

HTML Y JAVASCRIPT BÁSICO

CUESTIONES Y EJERCICIOS

1. Crea el código HTML de una página web muy básica con el título MI PÁGINA. En el cuerpo aparecerá únicamente tu nombre y apellidos.
2. Estamos modificando una página y tenemos abierto tanto su código fuente como el navegador con la página en sí. Si observamos que los últimos cambios realizados en el código no causan efecto en la página ¿a qué puede deberse? (Se supone que el código es correcto)
3. ¿Cuáles son las dos partes elementales de una página? ¿Cuál de ellas es la que alberga todos los contenidos importantes? (textos, imágenes, vínculos, etc.)
4. Escribe la etiqueta completa para definir una página cuyo fondo sea de color verde y el texto de color amarillo con tamaño 20 puntos y fuente Courier New.
5. Escribe el código que habría que añadir al estilo anterior para establecer en la página una imagen de fondo que reside en un archivo llamado fondo.jpg, encontrándose dicho archivo en una subcarpeta llamada IMÁGENES.
Sólo esta parte del código, no la etiqueta completa.
De aquí en adelante cuando mencionemos **subcarpeta** estaremos haciendo referencia a otra carpeta que se encuentra en la misma carpeta en la que alojamos el archivo HTML cuyo código estamos considerando.
6. Si has establecido una imagen de fondo en la página entonces el color de fondo ya no tiene sentido, pues en ningún caso éste será ya visible. ¿Es correcta esta afirmación? Razona la respuesta.
7. Si el cuerpo de una página se define sin formato alguno, escribiendo simplemente <BODY> sin parámetros, ¿Cuál será su estilo cuando el cliente abra ésta desde el navegador? (color de fondo, color, tamaño y fuente del texto)
8. Teniendo en cuenta lo anterior ¿En qué caso se le muestra al cliente la página con el formato establecido en su navegador?
9. Escribe el código necesario para insertar el siguiente texto en el cuerpo de una página:

Primer texto

Segundo texto

Tercer texto

(Escribe el código en una sola línea)

10. Escribe el código necesario para insertar el siguiente texto en el cuerpo de una página:

Frase separada

(Entre ambas palabras queremos dejar ocho espacios. Escribe el código en una sola línea)

11. Escribe el código necesario para insertar una imagen en el cuerpo de la página. La imagen se encuentra en un archivo llamado **paisaje.png** dentro de una subcarpeta llamada **imágenes**.

(No confundir **imagen de fondo**, que sólo puede haber una en la página, con **imagen insertada**, que puede haber tantas como se deseen en la página)

12. Amplía la respuesta anterior para insertar esa misma imagen con una anchura de 200 puntos y una altura de 150 puntos, mostrando el texto **“Texto emergente”** cada vez que el puntero se sitúe sobre dicha imagen en la página.

13. ¿Se puede tener texto e imagen en una misma línea del cuerpo una página? Pon un ejemplo sencillo (código).

14. Si en una página insertamos varias imágenes de forma consecutiva sin separarlas mediante la etiqueta
:

```
<IMG SRC="..."> <IMG SRC="..."> <IMG SRC="...">
```

¿Aparecerán siempre todas ellas en la misma línea? Razona la respuesta.

15. ¿Cuál es el parámetro más importante de la etiqueta IMG y por qué?

16. Si en el parámetro SRC de la etiqueta indicamos un nombre de archivo (o ruta) inexistente ¿Qué aparecerá en la página?

17. Escribe el código necesario para que en el cuerpo de una página aparezca lo siguiente:

Haz clic [aquí](#) para abrir otra página.

Es un vínculo a una página local llamada otra.htm, que se encuentra en la misma carpeta.

18. Repite el código anterior (sólo la etiqueta) para conseguir que la página vinculada se abra en una pestaña nueva (o en una página nueva, dependiendo del navegador).

19. ¿Qué ocurre si en el código de un vínculo se omite el cierre ? ¿Seguirá funcionando el vínculo? ¿De qué manera? Explícalo con claridad.

20. Escribe el código necesario para que en el cuerpo de una página aparezca lo siguiente:

Acceso a la web del [Padre Suárez](#)

Es un vínculo a una página remota.

21. Escribe el código necesario para que en el cuerpo de una página aparezca un vínculo hacia la web del Granada C.F., de manera que el vínculo no sea desde un texto sino desde una imagen residente en un archivo llamado **escudo.jpg**.

22. El archivo que se especifica en el parámetro HREF de la etiqueta <A> ¿Debe ser obligatoriamente un archivo de página web, con extensión HTM, HTML o similar? Razona y explica con claridad la respuesta.

23. Si el cliente hace clic en un vínculo hacia un archivo y no dispone de la aplicación necesaria para abrir éste ¿Qué sucederá?

24. Escribe el código para definir un **marcador** (con el nombre que desees) en un lugar concreto de la página, de manera que pueda accederse directamente a dicho lugar desde otras partes de la misma página mediante hipervínculos.

25. Teniendo en cuenta el marcador definido en el ejercicio anterior, escribe el código de un hipervínculo que dé acceso a él en la página.

26. En una página queremos un acceso directo desde el inicio hacia el final y otro desde el final hacia el inicio, de manera que ambos marcadores ejercen también de vínculos. Inserta los códigos necesarios, decidiendo tú los nombres de los marcadores:

```
<BODY>
```

... Todos los contenidos...

</BODY>

27. ¿Cuántos marcadores pueden establecerse en una página? ¿Qué hay que tener en cuenta con los nombres que se le asignen?
28. Escribe un hipervínculo desde una miniatura de una imagen hacia la misma imagen en su tamaño normal. La miniatura debe aparecer en la página con unas dimensiones de 100x100 puntos. Inventa el nombre del archivo de imagen.
29. Escribe el código completo para definir un estilo personalizado en una parte del texto de una página. Por ejemplo:
El siguiente texto tiene un estilo propio que no es el estilo global de la página
30. ¿Qué textos en una página adoptan el estilo definido en el parámetro STYLE de la etiqueta <BODY>?
31. Escribe el código de un texto breve en negrita utilizando etiqueta doble sencilla (sin utilizar ninguna etiqueta que admita el parámetro **style**).
32. Haz lo mismo con un texto breve en cursiva (itálica).
33. Haz lo mismo con un texto breve subrayado.
34. Las etiquetas , <i></i>, <u></u> son muy fáciles de utilizar para efectos en la fuente, pero su uso tiene un inconveniente. ¿Cuál?
35. Repasa las siguientes propiedades del parámetro STYLE y aprende los posibles valores que cada una puede admitir (especialmente en el caso de las cuatro últimas) y los valores por defecto en cada caso:

COLOR
FONT-SIZE
FONT-FAMILY
BACKGROUND-COLOR
BACKGROUND-IMAGE
FONT-WEIGHT
FONT-STYLE
TEXT-DECORATION

36. Repasa cómo se menciona cada una de las propiedades anteriores en código Javascript. Por ejemplo, **objeto.style.fontSize** para el caso del tamaño de fuente.
37. Escribe el código completo para insertar la siguiente tabla en una página, de manera que aparezca centrada horizontalmente ocupando la mitad de la anchura total de la ventana:

Uno	Dos	Tres
Cuatro	Cinco	Seis

38. El parámetro BORDER de la etiqueta <TABLE> ¿Qué tipos de valores acepta? ¿Cuál es su valor por defecto? ¿Qué indica este valor?
39. ¿Qué define el parámetro ALIGN de la etiqueta <TABLE>? ¿Qué valores acepta? ¿Cuál es el valor por defecto?

40. Si se establece `<TABLE ALIGN="CENTER">` ¿se conseguirá que en todas sus celdas cada contenido aparezca centrado? Razona la respuesta.
41. En algunos casos la tabla aparece con más anchura de la especificada en el parámetro `WIDTH`. ¿Cuáles? Explícalo con claridad.
(Se puede decir lo mismo en relación a la altura de la tabla y al parámetro `HEIGHT`)
42. En el parámetro `WIDTH` se puede establecer una medida **absoluta** o **relativa**. Pon un ejemplo de cada caso y explícalos. No es necesario que escribas el código de tablas completas, tan sólo `<TABLE WIDTH=...>` en cada ejemplo y comentarlo.
(Se puede decir lo mismo en relación a la altura de la tabla y al parámetro `HEIGHT`)
43. Escribe el código que habría que modificar en la tabla anterior para que la celda con el texto "Cinco" muestre dicho texto centrado en la celda. Escribe solamente el código correspondiente a dicha celda.
44. Escribe el código que habría que modificar en la tabla anterior para que todas las celdas de la primera fila aparezcan cada una de ellas con su texto centrado. Escribe solamente el código correspondiente a dicha fila y de la manera más sencilla posible, sin hacerlo manualmente en cada una de sus celdas.
45. El parámetro `VALIGN`, que puede utilizarse en la etiqueta `<TD>` (pero no en la etiqueta `<TABLE>`) ¿Qué establece en una celda? ¿Qué valores admite? ¿Cuál es el valor por defecto?
46. El parámetro `VALIGN` ¿Puede utilizarse en la etiqueta `<TR>`? ¿Qué establece en este caso?
47. Las etiquetas `<TABLE>`, `<TR>` y `<TD>` admiten el parámetro `STYLE` con todas sus propiedades. ¿Qué efecto tiene el estilo establecido en cada caso?
48. Crea la siguiente tabla irregular con celdas combinadas. En este ejercicio lo menos importante es el estilo. Recuerda que los parámetros `COLSPAN` y `ROWSPAN` sólo son aplicables a la etiqueta `<TD>`:

EINEM	AW	SEGIN
SIOPR		HLC
Recreo		
SEGIN	AW	

49. Utilizando una tabla sin borde, escribe el código para conseguir dos textos en la misma línea de manera que el primero quede alineado a la izquierda y el segundo a la derecha. Cada uno de ambos textos debe estar alineado siempre, independientemente del tamaño establecido por el cliente para la ventana.
50. Queremos utilizar una tabla para crear una galería de miniaturas, de forma que cada celda contenga una miniatura que a su vez sea enlace a la correspondiente imagen a su tamaño original. Crea el código completo de una celda, inventa el nombre del archivo de imagen y el tamaño de la miniatura. No olvides las etiquetas propias de la celda.
51. Explica la diferencia entre página estática y página dinámica.
52. Para que una página sea dinámica es necesario añadirle scripts. ¿Qué es un script?

53. Cada script puede asociarse a la ejecución de un **evento** en la página. ¿Qué es un evento?
54. Cita algunos eventos usuales (los vistos en clase hasta ahora). Escribe el parámetro HTML asociado en cada uno de ellos.
55. ¿En qué parte de una página se definen sus funciones internas? ¿Qué etiquetas HTML son necesarias para ello? Responde con claridad.
56. El código que se redacta entre las etiquetas <SCRIPT> </SCRIPT> de la cabecera, ¿corresponde a lenguaje HTML?
57. Escribe el código completo de una función sin parámetros llamada **saludo()**, que sólo se encarga de mostrar un cuadro de diálogo con el texto “Hola”.
(En lo sucesivo si no se especifica otra cosa se supone que la función solicitada se pide sin parámetros)
58. Escribe el código completo de un objeto **button** en el cual aparezca el texto OK, de manera que al hacer clic sobre él se ejecutará la función anterior.
59. En el código del ejercicio anterior, ¿qué parte del código corresponde a HTML y qué parte a Javascript? Señala ambas con claridad.
60. El evento ONCLICK ¿puede utilizarse únicamente en objetos de tipo **button**? Razona la respuesta.
61. Crea un objeto de tipo text en el cual aparezca escrito por defecto el texto “Escribe tu nombre”. Al hacer clic sobre este mismo objeto se ejecutará una función llamada borrar() que se encargará de vaciar el su valor. No olvides asignar un identificador (id) al objeto.
 - Escribe el código del objeto.
 - Escribe el código de la función.
62. Escribe el código de una función que se encargue de modificar el estilo del objeto **text** (no su valor) que creaste en el ejercicio anterior. Cuando esta función sea ejecutada el color del texto de este objeto se establecerá en azul y su color de fondo en amarillo. Inventa un nombre para la función.
(Sólo para dicho objeto, la función se define sin parámetros)
63. En relación a lo anterior, escribe el código de un botón que se encargue de ejecutar la función anterior al hacer clic sobre él.
64. En una página queremos establecer funciones que modifiquen el estilo global del **cuerpo** de la página, por lo tanto debemos asignarle un identificador. Escribe el código básico de la etiqueta <BODY> de esta página sin preocuparte por su estilo inicial.
65. Teniendo en cuenta el nombre que acabas de asignar en la etiqueta <BODY>, escribe el código de una función (inventa su nombre) que establezca color de fondo, color de texto, tamaño de texto y fuente que tú quieras para el cuerpo completo.
66. En relación a lo anterior, escribe el código de un botón que se encargue de ejecutar la función anterior al hacer clic sobre él.
67. Queremos definir un archivo de funciones externas en Javascript llamado **misfunciones**. Explica cómo debemos hacerlo: tipo de archivo, qué contenido debe tener y qué extensión debemos darle.
68. Escribe el código que hay que añadir en una página que hace uso de las funciones definidas en el archivo externo anterior.

69. ¿Puede una página hacer uso de dos o más archivos externos? ¿Cómo sería el código? Escríbelo (inventa el nombre de los archivos externos).
70. Escribe el código completo de una función llamada **ponernegrita()**. Esta función lleva un parámetro, correspondiente al objeto sobre el cual se desea aplicar el estilo negrita. Esta función podría definirse en un archivo externo **js**.
71. Escribe el código completo de una función llamada **quitarnegrita()**. Esta función lleva un parámetro, correspondiente al objeto sobre el cual se desea restablecer el estilo normal. Esta función podría definirse en un archivo externo **js**.
72. Escribe el código completo de un texto que aparece de forma estática “sin negrita”, de manera que al pasar el puntero sobre él se le aplica la negrita, volviendo a su estado normal al retirar el puntero. Haz uso de las dos funciones anteriores. Recuerda que el objeto hay que pasarlo como parámetro en la llamada de cada función.
(Pon un texto pequeño, eso es lo de menos)
73. En el código anterior se puede haber definido un **id** para el objeto. ¿Es obligatorio hacerlo en este caso? En caso de omitirlo ¿qué hay que hacer para mencionar el objeto en el código Javascript?
74. Define ahora una función única más “potente” que puede sustituir a las dos anteriores. Vamos a llamarla simplemente **negrita()** y servirá tanto como para aplicar negrita como para quitarla. Veamos ejemplos de llamadas:
negrita(mitexto,“bold”) establece que el objeto **mitexto** se pone en negrita.
negrita(this,“normal”) establece que el “mismo” objeto se pone en negrita.
Escribe el código de esta función. ¿Cuántos parámetros tiene?
75. Repite el ejercicio 72 haciendo únicamente llamadas a la función **negrita()**, tanto para aplicar como para restablecer.
76. Parecido al 74: Define una función única llamada **cursiva()** con dos parámetros. El primero de ellos hace referencia al objeto al que se desea aplicar (o quitar) estilo de cursiva. El segundo establece el valor a aplicar.
77. Escribe el código completo de un texto que aparece de forma estática “sin cursiva”, de manera que al pasar el puntero sobre él se le aplica la cursiva, volviendo a su estado normal al retirar el puntero. Haz uso de la función única anterior. Recuerda que en cada llamada a la función hay que especificar dos parámetros específicos.
(Pon un texto pequeño, eso es lo de menos)
78. Parecido al 74 y al 76: Define una función única llamada **subrayado()** con dos parámetros. El primero de ellos hace referencia al objeto al que se desea aplicar (o quitar) estilo subrayado. El segundo establece el valor a aplicar.
79. Escribe el código completo de un texto que aparece de forma estática “sin subrayar”, de manera que al pasar el puntero sobre él se le aplica el estilo subrayado, volviendo a su estado normal al retirar el puntero. Haz uso de la función única anterior. Recuerda que en cada llamada a la función hay que especificar dos parámetros específicos.
(Pon un texto pequeño, eso es lo de menos)
80. Escribe el código de una función llamada **estilo()** que tome cinco parámetros:
Objeto al que se aplicará un estilo.
Valor correspondiente a un color de texto.
Valor correspondiente a un color de fondo.
Valor correspondiente a un tamaño para el texto.

Valor correspondiente a un tipo de fuente para el texto

La función se encarga de aplicar sobre el objeto (indicado en el primer parámetro) el estilo indicado en los cuatro últimos parámetros. Un ejemplo de llamada a esta función sería:

estilo(mitexto,"red","pink","20pt","Century Gothic")

(que aplica las cuatro propiedades a un objeto que en este caso se llama **mitexto**)

81. Un texto está definido de forma estática de color verde, fondo amarillo, tamaño 14pt y fuente Comic Sans MS. Al acercar el puntero queremos que tome color rojo, fondo azul, tamaño 16pt y fuente Arial. Al retirar el puntero volverá al estilo indicado en primer lugar. Escribe el código completo para este objeto utilizando llamadas a la función del ejercicio anterior. No olvides definir el estilo estático para el objeto.
82. ¿Se pueden definir eventos en los objetos INPUT? ¿Y en las etiquetas ? ¿Y en las imágenes ?
83. ¿Se pueden definir eventos en la etiqueta <BODY>? ¿En qué caso se ejecutaría el script asociado a ONCLICK? ¿Y en ONMOUSEOVER?
84. ¿En qué caso se ejecuta el script asociado al evento ONLOAD de la etiqueta <BODY>?
85. Utilizando la función Math.random() de Javascript, escribe el código de una función llamada **aleatorio()** con un parámetro que especifica un objeto. Al ejecutarse la función el valor de dicho objeto tomará un número al azar entre cero y uno, sin incluir éste.
Por ejemplo, una llamada **aleatorio(mitexto)** hace que el valor del objeto **mitexto** tome un valor numérico al azar.
86. ¿Qué devuelve la ejecución de la función Math.random()?
(Prácticamente resuelto en el enunciado del ejercicio anterior)
87. Escribe el código de una función llamada **lanzardado()** con un parámetro que especifica un objeto INPUT de tipo TEXT. Al ejecutarse la función el valor de dicho objeto tomará un número **entero** al azar entre uno y seis, ambos inclusive.
Por ejemplo, una llamada **lanzardado(midado)** hace que el valor del objeto **midado** tome un número entero del 1 al 6.
88. Escribe el código de una función llamada **lanzarbyte()** con un parámetro que especifica un objeto INPUT de tipo TEXT. Al ejecutarse la función el valor de dicho objeto tomará un número **entero** al azar entre **cero** y 255, ambos inclusive.
Por ejemplo, una llamada **lanzarbyte(mibyte)** hace que el valor del objeto **mibyte** tome un número entero del 1 al 6.
89. Redacta de nuevo el código de la función **dado()** antes definida, de manera que tras generar un número del uno al seis se encargará de poner una imagen correspondiente en un objeto de imagen que le será indicado como segundo parámetro. Escribe la función completa de nuevo.
Se asume que las imágenes a cargar se llaman **1.jpg**, **2.jpg**, etc., y que todas ellas se ubican en una subcarpeta llamada **imágenes**.
90. Crea una función llamada **lanzarletra()** con un parámetro INPUT de tipo TEXT. Al ejecutarse la función el valor de dicho objeto tomará una letra al azar.
Por ejemplo, una llamada **lanzarletra(miletra)** hace que el valor del objeto **miletra** tome un valor al azar de la A a la Z.
(Puedes hacerlo con varias sentencias if, pero hay métodos más avanzados)