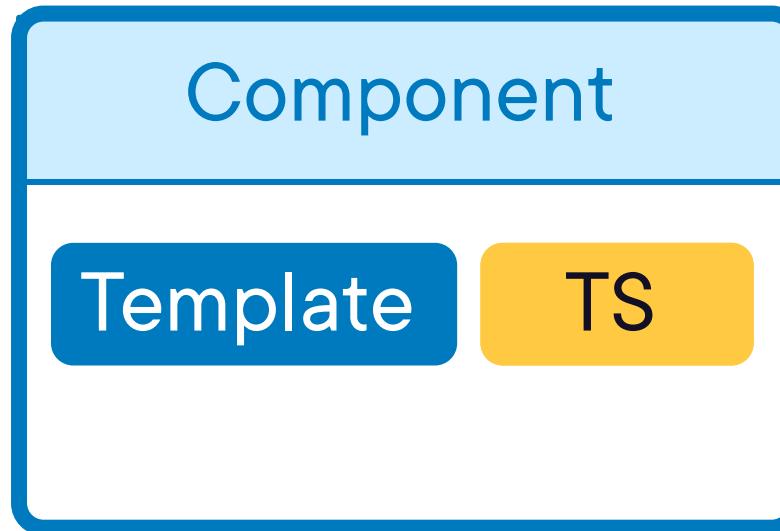
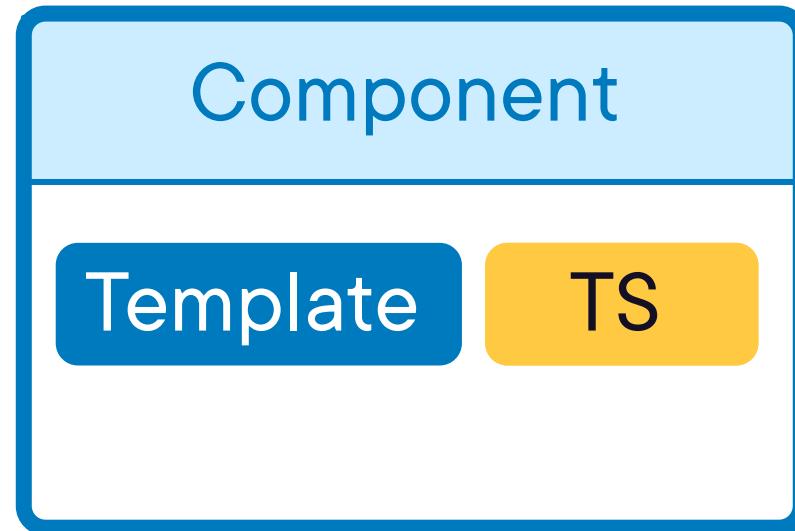
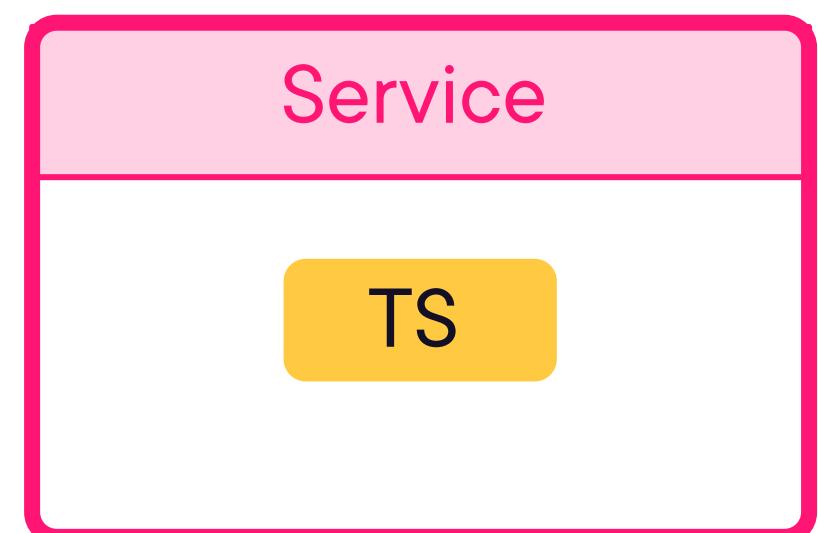
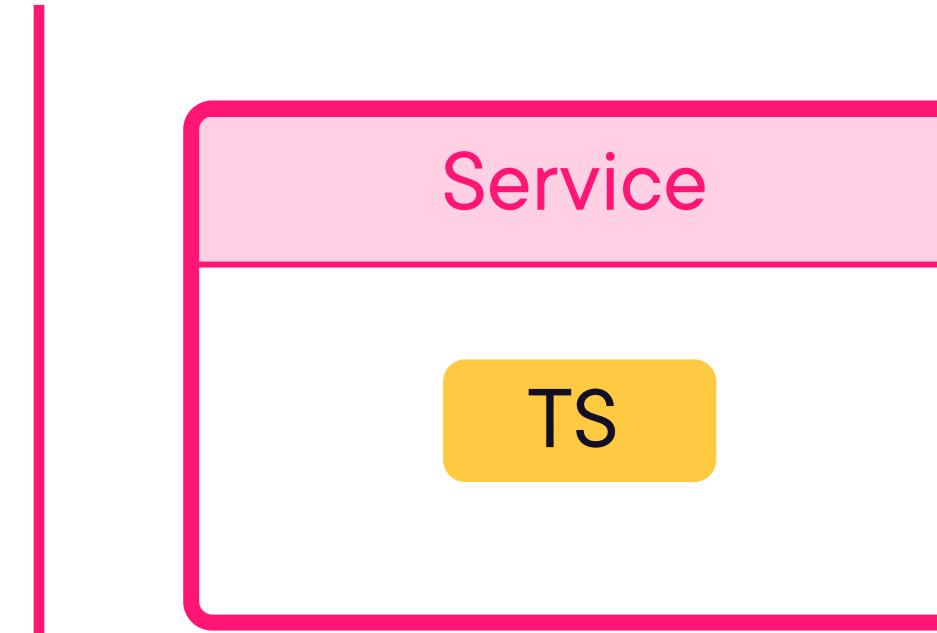
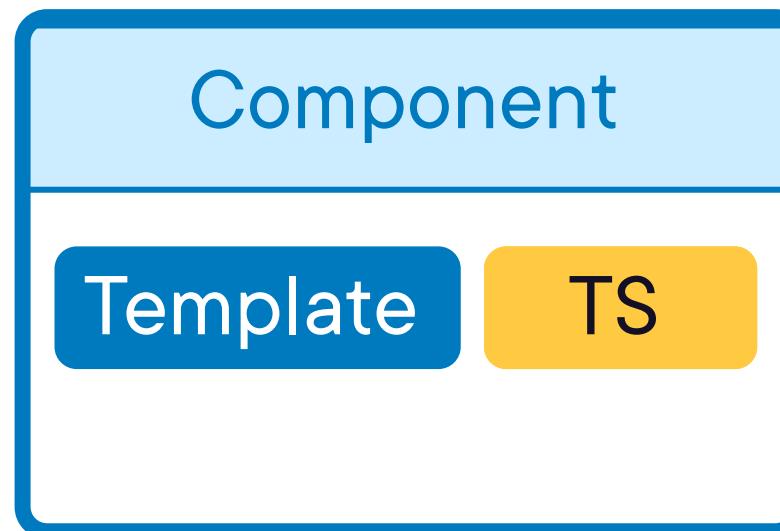
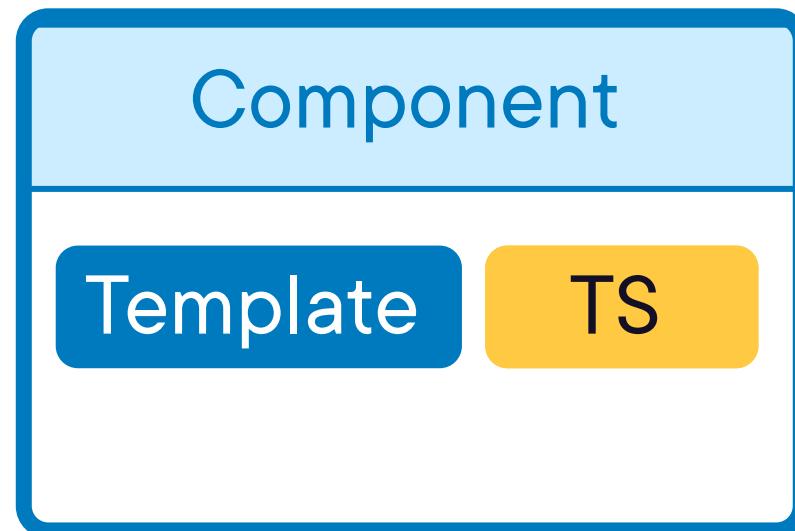
~~Basic TypeScript~~

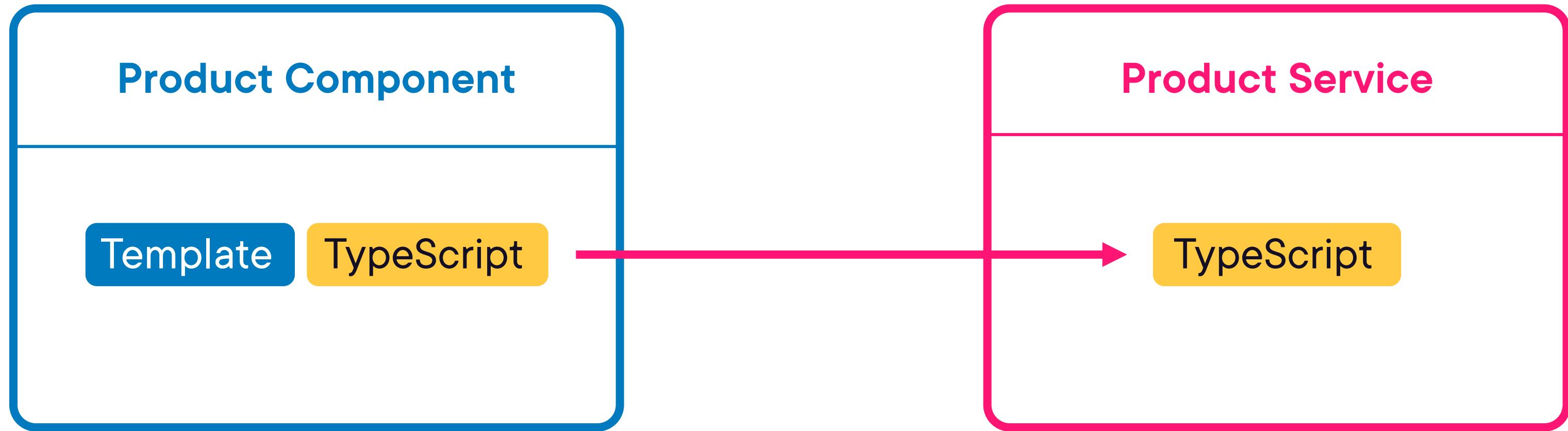


Angular Conceptual Overview

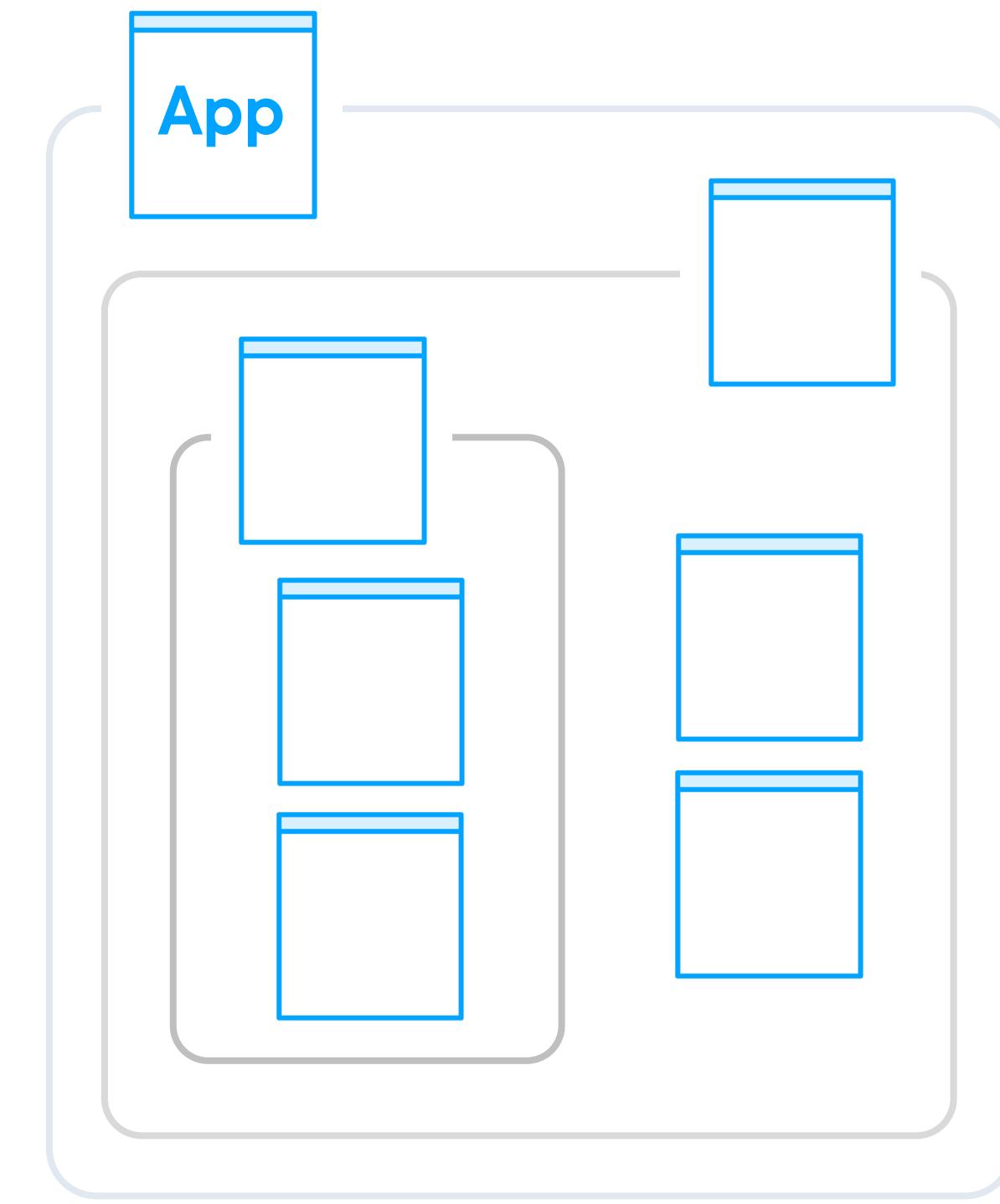


Angular Components and Services

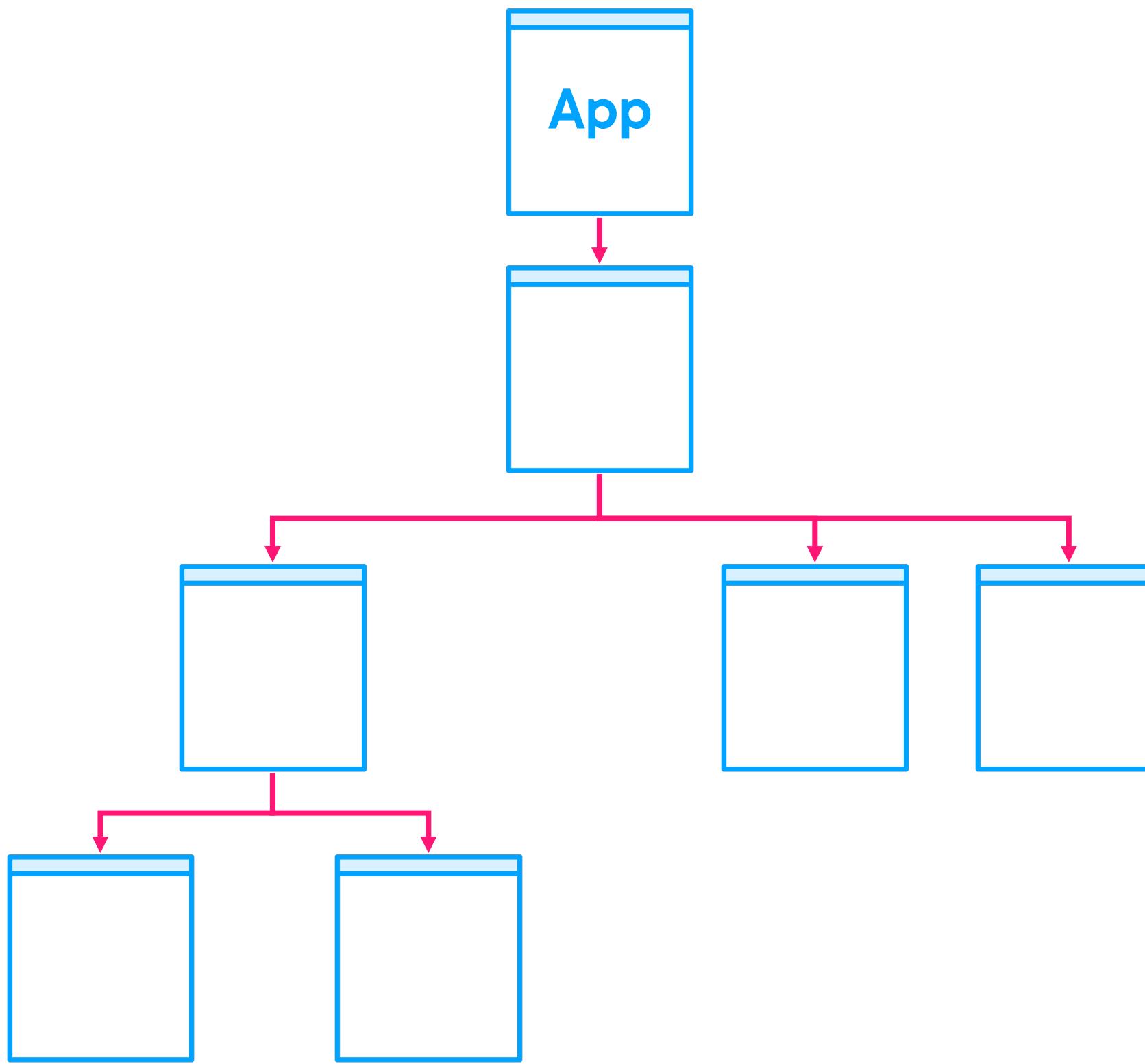




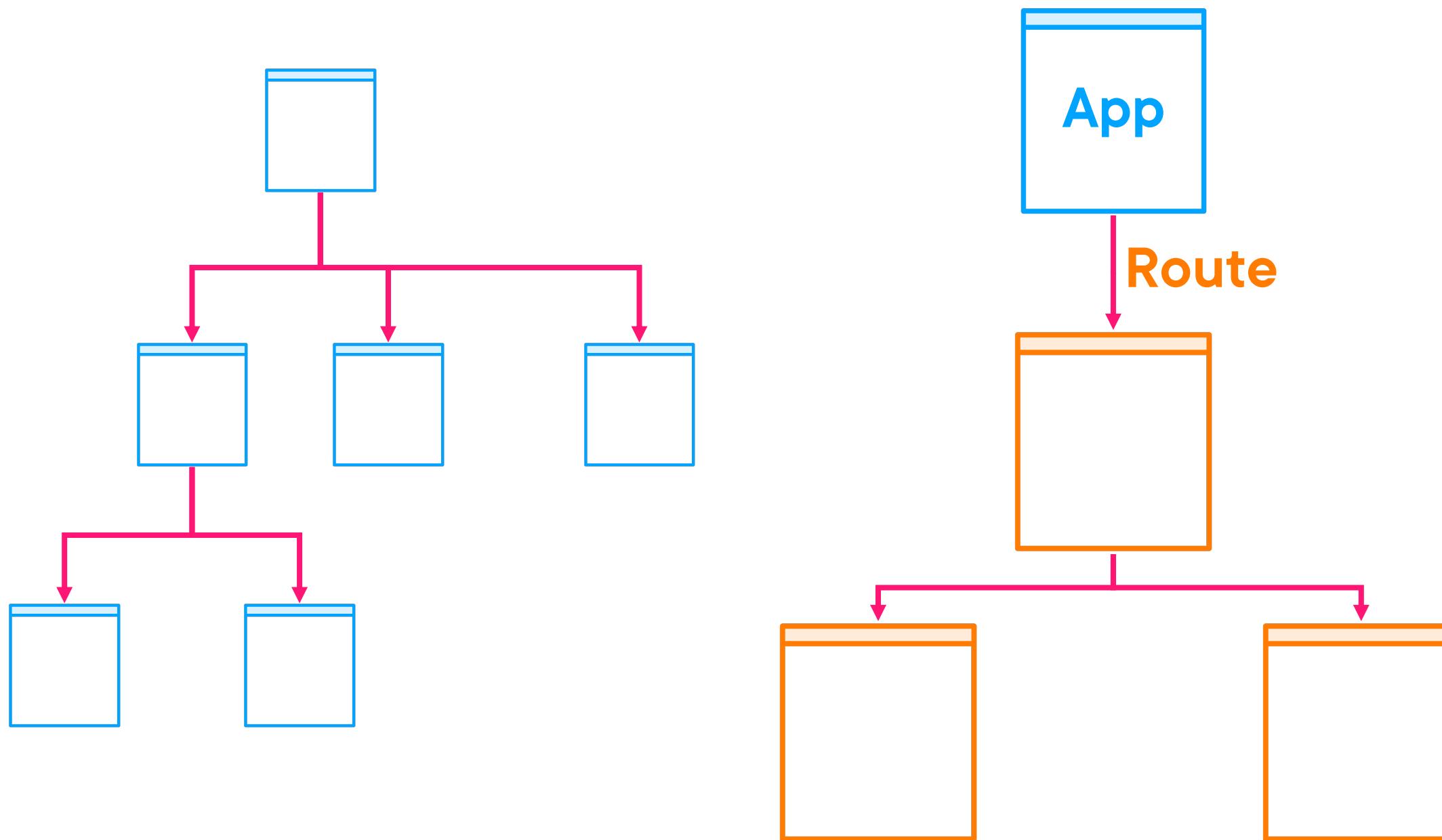
Angular Application



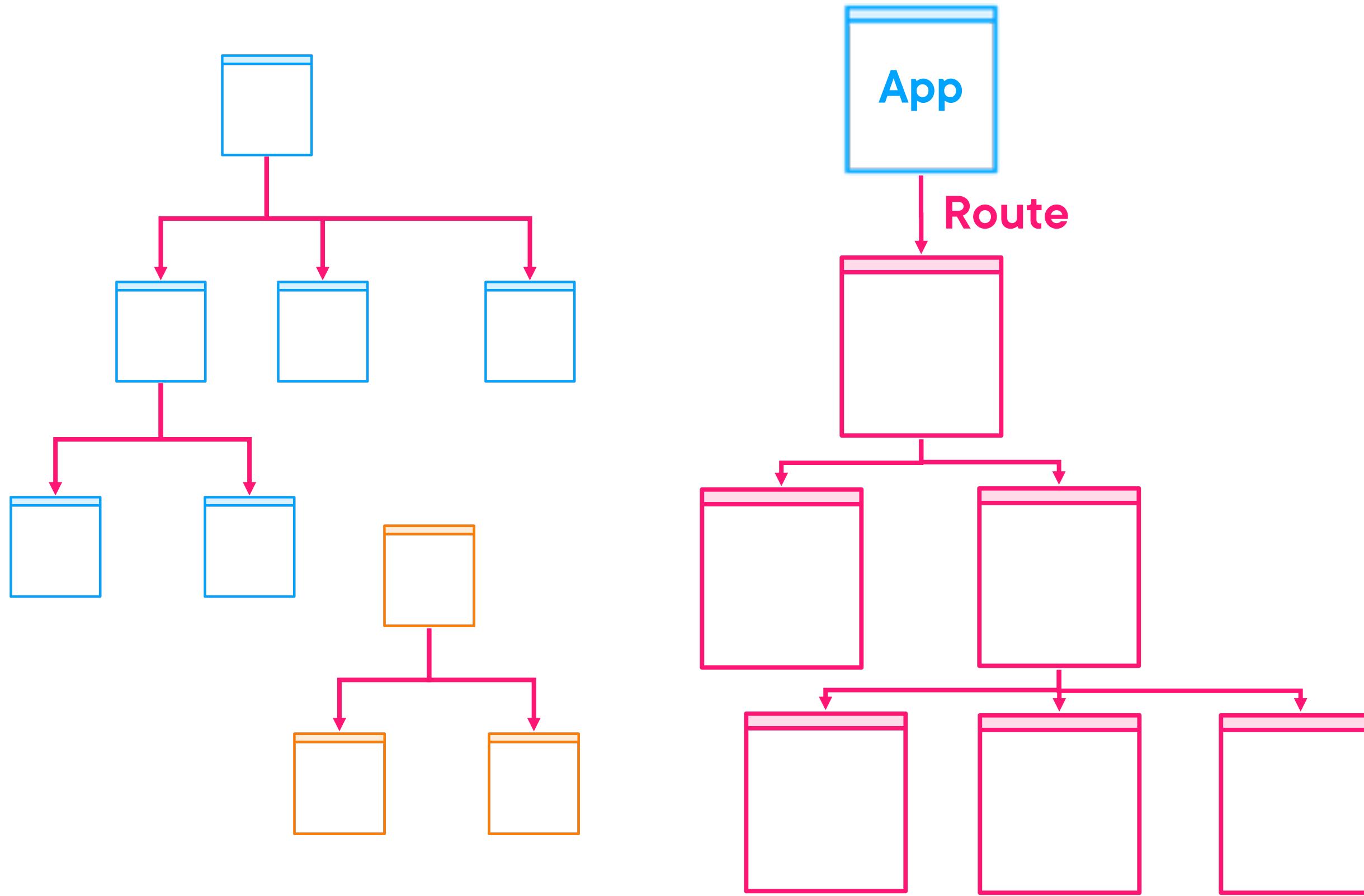
Angular Component Hierarchy



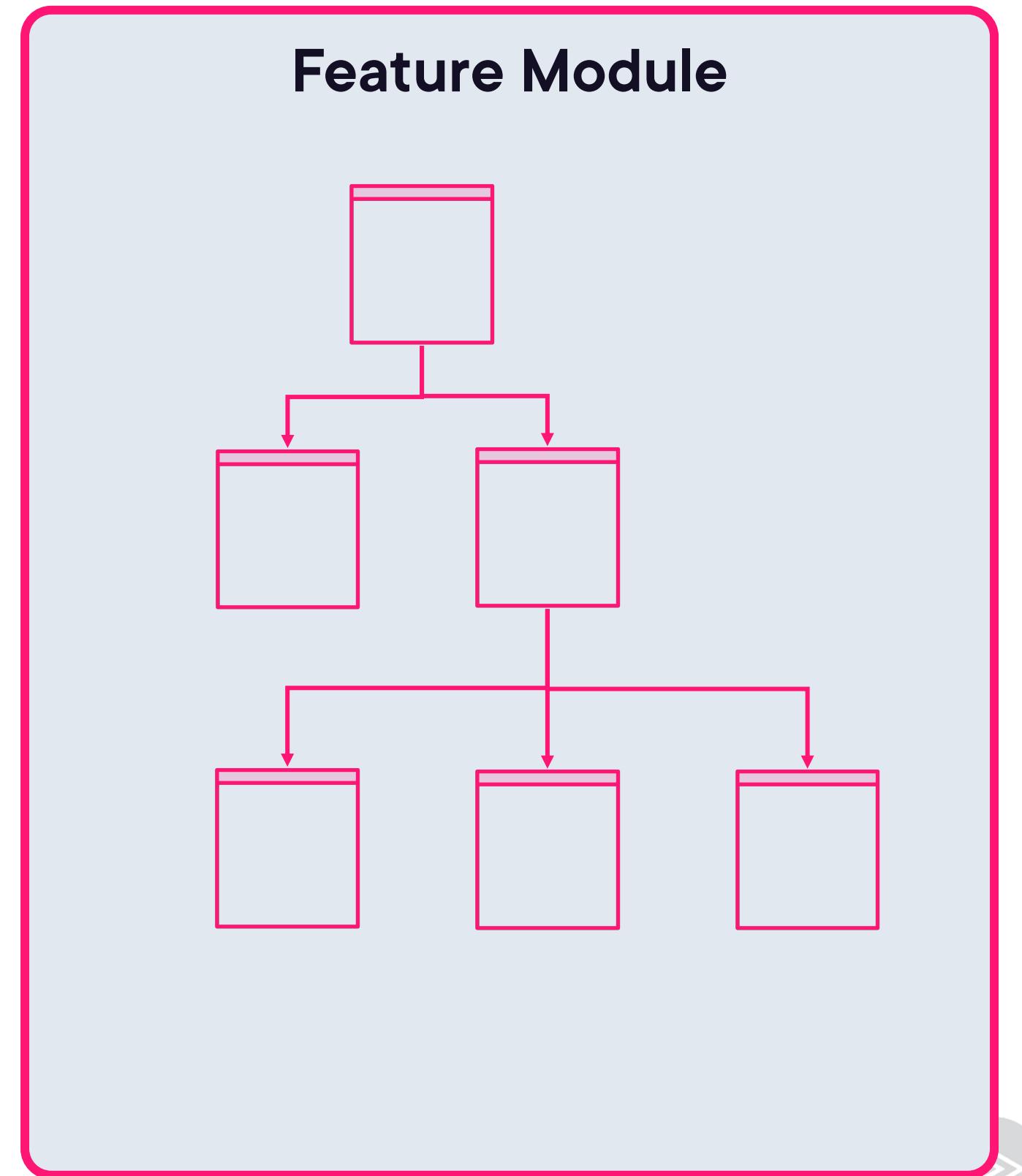
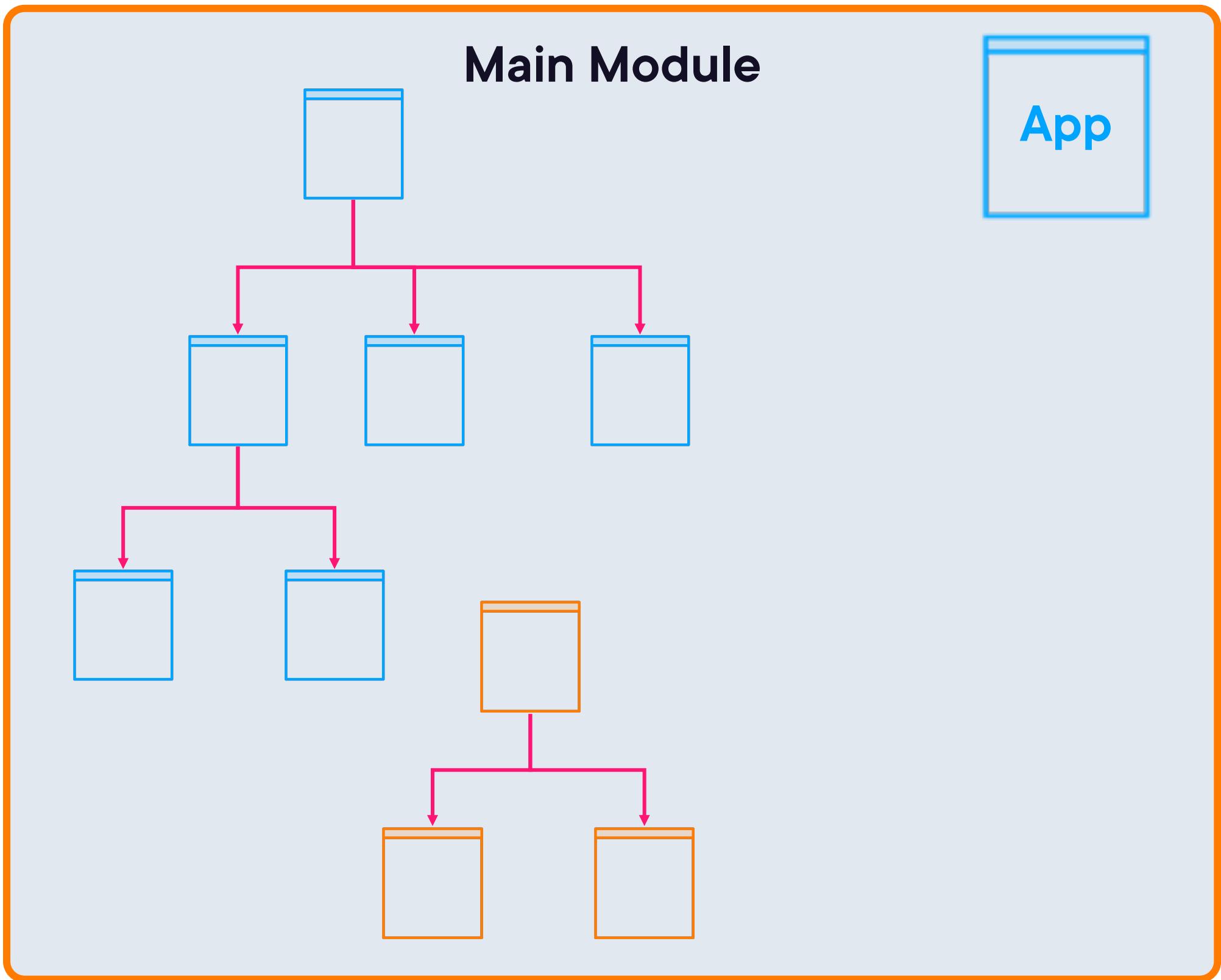
Angular Component Hierarchy



Angular Component Hierarchy



Angular Modules



TypeScript Features

Static typing

Interfaces

Class properties

Public/Private
accessibility



```
let name: string;  
let age: number;  
let birthDate: date;  
  
age = "seven"; //error
```

Static Typing



```
interface ICat {  
    name: string;  
    age: number;  
}
```

TypeScript Interfaces



```
interface ICat {  
    name: string;  
    → age: number;  
}  
let fluffy:ICat;  
                fluffy =  
                {  
                    name: 'Fluffy',  
                    → age: 'seven'  
                };
```

TypeScript Interfaces



```
interface ICat {  
    name: string;  
    age: number;  
}  
  
let fluffy:ICat = {  
    name: 'Fluffy'  
};
```

TypeScript Interfaces



```
interface ICat {  
    name: string;  
    age?: number;                fluffy =  
}  
let fluffy:ICat           {  
};  
                           name: 'Fluffy'
```

TypeScript Interfaces



```
class Cat {  
  constructor (name) {  
    this.name = name;  
  }  
}
```

TypeScript Class Properties



```
class Cat {  
    name: string;  
    constructor (name) {  
        this.name = name;  
    }  
}
```

TypeScript Class Properties



```
class Cat {  
    name: string;  
    color: string;  
    constructor (name) {  
        this.name = name;  
    }  
}
```

TypeScript Class Properties



```
class Cat {  
  → name: string;  
  constructor (name) {  
    this.name = name;  
  }  
  → speak() { console.log('meow'); }  
}
```

Public and Private Accessibility



```
class Cat {  
    name:string  
    constructor (name) {  
        this.name = name;  
    }  
    speak() { console.log('My name is: ' + this.name) }  
}
```

Public and Private Accessibility



```
class Cat {  
  name:string;  
  speak() { console.log('My name is: ' + this.name); }  
}  
  
let fluffy = new Cat();  
console.log(fluffy.name);  
fluffy.speak();
```

Public and Private Accessibility



```
class Cat {  
    private name:string;  
    private speak() { console.log('My name is: ' + this.name); }  
}
```

```
let fluffy = new Cat();  
console.log(fluffy.name);  
fluffy.speak();
```

Public and Private Accessibility



```
class Cat {  
    private name:string;  
    private speak() { console.log('My name is: ' + this.name); }  
}
```

```
let fluffy = new Cat();  
console.log(fluffy.name); //compile-time error  
fluffy.speak(); // compile-time error
```

Public and Private Accessibility



```
class Cat {  
    private name: string;  
    private color: string;  
    constructor(name, color) {  
        this.name = name;  
        this.color = color;  
    }  
}
```

```
class Cat {  
    constructor(private name, private color) {  
    }  
}  
let fluffy = new Cat('Fluffy', 'White');
```

Public and Private Accessibility



System Prerequisites

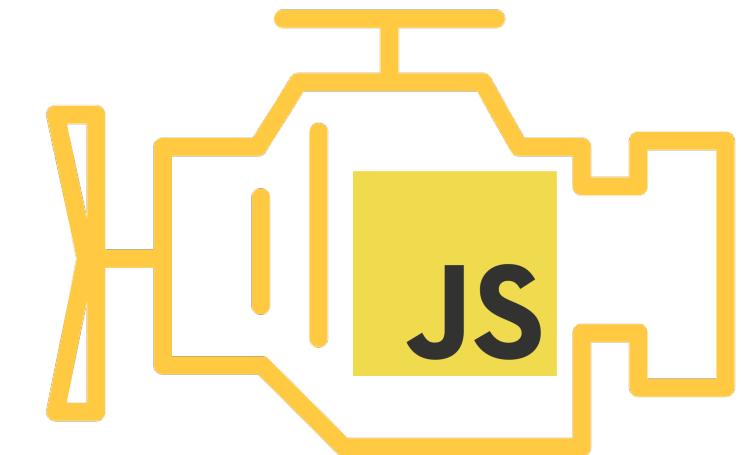
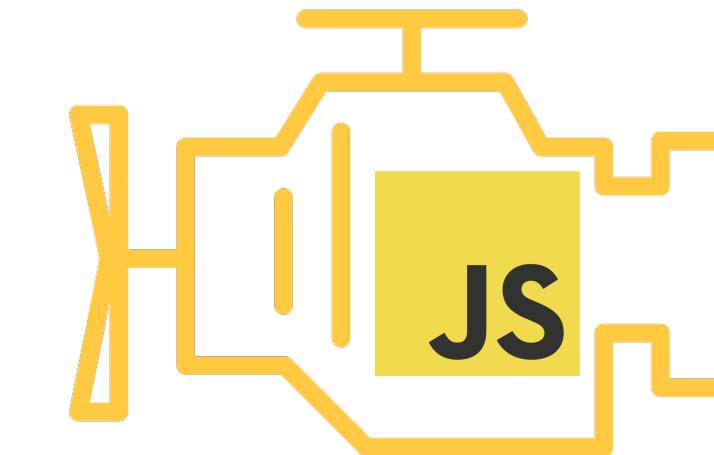
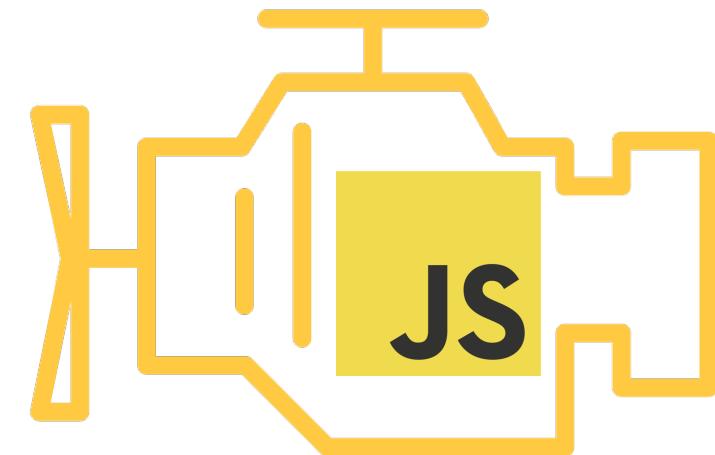
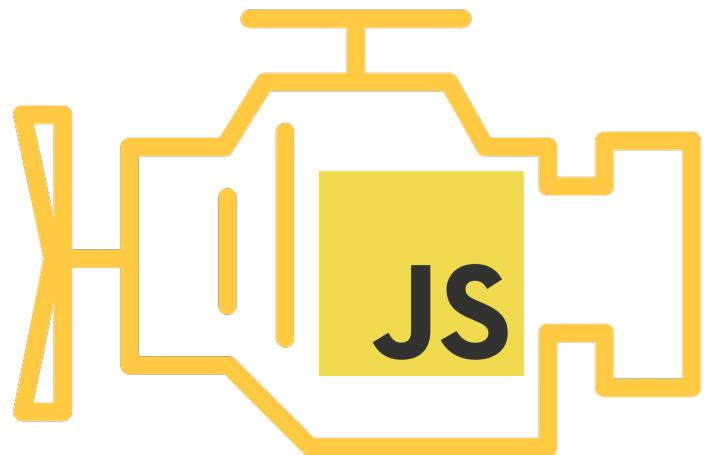
Install an editor

- Visual Studio Code
- <https://code.visualstudio.com>

Install Node.js / NPM

Install the Angular CLI





Up Next:

Creating Angular Components

