

{dev/talles}

django

Function-Based Views vs Class-Based Views



Por: Ricardo Cuéllar

¿Qué es Function-Based View (FBV)?

Vistas basadas en función

Una FBV es una función de Python que recibe un request y devuelve un HTTP Response (o un render)

Ventajas	Observaciones
Simples y fáciles de entender	Pueden volverse muy complejas y desordenadas
Muy flexibles	No se reutiliza tanto código



¿Qué es Function-Based View (FBV)?

Vistas basadas en función



```
# views.py
from django.http import HttpResponse

def hello_fbv(request):
    return HttpResponse("👋 Hola desde una vista basada en función (FBV)")
```



```
# urls.py
from django.urls import path
from .views import hello_fbv

urlpatterns = [
    path("hello-fbv/", hello_fbv, name="hello_fbv"),
]
```

¿Qué es Class-Based View (CBV)?

Vistas basadas en clases

Un CBV es una clase que hereda de View o alguna otra clase genérica, y define métodos como `get()` y `post()`.

Ventajas	Observaciones
Reutilización y herencia	Curva de aprendizaje puede ser compleja
Fáciles de extender	Pueden parecer muy abstractas



¿Qué es Class-Based View (CBV)?

Vistas basadas en clases

```
● ● ●
# views.py
from django.views import View
from django.http import HttpResponse

class HelloCBV(View):
    def get(self, request):
        return HttpResponse("👋 Hola desde una vista basada en clase (CBV)")
```

```
● ● ●
# urls.py
from .views import HelloCBV

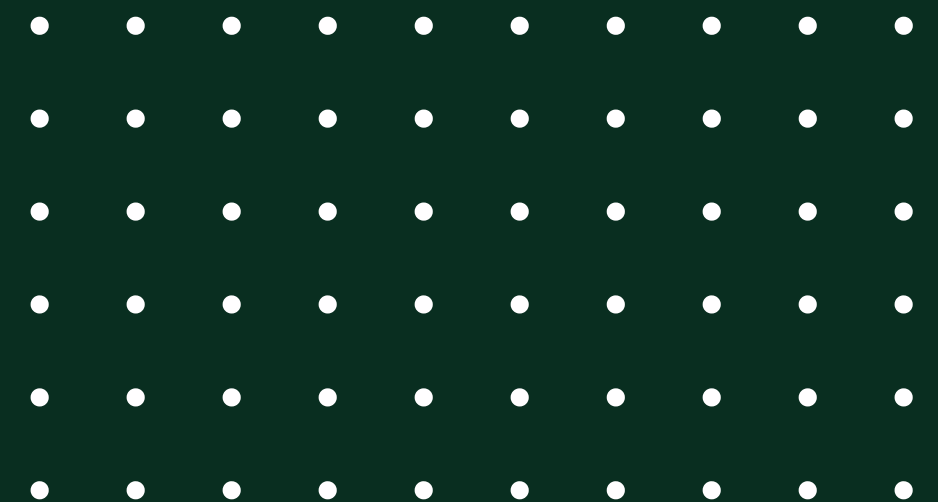
urlpatterns += [
    path("hello-cbv/", HelloCBV.as_view(), name="hello_cbv"),
]
```

¿Notaste el `.as_view()`?

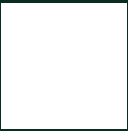


Es utilizado para **convertir la clase en una función** compatible con el enrutador de Django.

Podemos decir que funciona como puente que convierte la clase en una vista funcional.



¿Cuál usar?



Característica	FBV	CBV
Simplicidad	Más fácil de entender para principiantes	Más abstracto y avanzado
Control	Tú controlas el flujo	Heredas el comportamiento predefinido
Reutilización	Requiere funciones auxiliares	Puedes reutilizar mediante herencia
Organización	Difícil escalar en vistas grandes	Mejor estructura para vistas complejas
Rendimiento	Un poco más rápido	Un poco más pesado pero por poco



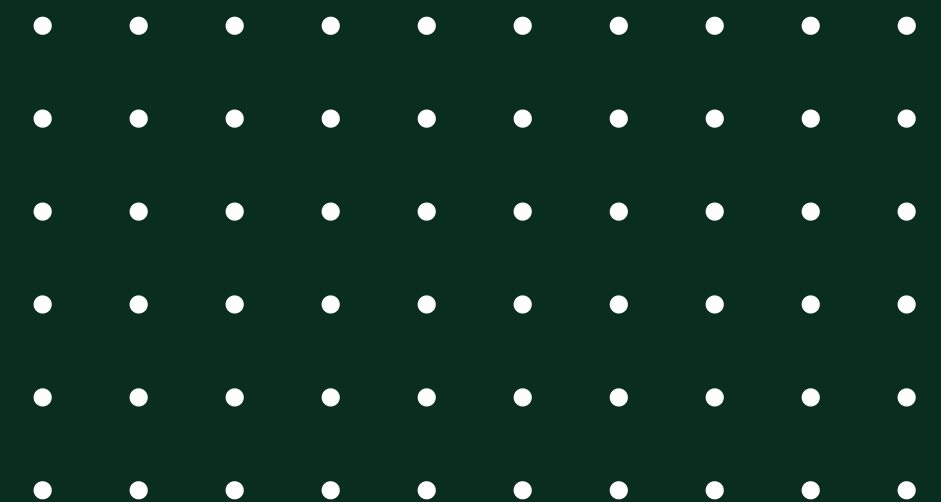
¿Cuál usar?

Debemos usar CBV cuando:

- Se requieren muchas vistas con comportamiento repetido como (Listar, crear, editar, eliminar, ósea un CRUD)
- Estás construyendo APIs o CRUDs.
- Quieres mantener el código DRY (Don't Repeat Yourself)

Debemos usar FBV cuando:

- Construyamos cosas pequeñas
- Necesitamos funciones muy específicas.



{dev/talles}



**¡A seguir
aprendiendo!**

