CURSO DE JAVA DIA 18

SPRING DATA JPA

Primero surge java jdbc y la utilizamos para conectarnos a bd relacionales, pero manejar las conexiones era tedioso, asi aparecieron los orms como hibernate, y tratan de simplificar el manejo del jdbc, es decir es un interprete entre el mundo relacional y el mundo de base de datos.

La lógica de negocio va en el service layer, pero toda esa lógica de negocio también la puedo implementar en la base de datos.una herramienta útil es pl/sql

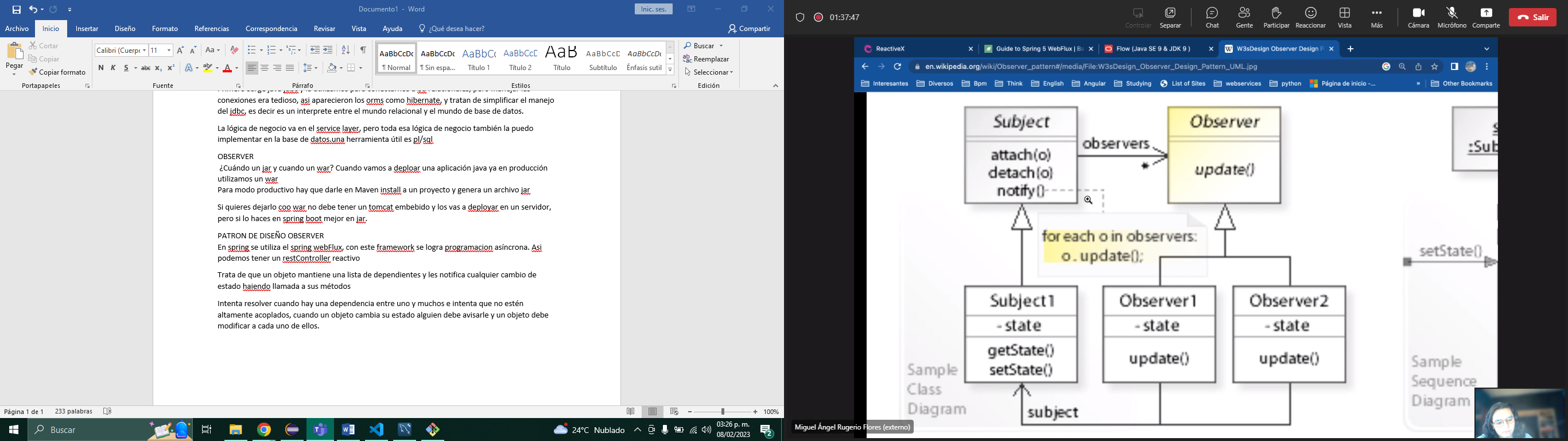
OBSERVER  
 ¿Cuándo un jar y cuando un war? Cuando vamos a deploar una aplicación java ya en producción utilizamos un war  
Para modo productivo hay que darle en Maven install a un proyecto y genera un archivo jar

Si quieres dejarlo coo war no debe tener un tomcat embebido y los vas a deployar en un servidor, pero si lo haces en spring boot mejor en jar.

PATRON DE DISEÑO OBSERVER  
En spring se utiliza el spring webFlux, con este framework se logra programacion asíncrona. Asi podemos tener un restController reactivo

Trata de que un objeto mantiene una lista de dependientes y les notifica cualquier cambio de estado haiendo llamada a sus métodos

Intenta resolver cuando hay una dependencia entre uno y muchos e intenta que no estén altamente acoplados, cuando un objeto cambia su estado alguien debe avisarle y un objeto debe modificar a cada uno de ellos.



El patrón te crea un subject y tus observers, es decir un sujeto hace un cambio y los observadores reaccionan a ese cambio que ejecuta el sujeto

Attach 🡪 Agregar, detach🡪 Elimnar, notify 🡪 notifica

La clase padre define que métodos puede hacer a variable de referencia