Desafío Mercado Libre

Alberto Luis Navarro

1. **crear un programa con un método o función con la siguiente firma:**

**boolean isMutant(String[] dna); // Ejemplo Java**

En donde recibirás como parámetro un array de Strings que representan cada fila de una tabla de (NxN) con la secuencia del ADN. Las letras de los Strings solo pueden ser: (A,T,C,G), las cuales representa cada base nitrogenada del ADN.

Para el desarrollo del punto 1 se desarrollo un algoritmo en java, se enfoco en dividir el problema en distintos métodos para simplificar la lógica.

Estos métodos utilizan el enfoque de ventana deslizante para verificar repeticiones consecutivas vertical, horizontal y diagonalmente, respectivamente. El algoritmo itera a través de cada dimensión, comparando elementos en una ventana deslizante de tamaño 4. Para ello se creó un función (isRepeated)que toma como parámetro una palabra, Si encuentra 4 caracteres repetidos consecutivamente, devuelve verdadero, lo que indica que se encontraron repeticiones consecutivas. De lo contrario, devuelve falso.

Teniendo esta función La función que determina si es mutante la utiliza recorriendo la matriz primero enviando las palabras horizontales, haya la matriz transpuesta para las verticales y las palabras en diagonal y diagonal inversa pero solo las palabras mayores a 3 letras. En cuanto encuentra una repetición, retorno verdadero

***Nivel 2:***

Crear una API REST

crear el servicio “/mutant/” en donde se pueda detectar si un humano es mutante enviando la secuencia de ADN mediante un HTTP POST con un Json el cual tenga el siguiente formato:

POST → /mutant/

{

“dna”:["ATGCGA","CAGTGC","TTATGT","AGAAGG","CCCCTA","TCACTG"] }

En caso de verificar un mutante, debería devolver un HTTP 200-OK, en caso contrario un 403-Forbidden

Para crear el api restful se utilizó SPRINT BOOT TOOL SUIT 4 con las dependencias de JPA y una base de datos de Mysql

Los datos de conexión a la base son los que están en el archivo proterties

Y son los siguientes

spring.datasource.url=jdbc:mysql://localhost:3306/db

spring.datasource.username=mutant

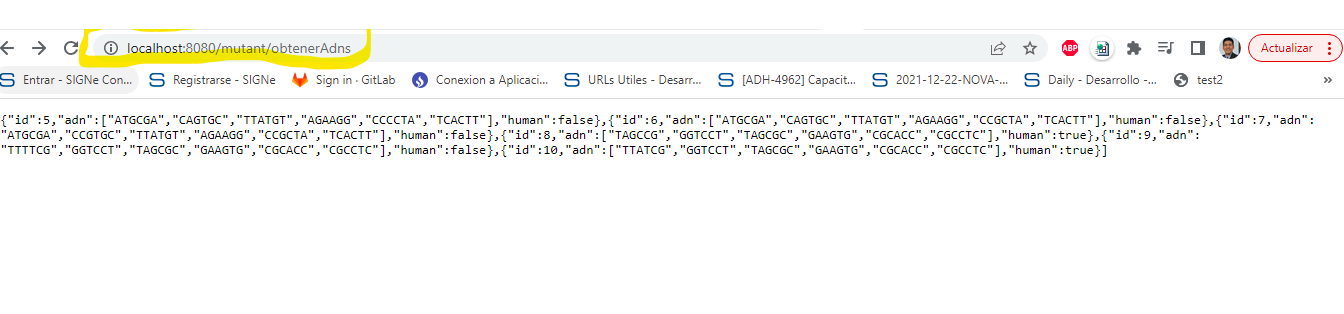
spring.datasource.password=Mut4nt2023

spring.jpa.hibernate.ddl-auto=update

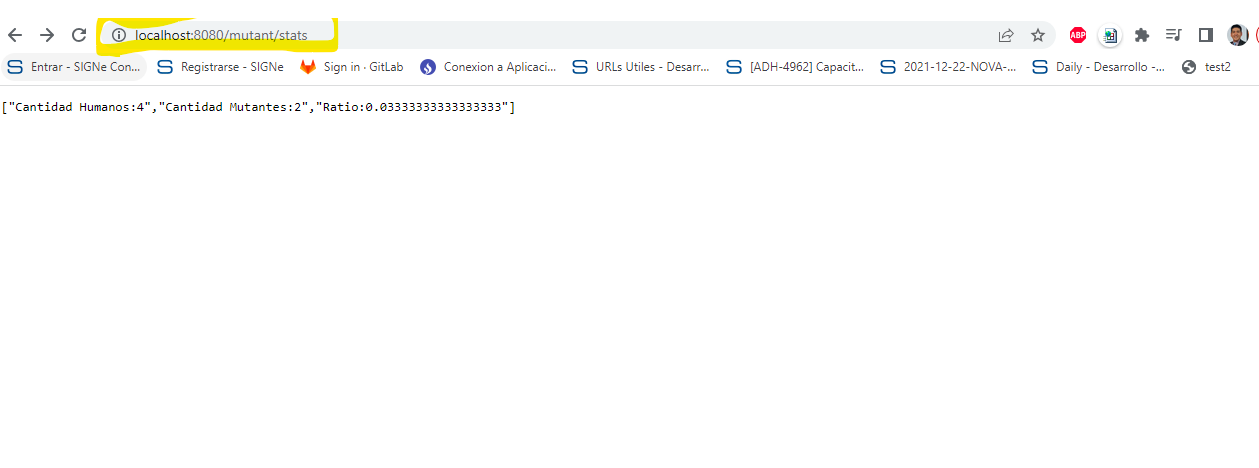
La api utiliza 4 packages ( controllers, Models, Repositories,services) y los correspondientes tests

Para las pruebas del método post se utilizó postman

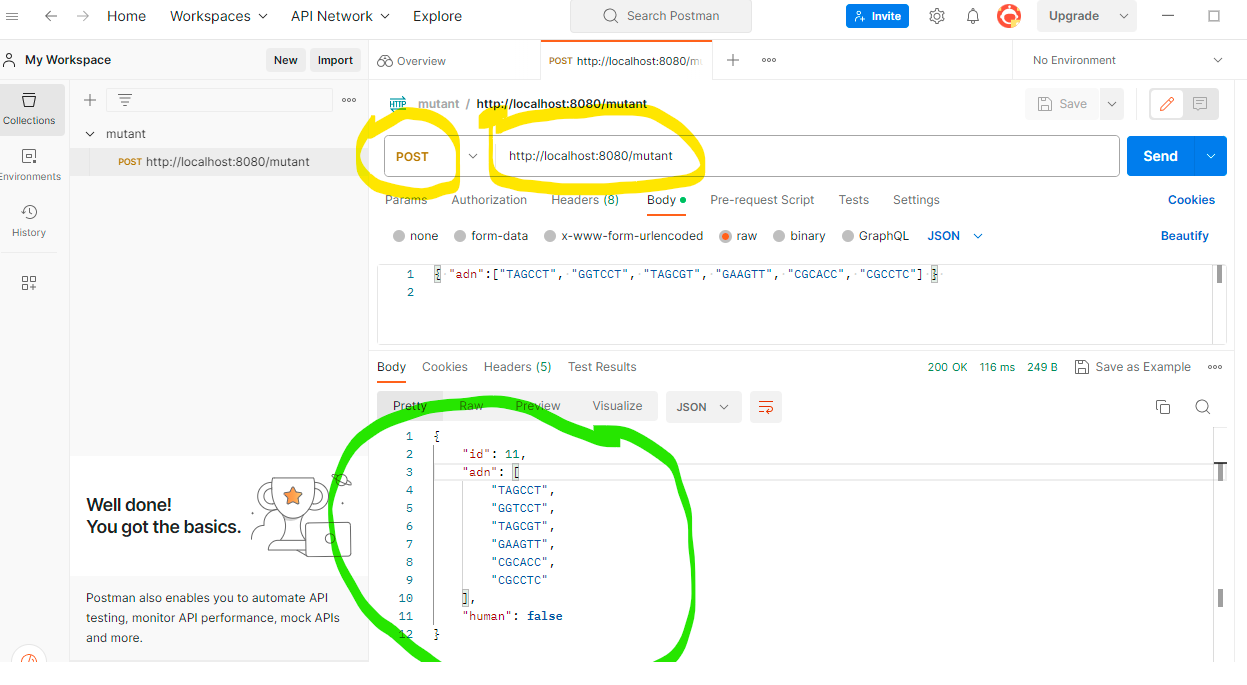
Resultado en el navegador de la consulta de Adns en el navegador



Resultado obtenido en el navegador de las estadísticas /stats



Resultado del método guardar mediante Postman



Resultado mostrado en la base de datos mysql

