

Actividad Práctica Integradora

Interfaz de desarrollo

Actividad 4

Situación

Con la aprobación del funcionamiento correcto del carrito en el último mockup presentado, el cual persiste los datos de la selección de productos que realiza el usuario, pidieron aplicar los últimos cambios para que el proyecto deje de ser un Mockup y se convierta en una Demo lo más próxima a un proyecto real.

Para cumplimentar esto último, solicitan los cambios finales que dejarán al proyecto cuasi funcional, listo para conectarse a una futura aplicación de backend. Se deberán aplicar unos últimos cambios en el proyecto evolutivo que vienen trabajando en las iteraciones anteriores.

Recuperar el código del proyecto en el cual vienen trabajando de manera evolutiva.

Consignas

1. Integrar el uso de la función `fetch()` utilizando promesas, para solicitar datos provenientes desde un archivo JSON. Para ello, se deberá convertir también el array de objetos a un archivo Json independiente.

- a) Convertir los datos del array productos, y guardarlos en un archivo llamado `productos.json`. Eliminar a continuación el archivo `productos.js` del proyecto.

Actividad Práctica Integradora

- b) Agregar la constante productos como un array vacío, en el archivo index.js. También agregar una constante URL, que tenga referenciada la ruta al archivo productos.json.
 - c) Crear una función llamada obtenerProductos(), la cual utilizará la función fetch() para leer el archivo productos.json, y volver a obtener de allí todo el array de productos original. Dentro de la función fetch() referenciar la constante URL, y definir los pasos para convertir los datos obtenidos a un formato JSON. Luego agregar todo su contenido al array productos. El Team Leader sugiere utilizar, para simplificar este paso, el uso del método productos.push() junto a spread operator.
 - d) Finalmente, llamar a la función cargarProductos(productos) dentro de un método .then() dedicado. El llamado a la función original debe ser quitado, porque ahora la función obtenerProductos() debe ser la función principal a llamar.
- 2) Los datos almacenados y recuperados de forma efectiva desde LocalStorage, se deben visualizar en el documento checkout.html. Para ello, realizar lo siguiente:
- a) desbloquear o agregar la referencia al archivo js/checkout.js en el archivo checkout.html. Recordemos sumar el atributo defer en la misma.
 - b) Dentro de este archivo, se debe copiar toda la estructura definida en <tr> y </tr>, dentro del documento checkout.html. Esta estructura corresponde a una fila de la tabla en el apartado <tbody>, y en ella se reflejarán todos los productos que estén

Actividad Práctica Integradora

cargados en el carrito.

c) Crear una función llamada `retornarTablaHTML()` y pasarle un parámetro llamado `producto`. Internamente, esta función retorna una plantilla HTML con toda la estructura copiada de la tabla HTML en cuestión. Aquí armarán la fila de la tabla intercalando con Template String, el producto que llegará a esta función por parámetro.

d) A continuación, verificar si `carritoFrutas` posee productos. De existir los mismos, recorrer este array utilizando el método `forEach` y por cada iteración, invocar a la función anteriormente creada, pasándole el parámetro correspondiente a cada producto.

Recuerden definir en esta función, como primer parámetro, el vaciado del cuerpo de la tabla HTML. Finalmente, declarar al inicio del archivo `checkout.js` una constante que permita conectarte con el elemento HTML `<tbody>`.

3) Deberán agregar una funcionalidad mínima en el botón **COMPRAR** de este último documento HTML para que, al ser pulsado, se muestre un mensaje de agradecimiento al usuario por la compra realizada, y se vacíe el array `carritoFrutas`, al mismo tiempo que también debe vaciarse `LocalStorage` y la tabla HTML con los productos visualizados.

a) Para el mensaje de agradecimiento, pueden utilizar `alert()` más algún texto ideado por ustedes.

b) Para vaciar `LocalStorage`, pueden aplicar el método `.removeItem()` o `.clear()`, ambos de `LocalStorage`, indistintamente.

Actividad Práctica Integradora

- c) Para vaciar la tabla HTML, pueden borrar (vaciar) directamente el contenido de la propiedad innerHTML, de acuerdo al nombre de la constante que creamos para enlazarnos desde JS a este tag del documento en cuestión.

FORMATO DE ENTREGA

Deberán presentar los siguientes archivos:

- El proyecto completo, 100% funcional.

Subirlo comprimido dentro de un archivo en formato .ZIP.

Entrega

¡Llegaste al final de la actividad de este módulo! Recuerda guardar tus respuestas y luego subirlas clicando en el botón "Enviar tarea".

Puedes consultar tus dudas con tus compañeros en el foro de la materia o con tu tutor.