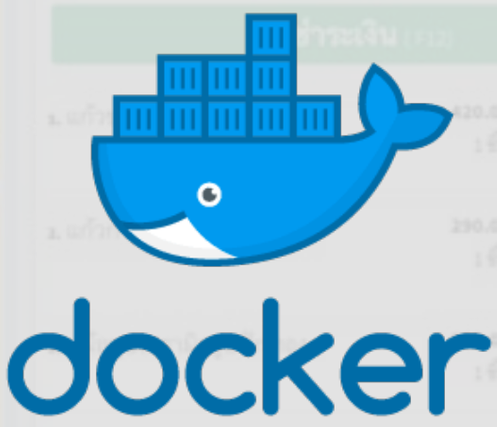


Angular + ASP .NET Core + Docker

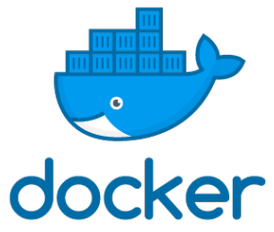
Point of Sale workshops

CODEMOBILES
COMPANY LIMITED

Day1

Install (latest version)

- Node (LTS)
- Angular CLI
- ASP .NET Core
- VSCode
  - C# (Extension)
- Docker
- Postman



Extensions

Install Extensions VSCode

Angular

- Angular2-switcher
- Angular 8 Snippets typeScript, Html, ….. -
- Angular Snippets (Version 8)
- Angular Language Service.
- Angular Schematics.
- TSLint // optional

CODEMOBILES COMPANY LIMITED

Install Extensions VSCode (cont.)

ASP .NET Core

- WilderMinds' ASP.NET Core Snippets
- SQL Server (mssql)
- C# Extensions
- Docker
- Dotnet core commands
- NuGet Package Manager

Install Extensions VSCode (cont.)

Additional

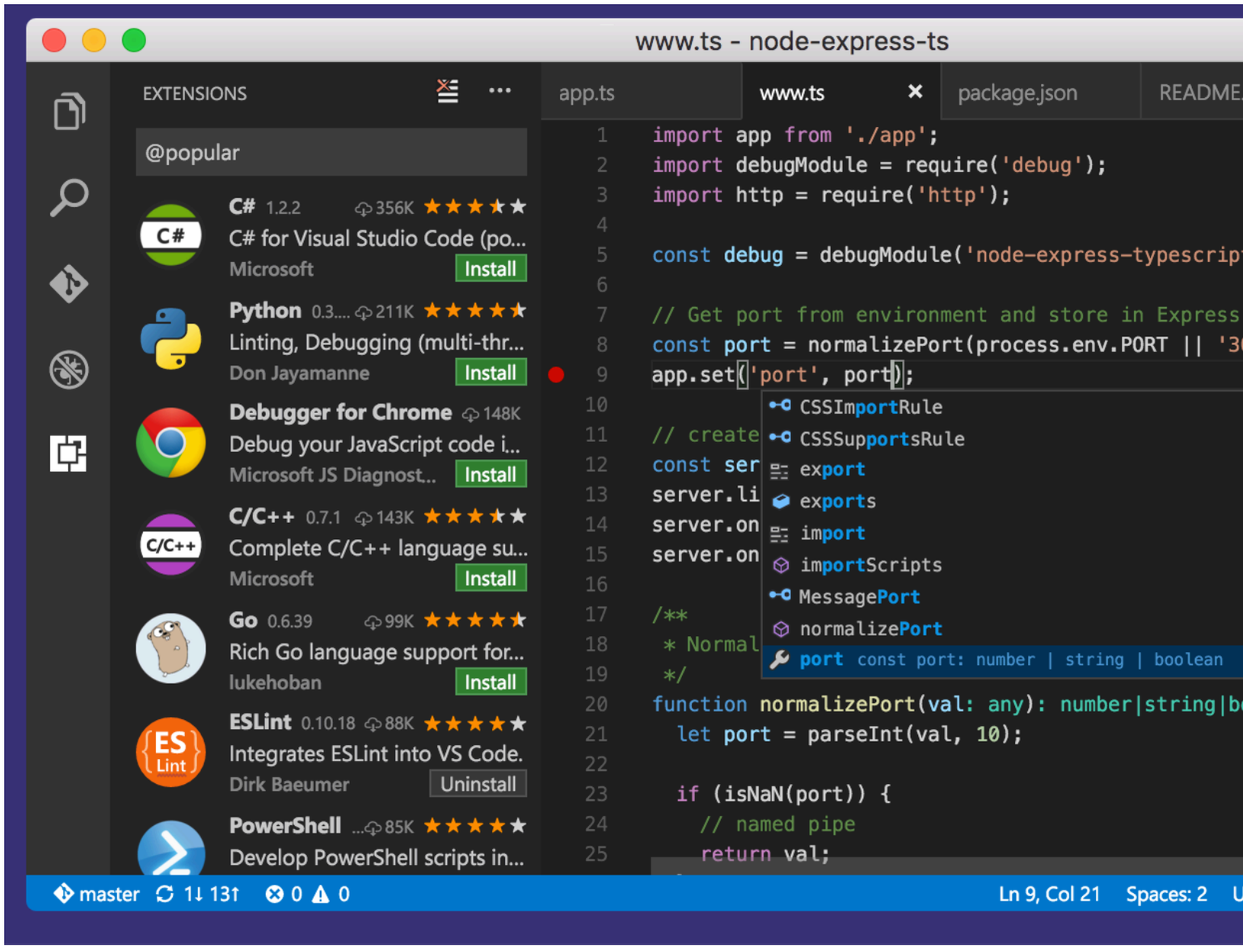
- Vscode-icon
- Auto close tag
- Auto rename tag
- vscode-faker
- Guides
- htmltagwrap
- npm Intellisense
- JavaScript (ES6) code snippets
- jQuery Code Snippets
- path-intellisense
- Terminal
- Terminal tabs
- Trailing Spaces
- Bootstrap 4, Font awesome 4, ….

Recommended

- Bracket-pair-colorizer
- Bookmarks
- Numbered Bookmarks
- Better Comments
- Peacock
- Beautify
- Change-case
- Dracula Official
- Material Theme
- EditorConfig for VS Code
- Indenticator
- Live Share
- Paste JSON as Code
- Code Runner
- Azure Cosmos DB
- Azure Account
- Prettier - Code formatter
- Project Manager
- REST Client
- Live Server
- TODO Highlight
- GitLens — Git supercharged
- Quokka.js
- vscode-spotify

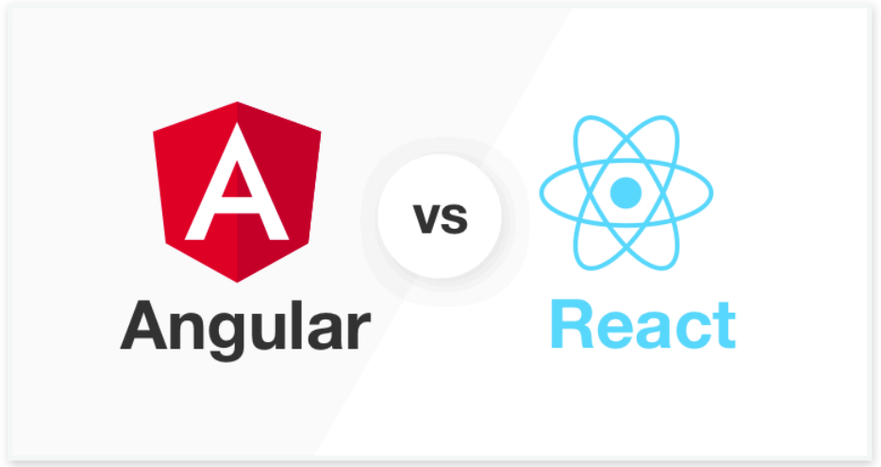
Setting VSCode

- Font size
- Auto save
- Icon
- Theme

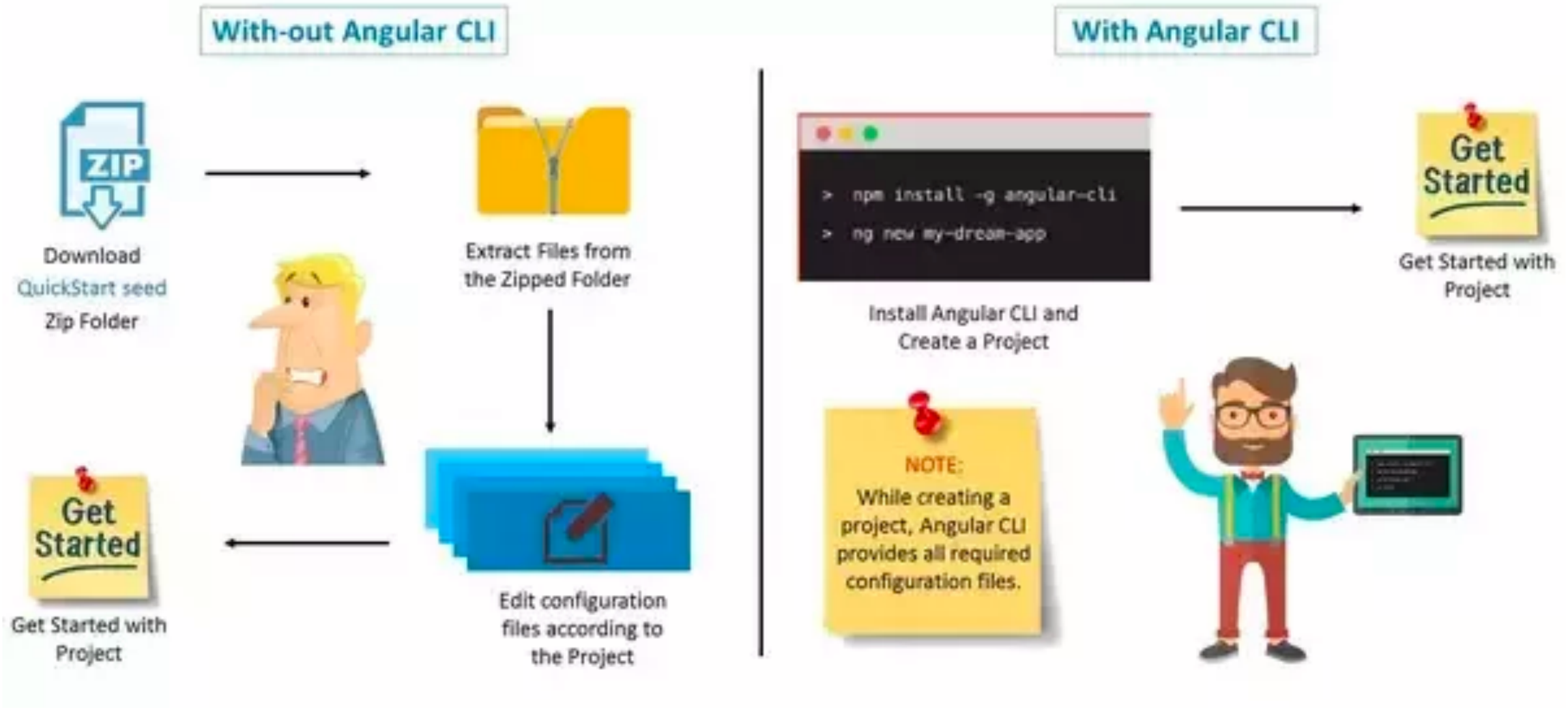


CODEMOBILES COMPANY LIMITED

- What & Why Angular
- Angular vs React
- Angular CLI
- Create project
  - ng n {project-name}
  - code . (open project on vscode)
  - ng s [-o l --port], npm start
- Project Structure
  - dependencies
  - index.html
  - components
  - app.module.ts



Angular CLI



NPM Install Error:Unexpected end of JSON input while parsing near '…nt-webpack-plugin“:”0'

npm cache clean --force

npm install -g @angular/cli@latest

Global package:

```
npm uninstall -g @angular/cli
npm cache verify
# if npm version is < 5 then use `npm cache clean`
npm install -g @angular/cli@latest
```

Upgrade version

Angular CLI Commands

By @yaircarreno with special thanks to @John_Papa and @DeborahKurata.

Generating new projects

Generate a new app in /my-app
ng new my-app

Don't write the files, but report them.
ng new my-app --dry-run

Style should use SASS
ng new my-app --style sass

Generate app with routing class included.
ng new my-app --ng4 --routing

Set the default selector prefix(i.e. mp).
ng new my-app --prefix mp

Linting

Lint the typescript code.
ng lint

Generating components

Generate a new component.
ng g c my-component

Generating directives

Generate a new directive.
ng g d my-directive

Generate a directive in your own folder.
ng g d my-directive --flat false

Generating pipes

Generate a new pipe.
ng g p my-pipe

Generating modules

Generate my-module.module.ts.
ng g module my-module

Generate my-module.module.ts and my-module-routing.module.ts.
ng g module my-module --routing

Generating services

Generate a new service.
ng g s my-service

Include the provider service into app.module class.
ng g s my-service -m app.module

Generating guards

CanActive guard auth.guard.ts.
ng g guard auth

Build commands

Output dev build files.
ng build

Output prod build files.
ng build --prod

Serving commands

Serve a dev build in memory.
ng serve

Serve and lanch the browser.
ng serve -o

Generating classes

Generate a new model class.
ng g cl my-class-model

Generating interfaces

Generate a new interface.
ng g i my-interface

Generating enums

Generate a new enum.
ng g e my-enum

Common commands

Don't create a /my-component folder.
--flat

Put the template in the .ts file.
--inline-template

Put the style in the .ts file.
--inline-style

Don't create a spec file.
--spec false

https://www.yaircarreno.com/2017/04/commands-in-angular-cli.html

src/app/app.component.html

```
<!-- "The sum of 1 + 1 is not 4" -->
<p>The sum of 1 + 1 is not {{1 + 1 + getVal()}}.</p>
```

src/app/app.component.html

```
<p>{{title}}</p>
<div><img src="{{itemImageUrl}}"></div>
```

src/app/app.component.html

```
<h3>Current customer: {{ currentCustomer }}</h3>
```

One-way from data source to view target

Event

src/app/app.component.html

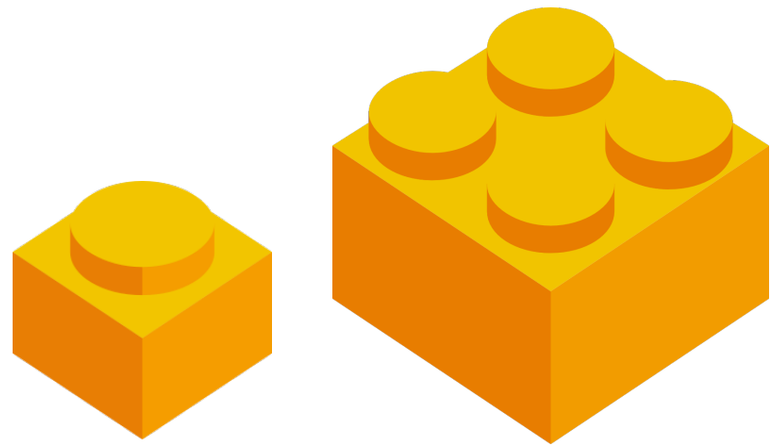
```
<button (click)="deleteHero()">Delete hero</button>
```

src/app/app.component.html

```
<button (click)="onSave($event)">Save</button>
<button *ngFor="let hero of heroes" (click)="deleteHero(hero)">{{hero.name}}</button>
<form #heroForm (ngSubmit)="onSubmit(heroForm)"> ... </form>
```

One-way from view target to data source

- What is component
- Generate Component
  - ng g c {folder}/{component-name}
    - option --spec=false
- Meta data
  - Selector (ref. selector in {component-name}.ts)
  - templateUrl
  - styleUrls



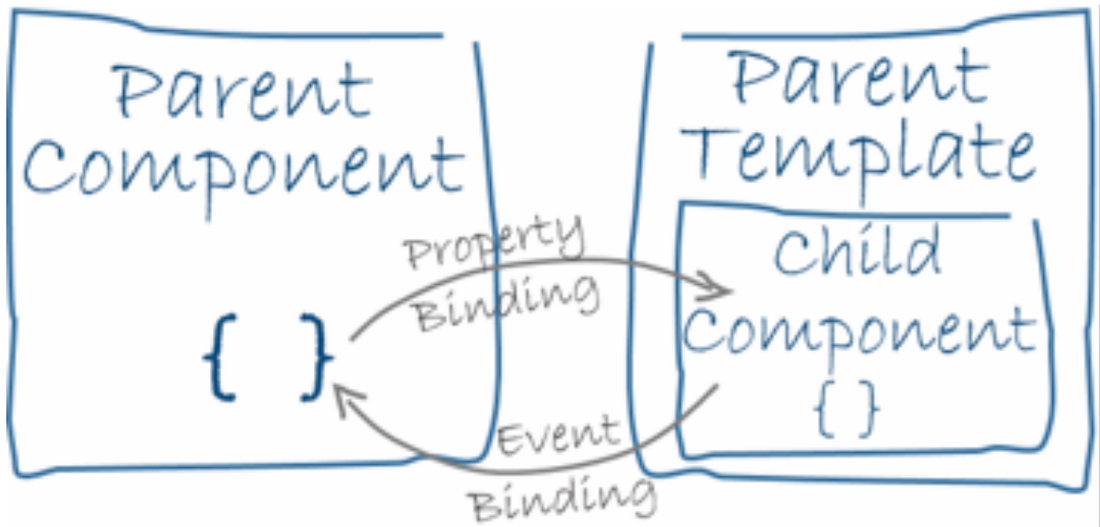
Shortcut key

- **Go to file**
  - command + p (mac)
  - ctrl + p (windows)

- **Switch between .ts and .html (angular2-switcher extension)**
  - alt + shift + o (mac)
  - alt + o (windows)

- **Go to symbol**
  - command + shift + o (mac)
  - ctrl + shift + o (windows)

- **Parent to child**
  - @Input() or Input('id')
  - @ViewChild (#id)
    - Html
    - Programming



Keyword : Property binding

- **Child to parent**
  - @Output() or Output('id') and EventEmitter
    - mark: make sure import
    - mark: beware symbol '$event' (html)
    - EventEmitter<datatype> datatype ref. emit
      - exam: EventEmitter<void>(); xxx.emit()
      - exam: EventEmitter<number>(); xxx.emit(1234)

Keyword : Event binding

Component Communication

child selector

source

property from parent

source

method from parent

```
<app-input-output [item]="currentItem" (deleteRequest)="crossOffItem($event)"></app-input-output>
```

target

@Input() property from child

target

@Output() event from child

Parent and children communicate via a service

component-interaction/src/app/mission.service.ts

```
import { Injectable } from '@angular/core';
import { Subject }    from 'rxjs';

@Injectable()
export class MissionService {
```

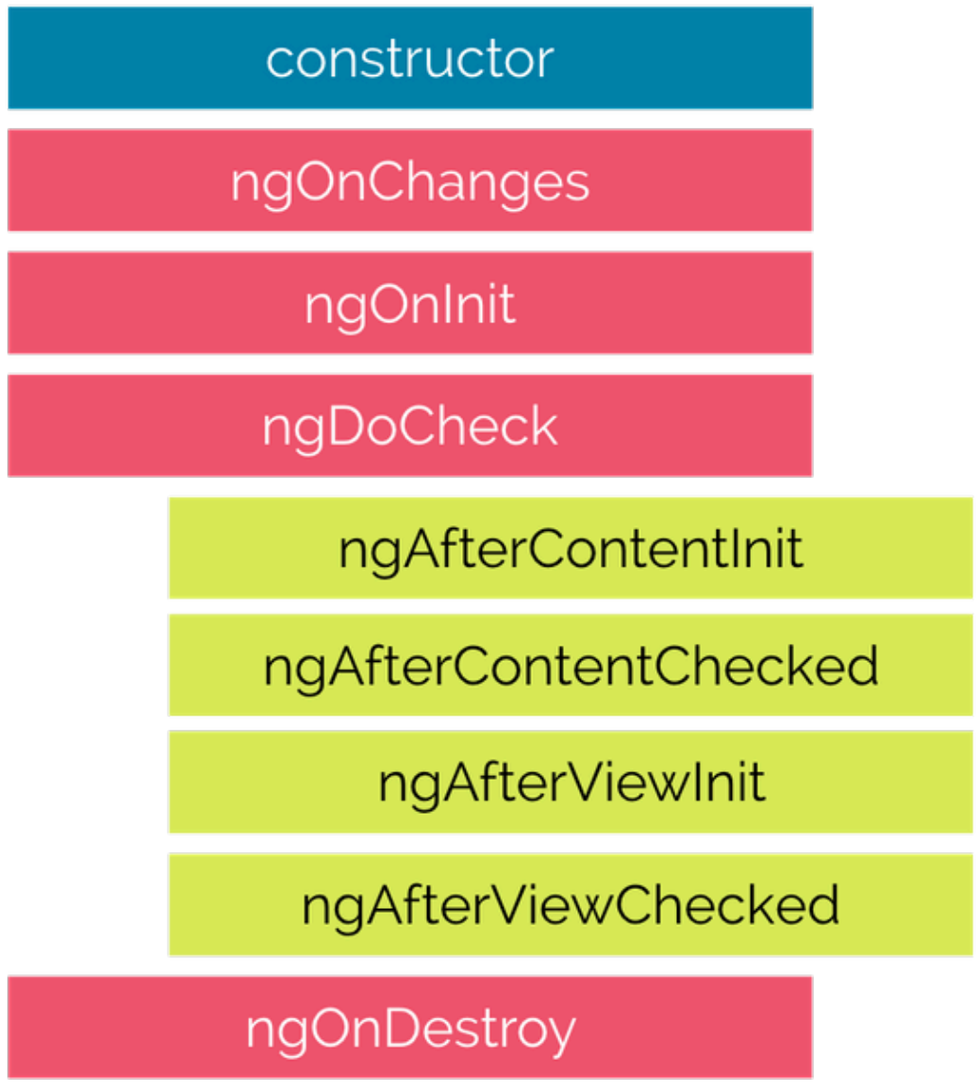
Service

```
constructor(private missionService: MissionService) {
  missionService.missionConfirmed$.subscribe(
    astronaut => {
```

Component

Injects the service in its constructor.

Life cycle hooks



Group1

- ngOnChanges
- ngOnInit
- ngDoCheck
- ngOnDestroy

Group2

- ngAfterContentInit
- ngAfterContentChecked
- ngAfterViewInit
- ngAfterViewChecked

ngDoCheck and **ngOnChanges** should not be implemented together on the same component.

CODEMOBILES COMPANY LIMITED

Hooks for the component

constructor

This is invoked when Angular creates a component or directive by calling new on the class.

ngOnChanges

Invoked every time there is a change in one of the input properties of the component.

ngOnInit

Invoked when given component has been initialized. This hook is only called once after the first ngOnChanges

ngDoCheck

Invoked when the change detector of the given component is invoked. It allows us to implement our own change detection algorithm for the given component.

ngOnDestroy

This method will be invoked just before Angular destroys the component. Use this hook to unsubscribe observables and detach event handlers to avoid memory leaks.

Hooks for the components children

ngAfterContentInit

Invoked after Angular performs any content projection into the components view

ngAfterContentChecked

Invoked each time the content of the given component has been checked by the change detection mechanism of Angular.

ngAfterViewInit

Invoked when the component’s view has been fully initialized.

ngAfterViewChecked

Invoked each time the view of the given component has been checked by the change detection mechanism of Angular.

Workshop

POS workshop

- New project
- Integrate Admin LTE
  - Copy bower_components, dist to assets folder
  - Copy code of index.html
  - Copy images and js to assets folder
- Create Components
  - ng g c {folder}/{component-name}
    - option --spec=false
- Copy and paste html files (header, footer, menu)
- Edit app.component.html

Workshop

Day2

- Edit routes (app-routing.module.ts)
- ng-template (app.component.html)
- Login component
  - Import FormsModule (app.module.ts)
  - Design html (Template-driven forms)

CODEMOBILES
COMPANY LIMITED

- Create services
  - ng g s {folder}/{service-name}
    - option --spec=false
  - Import HttpClientModule (app.module.ts)
  - Copy environment.ts, environment.prod.ts
  - Create models
    - ng g class {folder}/{class-name}
      - option --spec=false
    - http://json2ts.com/ or https://quicktype.io/

- Create services (cont.)
    - Implement rest.service
        - Beware import name
        - DI HttpClient to constructor
        - Implement login service
    - Implement login.component
        - Make sure attribute name of input tag (.html)
- Register component
    - Make sure attribute name of input tag (.html)

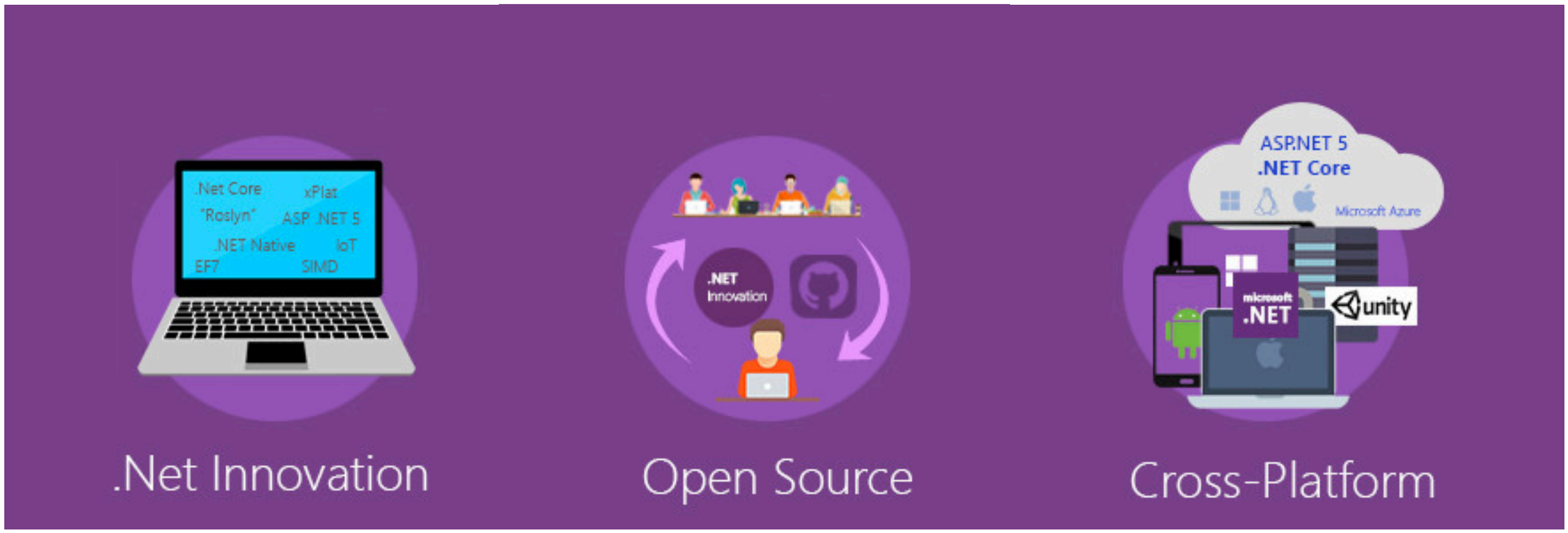
Workshop

Day3

Introduction ASP .NET Core

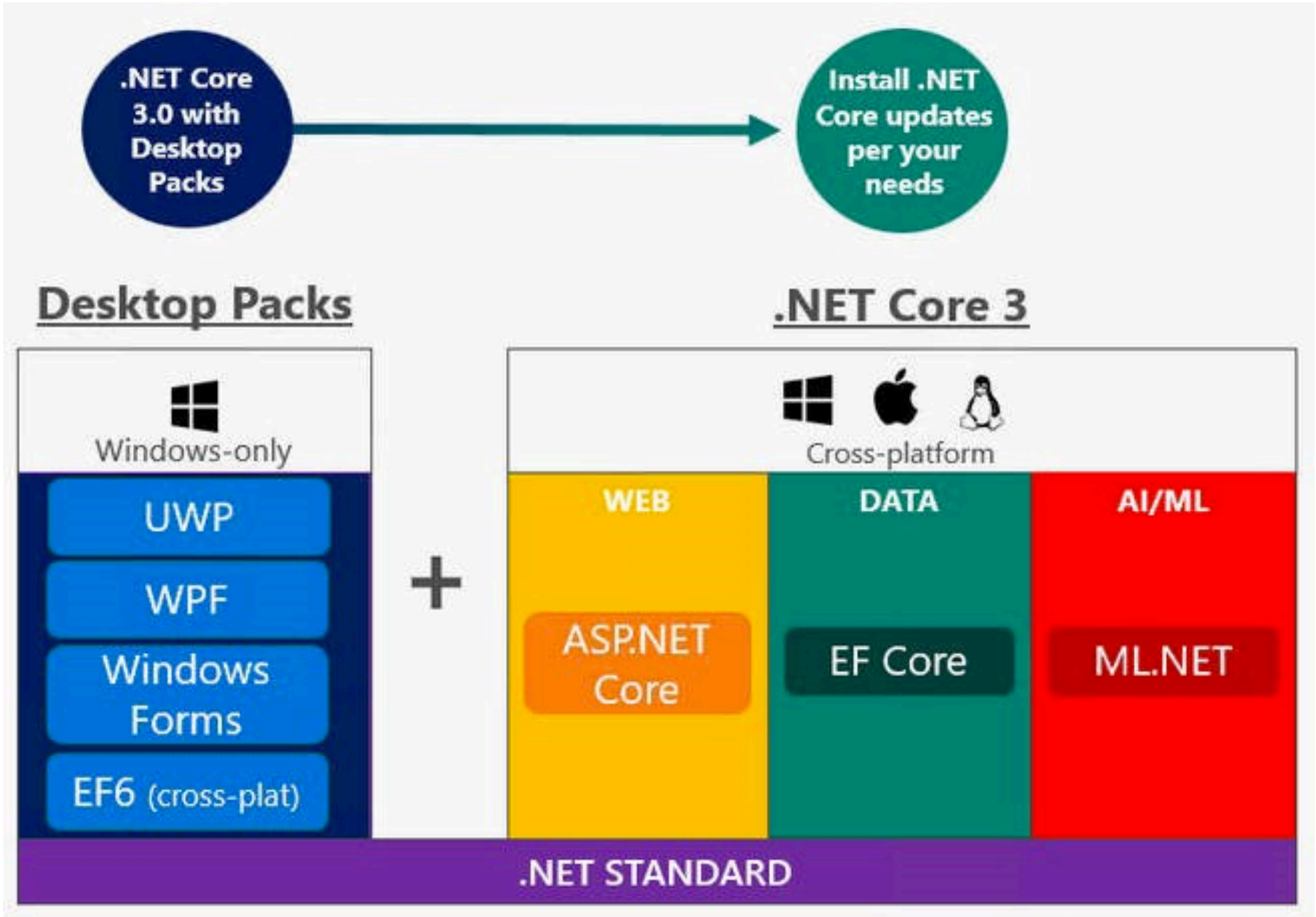
- What & Why ASP .NET Core
- Create project
  - Create folder and cd to folder
  - dotnet new webapi
  - code . (open project on vscode)

CODEMOBILES
COMPANY LIMITED



ASP.NET Core is a cross-platform, high-performance, open-source framework for building modern, cloud-based, Internet-connected applications.

ASP.NET Core was released in 2016.


.NET Core 3.0 with Desktop Packs
Install .NET Core updates per your needs
Desktop Packs
.NET Core 3
Windows-only
UWP
WPF
Windows Forms
EF6 (cross-plat)
+
Cross-platform
WEB
ASP.NET Core
DATA
EF Core
AI/ML
ML.NET
.NET STANDARD

- Project Structure
  - Controllers
  - launchSettings.json
  - *.csproj
  - Program.cs
  - Startup.cs
- Generate self cert
  - dotnet dev-certs https --trust
  - Close and open browser

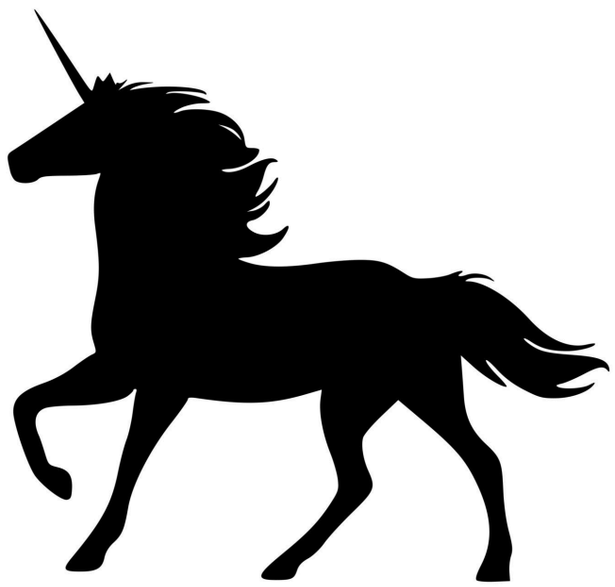
API Workshop

- Implement sql server service (database-first concept)
  - Models (Generate from ef-core command) // step 1
  - DatabaseContext (Generate from ef-core command) // step 2
  - Appsettings.json (ConnectionStrings) // step 3
  - Declare sql server service on ConfigureServices method (Startup.cs) // step 4
- Convert json key-name to snake case (optional)
  - Declare code on ConfigureServices method (Startup.cs)

Workshop

Entity Framework Core tools .NET CLI

dotnet tool install --global dotnet-ef --version 3.0.0



https://i.pinimg.com/originals/9b/52/d6/9b52d6ff047195c1190272ac10faf5d2.jpg

CODEMOBILES COMPANY LIMITED

```
dotnet ef dbcontext scaffold "Server=DESKTOPCDUD9DE\SQLEXPRESS;
    Database=CMPOS; Trusted_Connection=True;"
    Microsoft.EntityFrameworkCore.SqlServer -o Models -t products -t users
    -c DatabaseContext --context-dir Database
```

```
dotnet ef dbcontext scaffold "Server=localhost,1150;user id=sa;
    password=Tel1234!; Database=CMPOS;"
    Microsoft.EntityFrameworkCore.SqlServer -o Models -t products -t users
    -c DatabaseContext --context-dir Database
```

mark: macOs use ' (single quote)
mark: beware double backslash (used single backslash)
mark: ef-core command single line
mark: fix code error (use cli "dotnet build")
mark: stop run project

Workshop

Implement authen with JWT (JSON Web Tokens)

- Appsettings.json (jwt) // step 1
- Declare authen jwt service on ConfigureServices method (Startup.cs) // step 2
- Declare app.UseAuthentication(); on Configure method (Startup.cs) // step 3
  - Declare before app.UseEndpoints();
- [Authorize], [AllowAnonymous] on Controllers or Actions

mark: payload key is case-sensitive

Using Postman https://www.youtube.com/watch?v=wzNigwFmpdQ

- authorizataion tab > type (Bearer Token) > input jwt token

CORS (Cross-Origin Resource Sharing)

- Declare cors service on ConfigureServices method (Startup.cs)
- [EnableCors("...")] on Controllers or Actions or Middleware

Workshop

Day4

- Implement isLogin service
  - Login component
  - App component
- Stock component
  - Create model
  - Feed Data
  - Design html
  - Search
  - Delete product
    - npm i sweetalert2

- Stock component (cont.)
  - Edit product
    - ActivatedRoute
    - Don't forget import { Location }
  - Crate product
    - Don't forget import { Location }

- Http interceptor with JWT
  - ng g class {folder}/{class-name}
    - option --spec=false
    - jwt.interceptor (Easy to understand file names)

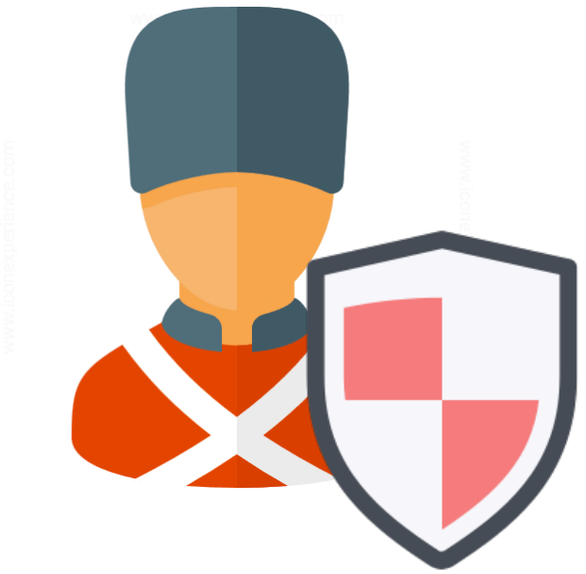
- Http interceptor with JWT (cont.)
  - Copy and paste code
  - Add @Injectable() above class name (jwt.interceptor.ts)
  - npm i rxjs-compat
  - import 'rxjs/add/operator/do';
  - Declare providers (app.module.ts)
    - Add http interceptor before http service
  - Try comment makeJWTManual() of rest.service.ts

Workshop

Day5

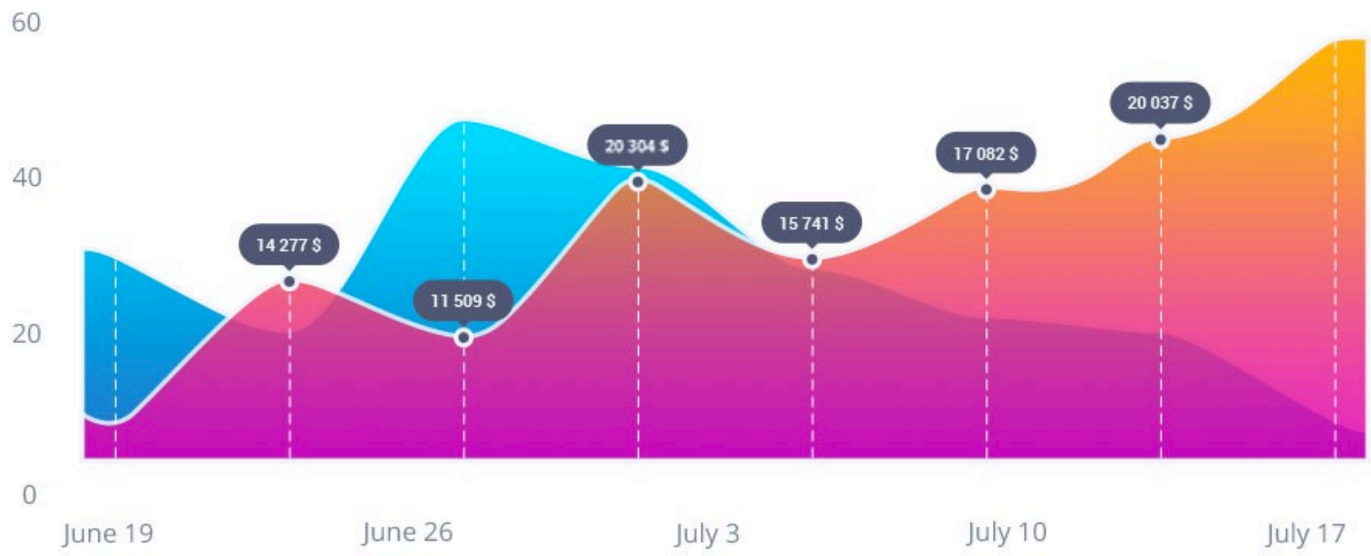
Guard

- ng g g {folder}/{guard-name}
  - option --implements=CanActivate
  - option --spec=false
- CanActivate
- CanDeactivate
- Declare guard (app-routing.module.ts)



- Shop component
  - Implement order and product item
  - Payment component
    - ng g p {folder}/{pipe-name}
    - option --spec=false
- Component Communication
  - @Input() // Parent to Child
  - @Output() // Child to parent
  - Declare payment-component.html in shop-component.html

- Transaction component
  - Home
  - Detail
    - ActivatedRoute
    - Don't forget import { Location }
- Report component (optional)
  - npm i chart.js (https://www.chartjs.org/)



Deploy Angular to IIS Server

- Install the URL Rewrite Module
  - If you are only interested in deploying your application to the web root, you can skip this step. (no sub-folder in wwwroot)
    - https://www.iis.net/downloads/microsoft/url-rewrite
- Create web.config in angular project ({project-name}/src)
  - Add path web.config in angular.json
    - build assets and test assets (src/web.config)

Deploy

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>

<system.webServer>
  <rewrite>
        <rules>
            <rule name="Redirect404" stopProcessing="true">
                <match url=".*" />
                <conditions>
                    <add input="{REQUEST_FILENAME}" matchType="IsFile" negate="true" />
                    <add input="{REQUEST_FILENAME}" matchType="IsDirectory" negate="true" />
                </conditions>
                    <action type="Redirect" url="/index.html" />
            </rule>
        </rules>
    </rewrite>
</system.webServer>

</configuration>
```

web.config

mark: url="/index.html" (root)
mark: url="./index.html" (sub folder)

CODEMOBILES COMPANY LIMITED

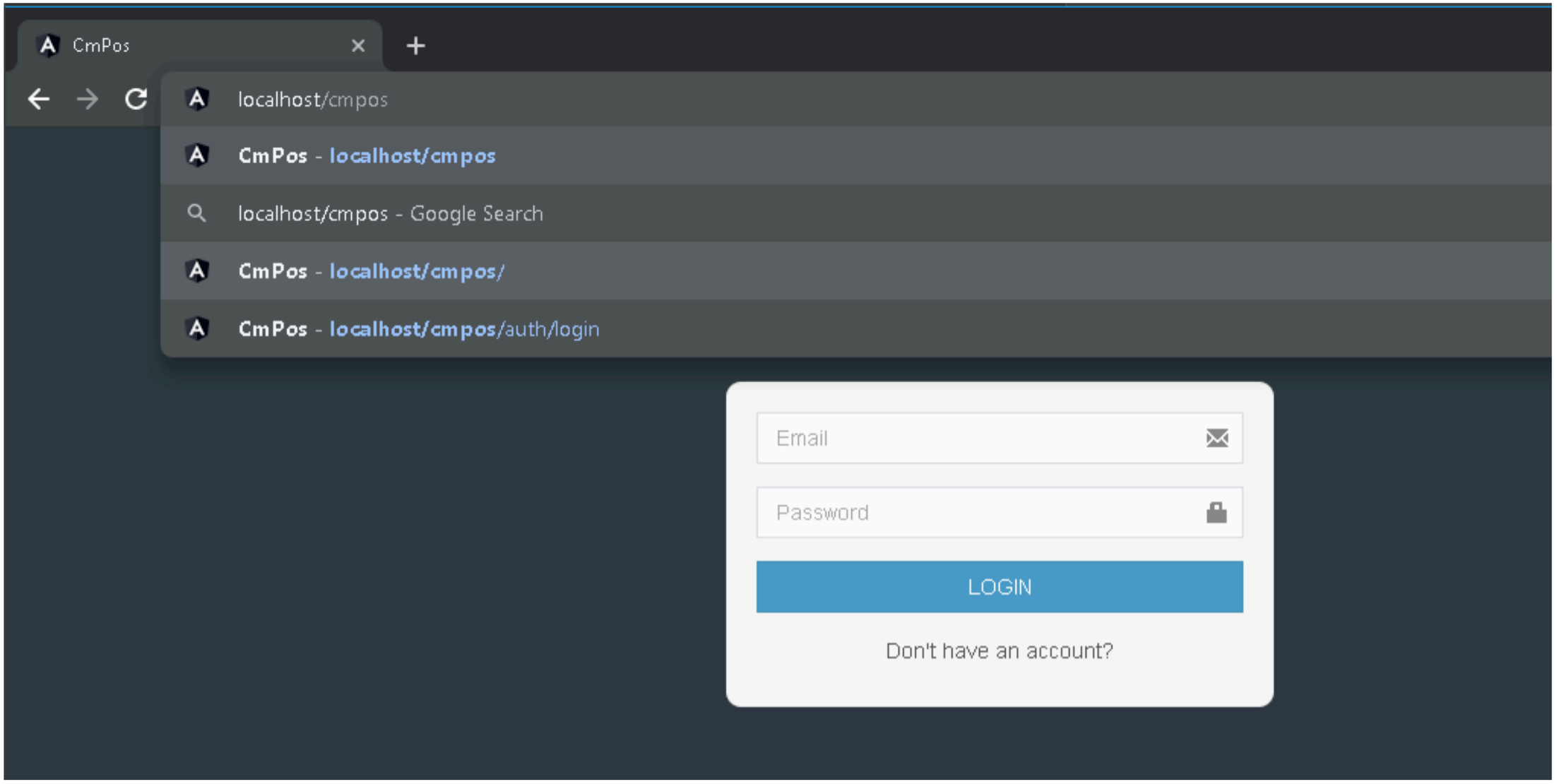
Deploy Angular to IIS Server (cont.)

- ng build --base-href "/{folder-name-in-wwwroot}/" --prod
  - e.g. wwwroot/cmpos on the iis server.

    ng build --base-href "/cmpos/" --prod

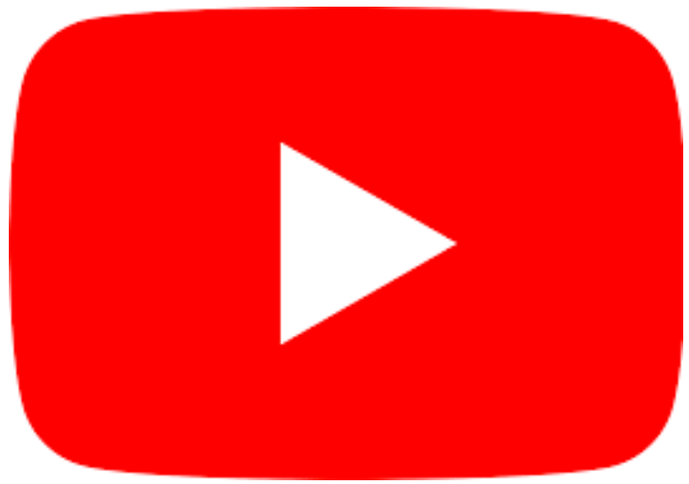
- Copy the contents of dist to our iis server (wwwroot)

Deploy



CODEMOBILES COMPANY LIMITED

Deploy



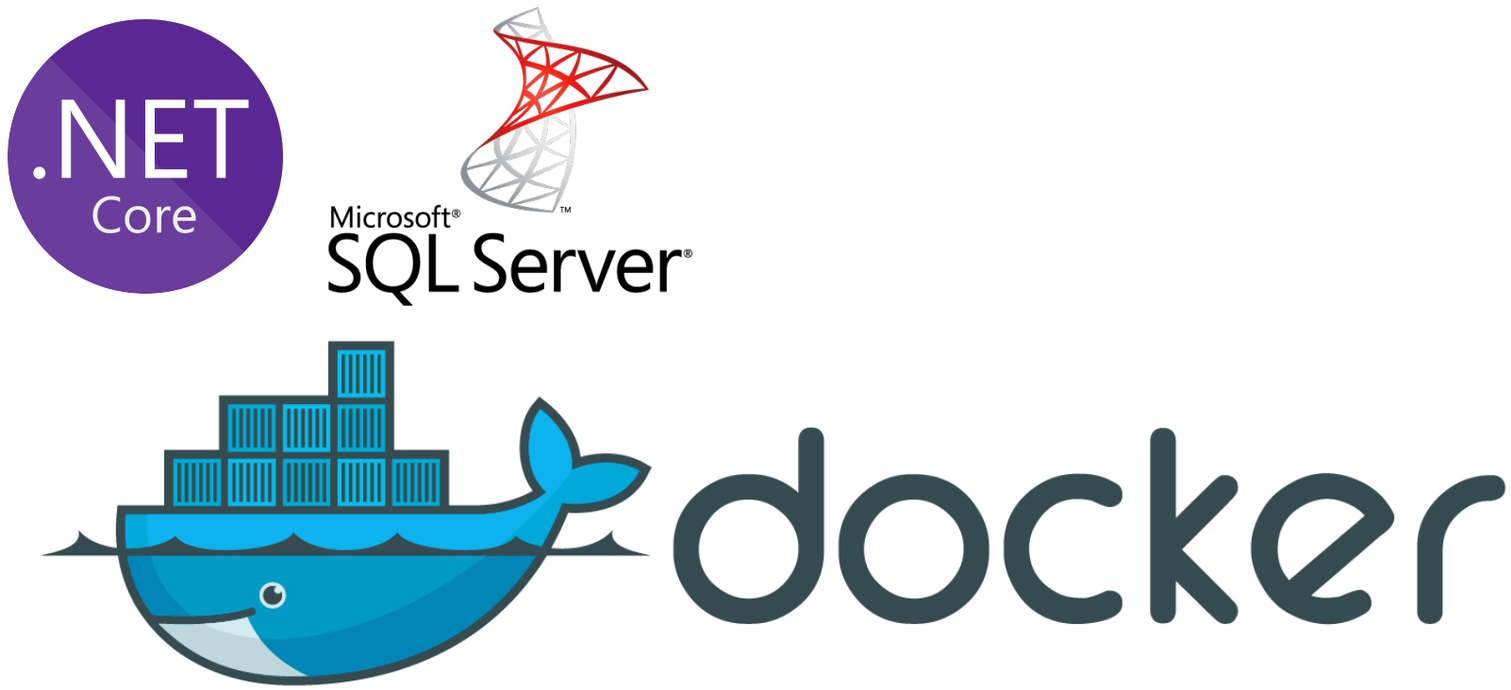
Deploy Angular + ASP .NET Core IIS Server

https://www.youtube.com/watch?v=xF05L_oACyM

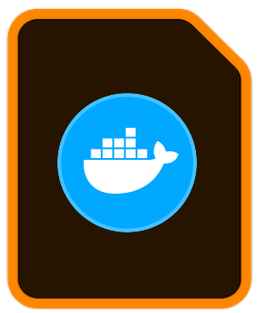
https://www.youtube.com/watch?v=Y9nxGybbbh0

CODEMOBILES COMPANY LIMITED

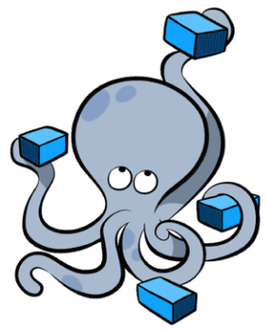
Docker Demo



Web API



Docker file



Docker compose

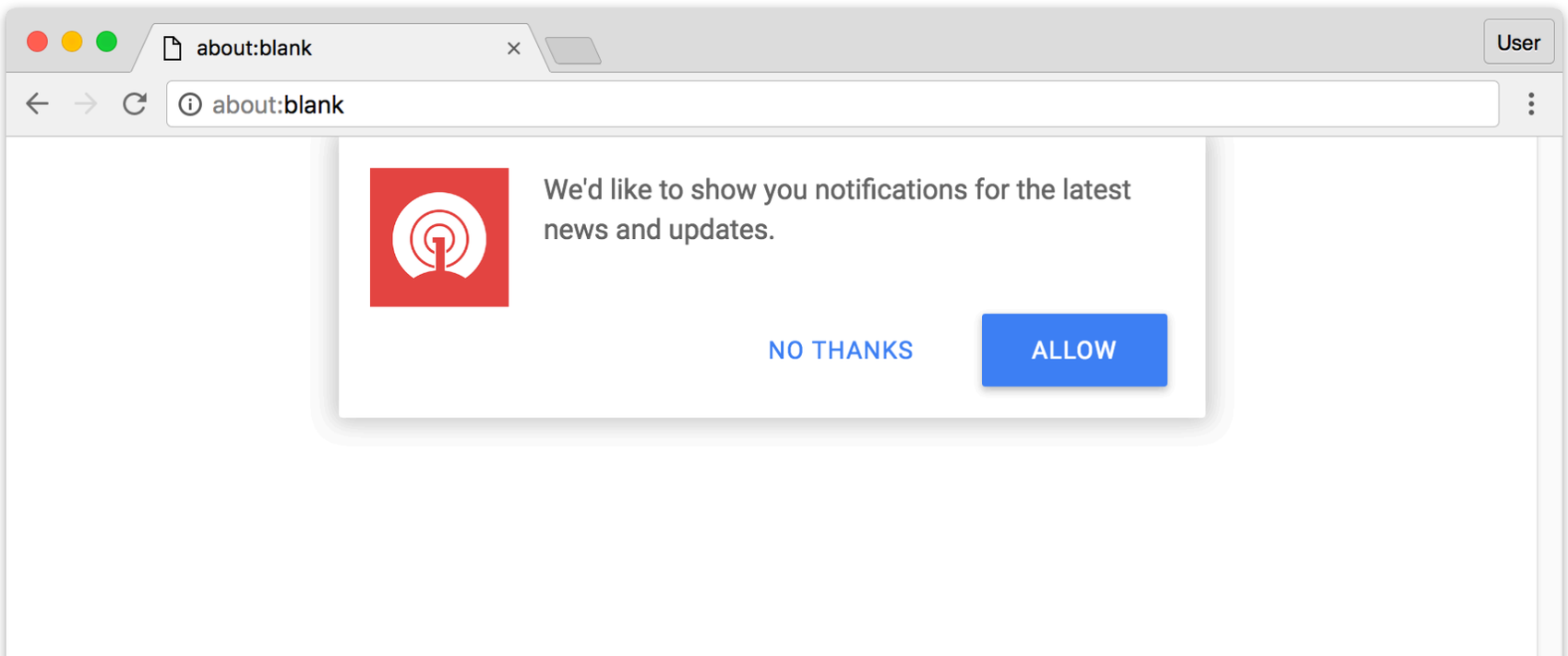
CODEMOBILES
COMPANY LIMITED

Loading spinner

- ngx-spinner (https://github.com/Napster2210/ngx-spinner)

Push notifications

- One signal (https://onesignal.com)



Introduction Docker

- Docker Desktop
- Docker Toolbox (vm ware)
- Enable VT-Technology in BIOS
- Windows 10 open hyper-v

SQL server linux on docker

- https://hub.docker.com/_/microsoft-mssql-server
- docker run -e 'ACCEPT_EULA=Y' -e 'SA_PASSWORD=yourStrong(!)Password' -p 1433:1433 -d mcr.microsoft.com/mssql/server:2017-CU14-ubuntu
  - yourStrong(!)Password // exam: Tel123456789!
  - Beware -e (environment). do not type wrong

mark: windows uses double quotation "..."
mark: mac uses single quotation '...'

Backup Database macOS

docker exec -it **d41392ad4b83** /opt/mssql-tools/bin/sqlcmd -S **localhost** -U SA -P '**Tel1234!**' -Q 'BACKUP DATABASE **cmpos** TO DISK = "/var/opt/mssql/**cmpos**.bak" WITH FORMAT, INIT, COMPRESSION, STATS = 10'

docker cp **d41392ad4b83**:/var/opt/mssql/**cmpos**.bak **cmpos**.bak

Backup Database Windows

docker exec -it **d41392ad4b83** /opt/mssql-tools/bin/sqlcmd -S **localhost** -U SA -P "**Tel1234!**" -Q "BACKUP DATABASE **cmpos** TO DISK = '/var/opt/mssql/**cmpos**.bak' WITH FORMAT, INIT, COMPRESSION, STATS = 10"

docker cp **d41392ad4b83**:/var/opt/mssql/**cmpos**.bak **cmpos**.bak

Docker SQL Sever

Restore Database macOS

docker cp **cmpos**.bak **d41392ad4b83**:/var/opt/mssql/

docker exec -it **4ffba** /opt/mssql-tools/bin/sqlcmd -S **localhost** -U SA -P '**Tel1234!**' -Q 'RESTORE DATABASE **cmpos** FROM DISK = "/var/opt/mssql/**cmpos**.bak"'

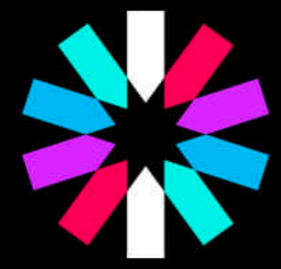
Restore Database Windows

docker cp **cmpos**.bak **d41392ad4b83**:/var/opt/mssql/

docker exec -it **d41392ad4b83** /opt/mssql-tools/bin/sqlcmd -S **localhost** -U SA -P "**Tel1234!**" -Q "RESTORE DATABASE **cmpos** FROM DISK = '/var/opt/mssql/**cmpos**.bak'"

CMO | CODEMOBILES COMPANY LIMITED

Revoking Tokens



JWT

eyJhbGciO

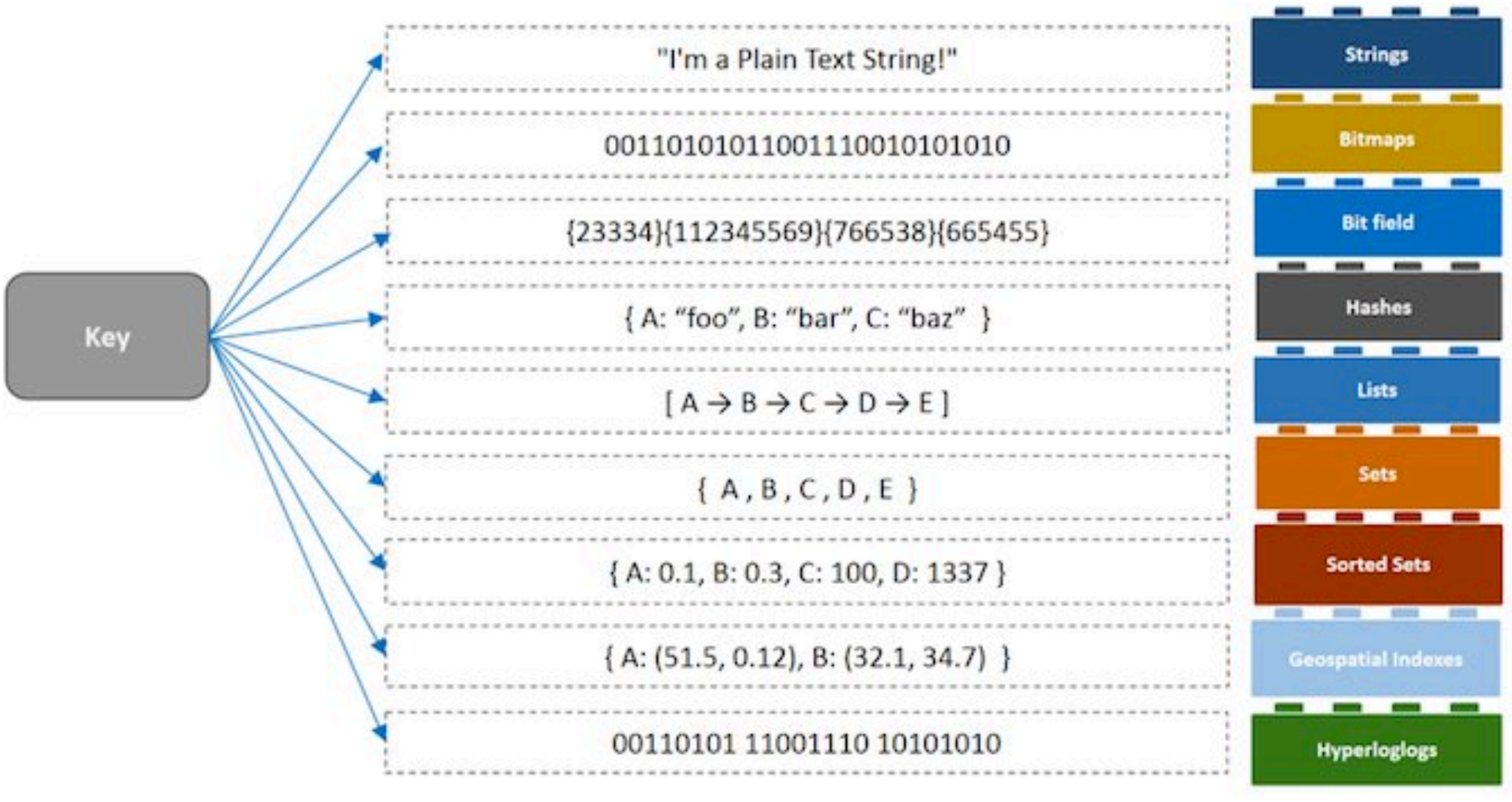
redis

eyJhbGciOiJIUzI1NiIsIn…

SQL Server

Revoking Tokens

Redis data structures



CODEMOBILES
COMPANY LIMITED

Observable

Asynchronous

SUBSCRIBE

Observer

https://www.youtube.com/watch?v=XbOuCBuQepI