


<b>Comenzado el</b>	Thursday, 10 de December de 2020, 14:07
<b>Estado</b>	Finalizado
<b>Finalizado en</b>	Thursday, 10 de December de 2020, 15:08
<b>Tiempo empleado</b>	1 hora
<b>Calificación</b>	

#### Información

##### Pautas Acordadas de Final

Los alumnos se conectarán a un Google Meet grupal donde podrán hacer consultas y demás.

Cada alumno deberá además conectarse a una llamada individual. En la llamada individual compartirá su pantalla y deberá (en caso que posea) tener la cámara encendida y micrófono. No habrá docentes conectados a dicha llamada. Esta llamada dejará grabado el examen en su totalidad para consulta posterior en caso de considerarse necesario. En caso de no tener cámara/micrófono pueden conectarse dos veces y usar la cámara del celular.

No se permite el uso del celular. Los alumnos deberán durante el tiempo de cada actividad estar solos, sin personas a su alrededor. Se permite utilizar internet para consultas puntuales, no para copiar ejemplos enteros (ni de trabajos realizados previamente).

En caso de no poseer cámara en la computadora ni celular, se hará una defensa de 10 minutos de lo presentado al finalizar el examen o día siguiente a coordinar. Dicha defensa podrá ser por llamada telefónica estándar.

Los alumnos que rinden libre deberán además presentar el Trabajo Práctico Especial que se trabaja durante la cursada finalizado completamente y correctamente y defenderlo oralmente en la videollamada individual.

**IMPORTANTE: Las respuestas con copia textual descontarán el puntaje de la pregunta, describa los conceptos con sus palabras.**

Las dudas de consigna deben pedirse permiso por chat.

Solo se pueden hacer preguntas de consigna.

Prestar atención al tiempo, una vez cerrado el examen en Moodle no se aceptarán excepciones. Cada uno es responsable de mirar su reloj y asegurarse de enviar el examen antes del horario límite.

Sugerimos tener capturas de pantalla / documento de lo que hagan por cualquier inconveniente técnico.

## Pregunta 1

Finalizado

Puntúa 2,00 sobre 2,00

Describe cómo es la estructura de un archivo html. Indique cual es la utilidad / funcionalidad de cada parte y donde incluye el vinculo con otras tecnologías web.

```
<!DOCTYPE html> -> aqui se indica que tipo de documento es este archivo
<html lang="en"> -> aqui se especifica el lenguaje en el que el documento estara
escrito
<head> -> dentro del "head" incluimos toda aquella informacion que es importante
para el documento pero que no sera visible al usuario
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title> -> el titulo que aparecera en la pestaña del navegador
  <link rel="stylesheet" href="/css/style.css"> -> referencia a la hoja de estilo css
</head>
<body>

(aca iria el contenido que queremos que aparezca en nuestra pagina)

<script src="js/script.js"></script> -> incluimos al final del body, los vinculos a JS
</body>
</html>
```

Comentario:

## Pregunta 2

Finalizado

Puntúa 1,00 sobre 2,00

Explique brevemente que es el Box Model (Modelo de Cajas). Describa cómo planifica los layouts y como otorga la funcionalidad para que un sitio sea responsive con tres breakpoints (puntos de quiebre).

El Box Model es el modelo por el cual podemos asignar una ubicación y tamaño concreto al contenido dentro de nuestra pagina web. Todos los proyectos por más simples que parezcan utilizan box model.

Mediante box model todos los elementos que observamos en nuestra pagina son representados dentro de un "rectangulo" el cual podemos reubicar y cambiar de tamaño utilizando propiedades de css tales como margin, padding, width, height..

Comentario:

NO describe proceso responsive

### Pregunta 3

Finalizado

Puntúa 2,00 sobre 2,00

Describe brevemente que es ECMAScript6 y que ventajas incorpora en JS. Mencione dos buenas prácticas en JS y explique por que las considera buenas prácticas.

ECMAScript6 es una version mas actualizada de JS la cual nos permite añadir nuevas funcionalidades tales como una version mas simplificada de escribir un "for" por ejemplo donde antes podiamos escribirlo tipo:

```
for (var i = 0; i < 9; i++) {  
  //lo que quiero hacer  
}
```

y con la incorporacion de ES6

```
for (elem of elements) {  
  //lo que quiero hacer  
}
```

Incorpora tambien la posibilidad de realizar peticiones REST con la funcion async/await reemplazando el fetch:

```
async function name(event){  
  event.preventDefault();  
  ..  
  try{  
    let r = await fetch(url);  
    if (r.ok){  
      ...
```

#### **BUENAS PRACTICAS EN JS:**

al inicio del JS escribir "use strict"; de esta manera evitamos que variables que no fueron inicializadas se inicialicen por error de sintaxis generando errores complejos de identificar.

#### **realizar una funcion que**

Guardar las peticiones al DOM en una variable de este modo luego utilizamos la variable por ejemplo btn y accedemos a distintas opciones como por ejemplo btn.innerHTML = ... u btn.value = ...

Cada peticion que realizamos al DOM es costosa por eso esta practica se considera una buena practica!

No declarar variables globales, NUNCA! Estas pueden evitarse en caso de ser necesarias, utilizando una variable de bloque en un ambito de funcion..

#### Pregunta 4

Finalizado

Puntúa 2,00 sobre  
2,00

Comentario:

En Javascript, que diferencia hay entre una variable simple, un arreglo y un objeto.  
Escriba un ejemplo de como declararía una variable simple, un arreglo y un objeto.

En JS una variable simple almacena unicamente un dato, por ejemplo un string, un number, una variable booleana..

Ej:

```
let contador= 0;
```



Un arreglo almacena un conjunto de datos simples y permite acceder a estos datos almacenados de manera individual utilizando un indice el cual comienza en 0.

Al arreglo se le pueden almacenar nuevos elementos de manera dinamica utilizando por ejemplo la sintaxis `arr.push= "nuevo elemento";`.

Para vaciar un arreglo se puede por ejemplo volver a declararlo de manera vacia y sobrescribirlo (`arr = []`) o utilizar el splice

Los arreglos son faciles de recorrer utilizando un for desde `i=0` hasta `i=arr.lenght..`

Un objeto puede almacenar tanto datos como comportamiento u otros objetos.

ej:

```
final = {
```



```
"materia": "web 1",
```

```
"materias_dadas"= ["arreglo de materias", "..."],
```

```
"comportamiento": function(),
```

```
"otro_objeto": {"sarasa":2,
```

```
                "sarasara": true,
```

```
            },
```

```
}
```

Comentario:

**Pregunta 5**

Finalizado

Puntúa 1,20 sobre  
2,00

Describe que es AJAX y para que se utiliza. Explique en que consiste una promesa.

AJAX

Metodo por el cual las paginas web no requieren de una recarga completa de la pagina cada vez que se realiza un evento.

Funciona del lado del cliente

Con la utilizacion de PARTIAL RENDER (nos permite cargar un fragmento de HTML y luego ser insertado en nuestro HTML dentro de un div

Con REST (Consultamos un objeto JSON y lo procesamos del lado del cliente con JS)

Comentario:

No describe promesa

**Facultad de Ciencias Exactas – UNICEN**

Contacto administradores plataforma: E-mail moodle@exa.unicen.edu.ar – Tel. +54 0249 4385650 int. 2098