# CONTROL PRIMERA EVALUACIÓN JAVASCRIPT.

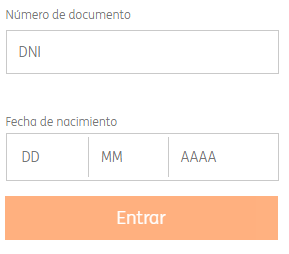
# Curso 2021/2022 PROGRAMACION EN ENTORNO CLIENTE. JAVASCRIPT

**Fecha: 24-11-2021 Duración: 3,45 HORAS**

**Nombre y apellidos: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Realizar un formulario de validación de los clientes registrados o no en una entidad bancaria japonesa llamada MIKEDOCONTO.**

El formulario de registro será el siguiente:



El formulario tendrá la estructura y aspecto del anterior. Se valorará la distribución de los elementos en el formulario. Colocación, colores, tipos de letra, etc.

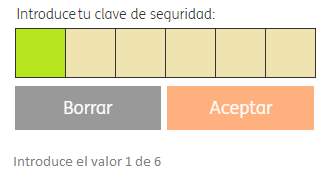
1. Se introducirá el DNI o NIF del cliente **rellenando las nueve cifras, ni más ni menos,** del mismo. Si el DNI tiene solo 8 cifras, el usuario tendrá que introducir un cero al principio. **La letra del NIF se mostrará automáticamente cuando se termine de teclear la novena cifra.**
2.  Se empezará desde el campo día, que es el primero por la izquierda. Solo podemos empezar a introducir la fecha por este campo. Cuando se rellene completamente el campo día, con las dos cifras de la fecha de nacimiento (se incluyen ceros al principio para valores inferiores a 10), se saltará automáticamente al siguiente campo mes, y cuando este esté completo, se saltará al año.

Cuando en el campo año se hayan introducido los 4 caracteres numéricos, es cuando se validará la fecha. Siempre se introducirá cero si el elemento día o mes es inferior a 10. Validar la fecha.

1. Botón que permite acceder a la pantalla en la que se introducirá la contraseña del usuario. Solo realizará la acción adecuada (no estará deshabilitado, no oculto, ni nada similar, simplemente no hará nada hasta que sea necesario) cuando los elementos 1 y 2 estén correctamente tecleados, aunque no correspondan al usuario, pero tienen que ser validos en cuando formato y longitudes

**Valoración:  *(2 puntos).***

**OPCION A: Valoración:  *(3 puntos).***

La siguiente pantalla es la de validación al sistema. Es la misma pantalla que la inicial, salvo que ahora en la zona de **Acceso clientes**, aparece los elementos de **Clave de acceso** para ***loguearse*** al sistema.

Estos elementos se mostrarán mediante InnerHTML eliminando la parte correspondiente a **Acceso clientes**, y mostrando los nuevos elementos.

1. Introduciremos los números correspondientes a la contraseña de usuario, de uno en uno de izquierda a derecha, saltando automáticamente de un campo a otro de la contraseña, solo si lo que se introduce es un número. En este caso, por cada número introducido correctamente, se pondrá de color verde la casilla correspondiente apareciendo un \* en la caja en la que se ha tecleado el número.

Si lo que se introduce es cualquier otra cosa, la casilla se mostrará en color rojo.

Solamente podemos introducir el elemento siguiente, si en el anterior de la izquierda, secuencialmente, se ha tecleado con anterioridad, es decir, no podemos introducir valores en los campos contraseña, empezando, por ejemplo, por el segundo, luego el cuarto y así. Solamente el primero, después el segundo y así hasta el final.

1. Cuando se pulsa este elemento, borraremos el elemento en el que estemos, y si lo volvemos a pulsar, borraremos el anterior y así hasta el primero de los 6 elementos del campo contraseña. En este caso el botón aceptar, no podrá realizar acción alguna (según se indica en el punto 8);
2. En la parte inferior de la pantalla, se irá mostrando el elemento que estamos introduciendo posicionalmente, es decir, se mostrará un mensaje que diga, “***Introduce el valor n de 6*”.**  Este valor se irá incrementando o decrementando, según avancemos al introducir la contraseña o al borrar elementos de esta.
3. Botón que se activará su acción cuando se haya introducido toda la contraseña. Antes de esto, el boton se verá, estará habilitado, activado, no oculto ni nada similar. Simplemente no tendrá acción asociada. Cuando pulsemos el botón aceptar, comprobaremos en la base de datos que los datos introducidos por el cliente son correctos: comprobaremos que la fecha de nacimiento, el DNI y la clave de acceso introducidos se corresponde con los que realmente hay en la base de datos.
4. Para terminar el proceso de validación, se introducirá el email del usuario, pero solamente una vez que hayamos verificado que fecha nacimiento, DNI y contraseña son correctas.
5. Para introducir el email, lo haremos por duplicado y verificaremos que la estructura de entrada de datos del mismo es adecuada. Es decir, que el email sea correcto.
6. Es entonces cuando contrastaremos el email introducido con el de la información que tenemos en el array. Si el email coincide sacaremos por pantalla un mensaje (no puede ser elemento emergente) en **el que se muestre toda la información del usuario introducida y un mensaje que indique que es correcta.**
7. Para grabar el registro en la base de datos, se mandará la consulta o query sql montado desde el JavaScript. Es decir, en PHP solamente tenemos que insertar el registro, pero la cadena INSERT la enviaremos montada desde el JavaScript.

**OPCION B: Valoración:  *(3 puntos).***

La siguiente pantalla es la de validación al sistema. Es la misma pantalla que la inicial, salvo que ahora en la zona de **Acceso clientes**, aparece los elementos del **Capcha** para ***loguearse*** al sistema.

Estos elementos se mostrarán mediante InnerHTML eliminando la parte correspondiente a **Acceso clientes**, y mostrando los nuevos elementos.

Es decir, el acceso al sistema se realizará mediante u8n control control de seguridad. Una vez introducidos todos los elementos anteriores y pulsado el botón , se activará la caja de texto comprobación  con los elementos numéricos mostrados tal y como se indica a continuación.

En esta caja de texto teclearemos los 4 números posicionalmente tal y como se muestran en la imagen en la zona del formulario que hay a tal efecto.

1. En la parte inferior de la pantalla, se irá mostrando el elemento que estamos introduciendo posicionalmente, es decir, se mostrará un mensaje que diga, “***Introduce el valor n de 4*”.**  Este valor se irá incrementando o decrementando, según avancemos al introducir la contraseña o al borrar elementos de esta.
2. Se mostrarán aleatoriamente 4 de las 10 imágenes, pudiéndose repetir como máximo dos veces cada una de ellas.
3.  Tecleado el código del **Captcha** en la caja de texto, se comprobará y entonces será cuando podremos pulsar realmente el botón***.*** Si pulsamos este botón y no hemos introducido los cuatro dígitos del código de validación, simplemente mostraremos un mensaje emergente que avise de tal circunstancia.
4. También avisaremos en el caso de introducir los cuatro elementos, pero que no coincidan con las imágenes mostradas. Estos dos mensajes de error serán diferentes indicando en cada uno de ellos la circunstancia que genera el error.
5. Si todo ha ido correctamente, al pulsar ***,*** se grabará el registro en la base de datos de alumnos con los datos introducidos en el formulario. Se mostrará una pantalla indicando que el registro se ha realizado de forma correcta.
6. Para terminar el proceso de validación, se introducirá el email del usuario, pero solamente una vez que hayamos verificado que fecha nacimiento, DNI y contraseña son correctas.
7. Para introducir el email, lo haremos por duplicado y verificaremos que la estructura de entrada de datos del mismo es adecuada. Es decir, que el email sea correcto.
8. Es entonces cuando contrastaremos el email introducido con el de la información que tenemos en el array. Si el email coincide sacaremos por pantalla un mensaje (no puede ser elemento emergente) en **el que se muestre toda la información del usuario introducida y un mensaje que indique que es correcta.**
9. Para grabar el registro en la base de datos, se mandará la consulta o query sql montado desde el JavaScript. Es decir, en PHP solamente tenemos que insertar el registro, pero la cadena INSERT la enviaremos montada desde el JavaScript.

**OPCION A y B: Valoración:  *1 punto).***

Para realizar todo el proceso de validación, utilizaremos una base de datos con la lista de datos que se pone en el punto 10.

La estructura de la base de datos es : DNI, fecha nacimiento, contraseña, email ***(introducir algún dni correcto para poder validar de forma adecuada)***

***Estos datos se importarán necesariamente desde un archivo CSV a la base de datos.***

11111111X,12-01-1994,888765,correo@domenicoscarlatti.es;

22222222X,12-01-2000,123456,miemail@domenicoscarlatti.es;

333333333X,12-01-1981,543251,correo3@domenicoscarlatti.es;

444444X,12-01-1964,426856,correo\_web@domenicoscarlatti.es;

11121111X,12-01-1944,888765,correo43@domenicoscarlatti.es;

2224222X,12-01-2000,133456,correo223.223@domenicoscarlatti.es;

33317933X,12-01-1951,543271,correo36\_empresa@domenicoscarlatti.es;

477644X,12-01-1964,426856,correo34@domenicoscarlatti.es;

1. **Preparar un proceso que nos permita introducir datos en una matriz bidimensional de cuatro filas por 5 columnas.**
   1. La tabla se creará por **HTML** dentro del archivo es.
   2. Una vez creada la tabla, prepararemos un proceso para rellenar las tres primeras filas, mostrando en pantalla, fila y columna del elemento que estamos introduciendo.
   3. Terminada la introducción de datos en cada una de las filas, cada una de ellas la mostraremos completa en pantalla y seguiremos introduciendo datos en el resto de filas hasta rellenar las tres.
   4. Una vez introducidos los datos en las tres primeras filas, y mostradas en pantalla, las ordenaremos manualmente (no se pueden utilizar métodos ni funciones del objeto array) de mayor a menor (no se pueden utilizar propiedades).
   5. Una vez ordenadas, sumaremos posicionalmente los elementos de cada fila y el resultado lo almacenaremos en la cuarta fila también posicionalmente.
   6. Mostraremos el elemento mayor del resultado de la suma y la posición (fila y columna) que ocupa en el elemento que está ubicado.

**Valoración:  *(2,5 puntos)***

1. **Explica y escribe 5 formas diferentes de dibujar tablas, con filas y columnas en HTML.**

**Valoración:  *(1,5 puntos)***