

Crea una API REST con Spring Boot que tenga las siguientes funcionalidades.

- La entidad que deberá llamarse con tu inicial y tu primer apellido, ejemplo David Mateos = dmateos
- Debe contener como mínimo 5 atributos, un ID autoincremental y un atributo tendrá que ser saldo (Float)
- Dicha API tendrá que estar documentada con Swagger de manera clara e inequívoca, todos los endpoints y los códigos que devuelven, igual que el tipo de dato (string, objeto, etc)
- Se deberá utilizar una arquitectura de mínimo 3 niveles, controller, repository y entity.
- Cada proyecto será **individual** y se entregará en un zip con el mismo nombre que la entidad, inicial y apellido.
- La base de datos se deberá llamar "di"
- Se valorará la claridad de código, la reusabilidad y los comentarios del mismo.
- **Se penalizará el uso de herramientas de generación de código mediante IA y el uso de funciones que no estén claramente detalladas y que no hayamos visto en clase.**

La API deberá proporcionar los siguientes métodos:

1. **crearElemento**: recibirá un objeto con toda la información para crear el elemento.
 - a. 200. Elemento creado con éxito
 - b. 205. Elemento creado con información parcial
2. **editarElemento**: recibirá un objeto con información total o parcial, y un identificador
 - a. 200. Elemento actualizado con éxito
 - b. 201. Elemento actualizado parcialmente con éxito
 - c. 209. Elemento no encontrado
3. **buscarElemento**: recibirá un ID y devolverá el elemento en formato JSON
 - a. 200. Elemento encontrado
 - b. 209. Elemento no encontrado
4. **borrarElemento**: recibirá un ID y eliminará el elemento
 - a. 200. Elemento eliminado
 - b. 209. Elemento no eliminado
5. **borrarTodo**: Borra todos los registros de la base de datos
 - a. 200: Todos los elementos se han eliminado correctamente
6. **aumentarSaldo**: recibirá un ID y una cantidad, y se aumentará el saldo bancario
 - a. 200: Saldo aumentado correctamente
 - b. 201: Saldo inicializado y aumentado con éxito
 - c. 209: No existe ese ID
7. **reducirSaldo**: recibirá un ID y una cantidad(float), y se reducirá el saldo bancario
 - a. 200: Saldo devengado correctamente y positivo
 - b. 201: Saldo devengado y negativo
 - c. 209: No existe ese ID
8. **todosElementos**: devolverá un JSON con todos los registros de la BD
 - a. 200: Todos los elementos son mostrados correctamente
 - b. 209: No hay ningún elemento para mostrar
9. **mediaSaldos**: devolverá la media de saldo de todos los registros almacenados

- a. 200: Saldo medio de todos los registros
- b. 201: El saldo medio es negativo