

Angular Fundamentals

day 3 - Short recap



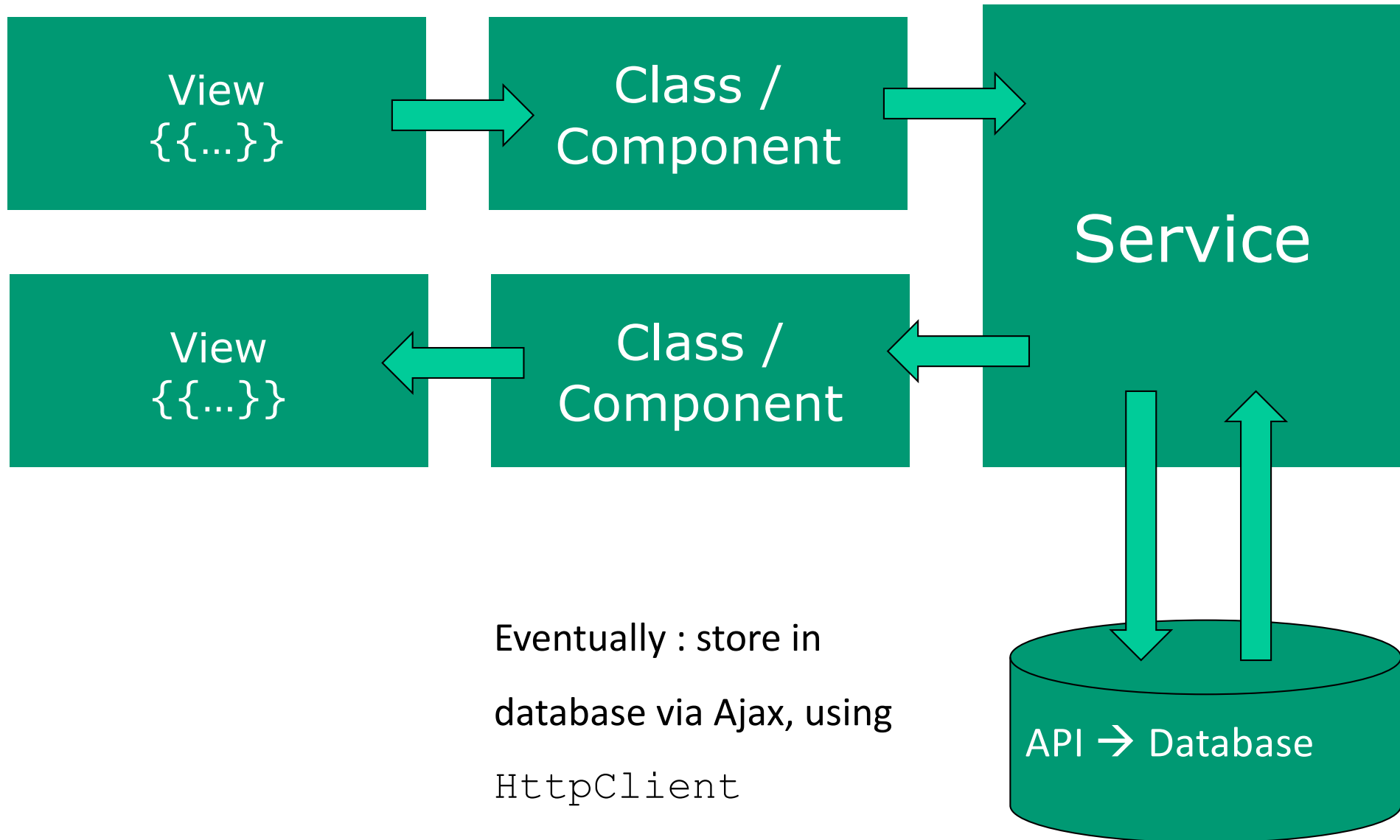
Atos

Peter Kassenaar –
info@kassenaar.com

Yesterday

- Data binding (continued)
 - Event binding (...)
 - Attribute binding [...]
 - Two-way Data Binding [(...)]
- Services
 - Static Services
 - RxJS / Observables
 - Live API's

Services with multiple components



Main concept:

Dependency Injection

Angular http-calls

Initial Output

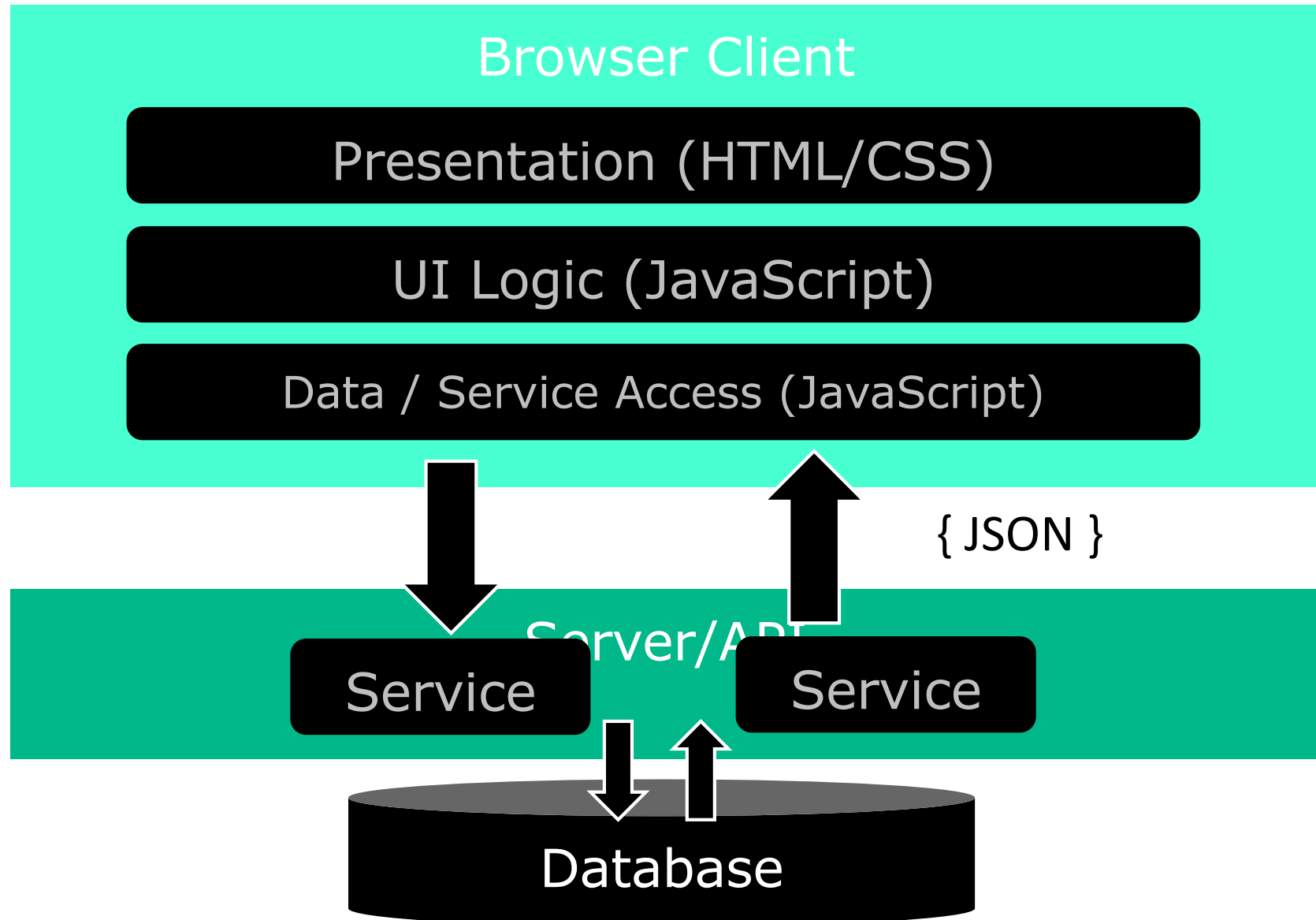
```
this.http.get<City[]>( 'assets/data/cities.json' )  
    .pipe(  
        filter(...),  
        map(...)  
    )  
    .subscribe((result) => {  
        //... Do something  
    });
```

Optional:
operator(s)

Final Input

Single Page Application

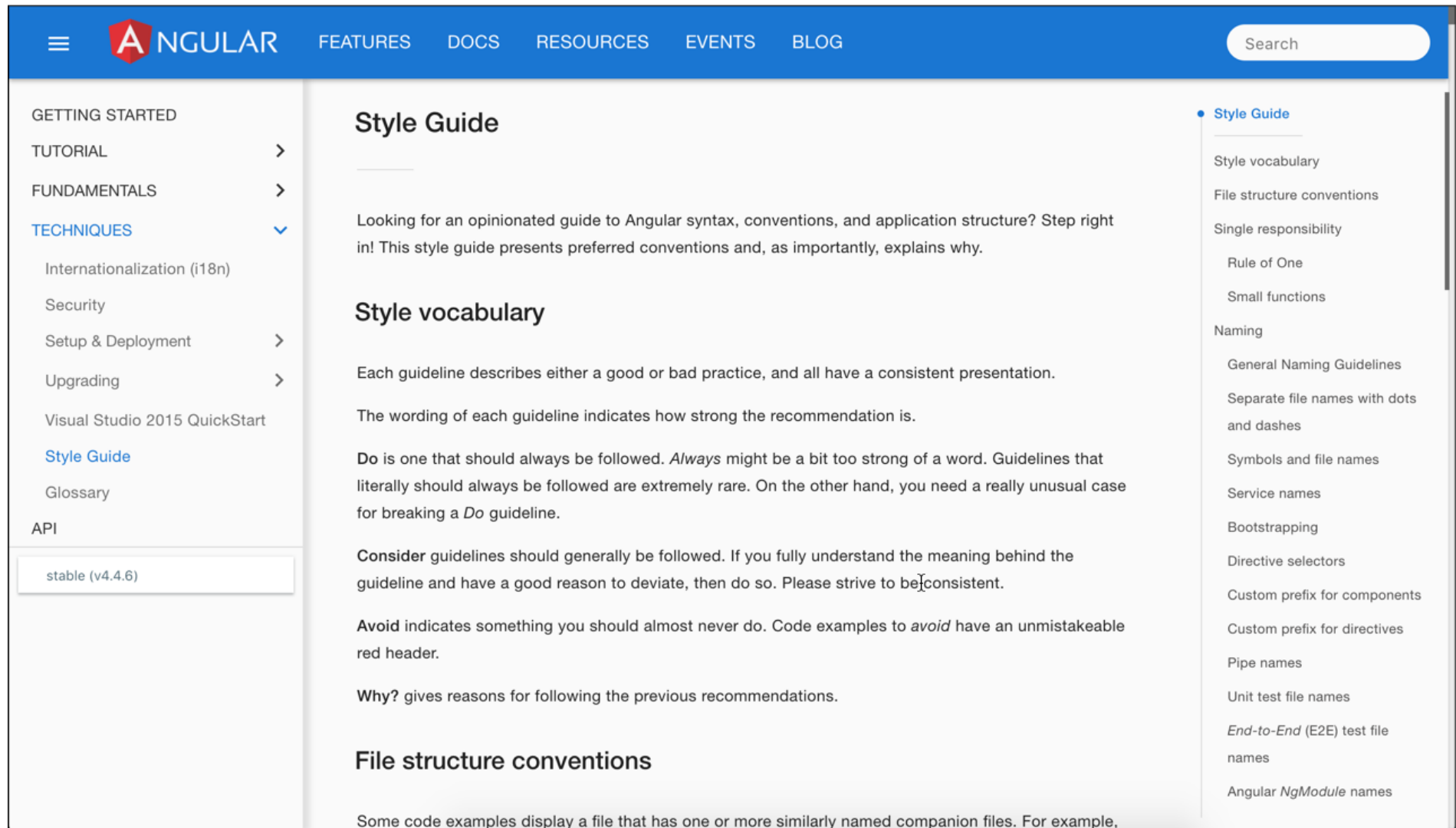
2010 – 20??



Today

- Component Communication using `@Input()` and `@Output()`
- Writing an Event bus using `Subject<Type>`
- Routing
- Forms
 - Template driven Forms
 - Model driven Forms

Angular Style Guide



The screenshot shows the Angular Style Guide page. The top navigation bar is blue with the Angular logo and links to FEATURES, DOCS, RESOURCES, EVENTS, and BLOG. A search bar is on the right. The left sidebar contains a list of categories: GETTING STARTED, TUTORIAL, FUNDAMENTALS, TECHNIQUES (highlighted with a blue checkmark), Internationalization (i18n), Security, Setup & Deployment, Upgrading, Visual Studio 2015 QuickStart, Style Guide (highlighted with a blue dot), Glossary, and API. Below the sidebar, there is a version selector showing 'stable (v4.4.6)'. The main content area has the title 'Style Guide' and a paragraph: 'Looking for an opinionated guide to Angular syntax, conventions, and application structure? Step right in! This style guide presents preferred conventions and, as importantly, explains why.' Below this is the section 'Style vocabulary' with a paragraph: 'Each guideline describes either a good or bad practice, and all have a consistent presentation. The wording of each guideline indicates how strong the recommendation is. Do is one that should always be followed. Always might be a bit too strong of a word. Guidelines that literally should always be followed are extremely rare. On the other hand, you need a really unusual case for breaking a Do guideline. Consider guidelines should generally be followed. If you fully understand the meaning behind the guideline and have a good reason to deviate, then do so. Please strive to be consistent. Avoid indicates something you should almost never do. Code examples to avoid have an unmistakable red header. Why? gives reasons for following the previous recommendations.' Below this is the section 'File structure conventions' with a paragraph: 'Some code examples display a file that has one or more similarly named companion files. For example,'. The right sidebar contains a list of links: Style Guide (highlighted with a blue dot), Style vocabulary, File structure conventions, Single responsibility, Rule of One, Small functions, Naming, General Naming Guidelines, Separate file names with dots and dashes, Symbols and file names, Service names, Bootstrapping, Directive selectors, Custom prefix for components, Custom prefix for directives, Pipe names, Unit test file names, End-to-End (E2E) test file names, and Angular NgModule names.

ANGULAR FEATURES DOCS RESOURCES EVENTS BLOG Search

GETTING STARTED
TUTORIAL >
FUNDAMENTALS >
TECHNIQUES ✓
Internationalization (i18n)
Security
Setup & Deployment >
Upgrading >
Visual Studio 2015 QuickStart
Style Guide
Glossary
API
stable (v4.4.6)

Style Guide

Looking for an opinionated guide to Angular syntax, conventions, and application structure? Step right in! This style guide presents preferred conventions and, as importantly, explains why.

Style vocabulary

Each guideline describes either a good or bad practice, and all have a consistent presentation.

The wording of each guideline indicates how strong the recommendation is.

Do is one that should always be followed. *Always* might be a bit too strong of a word. Guidelines that literally should always be followed are extremely rare. On the other hand, you need a really unusual case for breaking a *Do* guideline.

Consider guidelines should generally be followed. If you fully understand the meaning behind the guideline and have a good reason to deviate, then do so. Please strive to be consistent.

Avoid indicates something you should almost never do. Code examples to *avoid* have an unmistakable red header.

Why? gives reasons for following the previous recommendations.

File structure conventions

Some code examples display a file that has one or more similarly named companion files. For example,

- Style Guide
- Style vocabulary
- File structure conventions
- Single responsibility
- Rule of One
- Small functions
- Naming
 - General Naming Guidelines
 - Separate file names with dots and dashes
 - Symbols and file names
 - Service names
 - Bootstrapping
 - Directive selectors
 - Custom prefix for components
 - Custom prefix for directives
 - Pipe names
 - Unit test file names
 - End-to-End (E2E) test file names
 - Angular NgModule names

<https://angular.io/guide/styleguide>

Tomorrow - Advanced

- Multiple Modules,
- Advanced Routing - lazy loading
- Advanced components
- Unit Testing - E2E Testing
- Angular in the enterprise / architecture for bigger applications
- (Introduction - State Management with @ngrx/store)
- ...