

Tarea online

Título de la tarea: Tarea online para DWECC02

Unidad: UT02. Modelo de Objetos predefinidos en JavaScript.

Ciclo formativo y módulo: Desarrollo de Aplicaciones WEB -Desarrollo web en entornos cliente.

Curso académico: 2025/2026

¿Qué contenidos o resultados de aprendizaje trabajaremos?

RESULTADO DE APRENDIZAJE:

- ✓ RA3. Escribe código, identificando y aplicando las funcionalidades aportadas por los objetos predefinidos del lenguaje.

CONTENIDOS:

- Objetos de más alto nivel en Javascript: window, location, navigator y document.
- Marcos: jerarquías y comunicación.
- Objetos nativos en Javascript: String, Math, Number, Boolean, Date

1.- Descripción de la tarea



Caso práctico

Juan y María quieren ser desarrolladores front-end en entornos web, pero solamente tienen conocimientos de HTML. Por tanto, se ponen a investigar las tecnologías que se utilizan ahora en la empresa y finalmente se deciden por dar sus primeros pasos en JavaScript. ¿Podrías ayudarles en esta tarea?

Actividad 1. Mini panel de compatibilidad y navegación

Crea un mini panel que muestre información básica del navegador del usuario y dos acciones de navegación controlada.

Interfaz mínima:

- ✓ Tamaño de la ventana (*viewport*): ancho × alto
- ✓ URL actual (*location.href*)
- ✓ Sistema/Navegador: a partir de *navigator.userAgent* o *userAgentData*
- ✓ ¿Móvil?: detección sencilla (*regex* sobre *userAgent*)
- ✓ Estado de conexión: “online/offline” (*navigator.onLine*)
- ✓ Botones:
 - Abrir ayuda → ir a Google permitiendo volver atrás.
 - Salir → ir a Yahoo sin permitir volver atrás.

Comportamiento:

- ✓ El tamaño de ventana y de conexión deben actualizarse automáticamente.
- ✓ El botón de abrir ayuda debe permitir volver con el botón de atrás y el de salir no.

Actividad 2. Cifrado/Descifrado César (con ñ)

Cifrar y descifrar textos utilizando el [cifrado Cesar](#).

Interfaz mínima:

- ✓ Textarea editable: texto de entrada.
- ✓ Input numérico: clave (admite negativos).
- ✓ Botones: Cifrar y Descifrar.
- ✓ Textarea de resultado (solo lectura).
- ✓ Botón de copiar resultado del textarea de resultados al portapapeles.

Comportamiento:

- ✓ Cifra y descifra textos, soportando el alfabeto español (con ñ/Ñ).
- ✓ Desplazamiento circular (módulo longitud del alfabeto).
- ✓ Conserva espacios, signos y números sin alterarlos.
- ✓ Maneja mayúsculas/minúsculas (puedes preservar el caso o normalizar de forma consistente).

Pistas técnicas:

- ✓ Alfabeto mínimo:
 - ➔ `const ABC_MIN = "abcdefghijklmnñopqrstuvwxyz";`
 - ➔ `const ABC_MAY = ABC_MIN.toUpperCase();`
- ✓ Puedes omitir el uso de acentos o añadirlos directamente en el alfabeto: a á b c d e é f ...
- ✓ Usa `indexOf` y `(pos + clave) % alfabeto.length` con ajuste para negativos.

Actividad 3. Recordatorio de entregas

Crear una lista de “entregas” (tareas) con fecha y etiqueta breve.

Interfaz mínima:

- ✓ Input de texto corto (“Descripción de la entrega”, p. ej. “Práctica 2 de DWEC”).
- ✓ Input de fecha/hora (*datetime-local*).
- ✓ Botón Añadir.
- ✓ Lista visible de recordatorios.

Comportamiento:

- ✓ Permite añadir nuevas entradas a la lista de recordatorios.
- ✓ Cada elemento muestra: descripción, fecha/hora y una cuenta atrás (días/horas/min/seg) que se actualiza cada segundo.
- ✓ Si la fecha pasó, muestra “Hace X ...” (tiempo transcurrido) y aplica estilo diferenciado (p. ej., clase `.vencido`).

Pistas técnicas:

- ✓ Formatea la diferencia por unidades (días, horas, minutos y segundos).

Actividad 4. Almacenar recordatorios

Guardar y restaurar la lista de la Actividad 3 usando Web Storage (persistencia entre sesiones/pestañas del mismo navegador).

Requisitos:

- ✓ Al cargar la página, leer de `localStorage` (p. ej., clave `recordatoriosEntregas`) y cargar la lista en pantalla.
- ✓ Tras añadir una nueva entrada, actualizar el valor en `localStorage`.
- ✓ Botón de vaciar la lista de `localStorage` y recargar la página.

Demostración en DevTools (obligatoria):

- ✓ Abrir Herramientas del desarrollador → Application (Almacenamiento) → Local Storage y mostrar la clave/valor guardado.

Se deben tener en cuenta las siguientes consideraciones:

- Todo el código debe estar comentado y depurado. Este aspecto es fundamental para poder superar el criterio de evaluación RA3.h).
- Se deberá adjuntar el código de los ejercicios de programación.
- Se debe realiza un documento explicativo de la práctica en la que expliques que aproximación y elementos has utilizado en cada ejercicio, añade capturas de pantalla, portada y nombre y apellidos, curso y módulo. El documento será llamado DWEC02_NombreApellido1 y pasado a pdf.
- En las actividades de programación se deberán realizar capturas de pantalla mostrando su funcionamiento y se deberá adjuntar todo el código fuente necesario para su ejecución.

2.- Información de interés

Recursos necesarios y recomendaciones

RECURSOS NECESARIOS:

- Ordenador personal con, al menos, 4 Gigabyte de memoria RAM.
- Conexión a Internet.
- Procesador de textos.
- Navegador web.
- Entorno de desarrollo web (opcional).
- Software para comprimir los archivos de la Tarea.

RECOMENDACIONES:

Para los apartados de esta Tarea tendrás que:

- No olvides elaborar el documento explicativo.

Webs de interés:

- <https://www.w3schools.com/js/default.asp>
- <https://developer.mozilla.org/es/docs/Web/JavaScript>
- <https://developer.mozilla.org/es/docs/Web/HTML/Elemento/iframe>
- <https://stackoverflow.com/questions/11955413/accessing-elements-of-one-iframe-from-another-iframe>
- <https://developer.mozilla.org/es/docs/Web/API/Window>
- <http://www.adp-gmbh.ch/web/js/variables/opener.html>
- https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Objetos_globales/Math
- https://www.w3schools.com/jsref/met_document_getelementsbyclassname.asp
- https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Objetos_globales/String



Indicaciones de entrega

Una vez realizada la tarea, el envío se realizará a través de la plataforma. El archivo se nombrará siguiendo las siguientes pautas:

Apellido1_Apellido2_Nombre_DWEC_Tarea02

3.- Evaluación de la tarea

Criterios de evaluación implicados

- ✓ RA3.d) Se han generado textos y etiquetas como resultado de la ejecución de código en el navegador.
- ✓ RA3.f) Se han utilizado las características propias del lenguaje en documentos compuestos por varias ventanas.
- ✓ RA3.g) Se han utilizado mecanismos del navegador web para almacenar información y recuperar su contenido.
- ✓ RA3.h) Se ha depurado y documentado el código.

¿Cómo valoramos y puntuamos tu tarea?

Rúbrica de la tarea			
Actividad	Criterios de evaluación implicados	Calificación	Retroalimentación
Actividad 1: Mini panel	RA3.f)		
Actividad 2: Cifrado	RA3.d)		
Actividad 3: Recordatorio de fechas	RA3.d)		
Actividad 4: Almacenar recordatorios	RA3.g), RA3.h)		