



El futuro digital
es de todos

MinTIC



Enlazando Datos

Hola:

El concepto de vinculación o encadenamiento de datos/modelos es recurrente en el desarrollo de aplicaciones web. Este inicia cuando el usuario, a través de una página web que contiene un formulario en el que ingresa un conjunto de datos. En este se ingresa el número 10, que puede indicar el código de un producto. Luego, el navegador hace una petición al servidor web indicando la dirección URL del recurso, añadiendo el dato que se ingresó en el formulario web. Internamente la aplicación web debe hacer el encadenamiento de este dato con las entidades, en este caso es la clase producto. También es posible que deba realizarse alguna acción, como en el ejemplo, consultar el producto con ID 10 en los datos persistentes, que en este caso, está almacenado en un archivo de texto con formato JSON.



Estos formularios web o HTML permiten la entrada de información por parte del usuario, donde únicamente se tiene la entrada para un nombre de una persona. La etiqueta HTML para definir el formulario es `<form>`. Para enviar los datos se debe incluir siempre una etiqueta de tipo ‘enviar’ que le va a indicar al navegador lo que debe hacer cuando se haga click en cierto elemento y que clase de información debe retornar al servidor. Esto puede hacerse en etiquetas tipo botón o, puede ser cualquier elemento que permita entrada datos en el formulario.

Entre los datos que se envían se encuentran las etiquetas *input* del formulario, estas pueden ser texto, listas desplegables o datos seleccionados por en radio *boxes* o *check boxes*. Estos datos se envían a la dirección URL donde se obtuvo el formulario inicialmente a través de un elemento *submit* mediante una petición http.

Nombre

Grabar

```
<form>
```

```
    <label for="nombre">Nombre:</label>
    <input type="text" name="nombre" />
```

```
    <button type="submit">Save</button>
```

```
</form>
```

Una manera de cambiar la dirección hacia dónde va la petición al hacer un ‘get’ es mediante el atributo ‘action’, con el que se indica la dirección del recurso a través de un directorio llamado ‘update’. Es muy importante recordar que cuando se hace esto, se le debe indicar de manera explícita el método a utilizar, en este caso ‘get’. Además, nunca se debe utilizar un método ‘get’ para actualizar o eliminar el recurso, este se utiliza para indicar consultas.

Nombre

Oscar

Grabar

```
<form action="/update" method="get">
```

```
    <label for="nombre">Nombre:</label>
    <input type="text" name="nombre" />
```

```
    <button type="submit">Save</button>
```

```
</form>
```

Cuando el navegador envía una petición (request) ‘get’, este tiene lugar en una cadena de búsqueda con una tupla, es decir, un par de elementos, llave y el valor, que en este caso, la llave es nombre y el valor es Óscar. En caso de querer modificar el dato se utiliza un método ‘post’ que enviará los datos al servidor en tuplas llave y valor, pero ya no irán en la cadena de búsqueda, sino en el cuerpo (body) del mensaje que se envía al servidor.

Nombre

Oscar

Grabar

```
<form action="/update" method="post">
    <label for="nombre">Nombre:</label>
    <input type="text" name="nombre" />

    <button type="submit">Save</button>
</form>
```

En conclusión, entre los conceptos analizados en esta sesión se encuentra el concepto de encadenamiento o vinculación de datos, formularios web y su importancia para el ingreso de información por parte de los usuarios. Además, de dos métodos de envío de datos al servidor, ‘get’, que se utiliza para consultas y ‘post’, que se utiliza para modificación o actualización de datos.



**Mision
TIC2022**



Universidad de Caldas