

---

**Solucionario de**

# ejercicios de autoevaluación



---

## Ejercicios de autoevaluación

### Unidad de Aprendizaje 1

#### 1. Indica si las siguientes afirmaciones son verdaderas o falsas.

a. *JavaScript* es un lenguaje que no se usa en la programación de aplicaciones web.

- Falso
- Verdadero

b. El incremento en el uso de *JavaScript* obliga a establecer las reglas *Clean Code*.

- Falso
- Verdadero

c. La codificación limpia solo puede usarse cuando está finalizado el programa o aplicación.

- Falso
- Verdadero

d. El cumplimiento de las reglas *Clean Code* en *JavaScript* son obligatorias.

- Falso
- Verdadero

e. La utilización de módulos independientes para programar es un aspecto exclusivo de *JavaScript*.

- Falso
- Verdadero

f. Actualmente, los patrones de diseño no tienen nada que ver con los establecidos originariamente.

- Falso
- Verdadero

2. La base del principio *Clean Code* consiste en generar códigos simples de forma que creen funciones...

- a. ... sencillas, claras y pequeñas.
- b. ... complejas.
- c. ... privadas y sencillas.
- d. Todas las opciones son incorrectas.

3. En los módulos diferenciados la implementación de cada módulo...

- a. ... es privada y visible.
- b. ... es privada y no visible.
- c. ... no es privada ni visible.
- d. ... es privada, pero no visible.

4. La estructura de funcionamiento de una web y una aplicación ...

- a. ... se estructuran en 3 niveles en una web y 3 en una aplicación.
- b. ... se estructuran en 3 niveles en una web y 4 en una aplicación.
- c. ... se estructuran en 4 niveles en una web y 4 en una aplicación.
- d. ... se estructuran en 4 niveles en una web y 3 en una aplicación.

5. No es un patrón de diseño:

- a. Patrones creacionales
- b. Patrones estructurales
- c. Patrones de comportamiento
- d. **Patrones de análisis**

6. Los patrones de diseño que especifican la relación entre clases y objetos son:

- a. Patrones creacionales
- b. Patrones modulares
- c. Patrones de comportamiento
- d. **Patrones estructurales**

7. El modelo de patrón que más elementos incorpora es:

- a. El patrón creacional
- b. El patrón estructural
- c. El patrón modular
- d. El patrón de comportamiento**

8. Un motivo por el que se deben realizar pruebas unitarias al código es:

- a. Aumenta el tiempo de ejecución.
- b. Mejora la calidad del código.**
- c. Reduce la lógica del código.
- d. No es necesario probar el código.

9. El orden en el que se deben ejecutar las pruebas unitarias es:

- a. Organizar → Actuar → Afirmar**
- b. Afirmar → Actuar → Organizar
- c. Actuar → Organizar → Afirmar
- d. Afirmar → Organizar → Actuar

10. Una ventaja del uso de las pruebas unitarias es:

- a. No se sigue un esquema claro.
- b. Las pruebas unitarias no son independientes al resto del código.
- c. Se prueba un único código cada vez.**
- d. Solo hay que probar el código que supere las 30 líneas.



---

## Ejercicios de autoevaluación

### Unidad de Aprendizaje 2

#### 1. Indica si las siguientes afirmaciones son verdaderas o falsas:

a. Una variable es un dato que se define y que no se utiliza en el desarrollo de los *scripts*.

- Falso
- Verdadero

b. Los nombres de las variables se deben establecer mediante el uso de caracteres alfanuméricos.

- Falso
- Verdadero

c. Los nombres de las variables pueden comenzar por un número.

- Falso
- Verdadero

d. Las palabras reservadas pueden utilizarse para definir variables y funciones.

- Falso
- Verdadero

e. Se considera una buena práctica declarar las variables antes de utilizarlas.

- Falso
- Verdadero

f. En la declaración de las variables pueden declararse varias con el mismo nombre (alto = Alto).

- Falso
- Verdadero

2. Si se declara una variable `var alto = 50; alto = 40; var alto = 30; alto = 20`, el valor que se asignará a la función será:

- a. 50
- b. 20**
- c. 30
- d. 40

3. Las variables que se declaren usando la palabra reservada "let"...

- a. ... tienen un alcance ilimitado y no muestran mensajes de error.
- b. ... tienen un alcance limitado y muestran mensajes de error.**
- c. ... tienen un alcance ilimitado y muestran mensajes de error.
- d. ... tienen un alcance limitado y no muestran mensajes de error.

4. Para declarar varias variables en la misma línea, la manera correcta es:

- a. let alto = 50, ancho = 120, fondo = 70.**
- b. let alto = 50. ancho = 120. fondo = 70.
- c. let alto = 50; ancho = 120; fondo = 70.
- d. let alto = 50: ancho = 120: fondo = 70.

5. ¿Cuál de los siguientes tipos pueden definir una variable?

- a. Numéricos.
- b. Cadenas de texto.
- c. *Booleanas*.
- d. Todas las opciones son correctas.**

6. Los objetos se componen de:

- a. Propiedades y métodos.
- b. Métodos y constructores.
- c. Propiedades y constructores.
- d. Propiedades, métodos y constructores.**

7. Los objetos en *JavaScript* se organizan...

- a. ... dependiendo de la cantidad de veces que se utilicen.
- b. ... dependiendo del tipo de variable.
- c. ... por orden alfabético.
- d. ... **de manera jerárquica.**

8. El elemento jerárquico de mayor nivel en *JavaScript* es:

- a. *document*
- b. ***window***
- c. *option*
- d. *plugin*

9. Las variables que describen las características principales del objeto se denominan:

- a. **Propiedades.**
- b. Eventos.
- c. Funciones.
- d. Todas las opciones son incorrectas.

10. Las acciones que se producen en el sistema que definen un comportamiento cuando suceden se denominan:

- a. Propiedades.
- b. **Eventos.**
- c. Funciones.
- d. Todas las opciones son incorrectas.



---

## Ejercicios de autoevaluación

### Unidad de Aprendizaje 3

#### 1. Indica si las siguientes afirmaciones son verdaderas o falsas:

a. La programación funcional se basa en el estado de los objetos.

- Falso
- Verdadero

b. Internet se basa en un sistema de servidores y páginas web.

- Falso
- Verdadero

c. Los botones, desplegables, etc., son elementos propios de la programación del lado servidor.

- Falso
- Verdadero

d. La programación *FrontEnd* es más importante que la programación *BackEnd*.

- Falso
- Verdadero

e. Una función pura es aquella cuyo resultado es siempre el mismo para la misma variable de entrada.

- Falso
- Verdadero

f. Una buena práctica es utilizar todas las funciones de tipo impuras.

- Falso
- Verdadero

2. Un elemento que diferencia la programación funcional del resto de paradigmas es:

- a. Nunca devuelve el mismo resultado.
- b. Siempre devuelve el mismo resultado.**
- c. El valor de la función no depende de ella misma.
- d. No ocupan espacio de almacenamiento en la memoria.

3. Los lenguajes más habituales en la programación del lado cliente son:

- a. HTML.
- b. CSS.
- c. *JavaScript*.
- d. Todas las opciones son correctas.**

4. ¿Cuál de las siguientes opciones no es una ventaja del empleo del *BackEnd* y *FrontEnd*?

- a. Las actualizaciones son más complicadas.**
- b. Mayor escalabilidad.
- c. Diferentes equipos de desarrollo.
- d. Las opciones a y c son correctas.

5. Una característica de las funciones puras es:

- a. Siempre devuelven el mismo valor.
- b. No utiliza variables globales.
- c. No modifica el estado del programa.
- d. Todas las opciones son correctas.**

6. El método *objetc.assign()* tiene la característica de:

- a. Copiar los valores y propiedades de un único objeto a otro nuevo.
- b. Copiar los valores y propiedades de varios objetos a otro nuevo.
- c. Asigna y define propiedades en el objeto origen.
- d. Las opciones a y b son correctas.**

7. El operador cuya definición se realiza mediante tres puntos consecutivos (...) es:

- a. *for* o *while*
- b. *Array*
- c. *Script*
- d. ***Spread***

8. ¿Qué elemento de los siguientes no corresponde con el establecimiento de un bucle en *JavaScript*?

- a. Valor inicial
- b. **Número de repeticiones**
- c. Condición
- d. Valor actualizado

9. Cuando se crea un *array* utilizando una coma en su definición...

- a. **... debe evitarse dejar el campo del valor vacío.**
- b. ... debe realizarse escribiendo el valor o dato en mayúsculas.
- c. ... debe realizarse escribiendo el valor o dato en minúsculas.
- d. Todas las opciones son incorrectas.

10. Para realizar el filtrado de un *array* puede hacerse usando el método *filter*, pero...

- a. ... solo funciona para cadenas de texto.
- b. **... puede funcionar para cadenas de texto y números.**
- c. ... solo funciona para números enteros.
- d. Las opciones a y c son correctas.



---

## Ejercicios de autoevaluación

### Unidad de Aprendizaje 4

#### 1. Indica si las siguientes afirmaciones son verdaderas o falsas:

a. Los módulos no permiten agrupar funcionalidades y valores.

- Falso
- Verdadero

b. Los módulos se pueden utilizar tanto en el lado cliente como en el lado servidor.

- Falso
- Verdadero

c. Un beneficio de utilizar los módulos es el escalado de los nombres.

- Falso
- Verdadero

d. La mejor opción para optimizar el código es no utilizar los módulos.

- Falso
- Verdadero

e. Utilizando módulos el código queda más ordenado.

- Falso
- Verdadero

f. Una buena práctica es nombrar los módulos con mayúsculas.

- Falso
- Verdadero

#### 2. ¿Qué comando debemos ejecutar para conocer las funciones que incorpora un módulo por la consola?

- a. `return function(nombre_modulo).`
- b. `console.log(nombre_modulo).`

- c. No se puede, hay que acceder al archivo y leer todo el código.
- d. Se debe acceder al archivo y localizar las llamadas al módulo.

**3. El repositorio más utilizado por los programadores es:**

- a. *Bee Network*
- b. *Atom*
- c. *Google Drive*
- d. *Git***

**4. Para conocer la versión de un paquete que está instalado en un equipo se debe ejecutar el comando**

- a. *Nombre\_paquete --upgrade*
- b. *Nombre\_paquete -update*
- c. *Nombre\_paquete -revision*
- d. *Nombre\_paquete -install***

**5. Si se importa una librería mediante código HTML, deben incluirse los atributos:**

- a. No es necesario introducir atributos en la etiqueta `<script>`.
- b. Se recomienda declarar los atributos *type* y *src* aunque no son obligatorios.
- c. Es obligatorio declarar el atributo *type* pero no *src*.
- d. Es obligatorio declarar el atributo *src* pero no *type*.**

**6. Una ventaja que no corresponde a la implementación de un módulo asíncrono es:**

- a. Mejora el rendimiento del sitio.
- b. Reduce los errores de la página.
- c. Concatena y minimiza los módulos.
- d. Genera archivos de mayor tamaño.**

**7. Una de las características introducidas en la versión ECMAScript2015 fue...**

- a. ... la mejora de la velocidad de carga de las páginas.
- b. ... la reducción de la carga del servidor.

- c. ... que elimina el almacenamiento de variables.
- d. ... que permite separar el programa en archivos independientes.**

**8. La posibilidad de eliminar código inactivo se introdujo en la versión ECMAScript del año...**

- a. ... 2013.
- b. ... 2015.**
- c. ... 2017.
- d. ... 2019.

**9. La eliminación de código inactivo se denomina:**

- a. *Tree-Shaking*.**
- b. *Clean Code*.
- c. *Webpack*.
- d. Todas las opciones son incorrectas.

**10. En la exportación de variables se debe tener cuidado en que...**

- a. ... solo se pueden exportar tres variables.
- b. ... se pueden exportar todas las variables que se deseen.**
- c. ... solo se pueden exportar variables sin valor asignado.
- d. Las opciones a y c son correctas.



---

## Ejercicios de autoevaluación

### Unidad de Aprendizaje 5

#### 1. Indica si las siguientes afirmaciones son verdaderas o falsas:

a. Es obligatorio utilizar los patrones de diseño en la creación de un sitio o página web.

- Falso
- Verdadero

b. Los principios SOLID fueron establecidos por Robert C. Martin.

- Falso
- Verdadero

c. Según el principio de responsabilidad única, una responsabilidad no es un motivo de cambio.

- Falso
- Verdadero

d. La aplicación de los principios SOLID es exclusiva de *JavaScript*.

- Falso
- Verdadero

e. Conseguir un código 100 % *Open-closed* es prácticamente imposible.

- Falso
- Verdadero

f. El principio de segregación de la interfaz establece que cada clase debe depender de otras para su ejecución.

- Falso
- Verdadero

2. ¿Qué principio se basa en que una clase debe tener una única y exclusiva causa por la que puede ser modificada?

- a. Principio *Open-closed*.
- b. Principio de responsabilidad única.**
- c. Principio de segregación de interfaz.
- d. Todas las opciones son incorrectas.

3. La aplicación de los principios SOLID...

- a. ... ayuda a que el código tenga una buena interacción.
- b. ... ayuda a generar el código más rápidamente.
- c. ... ayuda a que el código esté aislado de las partes que no guardan relación.
- d. Las opciones a y c son correctas.**

4. ¿Qué principio se basa en que las clases no deben cambiar cuando lo haga un requisito?

- a. Principio *Open-closed*.**
- b. Principio de responsabilidad única.
- c. Principio de segregación de interfaz.
- d. Todas las opciones son incorrectas.

5. ¿Qué características tienen las clases que cumplen el principio *Open-closed*?

- a. Son cerradas para la extensión y cerradas para la modificación.
- b. Son cerradas para la extensión y abiertas para la modificación.
- c. Son abiertas para la extensión y cerradas para la modificación.**
- d. Son abiertas para la extensión y abiertas para la modificación.

6. ¿Qué principio puede confundirse con el de responsabilidad única?

- a. Principio *Open-closed*.
- b. Principio de responsabilidad única.
- c. Principio de segregación de interfaz.**
- d. Todas las opciones son incorrectas.

7. Dentro de la programación orientada a objetos es habitual establecer una jerarquía...

- a. ... en la que los objetos de alto nivel incluyen las instancias de bajo nivel.
- b. ... en la que los objetos de bajo nivel incluyen las instancias de alto nivel.
- c. ... en la que los módulos son independientes.
- d. Todas las opciones con incorrectas.

8. El patrón *Singleton* se engloba dentro del grupo de patrones de...

- a. ... comportamiento.
- b. ... diseño.**
- c. ... funcionamiento.
- d. ... reacción.

9. El patrón *Observer* se basa en...

- a. ... los objetos observados.
- b. ... el lenguaje *JavaScript*.
- c. ... los objetos observadores.
- d. Las opciones a y c son correctas.**

10. En el patrón estrategia intervienen:

- a. El cliente y el contexto.
- b. La interfaz de la estrategia y la concreta.
- c. El cliente y la interfaz.
- d. Las opciones a y b son correctas.**



---

## Ejercicios de autoevaluación

### Unidad de Aprendizaje 6

#### 1. Indica si las siguientes afirmaciones son verdaderas o falsas:

- a. La generación de código no implica organización.
- Falso
  - Verdadero
- b. Las pruebas unitarias consisten en aislar parte del código.
- Falso
  - Verdadero
- c. Las pruebas unitarias sirven para encontrar errores de código.
- Falso
  - Verdadero
- d. La herramienta que garantiza que el código es 100 % funcional es la experiencia del programador.
- Falso
  - Verdadero
- e. Las pruebas unitarias se estructuran dependiendo de las necesidades del cliente o usuario.
- Falso
  - Verdadero
- f. El único *framework* para comprobar lenguaje *JavaScript* es *Jasmine*.
- Falso
  - Verdadero

#### 2. ¿Qué característica debe reunir una prueba unitaria?

- a. No es necesario que tenga un objetivo.
- b. Tiene que ser independiente del resto del código.**

- c. No se tienen que poder repetir.
- d. Todas las opciones son incorrectas.

**3. Las pruebas unitarias se estructuran siguiendo el orden:**

- a. Actuar-Afirmar-Organizar.
- b. Afirmar-Organizar-Actuar.
- c. Organizar-Actuar-Afirmar.**
- d. Las pruebas unitarias no tienen estructura.

**4. ¿Por cuáles de los siguientes motivos se deben realizar pruebas unitarias?**

- a. Verifican el código.
- b. Aseguran la ejecución correcta del programa.
- c. Mejoran la calidad del código.
- d. Todas las opciones son correctas.**

**5. Una ventaja del uso del *framework Jasmine* es...**

- a. ... requiere un DOM para asegurar su funcionamiento.
- b. ... permite automatizar la realización de las tareas.
- c. ... tiene una sintaxis limpia que facilita la escritura de las pruebas.**
- d. ... aunque es de pago, su coste es asumible por cualquier desarrollador.

**6. El archivo responsable de guardar la configuración de *Jasmine* tiene la extensión:**

- a. *.conf*
- b. *.js*
- c. *.json***
- d. Todas las opciones son incorrectas.

**7. El desarrollo dirigido por pruebas se conoce también como:**

- a. *Test Driven Development*.**
- b. *Driven Development Test*.

- c. *Development Driven Test*.
- d. Pruebas colectivas de desarrollo.

8. El desarrollo dirigido por pruebas se caracteriza por la realización de ciclos \_\_\_\_\_ de código.

- a. largos
- b. cortos**
- c. abiertos
- d. cerrados

9. ¿Qué requisitos hay que definir para que funcionen las pruebas incorporadas al desarrollo dirigido por pruebas?

- a. Eliminar la duplicación.
- b. Tener experiencia en programación.
- c. Verificar que la prueba falla.
- d. Las opciones a y c son correctas.**

10. ¿Cuál de las siguientes no es una ventaja de la realización de un *Test Driven Development*?

- a. Arquitectura y código más simples.
- b. La refactorización promueve mejoras.
- c. Aumenta la confianza en el código.
- d. No se pueden efectuar cambios en los requerimientos.**



---

## Ejercicios de autoevaluación

### Unidad de Aprendizaje 7

#### 1. Indica si las siguientes afirmaciones son verdaderas o falsas.

a. Los errores deben ser contemplados cuando se desarrolla una aplicación o programa.

- Falso
- **Verdadero**

b. Como programadores debemos enfocarnos en un único camino de especialización.

- **Falso**
- Verdadero

c. Una ventaja que tenemos actualmente es la especialización en un único ámbito.

- **Falso**
- Verdadero

d. Las buenas prácticas deben integrarse en la corrección de errores de código.

- **Falso**
- Verdadero

e. *JavaScript* se define como un lenguaje asíncrono.

- Falso
- **Verdadero**

f. El fallo en un programa o aplicación se denomina coloquialmente como *log*.

- **Falso**
- Verdadero

**2. La incorporación de unas buenas prácticas mientras se programa...**

- a. ... aumenta el tiempo de comprobación de errores.
- b. ... redundante en la mejora del código.**
- c. ... aumenta el tiempo dedicado al desarrollo.
- d. Todas las opciones son incorrectas.

**3. La ejecución de unas buenas prácticas al desarrollar una aplicación...**

- a. ... mejora la arquitectura del código.
- b. ... facilita el trabajo colaborativo.
- c. ... obliga a utilizar una misma metodología de programación.
- d. Todas las opciones son correctas.**

**4. Se considera una buena práctica en JavaScript:**

- a. Colocar los *scripts* en la parte superior de la página.
- b. Establecer espacios en blanco.
- c. Utilizar el punto y coma al final de la línea.**
- d. Establecer líneas de código lo más largas posibles.

**5. La implementación del código JavaScript en los navegadores...**

- a. ... se realiza en todos igual.
- b. ... los navegadores no pueden cargar código *JavaScript*.
- c. ... cada navegador lo hace de forma diferente.**
- d. ... los navegadores tienen que estar actualizados a la última versión.

**6. El uso de "promise" en las operaciones asíncronas...**

- a. ... ayuda a localizar el momento en el que se produce el error.
- b. ... permite enfocarse en la resolución del problema.
- c. ... permite localizar los problemas de ejecución en equipos móviles.
- d. Las opciones a y b son correctas.**

7. Al programa que tiene como misión localizar los errores lo llamamos...

- a. ... ***debugger***.
- b. ... *logger*.
- c. ... *eye tracking system*.
- d. ... *log tracking system*.

8. ¿Cuál de los siguientes no es un nivel de gravedad de un error?

- a. Crítico
- b. Código**
- c. Medio
- d. Bajo

9. ¿Cuál de las siguientes opciones corresponde con el guardado del registro de errores?

- a. El orden de sucesión.
- b. El nombre del usuario.
- c. La categoría del acontecimiento.
- d. Las opciones a y c son correctas.**

10. Hay que implementar un registro de errores o *logs*...

- a. ... siempre.
- b. ... solo si el desarrollo nos ha llevado más de 40 h.
- c. ... para aumentar la confianza en el código.
- d. ... siempre que se considere necesario.**

