**Django Developer Assignment**

1: By default, are Django signals executed synchronously or

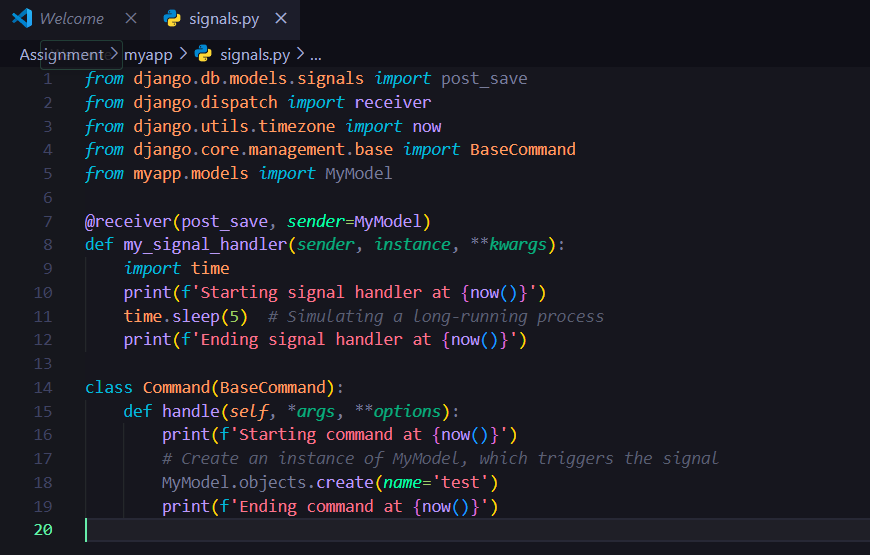
asynchronously? Please support your answer with a code snippet that conclusively

proves your stance. The code does not need to be elegant and production ready, we

just need to understand your logic.

Ans. By default, Django signals are executed synchronously. This means that when a signal is sent, the signal handlers are called immediately, and the processing of the signal handlers blocks the execution of the code that follows the signal emission until all the handlers have finished running.

Here’s a simple code snippet to demonstrate that Django signals are executed synchronously:

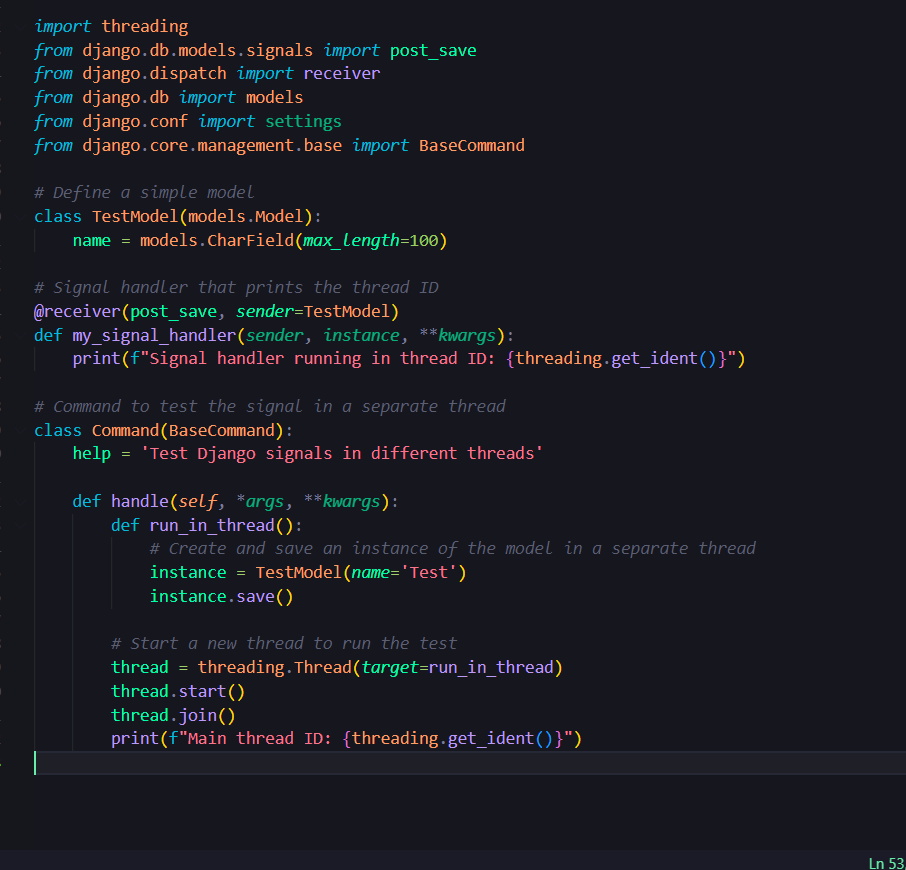


Question 2: Do Django signals run in the same thread as the caller? Please support your answer with a code snippet that conclusively proves your stance. The code does not need to be elegant and production ready, we just need to understand your logic.

Ans. Yes, Django signals run in the same thread as the caller. This means that when you emit a signal, the signal handlers are executed within the same thread that triggered the signal. To demonstrate this, you can use a simple example involving threading to show that the signal handlers are indeed run in the same thread that emits the signal.

Here’s a code snippet that demonstrates this:

Python



Question 3: By default, do Django signals run in the same database transaction as

the caller? Please support your answer with a code snippet that conclusively proves

your stance. The code does not need to be elegant and production ready, we just need

to understand your logic.

Ans. Yes, Django signals run in the same database transaction as the caller by default.

This means, when a signal is triggered, the operations performed within the signal handler

are part of the same transaction as the operation initiated by the caller.

Django signals are executed within the same atomic block as the caller's database

operation. An atomic block ensures that all tasks within it are treated as a single unit of

work, and that they all succeed or all fail together.

Below is the code snippet that demonstrates signals run in the same database transaction as

the caller:

