A yellow square containing the letters 'JS' in a bold, black, sans-serif font.

Desarrollo web en entorno cliente

UD – 3

Objetos predefinidos y Arrays

Actividades



1. JERARQUÍA DE OBJETOS NATIVOS DE JAVASCRIPT

2. OBJETOS IMPORTANTES

2.1 Windows

Act 2.1 Crea un buscador para que tus visitantes puedan encontrar un texto cualquiera en tus páginas. Pide al usuario que introduzca la cadena a buscar. Permitir las opciones de:

- Búsqueda de palabras completas.
- Distinguir entre mayúsculas y minúsculas

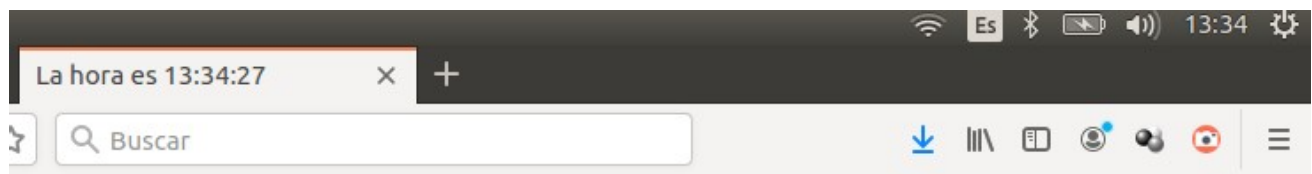
Texto a buscar	<input type="text"/>	<input type="button" value="Enviar"/>	Buscar solo palabras completas <input type="checkbox"/>	Distinguir entre mayusculas y minusculas <input type="checkbox"/>
----------------	----------------------	---------------------------------------	---	---

Act 2.2 Realiza un programa que cada 20 segundos (**mediante setInterval**) solicite un DNI hasta que alguien le escriba la cadena “-1”. En ese momento, el programa debe mostrar seguidas las letras de todos los DNIs introducidos. Aquí un enlace para saber como calcular la letra de DNI:

https://es.wikipedia.org/wiki/N%C3%BAmero_de_identificaci%C3%B3n_fiscal

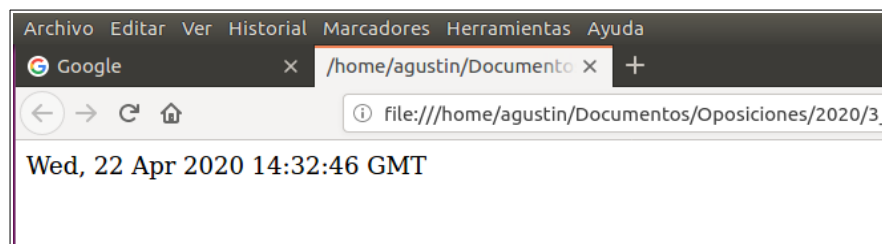
2.2 Document

Act 2.3 Establece un título del documento (document.title) de forma dinámica, mostrando cada segundo la hora actual, como vemos en la siguiente figura: Añade al documento un enlace que permita parar la hora:
[**Date** + **document.tittle**]



2.3 Generación de elementos HTML desde código

Act 2.4 Realizar un programa que pasados 20 segundos, nos muestre una vez la fecha actual del sistema.
[**SetTimeout** + **Date** + **Document.write**]



3. GESTIÓN DE VENTANAS

4. ARRAYS

Act 4.1 Crear un programa que permita introducir 10 nombres de los socios de un club (utilizar la estructura de array). Los elementos deben leerse de pantalla utilizando el método prompt(). En el mensaje de salida que permite la introducción de los nombres debe indicarse el número de elemento que se solicita. Después de introducir todos los elementos se tiene que mostrar la siguiente información por pantalla:

- mostrar los nombres de los socios tal y como se han introducido
- indicar el número socios que hay
- mostrar el nombre de los socios ordenados por orden alfabético
- mostrar el nombre de los socios en orden inverso al introducido

Act 4.2 Crear un script en el que se pregunte por el número de alumnos y en función de este dato pida el nombre de los alumnos y sus notas finales. Al final se debe mostrar, cada alumno con su nota y después la media de todos los alumnos.

Act 4.3 Crear un array para los nombres y otro para las edades. Introducir en el elemento 0 del primer array un nombre y en el elemento 0 del segundo array su edad y así sucesivamente hasta 6 nombres. Crear un enlace que permita mostrar por pantalla cada nombre con su edad y otro para ver la media de todas las edades. Utilizar una función para calcular la media de las edades.

Act 4.4 Añadir, al final del ejercicio anterior, un enlace que de la posibilidad de introducir un nombre por pantalla y conocer su edad si existe y si no un mensaje de advertencia.

Act 4.5 Completar el ejercicio anterior con la presentación de dos enlaces en pantalla con las siguientes opciones:

- Borrar un socio buscando su contenido
- Borrar un socio localizado por su índice

5. MAP

6. SET

7. NUMBER

8. MATH

Act 8.1 Hacer un programa que muestre un menú de opciones con las siguientes operaciones posibles:

- 1. Redondeo a su inmediato superior de un número
- 2. Redondeo a su inmediato inferior de un número
- 3. Redondeo al número más cercano
- 4. Cálculo del número máximo entre dos dados
- 5. Cálculo del número mínimo entre dos dados
- 6. Cálculo de un número elevado a otro
- 7. Cálculo de la raíz cuadrada de un número

El programa debe pedir una opción de las mostradas y en función de ello pedir los datos necesarios para realizar la operación y mostrar el resultado. Cada opción del menú debe ser ejecutada por una función diferente la cual debe pedir los datos y ejecutar las operaciones necesarias para obtener el resultado. El resultado debe ser devuelto por la función y se muestra en el programa principal.

9. STRING

Act 9.1 Solicitar una frase desde teclado, almacenar cada palabra en un array y mostrar los siguientes datos:

- La primera palabra de la frase.
- La última palabra de la frase.
- El número de palabras de la frase.



Act 9.2 Solicitar una cadena desde teclado con la siguiente estructura:

`"nombre:apellidos:telefono:email@servidor.codigopostal"`

Mostrar la información en la página principal:




Imagen del personaje

Información

- Nombre:
- Apellidos:
- Teléfono:
- Email:
- País:

Esta página dice

Mete cadena

Act 9.3 : Hacer un script que pregunte por el número de socios y pida, una vez por cada socio y en ventanas diferentes, el nombre y la dirección de correo electrónico. Cada vez que lea el dato del correo electrónico, una función deberá comprobar que el dato se corresponde con un dirección e-mail utilizando la función *indexOf()*. La función deberá devolver el valor *true* si la dirección tiene una *@* y *false* si no la tiene. En el caso de que sea una dirección electrónica debe almacenar el nombre y la dirección en un elemento de un array y si no debe pedirse de nuevo hasta que introduzca un dato correcto.

Act 9.4 Hacer un script que pida el número de teléfono y en función del prefijo nos indique si es de Granada(958), Almeria(957), Jaen(956) o Cordoba(955), si es de otro lugar mostrará el siguiente mensaje: "Ese prefijo aún no lo conozco". Se debe utilizar una función que tome por parámetro tres caracteres y nos muestre un mensaje en función de lo indicado anteriormente. Recordar el método *slice(inicio,[final])*.

10. DATE

Act 10.1 Crear un script que muestre un mensaje con la fecha y hora actual pero utilizando el siguiente formato: " Hoy es día 3 de Mayo de 2001 y en estos momentos son las 17:30"