**Video file :102**

**Code file : ch93**

**Async Class based view in django 5**

In class not sync and async both if we define one function in the class async so other aslo be the async

First install the uvicorn

Pip install unicorn

Pip install httpx

Now write some code to diffeneciate between async and sync :

Views.py:

from django.http import HttpResponse, JsonResponse

from django.views import View

import httpx

import time

import asyncio

# Home View (Async)

class HomeView(View):

    async def get(self, request, \*args, \*\*kwargs):

        return HttpResponse("Hello Home Page")

# Synchronous Class-Based View

class SyncView(View):

    def get(self, request, \*args, \*\*kwargs):

        start\_time = time.time()

        responses = []

        for \_ in range(5):

            response = httpx.get("https://jsonplaceholder.typicode.com/posts")

            responses.append(response.json())

        end\_time = time.time()

        time\_taken = end\_time - start\_time

        return JsonResponse({

            'status': 'success',

            'view\_type': 'sync',

            'total\_request': 5,

            'time\_taken': f"{time\_taken:.2f} seconds"

        })

# Asynchronous Class-Based View

class AsyncView(View):

    async def get(self, request, \*args, \*\*kwargs):

        start\_time = time.time()

        async with httpx.AsyncClient() as client:

            tasks = [client.get("https://jsonplaceholder.typicode.com/posts") for \_ in range(5)]

            responses = await asyncio.gather(\*tasks)

        end\_time = time.time()

        time\_taken = end\_time - start\_time

        return JsonResponse({

            'status': 'success',

            'view\_type': 'async',

            'total\_request': 5,

            'time\_taken': f"{time\_taken:.2f} seconds"

        })

Urls.py:

from django.urls import path

from .views import HomeView, SyncView, AsyncView

urlpatterns = [

    path('', HomeView.as\_view(), name='home'),

    path('sync/', SyncView.as\_view(), name='sync\_view'),

    path('async/', AsyncView.as\_view(), name='async\_view'),

]

For running this project we can hit :

uvicorn ch93.asgi:application --reload