

Técnico Superior en Desarrollo de Aplicaciones Web

Desarrollo Web en Entorno Cliente

Práctica: Objetos y Asincronía

Lee bien todo lo que se indica en este enunciado y procura respetarlo.

Si tienes alguna duda o crees que hay alguna incoherencia, avisa al profesor para que te la solucione.

Normas de Realización

- a) El ejercicio se realizará en clase en el tiempo indicado para ello. No se va a dejar tiempo para acabarlo en casa.
- b) No se pueden utilizar herramientas que generen código para completar la práctica. Si se detecta dicho uso, se considerará plagio.
- c) Puedes consultar tus apuntes o pero **NO fuentes de internet** para realizar el ejercicio.
- d) El trabajo se compone de varios ejercicios que hay que programar usando el lenguaje Javascript. Está totalmente prohibido editar los ficheros con código CSS y HTML, salvo para enlazar los ficheros Javascript correspondientes. Si necesitas cambiar/añadir algo más, pregunta al profesor.
- e) Los archivos Javascript deben enlazarse en el <head> de cada pagina.
- f) Los archivos Javascript que se creen para solucionar los ejercicios deben ir en la carpeta jscode de la estructura proporcionada. **NO SE CORREGIRÁ** ningún archivo que no esté en esa carpeta.
- g) Sólo se permite usar las herramientas del lenguaje que se han dado hasta este momento. *(Recuerda: yo evalúo competencias, no sólo conocimientos).* **Por tanto, el uso de cualquier cosa que no se haya dado en clase, invalidará el ejercicio donde se realice y se calificará con 0** (independientemente de que funcione correctamente o no)

El no cumplimiento de alguno de los puntos indicados en los apartados Normas de Realización y Entrega, supondrá una penalización de 0.5 puntos en la nota final por cada fallo cometido.

Copia o plagio

Tal y como se indica en los estatutos del centro, está totalmente prohibido copiar código de otro/a compañero/a y/o de internet. Si se detecta y se demuestra este tipo de comportamiento, la calificación de la actividad será de 0 puntos.

Entrega

El ejercicio puntuará para la evaluación práctica del 2o Trimestre de la asignatura.

- La estructura de directorios proporcionada con todos los archivos necesarios para el funcionamiento correcto de la práctica deben meterse en una carpeta cuyo nombre será: **Apellido1_Apellido2_Nombre**. Ej: Gomez_Luque_Rafael
- Al finalizar el ejercicio debes entregar un archivo comprimido en .zip .rar o .7z con la carpeta creada en el punto anterior. Sube ese archivo a la tarea indicada en el campus dentro de la asignatura.
- **Recuerda:** se puntúa lo que se entrega en la plataforma.

MUY IMPORTANTE

Internet solo está permitido para consumir/enviar datos al servidor correspondiente. Cualquier uso diferente al indicado será penalizado con un 0 en la práctica.

Toda la gestión del DOM que necesites hacer en esta práctica, debes hacerla usando las funciones correspondientes. Es decir, está PROHIBIDO el uso de la propiedad `innerHTML`. La única propiedad permitida es `textContent`, por lo que los nodos de texto puedes hacerlos usándola.

Ejercicio 1 (3 puntos)

La página *randomuser* es un sitio web que proporciona una API Rest gratuita para hacer pruebas. Por cada consulta que hagamos a esa API nos va a devolver un objeto con datos aleatorios de un usuario ficticio.

Usando los archivos proporcionados en la carpeta *ejercicio1*, programa la siguiente funcionalidad para el botón 'Obtener Usuario':

Apartado A (0.75 puntos)

Implementa la siguiente funcionalidad para el botón que pone "Apartado A".

Haciendo uso de alguna de las herramientas vistas en clase, obtén un usuario de la URL: <https://randomuser.me/api> y muestra por consola lo que se recibe.

Apartado B (2.25 puntos)

Implementa la siguiente funcionalidad para el botón que pone "Apartado B".

Para cada usuario que se obtenga haciendo uso de la API de la URL anterior, muestra su nombre, apellidos, edad y la imagen de tamaño medio¹.

Las propiedades a las que me refiero son:
















- Nombre: `first`
- Apellidos: `last`
- Edad: `age`
- Imagen: `medium`

Esas propiedades pueden estar en la raíz del objeto obtenido o en otras propiedades que a su vez sean objetos. Debes investigar tú dónde están para poder usarlas (si tienes alguna duda, pregunta al profesor)

La forma de mostrar esos datos va a ser la siguiente:

- Todos los usuarios obtenidos irán dentro del DIV que hay en el `<section>` del código HTML.
- Cada usuario debe ir dentro de un `<article>` de la clase 'usuario'.
- Cada `article` estará estructurado de la forma: *nombre apellido: edad años* y debajo la foto.
- No te preocupes del CSS. Si colocas la estructura como se te dice en los puntos anteriores, te saldrá todo como en la foto. HAZ PRUEBAS primero antes de preguntar al profesor.

¹ Sigue leyendo el ejercicio para saber cómo debes mostrarlo.

Creusa Carvalho: 28 años 	Liam Denys: 52 años 	Milja Laurila: 47 años 	Silas Garcia: 56 años 
Juana Lorenzo: 35 años 	Selma Schweitzer: 61 años 	Carmelo Ramirez: 50 años 	Finn Bourgeois: 46 años 
Peter Jimenez: 59 años 	Antonia Herrero: 36 años 	Julia Ennis: 38 años 	Arttu Kilpela: 47 años 
Jessie Bates: 37 años 	33 رالين ياسمى: años 	Jonas Jensen: 42 años 	

Nota: como se ve en la foto (última fila), los nombres en arábigo no respetan nuestra estructura de escritura (obviamente). No te preocupes por eso.

Ejercicio 2 (2 puntos)

Usando los archivos proporcionados en la carpeta ejercicio2, programa la siguiente funcionalidad para el botón 'Obtener Resultados':

- Haciendo uso de alguna de las herramientas vistas en clase, obtén los datos de la URL: <http://www.jaimeweb.es/medac/datos.json> y muestra por consola lo que obtienes. (0.5 puntos)
- Muestra POR CONSOLA cada objeto obtenido uno debajo de otro. (0.5 puntos)

```
► Object { id: 1, nombre: "Jaime Hormiga", email: "hormiga@jweb.es", estado: false }  
► Object { id: 2, nombre: "Carlos", email: "carlos@medac.es", estado: true }  
► Object { id: 3, nombre: "Cristian", email: "Cristian@medac.es", estado: true }  
► Object { id: 4, nombre: "Pablo", email: "Pablo@medac.es", estado: true }  
► Object { id: 5, nombre: "Alek", email: "Alek@medac.es", estado: true }
```

- En apartado diferente (no mezclado con el apartado anterior), muestra por consola todos los datos de cada siguiendo la plantilla que se muestra en la foto siguiente (1 punto)

```
campo 1: valor del objeto 1  
campo 2: valor del objeto 1  
campo 3: valor del objeto 1  
campo 4: valor del objeto 1  
=====  
campo 1: valor del objeto 2  
campo 2: valor del objeto 2  
campo 3: valor del objeto 2  
campo 4: valor del objeto 2  
=====  
campo 1: valor del objeto 3  
campo 2: valor del objeto 3  
campo 3: valor del objeto 3  
campo 4: valor del objeto 3  
=====  
campo 1: valor del objeto 4  
campo 2: valor del objeto 4  
campo 3: valor del objeto 4
```

Nota: donde aparece 'campo 1', 'campo 2', ... debe aparecer el nombre del campo/atributo/propiedad. Donde aparece 'valor del objeto X' debe aparecer el valor que tiene el objeto X en el campo/atributo/propiedad correspondiente.

Ejercicio 3 (5 puntos)

Usando los archivos proporcionados en la carpeta ejercicio3, programa la siguiente funcionalidad para los diferentes botones de esa página:

Botón 'Obtener Datos'

- a) (0,75 puntos) Haciendo uso de alguna de las herramientas vistas en clase, obtén los datos de los diferentes profesores usando la URL:
`http://www.jaimeweb.es/medac/getProfesores.php` y muestra por consola lo que obtienes
- b) (2 puntos) Muestra en el div salida los datos de cada profesor como indica el dibujo



A tener en cuenta:

- Cada profesor va en una "ficha". Elige la estructura que tú quieras para crearla y tan solo haz que pertenezca a la clase "ficha".
- Fijate que de cada profesor solo quiero el nombre y debajo el DNI. Me da igual como lo hagas mientras salga así (primero nombre y debajo el dni).
- Cada vez que se pulse el botón, debe eliminarse el contenido del DIV salida. No quiero que se vaya añadiendo profesores al DIV cada vez que se pulse el botón.

Botón 'Enviar Datos'

- a) **(1.25 puntos)** Haciendo uso de alguna de las herramientas vistas en clase, envía todos los datos del formulario a la URL:
`http://www.jaimeweb.es/medac/setProfesores.php`
- b) **(1 punto)** La respuesta que te devuelva la URL anterior debes mostrarla en el DIV 'salida'

Nota: el script PHP de la URL inserta un profesor con los datos del formulario (si ese profesor no existe). Por tanto puedes comprobar si te funciona o no pulsando el botón 'Obtener Datos'. Eso sí, recuerda que la BBDD la está tocando toda la clase...habrá un momento en el que van a salir muchos profesores (o cosas raras que meta la gente...).