

Final Web 1 – 25-02-2021

Anexar .zip de cada ejercicio.

Pregunta 1

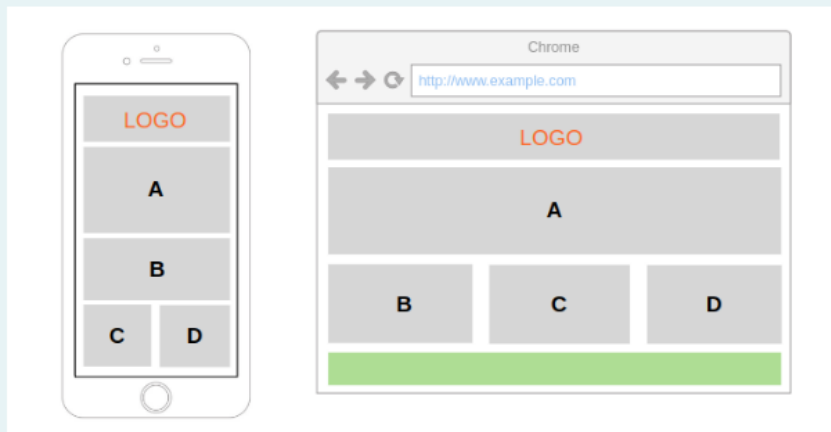
Pregunta **1**
Sin responder aún
Puntúa como 2,50
🚩 Marcar pregunta
⚙ Editar pregunta

Diseño Responsive

Realice un diseño **responsive** mobile first que se comporte según el siguiente diagrama.

NOTA

- **No se preocupe por las dimensiones exactas de las cajas**, solo se toma en cuenta la posición de las mismas.
- Si lo desea puede utilizar Bootstrap (ningún otro framework).



Pregunta 2

Pregunta **2**
Sin responder aún
Puntúa como 1,00
🚩 Marcar pregunta
⚙ Editar pregunta

CSS

Para cada una de las siguientes consignas escriba el código CSS correspondiente para que el sitio se visualice de la siguiente manera (No es necesario escribir código HTML pero debe suponer un correcto uso de los tags contemplando **semántica**):

NOTA: se evalúa el uso de buenas prácticas css. Utilice selectores anidados, combinados, reutilice clases, etc.

1. La tipografía del sitio debe ser "MS Serif" y color negro, los títulos h1 a h3 deben ser de color verde subrayado y en negrita, h4 y h5 en negrita y sin subrayar.
2. Los hiperlinks del menu de navegacion deben ser contexto de color blanco, fondo negro y cuando se posa el mouse deben invertirse (fuente negra y fondo blanco). Las imágenes de los artículos deben poseer un borde azul de 2 pixeles.
3. El pie de pagina debe ser de fondo azul con borde inferior rojo de 3px, y texto blanco.

Pregunta 3

Pregunta **3**
Sin responder aún
Puntúa como 3,00
🚩 Marcar pregunta
⚙ Editar pregunta

Realice una calculadora que permita calcular las 4 operaciones básicas entre dos operandos.
La calculadora debe cumplir los siguientes requerimientos.

- **No se debe permitir dividir por 0.**
- **Se debe tener una lista con las operaciones y resultados que se fueron haciendo.**
- **Se debe tener un botón para eliminar la lista de operaciones realizadas.**

Pregunta **4**

Sin responder
aún

Puntúa como
3,50

🚩 Marcar
pregunta

⚙ Editar
pregunta

El balneario "La Playita" desea implementar un servicio de pedidos de almuerzos del día. Para almacenar la información planean utilizar un servicio REST en la URL disponible en la URL "www.playita.com/api/menus". (la url es ficticia)

En una sección los veraneantes podrán ingresar los pedidos y en otra sección del sitio accederá el personal del servicio de almuerzos para consultar los pedidos y así saber cuantos almuerzos deben preparar y a que carpa deben acercarlos. Los pedidos de almuerzos y consultas interactúan sobre el mismo servicio REST. Dicho servicio devuelve objetos del tipo:

```
{
  "menus": [
    {
      "carpa": 1,
      "almuerzos": 8
    },
    {
      "carpa": 4,
      "almuerzos": 2
    },
    ...
  ]
}
```

Escriba el código JS para:

1. Escriba el código para obtener el total de almuerzos para la carpa 5 (u otra carpa indicada por parametro). Tenga en cuenta que puede haber más de un pedido de la misma carpa.
2. Insertar un pedido leído desde el HTML. Los objetos a enviar deben ser del tipo:

```
{
  "carpa": 3,
  "almuerzos": 3
},
```

Aclaración: el objeto JSON es de ejemplo, no suponga que las posiciones de los arreglos o los valores de los datos se respetan. Debe funcionar con cualquier objeto de este tipo.

Entrega: Se debe subir un archivo comprimido y TAMBIEN poner dentro del texto el código delimitando claramente cada una de las partes.