

Descargar el zip con las tres carpetas, una vez finalizado el examen se comprimirán en un archivo zip renombrando la carpeta con vuestro nombre y apellidos y se subirá a la plataforma para la calificación.

Se someterá la aplicación subida a análisis y pruebas oportunas para formular una valoración global que tenga en cuenta todos los aspectos implicados en su realización, con especial énfasis en la ejecución y en el estilo. Si lo estima conveniente, el profesor podrá citar al alumno para formularle cuestiones que considere relevantes para la valoración global reseñada.

- El examen práctico se compone de 3 ejercicios con una calificación total de 7
- Para aprobar el examen es necesario que la nota total obtenida en los 3 ejercicios sea mayor o igual que 3.5.
- Si el código solución de un ejercicio tiene errores de ejecución o no pasa todos los casos de prueba se penalizará con un 50% de la nota
- Si el código solución de un ejercicio compila sin errores y tiene la funcionalidad pedida, la puntuación en dicho ejercicio será máxima.
- Se pondrá especial énfasis en valorar positivamente la ejecución (funcionamiento, adecuación a las especificaciones, robustez, ...) y el estilo (formato, comentarios, elección de identificadores, buenas prácticas ...).
- Se valorará positivamente añadir pruebas unitarias propias, linting, etc.

Muy importante

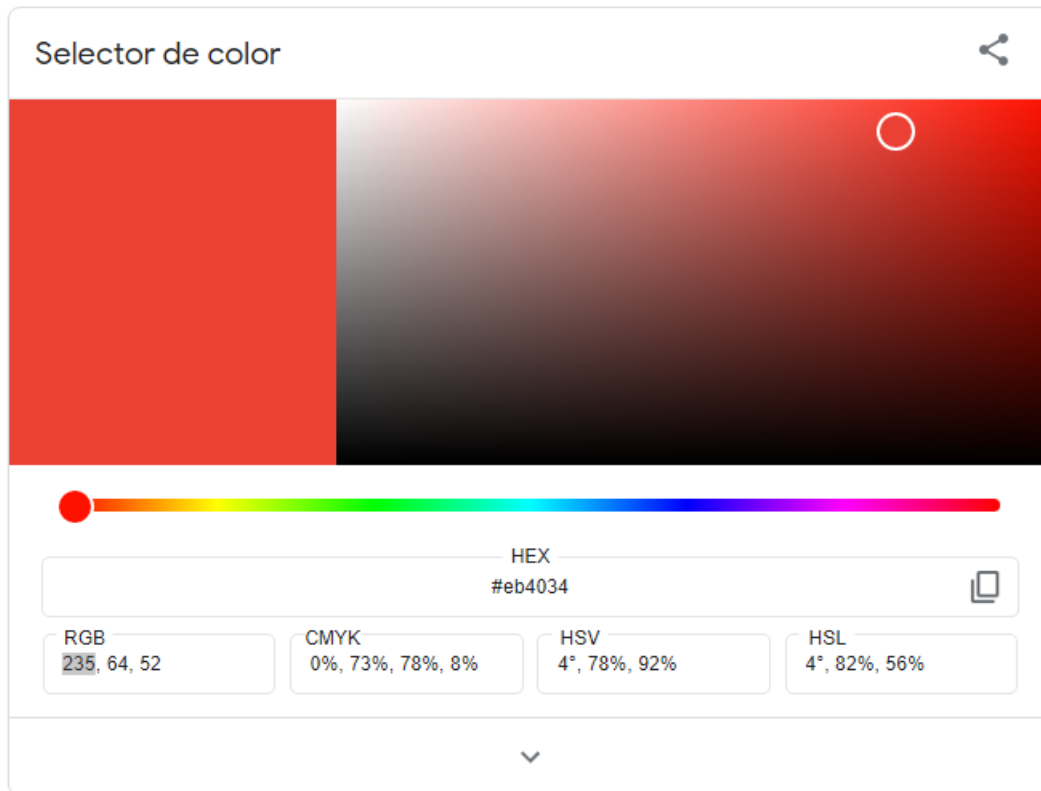
Lee con cuidado y atención las preguntas. Los ejercicios prácticos responden a lo que trabajamos en clase. Se pregunta sólo lo trabajado, confía en ti, tienes tiempo de sobra, puedes hacerlo.

Los exámenes son sólo fotos de tu conocimiento en el momento presente. Vas a dar lo mejor de tí, sin ninguna duda. Estás aquí por voluntad propia. No te presiones. Confía en ti.

Y recuerda: Esto solo es un parcial, simplemente estás escribiendo sobre lo que sabes de la profesión que has decidido seguir, desarrollador de páginas web. Todo va a salir bien. Respira hondo y sonríe, que en dos semanas es Navidad :o).

Primer ejercicio (2 puntos):

Si buscamos en google *rgb to hex* nos aparecerá una aplicación como la siguiente:



Tu objetivo es programar el método `rgb()` de esta aplicación que está incompleto. Completa el método para que al pasarle un valor RGB devuelva el resultado en representación hexadecimal. Los valores válidos para para RGB son de 0 a 255. **Cualquier valor (r,g,b) que salga de ese rango debería de devolverlo redondeado al valor válido más cercano.** Algunos ejemplos de los valores esperados son los siguientes :

```
rgb(255, 255, 255) // devuelve #FFFFFF
rgb(255, 255, 300) // devuelve #FFFFFF
rgb(0,0,0) // devuelve #000000
rgb(148, 0, 211) // devuelve #9400D3
```

1. Completar el método **`rgb()`** y pasar todos los casos de prueba. Puedes utilizar el método **`parseInt(string, base)`** para pasar de decimal a hexadecimal.
2. Crear una página web que permita introducir los valores rgb y calcule el valor hexadecimal con el método que acabas de completar.

Segundo ejercicio (2 puntos):

Estás esperando comenzar un proyecto, para que no estés sin hacer nada, tu SCRUM master te ha encargado implementar dos funciones en javascript que pueden ser útiles para proyectos en el que están trabajando.

- Implementar la función **validarDNI(dni)** Para validar un DNI se utilizan las letras "TRWAGMYFPDXBNJZSQVHLCKE". Su cálculo se realiza dividiendo el DNI entre 23 y el resto corresponde a la letra (puede estar en minúsculas y en mayúsculas) que está en esa posición. Deberemos verificar que es un número y la longitud no es mayor que 8
- Implementar una función llamada **calcularIMC(altura, peso, sexo, edad)**, que calcule el Índice de Masa Corporal, que se calcula multiplicando el peso en kilogramos por el cuadrado de la altura en metros. Además devolverá el imc con un decimal junto a la clasificación según la edad y el sexo ("hombre" o "mujer") siguientes:

Hombres

Edad	bajo peso	peso normal	sobrepeso	obesidad
16	<17.7	17.7 - 26.8	26.9 - 31.1	> 31.1
17	< 18.1	18.1 - 25.6	25.7 - 28.3	> 28.3
> = 18	< 18.1	18.1 - 25.1	25.2 - 30.1	> 30.1

Mujeres

	bajo peso	peso normal	sobrepeso	obesidad
16	<17.7	17.7 - 23.9	24 - 27.8	> 27.8
17	< 18.2	18.2 - 24.6	24.7 - 28.9	> 28.9
> = 18	< 18.0	18.0 - 25.7	25.8 - 28.3	> 28.3

Nota: Si tiene falta de tiempo, puede omitir la edad de 16 años.

Tercer ejercicio (3 puntos):

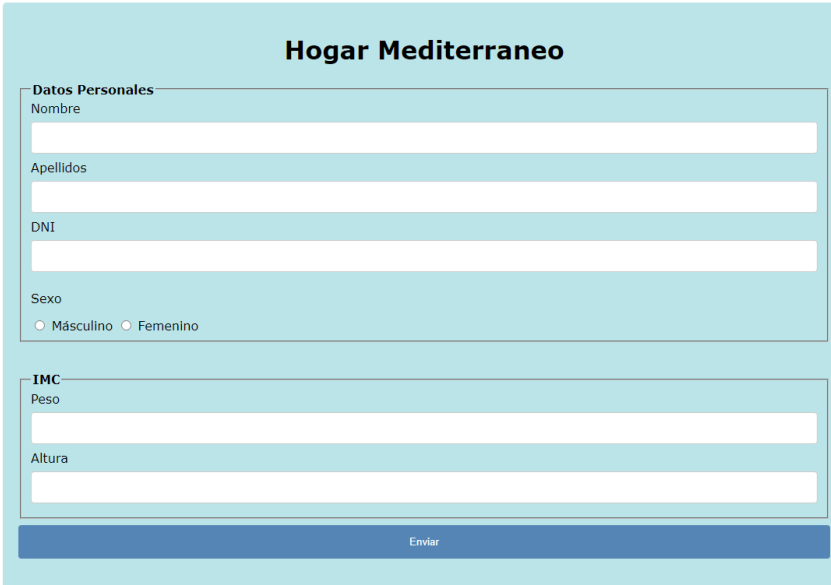
La empresa de adelgazamiento hogar Mediterraneo, han creado las siguientes historias de usuario:

“Cómo dueño del producto, quiero registrar usuarios rápidamente en hogarMediterraneo para empezar a venderle productos y ganar dinero”

“Cómo usuario, quiero obtener mi Índice de Masa Corporal antes de registrarme para saber si merece la pena dejarme un pastón en productos de la tienda”

a) Historia de usuario 1: Modificar el formulario html para validar los siguientes campos:

- Nombre : Obligatorio, máximo 20 caracteres
- Apellidos: Obligatorio, máximo 20 caracteres
- DNI: Obligatorio (usar función *validarDNI*)
- Sexo: Obligatorio. Devolverá “hombre” o “mujer”
- Fecha de nacimiento: Obligatorio
- Peso (Kg): Obligatorio. Validar para que esté entre 15 y 600
- Altura (metros): Obligatorio. Validar para que esté entre 0,5 y 2,5



Formulario de registro de Hogar Mediterraneo. El formulario está dividido en dos secciones: 'Datos Personales' y 'IMC'. La sección 'Datos Personales' contiene campos para Nombre, Apellidos, DNI, y Sexo (con botones de radio para Masculino y Femenino). La sección 'IMC' contiene campos para Peso y Altura. Un botón 'Enviar' está ubicado al final del formulario.

b) Historia de usuario 2: Utilizar la función *calcularIMC* para mostrar el resultado al final de la página una vez escrito/modificado el Peso y la Altura, sin tener que pulsar Enviar ni refrescar la página.

Puedes usar el evento **input** que se dispara cuando el valor (value) de un elemento `<input>` ha sido cambiado.