

Podrá descargar algunos elementos de este libro en la página web de Ediciones ENI: <http://www.ediciones-eni.com>.

Escriba la referencia ENI del libro **EIT2ANGNOD** en la zona de búsqueda y valide. Haga clic en el título y después en el vínculo de descarga.

Prólogo

Capítulo 1 Introducción

1. Introducción	13
2. La arquitectura MEAN para una aplicación web	15
2.1 El principio de las aplicaciones mono página (single page applications)	16
2.2 El paradigma de diseño modelo vista-controlador	17
3. Angular en el centro de la arquitectura MEAN	19
4. Presentación de El hilo rojo: una aplicación de e-commerce	21

Capítulo 2 El lenguaje JavaScript

1. Introducción a JavaScript.....	25
1.1 Breve repaso histórico.....	25
1.2 Panorama de la utilización de JavaScript	26
1.3 Las librerías y los frameworks aplicativos JavaScript.....	27
2. ¿Dónde escribir código JavaScript?	28
3. Las herramientas del navegador y la depuración	29
4. Los elementos de programación básicos	30
4.1 Las variables	30
4.1.1 Los tipos internos	30
4.1.2 El transtipado.....	31
4.2 Las estructuras de datos comunes	32

4.3	Aplicación de expresiones regulares	33
4.4	Los bloques de instrucciones	34
4.5	Las estructuras condicionales.	35
4.5.1	La estructura if ... else	35
4.5.2	La estructura switch de conmutador múltiple	35
4.6	Las estructuras iterativas	36
4.6.1	Las estructuras iterativas con índices de bucle	36
4.6.2	Las estructuras iterativas sin índices de bucle	37
5.	La programación funcional en JavaScript	38
5.1	Una función que se pasa como argumento (función de callback)	39
5.1.1	Ejemplo con el método forEach()	39
5.1.2	Ejemplo con el método map().	40
5.2	Una función devuelve una función (factory)	41
6.	La programación orientada a objetos con JavaScript	42
6.1	Los principios de la programación orientada a objetos con JavaScript	42
6.2	Los objetos literales	43
6.3	La herencia por encadenamiento de prototipos	44
6.3.1	La propiedad __proto__ del objeto del que heredan	44
6.3.2	La propiedad prototype	45
6.4	La creación de un objeto por medio de la llamada de una función constructora	45
6.5	Ejemplos de implementaciones de un método	47
6.6	La problemática del objeto actual (this)	47
6.7	La herencia	50
6.8	El encadenamiento de métodos	51
7.	Las principales aportaciones de la norme ECMAScript 6	52
7.1	La norma ECMAScript	52
7.2	La palabra reservada let	52
7.3	La interpolación de variables en las cadenas	53
7.4	Los argumentos por defecto	53

7.5 Una manipulación más cómoda de las listas	53
7.5.1 La estructura for (... of ...)	53
7.5.2 El método includes()	54
7.6 El operador "fat arrow" (=>)	54
7.7 Las clases	55
8. La programación reactiva, observables y promises (promesas)	56
8.1 Primer ejemplo: un observable en un botón	58
8.2 Segundo ejemplo: un observable sobre un entero	58
8.3 Tercer ejemplo: un observable sobre un timer	59
8.4 Las promises	60

Capítulo 3

Extensiones JavaScript para las clases

1. Presentación de las extensiones de JavaScript	63
2. El lenguaje TypeScript	64
2.1 El transpiler tsc	64
2.2 Tipado estático de las variables	65
2.2.1 Los tipos básicos	65
2.2.2 Tipado de variables no escalares	65
2.2.3 El tipo enum	66
2.2.4 El tipo genérico any	66
2.2.5 Tipado de las funciones	66
2.3 Las clases	67
2.3.1 La herencia	68
2.3.2 Las interfaces	69
2.3.3 La programación genérica	69
3. El lenguaje Dart	70
3.1 Instalación y prueba de Dart	70
3.2 Generación del código JavaScript con Dart	71
3.3 Las clases	71
3.3.1 La herencia	72
3.3.2 Las interfaces	73

Capítulo 4

La plataforma Node.js

1.	Presentación de Node.js	75
2.	Instalación y prueba de Node.js	76
2.1	Creación del archivo de prueba	76
2.2	Instalación y prueba de Node.js en Ubuntu	77
2.3	Instalación y prueba de Node.js en Windows	77
2.4	Instalación y prueba de Node.js en Mac OS	78
3.	La modularidad de Node.js	78
3.1	Los módulos y los paquetes	78
3.1.1	El administrador de módulos de Node.js: npm	79
3.1.2	Especificación de las dependencias: el archivo package.json	79
3.2	Creación de un primer servidor Node.js de prueba	80
3.3	Creación y reutilización de un módulo	82
3.4	Creación de un servidor que devuelva los datos	84
3.5	El módulo express	85
3.5.1	Gestión de rutas REST con el módulo express	85
3.5.2	Gestión de las plantillas con el módulo express	88
3.5.3	Especificación de las dependencias en un archivo package.json	90
3.5.4	Instalación del módulo express	91
3.6	El módulo fs (FileSystem)	91
3.7	Prueba de un servidor Node.js	93
3.7.1	Creación de un archivo de datos JSON	93
3.7.2	La problemática del control de acceso HTTP	94
3.7.3	Retorno al cliente de un archivo JSON	94
3.7.4	Configuración de las rutas	96
3.7.5	Gestión de los argumentos	100
4.	Securización de un servidor Node.js (protocolo HTTPS)	102
4.1	Creación de la autoridad de certificación	102
4.2	Creación del certificado	103
4.3	Creación del servidor	103

5. Conocimientos adquiridos en este capítulo	104
--	-----

Capítulo 5

El SGBD NoSQL MongoDB

1. Introducción	107
2. ¿Por qué utilizar una base de datos NoSQL?	108
3. Presentación de MongoDB.....	109
3.1 Las colecciones y los documentos	109
3.2 Los índices.....	109
4. Implementación de MongoDB	110
4.1 Instalación de MongoDB	110
4.1.1 Instalación de MongoDB en Linux.....	110
4.1.2 Instalación de MongoDB en Windows o en Mac OS ..	110
4.1.3 Utilización de MongoDB en línea de comandos	111
4.2 Visualización de la lista de las bases de datos.....	111
4.3 Creación de una base de datos.....	112
4.4 Visualización de la lista de las colecciones	112
4.5 Creación de una colección	112
4.5.1 Inserción de los documentos en una colección	112
4.5.2 Importación de documentos a partir de un archivo ..	113
4.5.3 Exportación de los documentos de una colección en un archivo JSON.....	114
4.6 Interrogación de una colección	114
4.6.1 Interrogación a través de un objeto "filtro"	114
4.6.2 Los operadores de comparación, los operadores de conjuntos y lógicos.....	116
4.6.3 El operador \$exists.....	116
4.6.4 El operador \$in.....	117
4.6.5 El operador \$nin.....	117
4.6.6 El operador \$or.....	117
4.6.7 El operador \$not	118
4.6.8 El operador \$nor.....	118

4.7	Aplicación de expresiones regulares	118
4.8	Las proyecciones y el método distinct()	119
4.8.1	Las proyecciones	119
4.8.2	El método distinct()	119
4.9	Referenciado los documentos y joins	120
4.9.1	Los objetos anidados (nested objects)	121
4.9.2	Los objetos referenciados	122
4.9.3	Los joins	123
4.10	Actualización y eliminación de un documento	127
4.10.1	Actualización de un documento	127
4.10.2	Eliminación de un documento	128
4.11	Eliminación de una colección.	128
5.	Utilización de MongoDB a través de Node.js	128
5.1	Instalación del módulo MongoDB para Node.js	128
5.2	Conexión al servidor MongoDB	130
5.3	Inserción de datos desde un servidor Node.js	131
5.4	Consultar datos desde un servidor Node.js	131
5.4.1	Explotación del resultado del método find()	132
5.4.2	Utilización del método toArray()	133
5.5	Sincronización de las consultas	135
5.5.1	Utilización de las funciones de callback	136
5.5.2	Utilización del módulo async	138
5.5.3	El método async.series()	139
5.5.4	El método async.waterfall()	140
6.	Interrogación de MongoDB con las rutas gestionadas por express .	142
6.1	La estructura de un servidor Node.js que consulta a MongoDB	142
6.2	La problemática del cross-origin resource sharing	143
6.3	Ejemplos de gestión de rutas	143
6.3.1	Gestión de una ruta para listar las marcas	144
6.3.2	Gestión de una ruta para filtrar los productos	145
6.3.3	Búsqueda de un producto a partir de su identificador interno	147

7.	El hilo rojo del lado servidor.....	148
7.1	Creación de la colección.....	148
7.2	Implementación de dos búsquedas sobre los productos	150
7.2.1	La superestructura del servidor.....	150
7.2.2	Gestión de la ruta que filtra los documentos usando diferentes criterios	152
7.2.3	Gestión de la ruta que devuelve un documento a través de su identificador interno	153
7.2.4	Ejemplos de consulta en el servidor	153
8.	Conocimientos adquiridos en este capítulo	154

Capítulo 6

Introducción al framework aplicativo Angular

1.	Presentación de Angular.....	155
1.1	Una evolución radical de AngularJS	155
1.2	La modularidad de la aplicación: los módulos y los componentes	157
1.2.1	Los módulos.....	157
1.2.2	Los componentes y los servicios	158
1.3	Manipulación de los componentes como etiquetas	159
1.4	Utilización de una extensión en JavaScript (TypeScript o Dart)	160
2.	Angular respecto al framework MVC (incluso VVM)	160
3.	Implementación de una aplicación Angular	163
3.1	Presentación de Angular CLI	164
3.2	Instalación de Angular CLI	164
3.3	Creación de un proyecto Angular con Angular CLI	165
3.4	Estructura de las carpetas de un proyecto Angular CLI	167
3.5	Un primer componente creado por Angular CLI	167
3.6	El root module creado por Angular CLI	170
3.7	Actualización de Angular a través de Angular CLI	171
4.	Generación del código JavaScript a partir de TypeScript.....	172

5.	Los decoradores.....	174
6.	Creación de un nuevo componente que muestra un mensaje.....	175
6.1	Creación del componente	175
6.1.1	La plantilla HTML.....	177
6.1.2	La hoja de estilos	177
6.2	Interfaz de componentes en el componente raíz	177
6.3	Especificación de los componentes en el módulo.....	178
6.4	Activación del módulo	179
6.5	La página web front	179
7.	El ciclo de vida de un componente	180
7.1	El hook ngOnChanges().....	181
7.2	El hook ngOnInit().....	182
7.3	El hook ngDoCheck().....	183
7.4	El hook ngOnDestroy()	183
8.	Conocimientos adquiridos en este capítulo	184

Capítulo 7

Angular: las plantillas, bindings y directivas

1.	Las plantillas	187
1.1	Anidación de las plantillas	189
1.2	Las plantillas insertadas (embedded templates)	195
1.3	Las plantillas externalizadas	196
2.	Data bindings entre el componente y la plantilla	196
2.1	Acceso a los elementos del DOM	197
2.2	Interpolación de una variable en una plantilla.....	197
2.3	Property binding.....	199
2.4	Event binding	200
2.5	Two-way data binding.....	203
3.	Las directivas.....	205
3.1	Las directivas estructurales	206
3.1.1	La directiva *ngFor	206
3.1.2	La directiva *ngIf.....	209

3.2 Las directivas de atributos	210
3.3 Envío de información de un componente a su padre (@Output())	215
4. Los pipes	218
5. Ejemplo de resumen: un formulario de autenticación	222
6. El hilo rojo: creación de un componente que muestra los productos	225
6.1 El componente	226
6.1.1 La clase que implementa el componente	227
6.1.2 La plantilla HTML	228
6.1.3 La hoja de estilos	230
6.2 El módulo que especifica el componente	230
6.3 Activación del módulo	231
6.4 La página web front	231
6.5 Lanzamiento de la aplicación	232
7. Conocimientos adquiridos en este capítulo	232

Capítulo 8

Angular y la conexión a Node.js: los servicios

1. Introducción	235
2. Inyección de dependencias	236
3. Utilización de los servicios para la transferencia de datos	237
3.1 Recuperación de datos formateados en JSON	237
3.2 Envío de datos JSON al servidor	238
3.3 Envío de datos a través de Query string	240
4. Implementación de los servicios en El hilo rojo	241
4.1 Declaración de las rutas del lado servidor	241
4.2 Gestión de los productos	244
4.2.1 Visualización de los selectores	244
4.2.2 Visualización de los productos según los criterios de búsqueda	247

4.2.3	Visualización de los productos asociados a las palabras clave.	250
4.2.4	Acceso a un producto por su identificador	253
4.3	Gestión de la cesta de la compra	254
4.3.1	Visualización de los identificadores de los productos de la cesta de la compra	254
4.3.2	Visualización de todos los productos de la cesta de la compra.	255
4.3.3	Adición de un producto en la cesta de la compra	257
4.3.4	Eliminación de un producto de la cesta de la compra	260
4.3.5	Reinicialización de la cesta de la compra	262
5.	La librería NgRx y los stores	263
6.	Conocimientos adquiridos en este capítulo	264

Capítulo 9

Angular y la gestión de las rutas internas

1.	Principio general del enrutado	267
1.1	¿Por qué establecer un enrutado?	267
1.2	Las rutas, el router, las tablas de enrutado	269
1.3	Las vistas activadas por las rutas.	271
1.4	Ejemplo de enrutado.	273
1.5	Definición de un árbol de vistas	276
1.6	Utilización de los outlets con nombre	281
2.	La sintaxis de las rutas	286
2.1	Las dos sintaxis de una ruta: cadena o link parameters array	287
2.2	Las rutas absolutas y las rutas relativas	288
2.3	Configuración de las rutas	289
2.4	Asociación de una ruta a un auxiliary outlet	289
3.	Selección de las rutas	290
3.1	La directiva routerLink.	291
3.2	El método navigate()	292
3.3	Ejemplo de ruta.	293

4.	Gestión de las rutas del controlador hacia el componente destino	294
4.1	Configuración de la tabla de enrutado	294
4.2	Las propiedades de una ruta.	295
4.3	Soporte de una ruta por varios módulos/tablas de enrutado.	297
4.4	Control de las rutas: guardián	299
4.5	Invocación de un componente.	308
4.6	Captura de una ruta durante la invocación de un componente	310
5.	Gestión de rutas en El hilo rojo	312
5.1	El módulo de enrutado asociado al root module	314
5.2	El módulo de enrutado asociado al feature module research	315
5.3	El módulo de enrutado asociado al feature module cart	316
6.	Conocimientos adquiridos en este capítulo	317

Capítulo 10

Angular y la visualización de información

1.	Introducción	319
2.	Creación de charts con D3.js y dc.js	320
2.1	Instalación de D3.js	321
2.2	El lenguaje SVG	321
2.3	Generación de elementos gráficos asociados a los objetos de una colección	324
2.4	Selección y modificación de elementos del DOM	325
2.4.1	Adición de elementos gráficos	326
2.4.2	Sustitución de elementos gráficos	327
2.5	Implementación de listeners de eventos	328
2.6	Integración de D3.js en Angular	329
2.6.1	Un servidor virtual de datos comerciales	329
2.6.2	El servicio que accede a los datos del servidor	331
2.6.3	La plantilla del componente que muestra un histograma	331

2.6.4	Implementación del componente que muestra un histograma	332
2.7	Las librerías dc.js y Crossfilter	336
2.7.1	Instalación de dc.js y de Crossfilter	337
2.7.2	Implementación del componente que muestra el histograma	338
3.	Integración de mapas Google Map en un proyecto Angular	340
3.1	Instalación de los requisitos previos técnicos	341
3.1.1	Instalación de los tipos TypeScript	341
3.1.2	Instalación de la librería PrimeNG	342
3.2	Presentación de un mapa Google Map estático	342
3.3	Un componente PrimeNG para gestionar un mapa Google Map	344
3.4	Gestión de un chart asociado a un Google Map	348
4.	Implementación de componentes gráficos	353
4.1	Otro ejemplo de componente PrimeNG (Calendar)	353
4.2	La librería de componentes Material	355
4.3	La librería ngx-bootstrap	357
5.	Conocimientos adquiridos en este capítulo	362

Capítulo 11

Prueba y despliegue

1.	Test	365
2.	Despliegue	369
2.1	Despliegue con Apache	369
2.2	Despliegue con Node.js	370
2.3	Despliegue con http-server (acceso directo a Node.js)	371
3.	Para ir más lejos	372