

MODELO EXAMEN 2ª EVALUACIÓN

Desarrollo Web entorno cliente
CFGS DAW

Sergio García Barea

sergio.garcia@ceedcv.es

2018/2019

Versión:190210.1512

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.Examen.....	3
----------------------	----------

MODELO EXAMEN 2ª EVALUACIÓN

1. EXAMEN

1) Realiza obligatoriamente en jQuery:

Tenemos una API resumida de la API que ofrece el Ayuntamiento de Valencia para conocer el estado de las estaciones de ValenBisi (servicio público de préstamo de bicicletas en Valencia).

La url ficticia donde obtenemos la información de ValenBisi se obtiene de "<http://valencia.com/valenbisi.php>".

(Tenéis acceso a un ejemplo de esta API en <http://hispabyte.net/DWEC/modeloExamenEj1.php>)

Esta url recibe parámetros por método POST. Recibe un único parámetro "day" que indica el día a consultar en formato dd/mm/aaaa. (Por ejemplo, si se le envía 01/11/2015 devuelve los valores de el 1 de noviembre de 2015).

Cuando se hace una petición AJAX a esta url ficticia se obtiene un JSON de este estilo:

```
{ "name": "Estacion 11", "available": "25", "free": "0", time: "20:15" },  
{ "name": "Estacion 12", "available": "24", "free": "1", time: "20:15"},  
{ "name": "Estacion 11", "available": "1", "free": "24", time: "20:00"},  
{ "name": "Estacion 12", "available": "22", "free": "3", time: "20:00"},
```

Donde:

- name es el nombre de la estación.
- available es el numero de bicis disponibles.
- free es el numero de puestos libres para dejar una bici en préstamo.
- time: es la hora en que se obtuvieron esos datos.

Realiza un programa usando jQuery que:

a) Realice la petición AJAX correspondiente obteniendo el día a consultar de un input de tipo texto con id="dia". **(0.5 puntos)**

b) Sobre la petición AJAX recibida, actualice un combobox con id="estaciones" con el nombre de las estaciones que se han obtenido datos. (OJO: si una estación esta varias veces en el JSON solo debe aparecer una vez en el combobox). **(1.5 puntos)**

c) Cuando el usuario modifique el resultado del combobox, deberá mostrar un alert diciendo que estación es, cual fue en cualquier momento el número máximo de puestos disponibles en dicha estación y cual fue el máximo de bicis disponibles en esa estación durante el día y cual fue el máximo de puestos libres en esa estación durante el día. **(1.5 puntos)**

d) Asocie al botón "Estación con más huecos libres" que tiene como id="masLibres" la acción que al hacer click diga con un alert el nombre de la estación que en algún momento del día ha tenido el máximo de huecos libres de todas las estaciones. **(0.75 punto)**

Si hay más de una estación en esta situación, cualquiera vale.

Ejemplo sobre el JSON propuesto:

El combobox tendrá dos entradas: "Estacion 11" y "Estacion 12".

Si elegimos "Estacion 11" se nos mostrará: "Estacion 11 tuvo como máximo 25 bicis y como máximo 24 libres".

Si elegimos "Estacion 12" se nos mostrará: "Estacion 12 tuvo como máximo 24 bicis y como máximo 3 libres".

Si pulsamos "Estación con más huecos libres" nos mostrará "Estacion 11" (que en algún momento tuvo 24 libres).

(4.25 puntos)

2) Utilizando Vue JS, realice una Web (o un componente en formato SFC) que:

- a) Tenga tres input de texto, con id="texto", id="orden" e id="numero" respectivamente.
- b) Habrá un elemento "P" que mostrará el contenido del input con id="texto" pasado a minúsculas. **(1 punto)**
- c) Si en algún momento el contenido del input de texto id="orden" es ocultar, se ocultaran los otros dos input de texto. Si en algún momento el contenido es "borrar", se borrara el contenido de los 3 input de texto. **(1.25 puntos)**.
- d) Habrá un Div con el contenido "Prueba for" que se mostrará tantas veces como nos marque el elemento con id="numero" siempre que dicho número sea par. Si es impar, se mostrará 0 veces el div. **(1.25 puntos)**.

(3.5 puntos)

3) Explica que hace este comando de Quasar CLI.

"quasar dev -m electron -t ios -T win32 -A ia32". ¿Que diferencia hay si se coloca build en lugar de dev?**(0.5 puntos)**

4) Explica con tus palabras los siguientes fragmento de código del Framework Quasar.

a) <q-page-container>

<router-view/>

</q-page-container>

b) <q-tabs color="primary">

<q-route-tab icon="school" to="/escuelaguay" name="escuelaguay" label="CEEDCV" />

</q-tabs>

(1 puntos)

5) Explica el código Phaser necesario para que un enemigo siga horizontalmente a un personaje.

(0.75 puntos)