

# Tarea Métodos GET y Post:

## Explicación:

### ¿Qué es el GET y el POST?

Estos últimos 2 son métodos HTTP que son utilizados para enviar solicitudes a un servidor, a continuación, explicare la función y diferencias entre ambos, incluidos algunos ejemplos para su uso.

#### *Método GET:*

Este método es utilizado para obtener datos o información y realizar acciones que no cambian el estado del servidor, estos datos viajan a través de la URL. Al momento de llamar al método get, la información solicitada viajara en la URL y nos desplegara su debida respuesta. Algunos ejemplos de aplicación del método GET podrían ser los siguientes:

- **Búsquedas en sitios web.** - Dentro de esto podemos ver que los parámetros de la búsqueda viajaran en la URL dándonos la debida respuesta
- **Navegación y Paginación.** - En las aplicaciones web suele ser empleada la paginación, se puede ver un ejemplo muy claro del método GET, cuando en la URL podemos identificar en que pagina nos encontramos.
- **Filtros y Clasificaciones.** - En algunas ocasiones se nos presentara el requerimiento de filtrar información en nuestra pagina web, gracias a la implementación del meto GET, podremos crear los filtros y los mimos serán visibles en la URL de la página web.
- **Acceso a recursos estáticos.** - Al momento de hacer énfasis en recursos estáticos podemos poner de ejemplo a una imagen en cualquier formato (.png, .jpg, etc.), implementado el método GET podremos mostrar una imagen en especifico y la misma se podrá ver en nombre dentro de la URL.
- **Visualización de Detalles de un ítem.** - Existen casos en los que, utilizando el método GET, se puede usar la URL para poder presentar los detalles que posee algun ítem que tengamos en nuestra página web, al hacer esto la URL presentará una modificación que hará referencia a que estamos obteniendo esos datos solicitados.

Estos son algunos de varios ejemplos en los que se puede emplear el método GET, a continuación, presento de manera gráfica, unos ejemplos del método GET que hacen énfasis a lo aprendido en clase:

#### *Buscar Empleados por palabra clave:*

En esta sección de código, se desarrolló una sección en la que se puede buscar empleados por una palabra clave, para lograr ello se creó el código siguiendo los pasos de a continuación:

Modelo → Vista → URL → Template

Gracias a ello obtenemos una URL que sería la siguiente:

`'buscar/empleado/'`

La cual va a ser actualizada utilizando el siguiente HTML del Template:

```
1 <h1>Buscar empleados por palabra clave</h1>
2
3 <form method="GET">
4   <input id="kword" name="kword" type="text" placeholder="Ingresa nombre">
5   <button type="submit">Buscar</button>
6 </form>
7
8 <h3>Lista Resultado</h3>
9
10 <ul>
11   {% for e in empleados %}
12   <li>{{e}} </li>
13   {% endfor %}
14
15 </ul>
```

En el HTML se implementa el método GET, lo que permite que el usuario al ingresar el input de la línea 4 y luego oprimir buscar, el URL obtendrá los datos ingresados y el método GET ayudara a la obtención de la información como se ve a continuación en la página web:



## Método POST:

La utilización de este método suele ser para enviar datos que pueden cambiar el estado del servidor, como crear o actualizar recursos en la base de datos. Esta información se envía en el cuerpo de la solicitud HTTP y no modifican el URL. Para tener una idea mas clara de este método presentare a continuación algunos ejemplos en los que suele ser empleado.

- **Formularios de registro.** - En las páginas web de lo más común suelen ser los formularios de registro para crear una nueva cuenta, gracias a la implementación del método POST podemos hacer el envío de esa información sin que la URL tenga esos datos visibles en ella
- **Formularios de inicio de sesión.** - Al igual que en la aplicación anterior, se utiliza el método POST para crear formularios en los que el usuario ingrese su información, inicie sesión y acceda a nuestra aplicación web

- **Envío de comentarios o reseñas.** - Este método suele ocuparse para la creación de formularios en los que el usuario envía su opinión o registra una reseña en algún contexto de nuestra página, toda esa información se envía con la solicitud HTTP sin modificar el URL
- **Actualización de Perfil.** - Un excelente ejemplo del método POST es la Actualización de un perfil, ya que gracias a la implementación de ese método estamos haciendo una solicitud de información y guardándola en la base de datos, sin necesidad de que la información sea vulnerable en la URL
- **Compra y Checkout.** - Dentro de este apartado su aplicación también es importante ya que gracias a ello podremos almacenar información valiosa como: Direcciones de envío, Métodos de Pago, etc. De manera más segura.

Existen más aplicaciones del método POST dentro de este tema, pero con estos ejemplos se describe de manera sencilla la utilización de este, a continuación, voy a mostrar ejemplos de este método que han sido implementados en la clase:

#### *Actualización de Datos de un Empleado:*

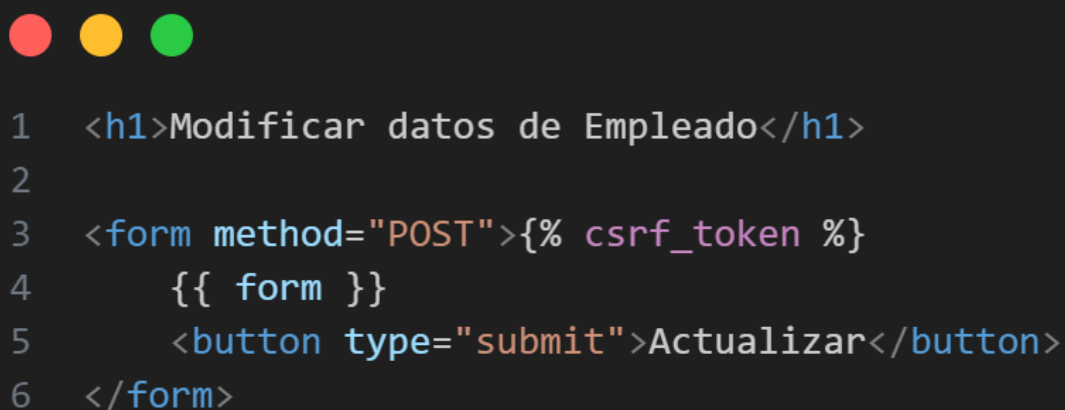
Dentro de esta sección creamos una función que nos permita editar los datos de un empleado ya existente en la base de datos, esto lo conseguimos siguiendo los siguientes pasos:

Modelo → Vista → URL → Template

Así obtenemos la siguiente URL:

‘/update-empleado/’

Como se puede ver es la misma estructura que en la parte anterior, con la diferencia que en la parte del Template, se utilizó el método POST, como se ve a continuación:



```
1 <h1>Modificar datos de Empleado</h1>
2
3 <form method="POST">{% csrf_token %}
4     {{ form }}
5     <button type="submit">Actualizar</button>
6 </form>
```

Lo que está haciendo el método post en esta sección de código HTML es acceder a la base de datos y actualizar los datos que hemos ingresado en la página web, como se puede ver a continuación:

Datos anteriores:

The screenshot shows a web browser with the address bar at 127.0.0.1:8000/ver-empleado/4. The page has a navigation bar with links: Empelados, Listar, Departamentos, Administrar, and Nuevo Departamento. The main content area displays a profile card for an employee named Maria Vicente, who works in the Deportes department. The card includes a placeholder for a profile picture labeled 'OTRO' and a button labeled 'Habilidades:'.

Formulario HTML implementando el método POST mostrado anteriormente:

The screenshot shows a web browser with the address bar at 127.0.0.1:8000/update-empleado/4/. The page title is 'Modificar datos de Empleado'. The form contains the following fields: 'Nombre:' with the value 'Maria', 'Apellidos:' with the value 'Vicente', 'Trabajo:' with a dropdown menu showing 'OTRO', and 'Departamento:' with a dropdown menu showing '4-Departamento de Deportes-Deportes'. Below these fields is a list of skills: '1-Cantar', '3-Correr', '4-Bailar', and '5-Tocar el Piano', with a dropdown menu showing '5-Tocar el Piano'. There is an 'Actualizar' button at the bottom of the form.

Formulario HTML implementando el método POST, marcando los nuevos datos para actualizar:

**Modificar datos de Empleado**

Nombre:

Apellidos:

Trabajo:

Departamento:

Habilidades:

Vista del Formulario HTML, implementando em método POST, con los datos ya actualizados:

Empelados   Listar   Departamentos   Administrar   Nuevo Departamento

OTRO

**Maria**  
Vicente Idalgo  
Deportes  
Habilidades:  
Cantar  
Correr  
Bailar

Como se pudo observar anteriormente, el URL no ha cambiado al realizar la actualización de datos sin embargo los datos si han sido actualizados con éxito.

Conclusión:

- **Método GET:** Es un método HTTP utilizado para obtener datos y realizar acciones que no cambian el estado del servidor, enviando parámetros a través de la URL.

- **Método POST:** Es un método HTTP utilizado para enviar datos del formulario que pueden cambiar el estado del servidor, enviando información en el cuerpo de la solicitud HTTP sin modificar la URL.