Scaffolding Apps with Angular CLI

Course Map: Day 1

1. ASP.NET Core Architecture
2. Getting Started with ASP.NET Core
3. Design Patterns, Unit Testing
4. Entity Framework Core

Course Map: Day 2

5. Introduction to TypeScript

6. Using VS Code for TypeScript

7. Angular 2 Architecture

8. **Using Angular CLI for Client Apps**

Agenda

- Your first app with Angular CLI
- Adding styles with bootstrap.js
- Generating models and components
- Adding data binding to the template
- Injecting services
- Connecting to the Web API

Get the Bits

github.com / tonysneed /

Kliant.AspNetCore-Angular

Beyond Hello World: Angular CLI

- **Command-line tool** for scaffolding Angular 2 applications
- Build **starter** app, then add items
- Run unit and end-to-end **tests**
- Lint and **format** code
- Create builds for both **dev** and **production**

Install and Use Angular CLI

```
npm install -g angular-cli

ng new hello-angular

cd hello-angular

ng serve
```

Adding Style

- Angular supports style libraries such as **Bootstrap**

```
npm install ng2-bootstrap bootstrap jquery --save
```

- Update **angular-cli.json**

```
"styles": "../node_modules/bootstrap/dist/css/bootstrap.min.css"
"scripts": [  "../node_modules/jquery/dist/jquery.js",
  "../node_modules/bootstrap/dist/js/bootstrap.js" ]
```

Scaffold Models and Components

- Create a new **model**

```
ng generate class shared/models/product
```

- Create a new **component**

```
ng g component products
```

Import Model into the Component

```
import { Product } from '../shared/models/product';

// Other code elided for clarity
export class ProductsComponent implements OnInit {

  title: string = 'Products'; products: Product[]; error: any;
  constructor() { }
  ngOnInit() {     // Initialize the products property
    this.products = [
      new Product(1, 'Product 1', 10),
      new Product(2, 'Product 2', 20),
    ]; } }
```

Create Template for the Component

```
<table class='table' *ngIf="products && products.length">
    <thead><tr><th>Product Name</th><th>Unit Price</th></tr></thead>
    <tbody>
        <tr *ngFor="let product of products">
            <td>{{ product.productId }}</td>
            <td>{{ product.productName }}</td>
            <td>{{ product.unitPrice }}</td>
        </tr>
    </tbody>
</table>
```

Scaffold Service

- Generate a **service** using Angular CLI

```
ng g service shared/services/products
```

Create Service: Imports

```
import { Injectable } from '@angular/core';
import { Http } from '@angular/http';
import 'rxjs/add/operator/toPromise';

import { Product } from '../models/product';
import { Urls } from '../constants';
```

Create Service: Methods

```
@Injectable()
export class ProductsService {

 constructor() { }

  getProducts(): Product[] {
    return [ new Product(1, 'Product 1', 10),
      new Product(2, 'Product 2', 20),];
  }
}
```

Register Service

```
// app.module.ts
import { ProductsService } from ‘./shared/services/products.service';

@NgModule({
 declarations: [AppComponent, ProductsComponent],

  // Code elided for clarity
  providers: [ProductsService],
  bootstrap: [AppComponent]
})
```

Use Service

```
// products.component.ts

export class ProductsComponent implements OnInit {

    constructor(private _productService: ProductsService) { }

    ngOnInit() {
        this.products = this._productService.getProducts();
}
```

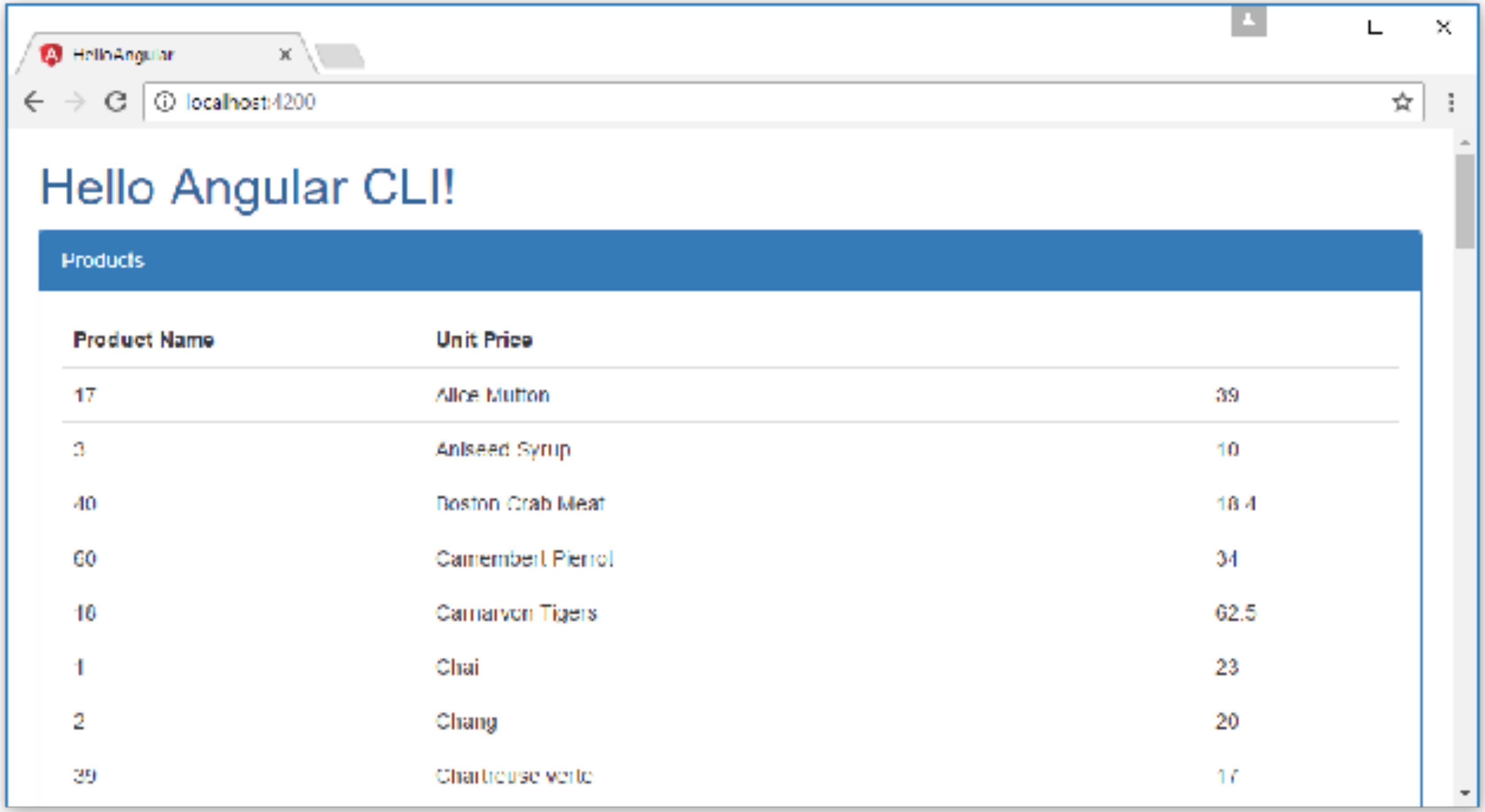
Use Service with HTTP

```
@Injectable()
export class ProductsService {
    constructor(private _http: Http) { }

async getProducts(): Promise<Product[]> {
  try {
    return (await this._http.get(this._productsUrl)
      .toPromise()).json() as Product[];
  } catch (error) {
    throw error.message || error;
  }
}
```

Use Async Service

```
export class ProductsComponent implements OnInit {
    constructor(private _productService: ProductsService) { }

    async ngOnInit() {
        try {
          this.products = await this._productService.getProducts();
        } catch (error) {
          this.error = error;
        }
    }
}
```

Hello Angular CLI with ASP.NET Core



Hello Angular CLI!

Products

Product Name	Unit Price	
17	Alice Mutton	39
3	Aniseed Syrup	10
40	Boston Crab Meat	18.4
60	Camembert Pierrot	34
18	Carnarvon Tigers	62.5
1	Chai	23
2	Chang	20
39	Chartreuse verte	17

Demo: Angular CLI with ASP.NET Core

Questions?