

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 **RIT3JASAP** en la zona de búsqueda y valide. Haga clic en el título y después en el botón de descarga.

Parte 1

Presentación de JavaScript y programación a partir de algoritmos

Capítulo 1

Presentación del lenguaje JavaScript

- 1. Definición y breve historia 19
- 2. Requisitos previos para un aprendizaje sencillo del lenguaje. 21
- 3. Herramientas necesarias. 22
- 4. Posicionamiento de JavaScript respecto a otras tecnologías de desarrollo web (HTML, CSS, PHP...) 23

Capítulo 2

Desarrollo a partir de algoritmos

- 1. Presentación de la noción de algoritmo. 25
- 2. Noción de variable 26
 - 2.1 Presentación de las nociones de variable y tipo 26
 - 2.2 Tipos básicos y operaciones asociadas. 28
 - 2.3 Interés de los tipos 29
 - 2.4 Uso de variables en las expresiones 31
 - 2.5 Tabla resumen de los operadores. 31
- 3. Manipulación de las variables 32
 - 3.1 Nomenclatura de las variables 32
 - 3.2 Asignación. 34
 - 3.3 Ejercicio n.º1: Inversión del contenido de dos variables en memoria 35

2 --- Aprender a desarrollar

con JavaScript

3.4	Visualización de los resultados	36
3.5	Ejercicio n.º2: Superficies de círculos.	39
3.6	Introducción por el teclado	39
3.7	Ejercicio n.º3: Superficie y volumen de una esfera	41
4.	Funciones predefinidas	41
4.1	Ejercicio n.º4: Visualización de la longitud de un nombre.	42
4.2	Ejercicio n.º5: Determinación de iniciales.	43
5.	Procesamientos condicionados	46
5.1	Ejercicio n.º6: Polinomio de segundo grado	48
5.2	Ejercicio n.º7: Etiqueta del mes sin cifrar	50
5.3	Ejercicio n.º8: Etiqueta del mes sin cifrar (Siguiendo ... Finsiguiendo)	53
6.	Estructuras iterativas	54
6.1	Concepto de iteraciones.	54
6.2	Estructuras iterativas básicas.	54
6.3	Ejercicio n.º9: Media de 10 números	57
6.4	Ejercicio n.º10: Media de una serie de n números.	58
6.5	Ejercicio n.º11: El común divisor más grande, por el método de las divisiones sucesivas	61
6.6	Estructura iterativa Para.	63
6.7	Ejercicio n.º12: Cálculo de la media de 10 números	65
6.8	Ejercicio n.º13: Deducir el número de vocales de una palabra	66
7.	Tabla de dimensión única	68
7.1	Ejercicio n.º14: Contar los números pares en una tabla.	69
8.	Tabla con dimensiones múltiples	71
8.1	Ejercicio n.º15: Minihoja de cálculo.	71
9.	Procedimientos, funciones y paso de argumentos	73
9.1	Los objetivos	73
9.2	Los procedimientos.	74
9.3	Ejercicio n.º16: Llamada de un procedimiento con paso de argumentos.	75
9.4	Las funciones.	77

9.5 Ejercicio n.º17: Llamada de una función con paso de argumentos 78

Parte 2
Comenzar bien con JavaScript

Capítulo 3
Conceptos principales del lenguaje JavaScript

1. Metodología de aprendizaje 81

2. Variables (declaración y tipado) 82

2.1 Ejercicio n.º2: Superficies de círculos 82

2.2 Ejercicio n.º3: Superficie y volumen de una esfera 89

2.3 Ejercicio n.º4: Número de letras de una palabra 92

2.4 Ejercicio n.º5: Determinación de las iniciales 92

Capítulo 4
Procesamiento condicionado

1. Presentación de la sintaxis 95

2. Ejemplos 97

2.1 Ejercicio n.º6: Polinomio de segundo grado 97

2.2 Ejercicio n.º8: Impresión de la etiqueta de un mes 98

Capítulo 5
Procesamiento iterativo (bucles)

1. Presentación de la sintaxis de los bucles 101

2. Bucle while 102

2.1 Sintaxis 102

2.2 Ejercicio n.º9: Media de 10 números que se introducirán con el teclado 103

2.3 Ejercicio n.º10: Media de una serie de n números que se introducirán con el teclado 104

4 --- Aprender a desarrollar

con JavaScript

3.	Bucle do while	106
3.1	Sintaxis	106
3.2	Ejercicio n.º11: Media de una serie de n números que se introducirán con el teclado	106
4.	Bucle for	107
4.1	Sintaxis	107
4.2	Ejercicio n.º12: Media de una serie de 10 números que se introducirán con el teclado	108
4.3	Ejercicio n.º13: Contar el número de vocales en una palabra . .	109

Capítulo 6

Tablas

1.	Tablas de dimensión única	111
1.1	Sintaxis	111
1.2	Ejercicio n.º14: Contar los números pares en una tabla	113
2.	Tablas de dimensiones múltiples	114
2.1	Sintaxis	114
2.2	Ejercicio n.º15: Minihoja de cálculo	115

Capítulo 7

Procedimientos y funciones

1.	Los procedimientos	119
1.1	Sintaxis	120
1.2	Ejercicio n.º16: Llamada de un procedimiento con argumentos	120
2.	Las funciones	122
2.1	Sintaxis	122
2.2	Ejercicio n.º17: Llamada de una función con argumentos . . .	123

Parte 3
Aproximación a la POO en JavaScript

Capítulo 8
Enfoque orientado a "objetos" en JavaScript

- 1. Introducción 125
- 2. Programación orientada a objetos a través de ejemplos..... 126
 - 2.1 Secuencia 1: Declaración de los objetos JavaScript de manera "Inline". 126
 - 2.2 Secuencia 2: Creación de objetos JavaScript con un constructor 127
 - 2.3 Secuencia 3: Variables privadas en una instancia de objeto. . 128
 - 2.4 Secuencia 4: Paso de argumento(s) a un constructor 129
 - 2.5 Secuencia 5: No compartición de los métodos por las instancias de objetos. 130
 - 2.6 Secuencia 6: Noción de prototipo 131
 - 2.7 Secuencia 7: Sobrecarga de un método 133
 - 2.8 Secuencia 8: Extensión de un prototipo 134
 - 2.9 Secuencia 9: Mecanismo de la herencia. 135
 - 2.10 Secuencia 10: Límite de la herencia de la secuencia 9. 137
 - 2.11 Secuencia 11: Un segundo límite a nuestra herencia 139

Capítulo 9
Principales objetos en JavaScript

- 1. Presentación 141
- 2. Los objetos básicos 142
 - 2.1 Objeto Array 142
 - 2.2 Objeto Date 142
 - 2.3 Objeto Math 150
 - 2.4 Objeto window. 153
 - 2.5 Objeto navigator. 163
 - 2.6 Objeto String 165

6 --- Aprender a desarrollar

con JavaScript

Capítulo 10

Las novedades de EcmaScript 6

1. Presentación general	171
2. Aportaciones a nivel de la Programación Orientada a Objetos . . .	172
2.1 Noción de prototipo	172
2.2 Sobrecarga de un método	176
2.3 Extensión de prototipo	179
2.4 Herencia	182
2.5 Primer ejemplo POO en EcmaScript 6	186
2.6 Herencia en EcmaScript	190
2.7 Métodos getter, setter y static en EcmaScript 6	194
3. Funciones flecha (arrow functions)	198
3.1 Ventajas de las funciones flecha	198
3.2 Ejemplo	199
4. Estructuras Map, Set y bucle for of	205
4.1 Presentación general	205
4.2 Ejemplo	205
5. Ámbito de las variables (var o let)	215
5.1 Presentación general	215
5.2 Ejemplo	215
6. Promesas (promise)	219
6.1 Presentación general	219
6.2 Ejemplo	220
7. Desestructuración	224
7.1 Presentación general	224
7.2 Ejemplo	225

Parte 4 Gestión de formularios y modelo DOM

Capítulo 11 Utilizar formularios para introducir datos

- 1. Controles en la entrada de datos con JavaScript 229
 - 1.1 Control de entrada de datos en un campo de texto 229
 - 1.2 Control del tipo numérico de una entrada de datos en un campo de texto 236
 - 1.3 Control de caracteres alfabéticos de una entrada de datos en un campo de texto 239
 - 1.4 Control de caracteres alfabéticos y numéricos de una entrada de datos en un campo de texto 239
 - 1.5 Control de la longitud de una entrada de datos en un campo de texto 239
 - 1.6 Control de entrada de datos en una dirección de e-mail 240
 - 1.7 Control de la selección de una lista desplegable (versión simplificada) 241
 - 1.8 Control de la selección en una lista desplegable (versión extendida) 245
 - 1.9 Control de selección con un botón de radio 248
 - 1.10 Control de la selección con casillas de selección 252

Capítulo 12 Modelo DOM

- 1. Introducción 257
 - 1.1 Definición del DOM 257
 - 1.2 Definición de la arborescencia 258
- 2. Aprendizaje del modelo DOM 261
 - 2.1 Script "Hello World" 261
 - 2.2 Diferencia entre write y writeln 262
 - 2.3 Gestión de los enlaces hipertextos 263
 - 2.4 Gestión de las imágenes 265

8 --- Aprender a desarrollar

con JavaScript

2.5	Gestión de formularios y de sus etiquetas	267
2.6	Gestión de los anchors	270
2.7	Gestión de la navegación entre páginas web	272
2.8	Visualización de las características generales del documento.	277
2.9	Gestión de los botones en los formularios	279
2.10	Gestión de las tablas (etiqueta HTML table)	290

Capítulo 13

Exploración de flujo XML vía DOM

1.	Noción de flujo XML	319
2.	Ejemplos	320
2.1	Ejemplo 1: Visualización del contenido de un e-mail codificado en XML	320
2.2	Ejemplo 2: Lista de marcas de coches (archivo coches.xml)	324
2.3	Ejemplo 3: Lista de las marcas de coches con un bucle	326
2.4	Ejemplo 4: Lista de nodos conectados a la raíz.	327
2.5	Ejemplo 5: Lista de los campos (nodos) de cada coche.	329
2.6	Ejemplo 6: Sustitución de un valor de nodo	332
2.7	Ejemplo 7: Acceso a los atributos	332
2.8	Ejemplo 8: Acceso a un nodo padre.	334
2.9	Ejemplo 9: Recorrer hacia atrás los nodos.	335
2.10	Ejemplo 10: Sustitución sistemática de un valor de atributo	336
2.11	Ejemplo 11: Conversión XML en HTML	337
2.12	Ejemplo 12: Eliminación de un nodo en un flujo XML	339

Parte 5
Cookies y mecanismos de persistencia

Capítulo 14
Gestión de las cookies en JavaScript

- 1. Noción de cookie 343
- 2. Escritura de una cookie..... 344
- 3. Lectura de una cookie..... 346
- 4. Eliminación de una cookie 348

Capítulo 15
Almacenamiento local de datos

- 1. Presentación general de las soluciones..... 351
 - 1.1 Almacenamiento con sessionStorage 352
 - 1.2 Almacenamiento con localStorage 352
- 2. Implementación de Web Storage a través de ejemplos 353
 - 2.1 Ejemplo 1: Almacenamiento por localStorage de cadenas de caracteres..... 353
 - 2.2 Ejemplo 2: Almacenamiento en el localStorage de un objeto JavaScript 362

Capítulo 16
Almacenamiento remoto: formato XML

- 1. Presentación general de la solución 373
- 2. Implementación del almacenamiento remoto a través de ejemplos 374
 - 2.1 Ejemplo 1: Acceso Ajax sobre BDD MySQL (lista de los coches)..... 375
 - 2.2 Ejemplo 2: Acceso MySQL vía Ajax 400

Capítulo 17

Almacenamiento remoto: formato JSON

- 1. Presentación general de la solución 407
- 2. Implementación del almacenamiento remoto a través de ejemplos 409
 - 2.1 Ejemplo 1: Presentación del sistema de notación JSON 409
 - 2.2 Ejemplo 2: Lectura de un archivo JSON con XMLHttpRequest 414
 - 2.3 Ejemplo 3: Lectura de un archivo JSON con XMLHttpRequest y un script servidor en PHP 419
 - 2.4 Ejemplo 4: Lectura de una tabla MySQL con XMLHttpRequest (servidor PHP y flujo JSON) 421
 - 2.5 Ejemplo 5: Recodificación del ejemplo 4 con una lista desplegable 427

Parte 6

Geolocalización, diseño y gráficos de gestión

Capítulo 18

Geolocalización

- 1. Principio de la geolocalización 433
- 2. Ejemplos de aplicaciones de geolocalización 434
 - 2.1 Ejemplo 1: Visualización del mapa del centro de España. . . . 434
 - 2.2 Ejemplo 2: Visualización del mapa del centro de España (marcador) 441
 - 2.3 Ejemplo 3: Visualización del mapa del centro de España (marcador y círculos de población) 445
 - 2.4 Ejemplo 4: Visualización del mapa del centro de España (información meteorológica) 449
 - 2.5 Ejemplo 5: Visualización del mapa de Madrid (con fotos de los sitios importantes) 456
 - 2.6 Ejemplo 6: Visualización del mapa de Madrid (Street View) . . 458

Capítulo 19
Diseño (HTML5 CANVAS)

- 1. Presentación del API HTML5 CANVAS 465
- 2. Ejemplos de aplicaciones del elemento <canvas> 466
 - 2.1 Ejemplo 1: Dibujar un cuadrado sencillo 466
 - 2.2 Ejemplo 2: Dibujar una matriz de TicTacToe 470
 - 2.3 Posibles mejoras en el conjunto del TicTacToe 477

Capítulo 20
Gráficos de gestión

- 1. Diferentes soluciones de diseño de gráficos de gestión 479
- 2. Ejemplos de uso de los API de Google Charts 480
 - 2.1 Ejemplo 1: Dibujar un histograma 480
 - 2.2 Ejemplo 2: Representación de un gráfico de sectores 485
 - 2.3 Ejemplo 3: Representación de un mapa 488

Parte 7
Frameworks JavaScript y cliente/servidor

Capítulo 21
Framework AngularJS

- 1. Presentación general 493
- 2. Nociones básicas 494
 - 2.1 Primer ejemplo 494
 - 2.2 Directiva ng-bind 497
 - 2.3 Directiva ng-init 499
 - 2.4 Evaluación de una expresión 500
 - 2.5 Ejemplo de resumen 502
- 3. Manipular tablas de memoria y objetos 505
 - 3.1 Uso de una tabla 505
 - 3.2 Uso de un objeto 507

4.	Módulo y controlador	509
4.1	Ejemplo utilizando un módulo y un controlador.	511
4.2	Directiva personal.	513
4.3	Método personal.	515
4.4	Controlador en un archivo externo.	518
4.5	Introducir datos con zonas de texto	520
5.	Bucles y visualización en modo tabla	522
5.1	Iteración sobre una tabla de datos.	522
5.2	Iteración sobre una tabla de datos y un filtro.	525
5.3	Iteración sobre una tabla de datos y una ordenación.	528
5.4	Iteración sobre una tabla de datos y una tabla HTML	529
5.5	Filtrado de una tabla mediante una zona de texto.	533
6.	Acceso a un servidor de datos remoto (servidor PHP)	536
6.1	Lista simple a partir de una tabla MySQL	536
6.2	Lista filtrada a partir de una tabla MySQL.	546
6.3	Insertar un registro en una tabla MySQL.	552
7.	Controles de introducción de datos en los formularios	559
7.1	Control para introducir datos en una zona de texto	559
7.2	Control para introducir los datos de una dirección de correo electrónico.	561
7.3	Control para introducir datos en una zona de texto requerida	562
7.4	Lista desplegable para escoger una marca	564
7.5	Lista desplegable más avanzada para seleccionar una marca	567
7.6	Lista desplegable todavía más avanzada para seleccionar una marca	568
7.7	Lista desplegable asociada a una búsqueda MySQL.	570
7.8	Directivas ng-click y ng-mouseover	574
7.9	Directivas ng-show y ng-mouseleave	578
7.10	Gestión del tiempo (temporizador).	580
7.11	Gestión de las opciones mediante botones de radio.	584
7.12	Gestión de las opciones mediante casillas de selección	586

Capítulo 22
Framework Node.js

- 1. Presentación general 591
- 2. Instalación de Node.js 592
- 3. Ejecución del entorno Node.js 594
- 4. Primer script Node.js que muestra "Hello World" 595
 - 4.1 Presentación general 595
 - 4.2 Código fuente 596
 - 4.3 Ejecución 598
- 5. Segundo script que muestra «Hello World» (versión con Content-Type) 599
 - 5.1 Presentación general 599
 - 5.2 Código fuente 599
 - 5.3 Ejecución 600
- 6. Tercer script que muestra «Hello World» (versión HTML) 601
 - 6.1 Presentación general 601
 - 6.2 Código fuente 601
 - 6.3 Ejecución 603
- 7. Gestión de las URL 603
 - 7.1 Presentación general 603
 - 7.2 Código fuente 603
 - 7.3 Ejecución 605
- 8. Recuperar parámetros de la URL 606
 - 8.1 Presentación general 606
 - 8.2 Código fuente 606
 - 8.3 Ejecución 608
- 9. Detectar el evento close en el servidor 609
 - 9.1 Presentación general 609
 - 9.2 Código fuente 609
 - 9.3 Ejecución del servidor node_06.js 610

14 --- Aprender a desarrollar

con JavaScript

10. Creación de un objeto con su propio evento	611
10.1 Presentación general	611
10.2 Código fuente	611
10.3 Ejecución	613
11. Mostrar el primer registro de una tabla MySQL (versión 1)	613
11.1 Presentación general	613
11.2 Código fuente	614
11.3 Ejecución	617
12. Mostrar el conjunto de registros de una tabla MySQL	618
12.1 Presentación general	618
12.2 Código fuente	619
12.3 Ejecución	621
13. Mostrar el primer registro de una tabla MySQL (versión 2)	621
13.1 Presentación general	621
13.2 Código fuente	622
13.3 Ejecución	624

Parte 8

Desarrollo híbrido con Ionic 3

Capítulo 23

Desarrollo híbrido en JavaScript

1. Enfoques de desarrollo para las aplicaciones móviles	625
1.1 Desarrollos web, nativo y híbrido	626
1.1.1 Aplicaciones web	626
1.1.2 Aplicaciones nativas	627
1.1.3 Aplicaciones híbridas	629
1.2 Las tres principales plataformas	629
1.2.1 Apple iOS	629
1.2.2 Android	630
1.2.3 Windows Phone, Windows 10 Móvil	630

- 2. Panorama de las principales plataformas «híbridas» 631
 - 2.1 Ionic 631
 - 2.2 React Native 631
 - 2.3 Otras soluciones 632

Capítulo 24

Ionic 3, JavaScript en modo híbrido

- 1. Presentación detallada del entorno Ionic 633
- 2. Instalación del entorno Ionic 3 634
 - 2.1 La instalación de Node.js 635
 - 2.2 La instalación de los frameworks Ionic y Cordova 636
 - 2.3 Una primera aplicación de prueba 636
 - 2.4 Un IDE para modificar las scripts,
Microsoft Visual Studio Code 639
 - 2.5 El código del proyecto ionic3-blank en detalle 640

Capítulo 25

Gestión del Hardware en Ionic 3

- 1. Introducción 659
- 2. Aplicación ioni3-infos硬件 659
 - 2.1 Informe de la ejecución de la aplicación 659
 - 2.2 Análisis de los scripts principales 660
 - 2.3 Despliegue de la aplicación en Android 668
 - 2.3.1 Descarga de Android Studio 670
 - 2.3.2 Preparación de la aplicación Ionic
para su transferencia en Android Studio 670
 - 2.3.3 Importación de la aplicación Ionic en Android Studio 672
 - 2.3.4 Prueba de la aplicación Ionic en Android Studio
en modo AVD 675
 - 2.3.5 Prueba de la aplicación Ionic en el dispositivo «físico» 678

16 --- Aprender a desarrollar

con JavaScript

3.	Aplicación ionic3-battery_status	680
3.1	Resultado de la ejecución de la aplicación.	680
3.2	Análisis de los scripts principales.	681
4.	Aplicación ionic3-screen_orientation	686
4.1	Resultado de la ejecución de la aplicación.	686
4.2	Análisis de los scripts principales.	687

Capítulo 26

Componentes principales en Ionic 3

1.	Introducción	693
2.	Aplicación ionic3-paso_argumentos	694
2.1	Resultado de la ejecución de la aplicación.	694
2.2	Análisis de los scripts principales.	696
3.	Aplicación que integra ventanas modales	704
3.1	Aplicación ionic3-alert_basic	706
3.2	Aplicación ionic3-alert_confirm	710
3.3	Aplicación ionic3-alert_checkbox	714
3.4	Aplicación ionic3-alert_prompt	721
3.5	Aplicación ionic3-alert_radio	725

Capítulo 27

Aplicaciones básicas en Ionic 3

1.	Introducción	729
2.	Aplicación ionic3-raices_polinomio	729
2.1	Resultado de la ejecución de la aplicación.	729
2.2	Análisis de los scripts principales.	731
3.	Aplicación ionic3-MCD	739
3.1	Resultado de la ejecución de la aplicación.	739
3.2	Análisis de los scripts principales.	741

Capítulo 28

Gestión de la persistencia en Ionic 3

- 1. Introducción 745
- 2. Aplicación ionic3-clipboard 746
 - 2.1 Resultado de la ejecución de la aplicación. 746
 - 2.2 Análisis de los scripts principales. 747
- 3. Aplicación ionic3-local_storage 751
 - 3.1 Resultado de la ejecución de la aplicación. 751
 - 3.2 Análisis de los scripts principales. 752
- 4. Aplicación ionic3-sqlite 756
 - 4.1 Resultado de la ejecución de la aplicación. 756
 - 4.2 Análisis de los scripts principales. 759

Capítulo 29

Gestión de las listas en Ionic 3

- 1. Introducción 767
- 2. Aplicación ionic3-list_basic 768
 - 2.1 Resultado de la ejecución de la aplicación. 768
 - 2.2 Análisis de los scripts principales. 769
- 3. Aplicación ionic3-list_avatar 771
 - 3.1 Resultado de la ejecución de la aplicación. 771
 - 3.2 Análisis de los scripts principales. 772
- 4. Aplicación ionic3-list_slider 773
 - 4.1 Resultado de la ejecución de la aplicación. 773
 - 4.2 Análisis de los scripts principales. 776

Capítulo 30 Gráficos de gestión en Ionic 3

1. Introducción	785
2. Aplicación ionic3-chartjs_bar	786
2.1 Resultado de la ejecución de la aplicación.	786
2.2 Análisis de los scripts principales.	787
2.3 Script home.html	787
3. Aplicación ionic3-chartjs_pie.	792
3.1 Resultado de la ejecución de la aplicación.	792
3.2 Análisis de los scripts principales.	794
4. Aplicación ionic3-chartjs_doughnut_json	798
4.1 Resultado de la ejecución de la aplicación.	798
4.2 Análisis de los scripts principales.	800
Índice	811

Prefacio

- 1. ¿Por qué este libro? 13
- 2. ¿A quién se dirige este libro? 14
- 3. Estructura del libro 14

Capítulo 1 Introducción

- 1. Situación actual del desarrollo web 17
- 2. Angular 18
 - 2.1 Organización por componentes 18
 - 2.2 TypeScript 19
 - 2.3 Las especificaciones ES6 19
 - 2.4 DOM virtual 19
 - 2.5 Renderizado posible en el lado del servidor 20
- 3. AngularJS vs Angular 21

Capítulo 2 Mi primera aplicación

- 1. Introducción 23
- 2. El setup 23
 - 2.1 Instalación de npm 23
 - 2.2 Instalación de Angular CLI e inicialización del proyecto 25
- 3. Mi primer componente 27
- 4. Mi primer módulo 29
- 5. Ejecución de la aplicación 30

Capítulo 3

Fundamentos de Angular

1. Introducción	33
2. Los componentes	33
3. Los templates	35
3.1 Binding	36
3.2 Eventos JavaScript	38
3.3 Listas	39
3.4 Mostrar/ocultar un elemento	41
3.5 Enlaces	44
3.6 Estilos	45
3.7 Directivas	49
4. Los decoradores	51
5. Los pipes	52
5.1 Utilizar un pipe	52
5.2 Los pipes del framework	54
5.3 Crear un pipe	56
5.3.1 Los pipes de transformación	56
5.3.2 Los pipes filtros	58
5.4 Los pipes puros e impuros	59
6. Los módulos	61
7. Observable y promesa	62
8. Preparar la aplicación para un entorno de producción	64

Capítulo 4 TypeScript

1. JavaScript	65
2. TypeScript	67
2.1 Sintaxis	68
2.1.1 Variables	68
2.1.2 Funciones	69
2.1.3 Clases	69
2.1.4 Arrow Function	71
2.1.5 Unión de tipos	72
2.1.6 Interfaz	73
2.1.7 Genérico	74
2.1.8 Los decoradores	74
2.2 Typings	75
2.3 Compilación	76
2.4 EcmaScript 6 y 7 a partir de ahora	77
2.5 Angular y TypeScript	78

Capítulo 5 Angular CLI

1. Introducción	79
2. ¿Qué es Angular CLI?	79
2.1 La definición	79
2.2 Los comandos principales	81
2.2.1 Crear un proyecto nuevo Angular CLI: ng new	81
2.2.2 Compilar la aplicación: ng build	82
2.2.3 Ejecutar la aplicación con ng serve	85
2.2.4 Ejecutar pruebas con Angular CLI	87
2.3 Los comandos de scaffolding	90
2.4 La generación de un componente	91

3.	Configurar un proyecto Angular CLI	98
3.1	Esquema del archivo de configuración de Angular CLI	98
3.2	Gestionar los assets durante la compilación	102
3.3	Integrar una librería externa	105
3.4	Integrar un pre-procesador CSS	106
3.5	Añadir entidades de manera global	107
3.5.1	Añadir un script globalmente	107
3.5.2	Añadir una hoja de estilo globalmente	108
3.6	Eliminar la configuración Webpack	110
4.	Compilación Ahead-Of-Time (AOT)	111

Capítulo 6

Los componentes

1.	Introducción	115
2.	¿Qué es un componente?	115
2.1	Una primera definición	115
2.2	Crear un primer componente	117
2.2.1	Sintaxis inline	117
2.2.2	Una sintaxis real, descompuesta en varios archivos	118
2.2.3	Utilizar un componente en su aplicación	119
2.3	Mostrar las propiedades de un componente	121
3.	Los inputs y outputs	123
3.1	Los inputs de un componente	123
3.1.1	Declarar una variable como Input	123
3.1.2	Un ejemplo concreto con una lista	124
3.1.3	Dar un nombre personalizado a su input	126
3.2	Los outputs de un componente	127
3.3	Dar un nombre personalizado a su output	130

- 4. Interacción entre componentes 130
 - 4.1 Pasar un dato del padre al hijo con ayuda de un input 131
 - 4.2 Interceptar un cambio de valor con ayuda de un setter. 133
 - 4.3 Notificar al padre con ayuda de un EventEmitter en output . 135
 - 4.4 Observar los cambios de input con ngOnChanges. 137
 - 4.5 Utilizar una variable local 140
- 5. Los decoradores @ViewChild y @ViewChildren 142
 - 5.1 Recuperar una referencia con @ViewChild 142
 - 5.2 Recuperar varias referencias con @ViewChildren 145
- 6. Los componentes Angular y la View Encapsulation 147
 - 6.1 El Shadow DOM 147
 - 6.2 Especificar una View Encapsulation 149
 - 6.3 Tipos de View Encapsulation 149
 - 6.3.1 View Encapsulation emulada 150
 - 6.3.2 La View Encapsulation nativa 152
 - 6.3.3 Ninguna View Encapsulation 154

Capítulo 7
Los servicios

- 1. Introducción 157
- 2. ¿Qué es un servicio? 157
- 3. Declarar su servicio. 158
- 4. Utilizar su servicio 158
- 5. Hacer su servicio asíncrono 161
 - 5.1 Las promesas 161
 - 5.2 Los observables 164
 - 5.3 ¿Que elegir?..... 170
- 6. Notificar cuando los datos cambian 170

Capítulo 8

La inyección de dependencias

1. Principio básico	177
2. Inyección de dependencias en Angular	178
2.1 Registro global	179
2.2 Registro en un componente	180
2.3 Inyectar una dependencia	181
3. Provider	182
3.1 useClass	182
3.2 useExisting	183
3.3 useFactory	185
3.4 useValue	188
3.5 opaqueToken	189
3.6 Dependencia opcional	192
3.7 Inyección limitada	194
3.8 Restricción del registro de una dependencia	195

Capítulo 9

La petición HTTP

1. Introducción	197
2. Obtener y enviar datos	197
3. Transformar datos	200
4. Comunicar de manera segura	201
5. Simular la consulta HTTP	204

Capítulo 10
Las interacciones de usuario

- 1. ¿Qué es el event binding? 207
- 2. Suscribirse a un evento 208
- 3. Recuperar una entrada de usuario 209
 - 3.1 ¿Cómo manipular el objeto \$event? 209
 - 3.2 Utilizar un tipado fuerte para \$event 210
 - 3.3 Una alternativa con template reference variable 211
 - 3.4 Utilizar un template reference variable con un evento 212
 - 3.5 Filtrar las entradas de usuario 213

Capítulo 11
Los formularios

- 1. Los formularios basados en un template 217
- 2. Crear un componente de formulario 218
 - 2.1 El componente 218
 - 2.2 La vista y el data binding 219
 - 2.2.1 La sintaxis ngModel 220
 - 2.2.2 ngModel en detalle 220
 - 2.2.3 La utilización de ngModel en un caso concreto 220
 - 2.3 Integrar el formulario en la aplicación 221
- 3. Los estados y la validez de un campo 222
 - 3.1 Los estados de un input 222
 - 3.2 Aplicar estilos según la validez 224
- 4. Enviar el formulario 225
- 5. Los formularios y los FormControls 226
 - 5.1 Los controles y los grupos de controles 226
 - 5.2 Los validadores integrados 229
 - 5.3 Crear un validador personalizado 230
 - 5.4 Los validadores asíncronos 232

Capítulo 12

El enrutamiento

1. Introducción	233
2. Definir las rutas de una aplicación	233
3. El renderizado de componentes	235
4. Navegar en su aplicación	237
5. Recuperación de los datos de enrutamiento	238
6. Outlet con nombre	242
6.1 Definir outlets con nombre	242
6.2 Navegar con outlets con nombre	244
7. Resolver	246

Capítulo 13

Las directivas

1. Introducción	251
2. ¿Qué es una directiva?	251
2.1 Introducción	251
2.2 Directivas comunes	252
2.2.1 ngIf	252
2.2.2 ngFor	252
2.2.3 ngStyle	253
2.2.4 ngClass	253
3. Las directivas de atributo	254
3.1 Crear una directiva de atributo	254
3.2 Interactuar con los eventos del DOM	256
3.3 Pasar valores a las directivas de atributo	257
4. Las directivas estructurales	260
4.1 La etiqueta <template> y el asterisco	260
4.2 Crear una directiva estructural	261

Capítulo 14
Probar su aplicación

- 1. Introducción 267
- 2. Las pruebas unitarias 267
 - 2.1 Introducción a las pruebas con Jasmine 268
 - 2.2 Ejecutar código antes o después cada prueba 271
 - 2.3 Los matchers disponibles 273
 - 2.3.1 ¿Cómo utilizar un matcher? 273
 - 2.3.2 Ejemplos de matchers. 273
 - 2.3.3 Negación de un matcher 276
 - 2.4 Crear un matcher personalizado 277
 - 2.4.1 Crear una librería de matchers personalizados. 277
 - 2.4.2 Utilizar un matcher personalizado 279
 - 2.5 Los componentes 280
 - 2.5.1 El componente a probar 280
 - 2.5.2 El TestBed 281
 - 2.5.3 Comprobar que el componente está bien instanciado. . 282
 - 2.5.4 Controlar las propiedades del componente. 283
 - 2.5.5 Asegurarse de que el renderizado del componente sea coherente. 283
 - 2.6 Los servicios 285
 - 2.6.1 El servicio a probar 285
 - 2.6.2 Probar el servicio. 285
 - 2.6.3 Asegurarse de que el servicio se ha inyectado correctamente. 286
 - 2.7 Las directivas. 287
 - 2.7.1 La directiva a probar 287
 - 2.7.2 Probar la directiva. 288
 - 2.8 Inyectar un falso servicio 290

3.	Las pruebas e2e	291
3.1	Ejecutar las pruebas e2e	292
3.2	Escribir una prueba e2e	293
3.2.1	Interactuar con el navegador	294
3.2.2	Recuperar un elemento del DOM e interactuar con él	295
3.2.3	Interactuar con los elementos DOM.	296
3.2.4	Probar el componente.	298
3.3	Escribir una prueba e2e mejorada	299

Capítulo 15

La cross-platform con Angular

1.	Apache Cordova	303
1.1	Crear un proyecto Apache Cordova	304
1.2	Angular en Apache Cordova	306
1.3	Utilizar un plug-in Apache Cordova	311
2.	Ionic 2	316
2.1	Crear un proyecto Ionic	316
2.2	Utilizar los componentes Ionic	317
2.3	Utilizar Ionic DevApp	319
2.4	Utilizar un plug-in con Ionic	320

Capítulo 16

Para llegar mas lejos

1.	Introducción	323
2.	Renderizado en el lado del servidor	323
2.1	Principio de la implementación	324
2.2	Añadir el renderizado en el lado del servidor	325

- 3. La detección de cambios 335
 - 3.1 ¿Por qué la detección de cambios? 335
 - 3.2 Las zonas, o ¿cómo notificar a Angular? 338
 - 3.2.1 El comportamiento de la detección de cambios 340
 - 3.2.2 Inmutabilidad y estrategia de detección de cambios OnPush 341
 - 3.3 Todavía más control sobre la detección de cambios. 344
- 4. El ciclo de vida de un componente 345
 - 4.1 Presentación de los lifecycle hooks 345
 - 4.1.1 Los diferentes hooks 345
 - 4.1.2 Utilizar un lifecycle hook. 346
 - 4.2 El ciclo de vida de un componente 347
 - 4.2.1 El constructor 350
 - 4.2.2 ngOnInit 350
 - 4.2.3 ngOnChanges 351
 - 4.2.4 ngDoCheck 352
 - 4.2.5 ngAfterContentInit 352
 - 4.2.6 ngAfterContentChecked 354
 - 4.2.7 ngAfterViewInit 354
 - 4.2.8 ngAfterViewChecked 355
 - 4.2.9 ngOnDestroy 355
- Índice 357

