

angular: estructura proyecto

- **e2e:**
 - pruebas extremo a extremo, simulación interacción con humano
 - angular-cli configura este directorio de forma automática
- **node_modules:**
 - multitud paquetes de ayuda al desarrollo
 - cuando se genera la aplicación, solo se coge lo estrictamente necesario
 - esta carpeta ha de quedar fuera del repositorio o copia de seguridad
 - `npm install` regenera/descarga los ficheros según la información de **package.json**
- **angular.json**
 - como es y como funciona nuestra app
 - partes importantes: assets, styles, scripts
- **package-lock.json**
 - generación automática
 - le indica a la parte de Node.js como se creó package.json
 - deja rastro
- **package.json**
 - muy importante
 - generado y mantenido de manera automática
 - información, dependencias de nuestro proyecto
- **tsconfig.json**
 - indica a TypeScript como trabajar
- **tslint.json**
 - define reglas de sintaxis para escribir código más limpio
 - genera error en el IDE si no las cumplimos
- **src/**
 - código fuente de la app
 - `src/index.html`
 - fichero de inicio, html 5, con tags específicos (placeholder) para colocar nuestra app
 - `<app-root></app-root>` por defecto
 - `src/app/`
 - app angular
 - `src/app/app.component.spec.ts:`
 - pruebas unitarias (se usa **spec** para identificarlos)
 - `src/app/app.component.ts`
 - contiene el código que rellenará la etiqueta `<app-root/>` indicado en la directiva **selector**

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})

export class AppComponent {
  nombre = 'Peter';
  apellido = 'Parker';
}
```

- `src/app/app.component.html`
 - contiene el HTML que se usa desde el .ts que será inyectado en `<app-root/>` según

la directiva **templateURL**

```
• <h1>Hola mundo</h1>
  <ul>
    <li>Nombre: {{nombre}}</li>
    <li>Apellido: {{apellido}}</li>
  </ul>
```

- src/app/app.component.css: contiene el código CSS para **app.component.html**
- src/app/app.module.ts:
 - clase que define un decorador llamado **ngModule** (se verá más adelante)
 - lo manipularemos a conveniencia
- src/assets/
 - recursos estáticos: imágenes, audios, vídeos...
 - src/assets/.gitkeep:
 - fichero para que git siempre suba el directorio **assets**
 - nombre por convención
- src/environments:
 - archivos con variables de entorno según estemos en producción o desarrollo
- src/browserlist
 - desde Angular 6
 - casos de incompatibilidad con navegadores (normalmente versiones antiguas)
- src/karma.conf.js
 - archivo configuración test unitarios
- src/main.ts
 - primer fichero analizado por Angular
 - define, entre otras cosas, el tipo de aplicación (web, móvil, ...)
- src/polyfills.ts
 - ayuda a compatibilizar versiones viejas de navegadores
- src/styles.css
 - estilos globales de la app
- src/test.ts
 - relativo a los test unitarios con **karma**
 - requerido por **karma.conf.js** para cargar recursivamente los ficheros **.spec**
- src/tsconfig.app.json:
 - configuraciones de TypeScript relativas a la app
 - amplía **tsconfig.json**
- src/tsconfig.spec.json
 - configuraciones relativas a las pruebas unitarias
 - amplía el fichero **tsconfig.json**
- src/tslint.json
 - especifica como se van a mostrar los errores de JS/TS en el IDE
 - amplía el fichero **tslint.json**

From:
<https://matebcn.eu/> - miguel angel torres egea

Permanent link:
<https://matebcn.eu/development:angular:estructura>

Last update: **26/01/2020 10:28**

