

## 03-Rutas en Flask

### ¿Qué son las Rutas?

En Flask, una "ruta" se refiere a la URL a la que un usuario puede acceder en una aplicación web. Cada ruta está asociada con una función, conocida como view function, que maneja la lógica necesaria para responder a las solicitudes en esa URL específica.

### Definiendo Rutas

Las rutas se definen utilizando el decorador `@app.route()`, que se asocia con una función. Este decorador acepta la URL de la ruta como primer argumento y opcionalmente, métodos como `GET`, `POST`, etc., que define qué tipos de solicitudes HTTP puede aceptar esa ruta.

Esta función responderá a las solicitudes HTTP GET en la URL raíz (/) de tu aplicación.

```
@app.route('/')
def home():
    return "Bienvenido a la Página Principal!"
```

### Métodos HTTP

Por defecto, las rutas en Flask solo responden a solicitudes GET. Si quieres manejar otros métodos, como POST (usado comúnmente para enviar formularios), debes especificarlo usando el argumento `methods` del decorador `@app.route()`.

Ejemplo con Métodos Múltiples:

```
@app.route('/submit', methods=['GET', 'POST'])
def submit():
    if request.method == 'POST':
        # lógica para manejar datos del formulario
        return "Formulario enviado"
    return "Envía un formulario"
```

En este ejemplo, la misma función maneja tanto GET como POST. Si el método es POST, se procesa el formulario; si es GET, simplemente se muestra una página.

## Pasar Variables

Flask permite pasar variables desde la URL a la función de vista. Esto se hace colocando partes variables de la URL entre `<variable>`.

Ejemplo con Variables:

```
@app.route('/user/<username>')
def show_user_profile(username):
    return f"Perfil del usuario {username}"
```

Aquí, cualquier valor que se coloque en la posición de `username` en la URL será pasado a la función `show_user_profile`.

## Redirecciones y URLs

Para redireccionar a los usuarios a otra ruta o para construir URLs a funciones de vista específicas, Flask proporciona las funciones `redirect()` y `url_for()` respectivamente.

Ejemplo de Redirección y Construcción de URL:

```
from flask import redirect, url_for
```

```
@app.route('/admin')
def admin():
    return redirect(url_for('home'))
```

Este ejemplo redirige a cualquier usuario que acceda a `/admin` a la ruta asociada con la función `home`.

## Consideraciones Finales

- Blueprints: Para aplicaciones más grandes, Flask ofrece "Blueprints" que permiten modularizar la gestión de rutas en componentes reutilizables y escalables.
- Errores y Páginas Personalizadas: Flask también permite definir funciones personalizadas para manejar errores específicos como 404 o 500.

El manejo eficiente de las rutas es crucial para crear aplicaciones web funcionales y fáciles de mantener. Flask proporciona herramientas flexibles y potentes que facilitan la gestión de estas rutas, ayudando a los desarrolladores a construir aplicaciones web robustas y eficientes.