

12-11-2023

# Modelo de objetos predefinidos en JavaScript

## Tarea 03

ROBERTO RODRÍGUEZ JIMÉNEZ  
roberto.rodjim.1@educa.jcyl.es

## Contenido

Tarea online DWEC02.....	3
¿Qué contenidos o resultados de aprendizaje trabajaremos? .....	3
Resultados de Aprendizaje.....	3
Contenidos .....	3
1.- Descripción de la tarea.....	4
<b>Caso práctico</b> .....	4
¿Qué te pedimos que hagas? .....	4
2.- Información de interés .....	5
Recursos necesarios .....	5
Consejos y recomendaciones.....	5
3.- Evaluación de la tarea .....	5
Criterios de evaluación implicados .....	5
¿Cómo valoramos y puntuamos tu tarea? .....	5
RESOLUCIÓN .....	6
Abra una nueva ventana no redimensionable. ....	6
Hacer una función y dentro de esa función: .....	8
Escribir en la nueva ventana <h3>Ejemplo de Ventana Nueva</h3> .....	9
URL Completa: XXXXX .....	9
Protocolo utilizado: XXXXX.....	9
Nombre en código del navegador: XXXXX.....	9
Que detecte si está JAVA disponible .....	9
Que abra un iframe con el contenido de www.google.es y de 800x600. ....	9
Y ahora fuera del código de la función que siga haciendo lo siguiente: .....	11
Que escriba en la ventana principal <h1>TAREA DWEC03</H2><HR /> .....	11
Que solicite: introduzca su nombre y apellidos. ....	11
Que solicite: introduzca DIA de nacimiento.....	11
Que solicite: introduzca MES de nacimiento. ....	11
Que solicite: introduzca AÑO de nacimiento. ....	11
Una vez solicitados esos datos imprimirá en la ventana principal:.....	11
Buenos días XXXXX.....	11
Tu nombre tiene XX caracteres, incluidos espacios. ....	12
La primera letra A de tu nombre está en la posición: X .....	12
La última letra A de tu nombre está en la posición: X .....	12
Tu nombre menos las 3 primeras letras es: XXXXXXXX .....	12
Tu nombre todo en mayúsculas es: XXXXXXXX .....	12

Tu edad es: XX años.....	12
Naciste un feliz XXXXXX del año XXXX.....	13
El coseno de 180 es: XXXXXXXXXX .....	13
El número mayor de (34,67,23,75,35,19) es: XX.....	13
Ejemplo de número al azar: XXXXXXXXXX.....	13
Código .....	14
index.html .....	14
aplicacion.js.....	16
Recursos .....	18

## Tarea online DWEC02

**Título de la tarea:** Estructura del lenguaje JavaScript

**Unidad:** DWEC02

**Ciclo formativo y módulo:** Desarrollo de Aplicaciones Web - Desarrollo web en entorno cliente

**Curso académico:** 2020/2021

### ¿Qué contenidos o resultados de aprendizaje trabajaremos?

#### Resultados de Aprendizaje

- ✓ RA 3: Escribe código, identificando y aplicando las funcionalidades aportadas por los objetos predefinidos del lenguaje.

#### Contenidos

Modelo de objetos predefinidos en JavaScript.

1. Objetos de más alto nivel en Javascript.
  1. Objeto `window`.
    1. Gestión de ventanas.
    2. Propiedades y métodos.
  2. Objeto `location`.
  3. Objeto `navigator`.
  4. Objeto `document`.
2. Objetos nativos en Javascript.
  1. Objeto `String`.
    1. Propiedades y métodos del objeto `String`.
  2. Objeto `Math`.
  3. Objeto `Number`.
  4. Objeto `Boolean`.
  5. Objeto `Date`.

## 1.- Descripción de la tarea

### Caso práctico

Antonio ha completado correctamente la fase de introducción y fundamentos básicos del lenguaje JavaScript, y ahora comienza a investigar en las características de los objetos predefinidos en JavaScript.

Estos objetos le van a permitir gestionar ventanas, marcos, propiedades de los navegadores, de las URL, etc. en JavaScript.

Además, también va a poder realizar operaciones matemáticas, de fecha y de cadenas, con otros tantos objetos nativos del lenguaje JavaScript.

Antonio tiene una pequeña reunión con Ada y con su tutor Juan, para comentar los progresos realizados hasta este momento y se pone manos a la obra con esta nueva sección.

### ¿Qué te pedimos que hagas?

Realizar una pequeña aplicación en JavaScript que realice lo siguiente:

- ✓ Abra una nueva ventana no redimensionable.
  - Hacer una función y dentro de esa función:
  - Escribir en la nueva ventana `<h3>Ejemplo de Ventana Nueva</h3>`
  - URL Completa: XXXXX
  - Protocolo utilizado: XXXXX
  - Nombre en código del navegador: XXXXX
  - Que detecte si está JAVA disponible en esa ventana del navegador y si es así que escriba:
    - Java SI disponible en esta ventana, o bien.
    - Java NO disponible en esta ventana.
  - Que abra un iframe con el contenido de [www.google.es](http://www.google.es) y de 800x600.

Y ahora fuera del código de la función que siga haciendo lo siguiente:

- Que escriba en la ventana principal `<h1>TAREA DWEC03</H2><HR />`
- Que solicite: introduzca su nombre y apellidos.
- Que solicite: introduzca DIA de nacimiento.
- Que solicite: introduzca MES de nacimiento.
- Que solicite: introduzca AÑO de nacimiento.
- Una vez solicitados esos datos imprimirá en la ventana principal:
  - Buenos días XXXXX
  - Tu nombre tiene XX caracteres, incluidos espacios.
  - La primera letra A de tu nombre está en la posición: X
  - La última letra A de tu nombre está en la posición: X
  - Tu nombre menos las 3 primeras letras es: XXXXXXXX
  - Tu nombre todo en mayúsculas es: XXXXXXXX
  - Tu edad es: XX años.
  - Naciste un feliz XXXXXX del año XXXX.
  - El coseno de 180 es: XXXXXXXXXX
  - El número mayor de (34,67,23,75,35,19) es: XX

- Ejemplo de número al azar: XXXXXXXXXX

**Donde aparecen las XXXX tendrá que aparecer el cálculo o texto que corresponda.**

## 2.- Información de interés

### Recursos necesarios

Editor de texto para teclear el código de la aplicación y un navegador web.

### Consejos y recomendaciones

Mirar las referencias y códigos de ejemplo de la página W3Schools:

[Página de referencia de w3schools.](https://www.w3schools.com/)

## 3.- Evaluación de la tarea

### Criterios de evaluación implicados

- a) Se han identificado los objetos predefinidos del lenguaje.
- b) Se han analizado los objetos referentes a las ventanas del navegador y los documentos web que contienen.
- c) Se han escrito sentencias que utilicen los objetos predefinidos del lenguaje para cambiar el aspecto del navegador y el documento que contiene.
- d) Se han generado textos y etiquetas como resultado de la ejecución de código en el navegador.
- e) Se han escrito sentencias que utilicen los objetos predefinidos del lenguaje para interactuar con el usuario.
- f) Se han utilizado las características propias del lenguaje en documentos compuestos por varias ventanas y marcos.
- g) Se han utilizado “cookies” para almacenar información y recuperar su contenido.
- h) Se ha depurado y documentado el código.

¿Cómo valoramos y puntuamos tu tarea?

### Rúbrica de la tarea

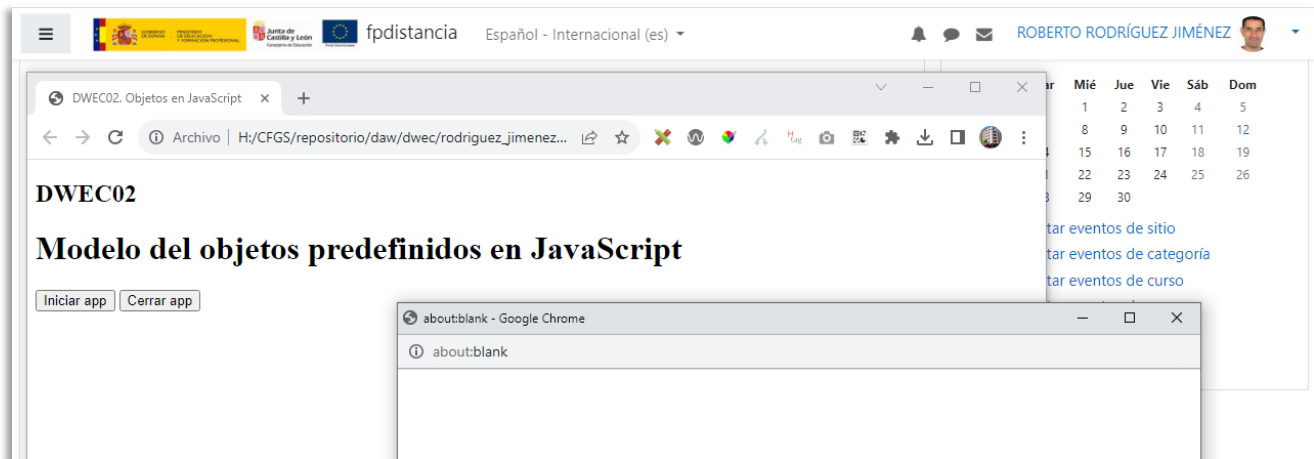
Creación de la función que se pide.	5 puntos.
Petición de los datos.	2 puntos.
Visualización de los mismos.	2 puntos.
Claridad y presentación del código del ejercicio, comentarios en el código y su indentación.	1 punto.

## RESOLUCIÓN

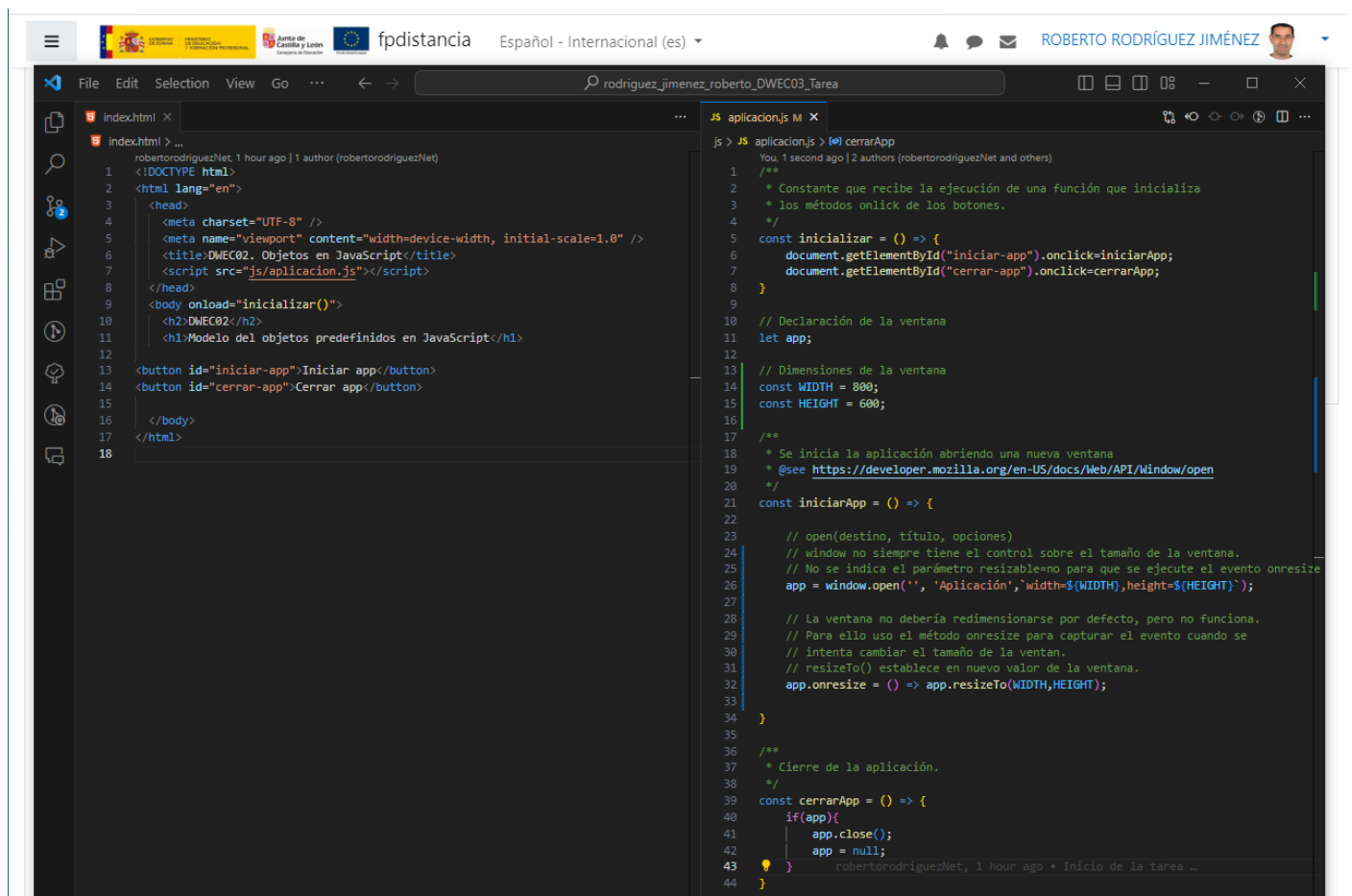
Abra una nueva ventana no redimensionable.

index.html contiene los controles para iniciar y cerrar la aplicación.

js/aplicacion.js es el script que ejecuta la aplicación.



*Página con los botones y ventana nueva abierta.*



*Código para index.html y aplicacion.js*

### Problemas encontrados

El objeto `window` no siempre tiene el control sobre el tamaño de la ventana. Los navegadores dan el control al usuario, que es quien decide si redimensiona o no la ventana.

Las ventanas creadas con `window.open()` no pueden, por defecto, redimensionarse a no ser que se indique con el parámetro `resizable=yes`. También es posible indicar `resizable=no`.

Para mantener el tamaño original, que no evitar el redimensionamiento, se escucha el evento `onresize` y se llama a la función `resizeTo(w,h)` que establece un nuevo valor para la dimensión de la ventana, en este caso, la original.

Sin embargo, si se establece `resizable=no`, se evita el redimensionamiento (aunque no sea efectivo) por lo que no se invoca el evento `onresize` y, por lo tanto, la ventana puede cambiar sus dimensiones.

### Solución

No impedir el redimensionamiento en la construcción de la ventana y capturar ese redimensionamiento para poder devolver el valor mediante `resizeTo()`.

Fuentes:

<https://developer.mozilla.org/en-US/docs/Web/API/Window/open>

<https://es.javascript.info/popup-windows>

```
const inicializar = () => {
  document.getElementById("iniciar-app").onclick=iniciarApp;
  document.getElementById("cerrar-app").onclick=cerrarApp;
}

let app;

const WIDTH = 800;
const HEIGHT = 600;

const iniciarApp = () => {
  app = window.open('', 'Aplicación', `width=${WIDTH},height=${HEIGHT}`);
  app.onresize = () => app.resizeTo(WIDTH,HEIGHT);
}

const cerrarApp = () => {
  if(app){
    app.close();
    app = null;
  }
}
```

*Código para aplicacion.js*



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>DWEC02. Objetos en JavaScript</title>
    <script src="js/aplicacion.js"></script>
  </head>
  <body onload="inicializar()">
    <h2>DWEC02</h2>
    <h1>Modelo del objetos predefinidos en JavaScript</h1>

    <button id="iniciar-app">Iniciar app</button>
    <button id="cerrar-app">Cerrar app</button>

  </body>
</html>
```

*Código para index.html*

*Hacer una función y dentro de esa función:*

Función `crearContenido()`

```
function crearContenido() {

  // Obtener los valores de los objetos.
  // Algunas de estas propiedades, y algún método, están obsoletas (noviembre de 2023).
  let documento = app.document;
  let url = location.href;
  let protocolo = location.protocol;

  // Para obtener la información del navegador se usa userAgent,
  // que contiene la información que da appCodeName() (obsoleta).
  let navegador = navigator.userAgent;

  // javaEnabled() en un método obsoleto.
  // No se recomienda su uso y la alternativa es consultar
  // la compatibilidad del navegador.
  let java = navigator.javaEnabled();

  // Header con el título.
  documento.write("<h3>Ejemplo de nueva ventana</h3>");

  // Lista de definiciones con los datos que se piden.
  documento.write("<dl>");
  documento.write(`<dt>URL:</dt><dd>${url}</dd>`);
  documento.write(`<dt>Protocolo:</dt><dd>${protocolo}</dd>`);
```

```

documento.write(`<dt>Navegador:</dt><dd>${navegador}</dd>`);
documento.write(
    "<dt>Java habilitado:</dt><dd>Java " +
    (java ? "SI" : "NO") +
    " disponible en esta ventana</dd>"
);
documento.write("</dl>");

// iframe solicitando una url que es rechazada.
documento.write(`<iframe
id="iframe-goole"
title="Google"
width="800"
height="600"
src="www.google.es">
</iframe>`);
}

```

Escribir en la nueva ventana <h3>Ejemplo de Ventana Nueva</h3>

```
documento.write("<h3>Ejemplo de nueva ventana</h3>");
```

URL Completa: XXXXX

```
documento.write(`<dt>URL:</dt><dd>${url}</dd>`);
```

Protocolo utilizado: XXXXX

```
documento.write(`<dt>Protocolo:</dt><dd>${protocolo}</dd>`);
```

Nombre en código del navegador: XXXXX

```
documento.write(`<dt>Navegador:</dt><dd>${navegador}</dd>`);
```

Que detecte si está JAVA disponible

```

documento.write(
    "<dt>Java habilitado:</dt><dd>Java " +
    (java ? "SI" : "NO") +
    " disponible en esta ventana</dd>"
);

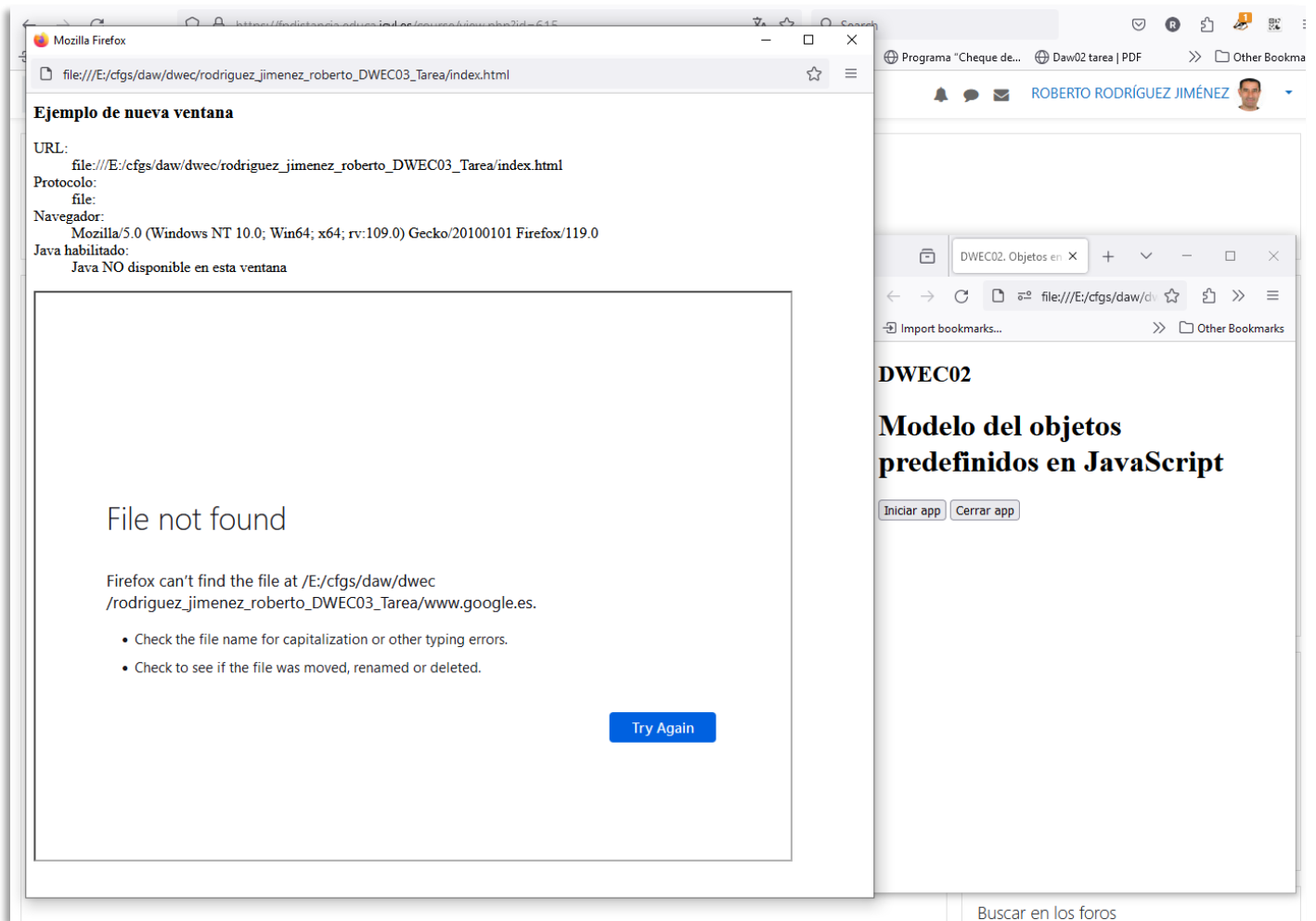
```

Que abra un iframe con el contenido de www.google.es y de 800x600.

```

documento.write(`<iframe
id="iframe-goole"
title="Google"
width="800"
height="600"
src="www.google.es">
</iframe>`);
}

```



Ejecución de la primera parte del ejercicio

Y ahora fuera del código de la función que siga haciendo lo siguiente:

Delimitamos es código JavaScript

```
<!DOCTYPE html>
. . .
<body onload="inicializar()">

  <script type="text/JavaScript">
    . . .
  </script>
  . . .
</body>
</html>
```

Que escriba en la ventana principal <h1>TAREA DWEC03</H2><HR />

```
document.write("<h1>TAREA DWEC03</h1>")
```

Que solicite: introduzca su nombre y apellidos.

```
nombre_apellidos = window.prompt("Escriba su nombre y apellidos");
```

Que solicite: introduzca DIA de nacimiento.

```
dia = parseInt(window.prompt("Día de nacimiento"));
```

Que solicite: introduzca MES de nacimiento.

```
mes = parseInt(window.prompt("Mes de nacimiento (número)"));
```

Que solicite: introduzca AÑO de nacimiento.

```
anno = parseInt(window.prompt("Año de nacimiento"));
```

Obtenemos un objeto de tipo Date a partir de los datos de la fecha para poder trabajar mejor con ella.

```
fecha = new Date(anno, mes, dia);
```

El código HTML se guarda en un solo bloque y se le asigna una variable.

Este código va recogiendo los diferentes datos que se piden.

Una vez solicitados esos datos imprimirá en la ventana principal:

Buenos días XXXXX

```
html = `Buenos días ${nombre_apellidos}.
```

Esta línea muestra directamente el valor recogido por el prompt. Si el campo está en blanco, no se muestra nada.

*Tu nombre tiene XX caracteres, incluidos espacios.*

```
Tu nombre tiene ${nombre_apellidos.length} caracteres, incluidos espacios.
```

Para mostrar los caracteres accedemos a la propiedad `length` de los objetos `String`.

*La primera letra A de tu nombre está en la posición: X*

```
La primera letra A de tu nombre está en la posición ${nombre_apellidos.indexOf('a') +1}.
```

*La última letra A de tu nombre está en la posición: X*

```
La última letra A de tu nombre está en la posición: ${nombre_apellidos.lastIndexOf('a') +1}.
```

El índice de la posición de los caracteres en un `String` comienza por 0, pues no deja de ser un array de caracteres, por lo que se le debe sumar 1 para obtener la posición real.

*Tu nombre menos las 3 primeras letras es: XXXXXXXX*

```
Tu nombre menos las 3 primeras letras es: ${nombre_apellidos.substring(3)}.
```

A `substring(n)` debemos pasarle la posición desde la que comenzar a contrar.

*Tu nombre todo en mayúsculas es: XXXXXXXX*

```
Tu nombre todo en mayúsculas es: ${nombre_apellidos.toUpperCase()}.
```

*Tu edad es: XX años.*

```
Tu edad es: ${isNaN(getEdad(fecha))? ' -- ' : getEdad(fecha) } años.
```

Para obtener el año se ha creado una función que devuelve el año a partir de una cantidad de milisegundos que son el resultado de restar dos fechas: la de nacimiento de la de el momento actual.

Para poder restar las fechas n milisegundos, estas deben ser objetos del tipo `Date`.

```
/**
 * @param Date fecha es la fecha en tipo Date
 * @return int el año
 */
function getEdad(fecha){

    // Obtener la edad en milisegundos
    // Date.now() obtiene el momento actual en milisegundos
    // edadEnMilisegundos es un objeto Date con el poder trabajar con fechas.
    let edadEnMilisegundos = new Date( Date.now() - fecha );

    // Devolvemos los años entre la edadEnMilisegundos y 1970.
    // El 1/1/1970 se toma como en inicio de la cuenta del tiempo (Timestamp)
    return Math.abs(edadEnMilisegundos.getUTCFullYear() - 1970);
}
```

*Naciste un feliz XXXXXX del año XXXX.*

```
Naciste un feliz ${typeof semana[fecha.getDay()] === 'undefined'? ' -- ' :
semana[fecha.getDay()]} del año ${isNaN(anno)? ' ---- ' : anno}.
```

Como ya tenemos el objeto de tipo Date *fecha*, resulta sencillo obtener datos a través de sus métodos.

Ya que `getDay()` devuelve el número del día de la semana, he declarado un array con los nombres de los días, ordenados dependiendo de la configuración del primer día de la semana.

```
// Días de la semana
const semana = ["jueves", "viernes", "sábado", "domingo", "lunes", "martes", "miércoles"];
```

*El coseno de 180 es: XXXXXXXXXXXX*

```
El coseno de 180 es: ${Math.cos(180)}.
```

*El número mayor de (34,67,23,75,35,19) es: XX*

```
El número mayor de 6(34,7,23,75,35,19) es: ${Math.max(34,7,23,75,35,19)}
```

*Ejemplo de número al azar: XXXXXXXXXXXX*

```
Ejemplo de número al azar: ${parseInt(Math.random() * 1000)}
```

El `Math.random()` devuelve un número entre 0 y 1, por lo que aparece un número con muchos decimales, es por ello por lo que multiplica por 1.000 y se convierte a entero.

## Resultado

The screenshot shows a web application interface with a sidebar on the left and a main content area on the right. The sidebar contains a navigation menu with links like 'Página Principal', 'Mis cursos', and 'Desarrollo Web'. The main content area displays the results of a task, including a greeting, user name, age, birth date, and various calculations. The browser window shows the URL 'file:///E:/cfigs/daw/dwec/roc' and the user 'ROBERTO RODRÍGUEZ JIMÉNEZ'.

**TAREA DWEC03**

Buenos días .  
 Tu nombre tiene 0 caracteres, incluidos espacios.  
 La primera letra A de tu nombre está en la posición 0.  
 La última letra A de tu nombre está en la posición: 0.  
 Tu nombre menos las 3 primeras letras es: .  
 Tu nombre todo en mayúsculas es: .  
 Tu edad es: 54 años.  
 Naciste un feliz domingo del año 1969.  
 El coseno de 180 es: -0.5984600690578581.  
 El número mayor de 6(34,7,23,75,35,19) es: 75  
 Ejemplo de número al azar: 502

Buttons:

*Captura de ejecución de la segunda parte de la tarea*

## Código

index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>DWECE02. Objetos en JavaScript</title>
    <script src="js/aplicacion.js"></script>
  </head>
  <body onload="inicializar()">
    <script type="text/JavaScript">

      /**
       * @param Date fecha es la fecha en tipo Date
       * @return int el año
       */
      function getEdad(fecha){

        // Obtener la edad en milisegundos
        // Date.now() obtiene el momento actual en milisegundos
        // edadEnMilisegundos es un objeto Date con el poder trabajar con fechas.
        let edadEnMilisegundos = new Date( Date.now() - fecha );

        // Devolvemos los años entre la edadEnMilisegundos y 1970.
        // El 1/1/1970 se toma como en inicio de la cuenta del tiempo (Timestamp)
        return Math.abs(edadEnMilisegundos.getUTCFullYear() - 1970);
      }

      // Dias de la semana
      const semana = ["jueves","viernes","sábado", "domingo", "lunes", "martes",
"miércoles"];

      // Variable que se utiliza como un comodín para parsear los datos
      // recogidos del prompt
      let data;

      // Variable para html que se mostrará
      let html;

      // Para la fecha de tipo Date
      let fecha;

      // Variables para los datos solicitados.
      let nombre_apellidos;
      let dia;
      let mes;
      let anno;
```

```

document.write("<h1>TAREA DWEC03</h1>");

// Solicitud de datos
nombre_apellidos = window.prompt("Escriba su nombre y apellidos");
dia = parseInt(window.prompt("Día de nacimiento"));
mes = parseInt(window.prompt("Mes de nacimiento (número)"));
año = parseInt(window.prompt("Año de nacimiento"));
fecha = new Date(año, mes, dia);

html = `Buenos días ${nombre_apellidos}.
<br>
Tu nombre tiene ${nombre_apellidos.length} caracteres, incluidos espacios.
<br>
La primera letra A de tu nombre está en la posición ${nombre_apellidos.indexOf('a')
+1}.
<br>
La última letra A de tu nombre está en la
posición: ${nombre_apellidos.lastIndexOf('a') +1}.
<br>
Tu nombre menos las 3 primeras letras es: ${nombre_apellidos.substring(3)}.
<br>
Tu nombre todo en mayúsculas es: ${nombre_apellidos.toUpperCase()}.
<br>
Tu edad es: ${isNaN(getEdad(fecha))? ' -- ' : getEdad(fecha) } años.
<br>
Naciste un feliz ${typeof semana[fecha.getDay()] === 'undefined'? ' -- ' :
semana[fecha.getDay()]} del año ${isNaN(año)? ' ---- ' : año}.
<br>
El coseno de 180 es: ${Math.cos(180)}.
<br>
El número mayor de 6(34,7,23,75,35,19) es: ${Math.max(34,7,23,75,35,19)}
<br>
Ejemplo de número al azar: ${parseInt(Math.random() * 1000)}
`;

document.write(html);

</script>

<div style="margin-top: 20px">
  <button id="iniciar-app">Iniciar app</button>
  <button id="cerrar-app">Cerrar app</button>
</div>
</body>
</html>

```



aplicacion.js

```
// Declaración de la ventana
let app;

/**
 * Constante que recibe la ejecución de una función que inicializa
 * los métodos onclick de los botones.
 */
const inicializar = () => {
  document.getElementById("iniciar-app").onclick = iniciarApp;
  document.getElementById("cerrar-app").onclick = cerrarApp;
};

// Dimensiones de la ventana
const WIDTH = 900;
const HEIGHT = 900;

/**
 * Se inicia la aplicación abriendo una nueva ventana
 * @see https://developer.mozilla.org/en-US/docs/Web/API/Window/open
 */
const iniciarApp = () => {
  // Cerramos la ventana si ya tenemos una instancia
  if (app != null) {
    app.close();
  }

  // open(destino, título, opciones)
  // window no siempre tiene el control sobre el tamaño de la ventana.
  // No se indica el parámetro resizable=no para que se ejecute el evento onresize
  app = window.open("", "Aplicación", `width=${WIDTH},height=${HEIGHT}`);

  // La ventana no debería redimensionarse por defecto, pero no funciona.
  // Para ello uso el método onresize para capturar el evento cuando se
  // intenta cambiar el tamaño de la ventana.
  // resizeTo() establece en nuevo valor de la ventana.
  app.onresize = () => app.resizeTo(WIDTH, HEIGHT);

  // Llamar a la función que creará el contenido
  crearContenido();
};

/**
 * Cierre de la aplicación.
 */
const cerrarApp = () => {
  if (app) {
    app.close();
    app = null;
  }
};
```

```
    }  
};  
  
/**  
 * Función para dar contenido a la ventana abierta  
 */  
function crearContenido() {  
  
    // Obtener los valores de los objetos.  
    // Algunas de estas propiedades, y algún método, están obsoletas (noviembre de 2023).  
    let documento = app.document;  
    let url = location.href;  
    let protocolo = location.protocol;  
  
    // Para obtener la información del navegador se usa userAgent,  
    // que contiene la información que da appCodeName() (obsoleta).  
    let navegador = navigator.userAgent;  
  
    // javaEnabled() en un método obsoleto.  
    // No se recomienda su uso y la alternativa es consultar  
    // la compatibilidad del navegador.  
    let java = navigator.javaEnabled();  
  
    // Header con el título.  
    documento.write("<h3>Ejemplo de nueva ventana</h3>");  
  
    // Lista de definiciones con los datos que se piden.  
    documento.write("<dl>");  
    documento.write(`<dt>URL:</dt><dd>${url}</dd>`);  
    documento.write(`<dt>Protocolo:</dt><dd>${protocolo}</dd>`);  
    documento.write(`<dt>Navegador:</dt><dd>${navegador}</dd>`);  
    documento.write(  
        "<dt>Java habilitado:</dt><dd>Java " +  
            (java ? "SI" : "NO") +  
            " disponible en esta ventana</dd>"  
    );  
    documento.write("</dl>");  
  
    // iframe solicitando una url que es rechazada.  
    documento.write(`<iframe  
id="iframe-goole"  
title="Google"  
width="800"  
height="600"  
src="www.google.es">  
</iframe>`);  
}
```

## Recursos

Documentar JavaScript (jsdoc)

<https://www.gradiweb.com/es/marketing-digital/documentacion-de-codigo-para-javascript-con-jsdoc/>

Manual JavaScript

<https://www.w3schools.com/js/default.asp>

Referencias sobre JavaScript

<https://developer.mozilla.org/es/>