



MODELO EXAMEN 1ª EVALUACIÓN

Desarrollo Web entorno cliente
CFGS DAW

Sergio García Barea
sergio.garcia@ceedcv.es

2018/2019

Versión:181108.1732


Licencia



Reconocimiento - NoComercial - CompartirIgual (by-nc-sa): No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Nomenclatura

A lo largo de este tema se utilizarán distintos símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

 Importante

 Atención

 Interesante

ÍNDICE DE CONTENIDO

1.Boletín de ejercicios.....	3
------------------------------	---

MODELO EXAMEN 1ª EVALUACIÓN

1. EXAMEN

1) El bingo es un juego que consiste en comprobar números que van saliendo aleatorios de un cartón (que podemos representar como un array bidimensional).

Cuando todos los números de una fila han salido, se puede cantar línea.

Cuando todos los números de un cartón han salido, se puede cantar bingo.

Realiza las siguientes funciones:

a) Función `compruebaLinea`, que recibe un cartón de bingo (representado por un array bidimensional) y un array unidimensional desordenado, con los números que han ido saliendo. Devuelve `true` si se canta línea, `false` en caso contrario.

b) Función `compruebaBingo`, que recibe un cartón de bingo (representado por un array bidimensional) y un array unidimensional desordenado, con los números que han ido saliendo. Devuelve `true` si se canta bingo, `false` en caso contrario.

c) Función `sacaNumeroNuevo`, que recibe un array unidimensional con los números que han ido saliendo. Esta función devuelve un número aleatorio entre 1 y 99 que no este presente en el array unidimensional recibido.

d) Guarda en `LocalStorage` cuantas veces se ha llamado a la función `compruebaBingo`. Debe cargarse este valor desde `localStorage` a una variable global cada vez que se cargue la página para hacer un conteo permanente (No es necesario mostrarlo por pantalla).

(4.5 puntos)

2) Suponiendo que tenemos dos input “usuario” y “password” y un botón “meterPassword” y un segundo botón “compruebaUser”.

a) Asocia al evento `onfocus` de password que al poner el foco se vacíe dicho elemento.

b) Asocia al evento `onblur` de usuario que al perder el foco comprueba que tiene entre 7 y 15 caracteres.

c) Asocia al evento `onblur` de password que al perder el foco comprueba que tiene al menos un letra y un numero.

d) Cada vez que pasen 10 segundos saldrá un mensaje “Corre, mete ya el password”. Cuando alguien haga click en el botón “meterPassword”, deberá cancelarse la aparición del mensaje.

e) Al pulsar el botón “compruebaUser”, realiza una petición por AJAX enviando mediante metodo `GET` el dato “user” con el contenido del input “usuario” a la url “hola.com/pruebauser.py”. Esta responderá con un JSON con el formato “{‘respuesta’:‘SI’}” si es correcto el usuario y con el formato “{‘respuesta’:‘NO’}” si no lo es.

Muestra un alert diciendo si el usuario es correcto o no cuando la respuesta sea recibida.

(4.5 puntos)

3) Crea la clase `consultaSQL`, con un constructor con un único atributo que sea `cadenaConsulta`.

a) Asocia a esa clase un método `comprobarSeguridadConsulta` que compruebe si la consulta de nuestra clase tiene algún símbolo ‘ o #. Si es así, devolverá `false`, si no tiene ninguno de esos símbolos devolverá `true`.

b) Asocia a esta clase un método `comprobarCadenaMayusculas`, que devolverá `true` si la cadena esta toda en mayúsculas, `false` en caso contrario.

(1 puntos)