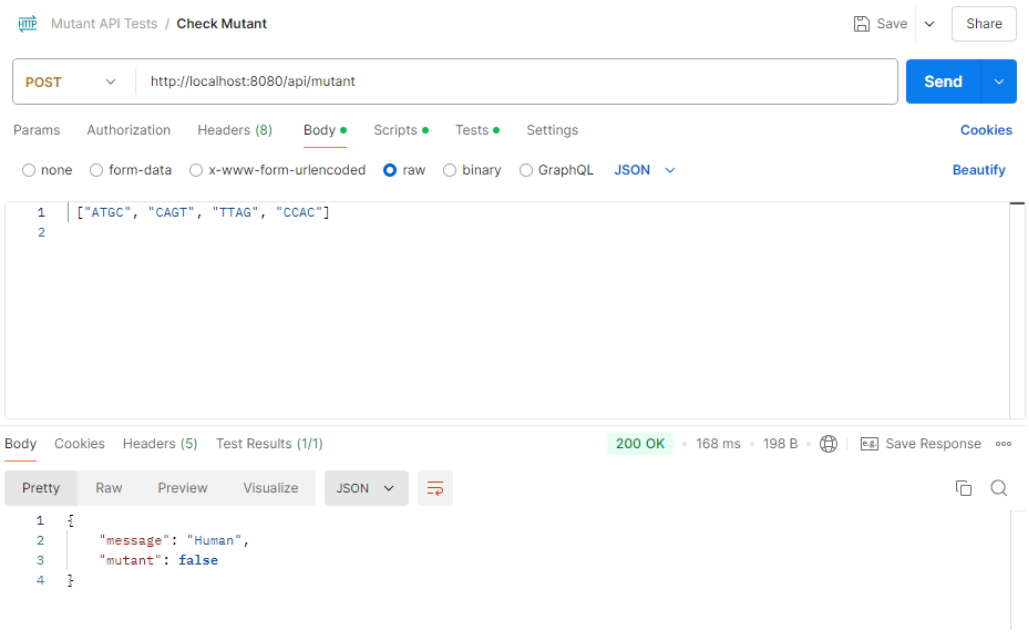
Parcial 1 – Programación III

Jennifer Contreras – 14/10/2024

Documentación Nivel 3

Flujo general del sistema:

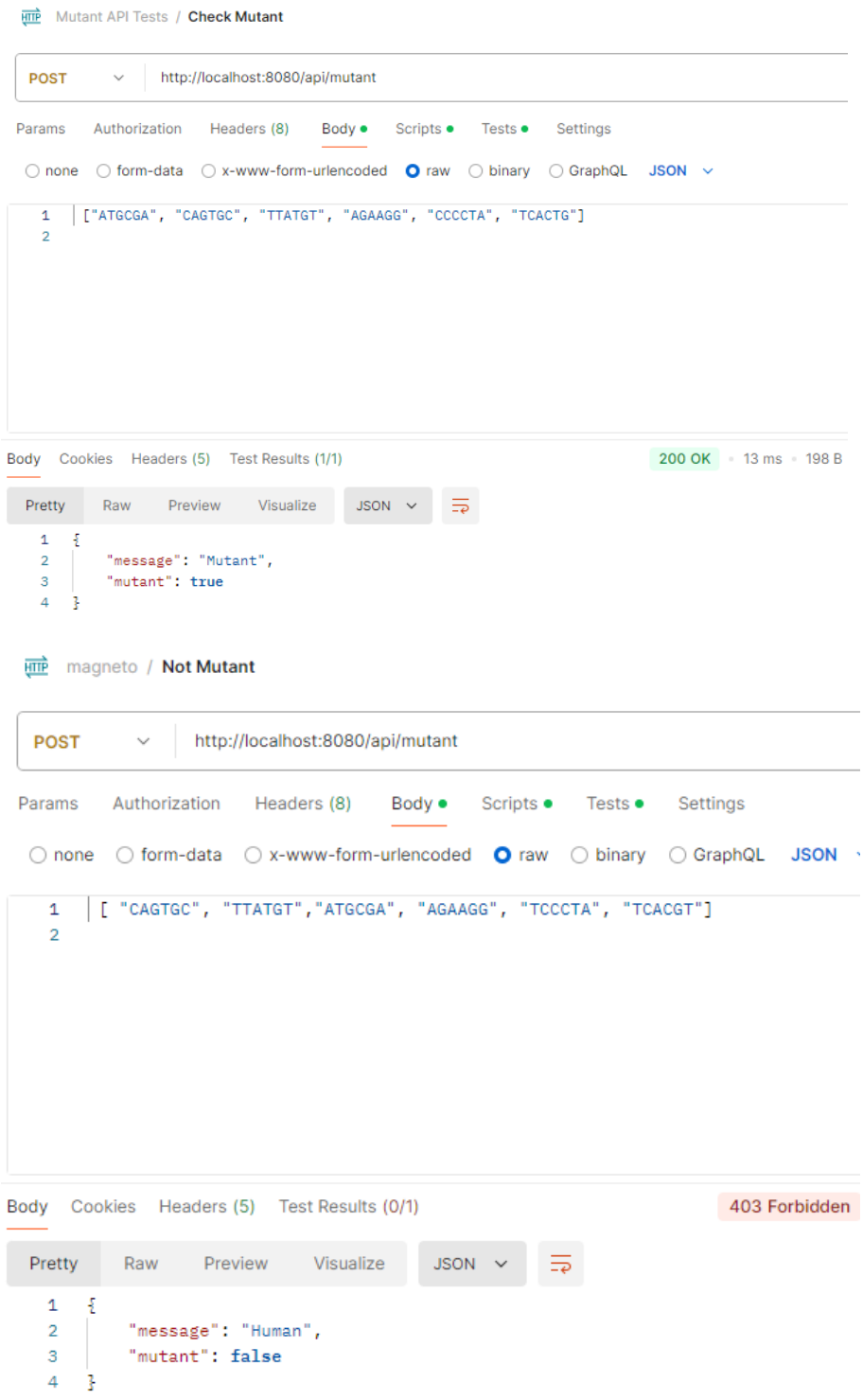
1. **Solicitud HTTP**:
   - Se realiza una solicitud HTTP POST a la ruta /api/mutant, con un cuerpo que contiene una matriz de cadenas que representan una secuencia de ADN (por ejemplo, ["ATGC", "CAGT", "TTAG", "CCAC"]).
   - La solicitud es enviada por un cliente



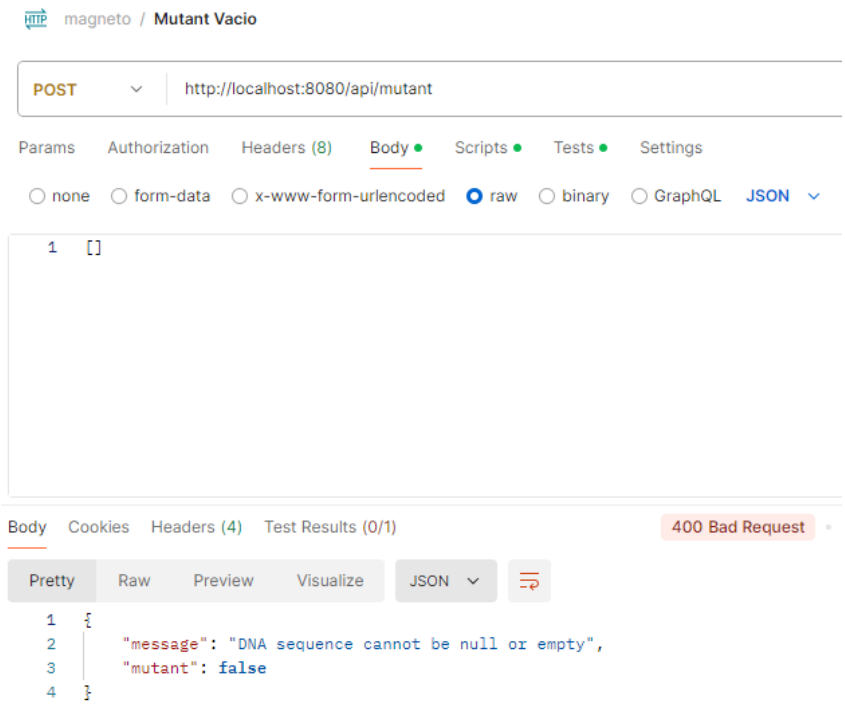
2. **Controlador**:
   - El controlador MutantController recibe la solicitud en el método isMutant().
   - Se verifica si la secuencia de ADN ya existe en la base de datos usando el repositorio DnaSequenceRepository.
   - Si ya existe, se obtiene la información existente y se retorna una respuesta con un mensaje (ya sea "Mutant" o "Human") junto con el valor booleano de isMutant.

- Si no existe, el controlador llama al servicio para determinar si la secuencia es mutante.

3. **Servicio**
   - El método isMutant() del servicio MutantService recibe la secuencia de ADN.
   - Se realizan varias validaciones para asegurar que la entrada sea correcta.
   - Se cuenta cuántas secuencias mutantes hay al verificar filas, columnas y diagonales de la matriz de ADN.
   - La lógica de negocio se basa en buscar secuencias mutantes definidas por la presencia de cuatro letras iguales consecutivas (AAAA, TTTT, CCCC, GGGG).
4. **Repositorio**
   - Después de determinar si la secuencia es mutante, se crea un nuevo objeto DnaSequence con la información correspondiente.
   - Este objeto se guarda en la base de datos a través del repositorio DnaSequenceRepository.
   - Si la secuencia ya existía, se retorna el objeto encontrado de la base de datos sin necesidad de guardarlo de nuevo.
5. **Respuesta HTTP**:
   - Si el ADN es mutante, se retorna una respuesta con el estado HTTP 200 y el DTO ResponseDto con el mensaje "Mutant" y isMutant como true.
   - Si el ADN no es mutante, se retorna una respuesta con el estado HTTP 403 y el DTO ResponseDto con el mensaje "Human" y isMutant como false.
   - Si ocurre una excepción, como una secuencia de ADN vacía o inválida, se lanza una excepción IllegalArgumentException, y se devuelve una respuesta con el estado HTTP 400 y un mensaje de error detallado en el cuerpo de la respuesta. En este caso, se utiliza ResponseEntity.badRequest() para retornar un código de estado 400.


Mutant API Tests / Check Mutant
POST
http://localhost:8080/api/mutant
Params
Authorization
Headers (8)
Body
Scripts
Tests
Settings
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
["ATGCGA", "CAGTGC", "TTATGT", "AGAAGG", "CCCCTA", "TCACTG"]
Body
Cookies
Headers (5)
Test Results (1/1)
200 OK
13 ms
198 B
Pretty
Raw
Preview
Visualize
JSON
{
"message": "Mutant",
"mutant": true
}

magneto / Not Mutant
POST
http://localhost:8080/api/mutant
Params
Authorization
Headers (8)
Body
Scripts
Tests
Settings
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
["CAGTGC", "TTATGT","ATGCGA", "AGAAGG", "TCCCTA", "TCACGT"]
Body
Cookies
Headers (5)
Test Results (0/1)
403 Forbidden
Pretty
Raw
Preview
Visualize
JSON
{
"message": "Human",
"mutant": false
}



Estadísticas en el sistema

1. **Recolección de estadísticas**:
   - Se realiza una solicitud HTTP GET a la ruta /api/stats.
   - El controlador StatsController maneja esta solicitud y utiliza el repositorio para obtener la cantidad de secuencias de ADN mutantes y humanas.
2. **Interacción del DnaSequenceRepository**:
   - Los datos que se recopilan de las secuencias mutantes y humanas se almacenan en la base de datos a través del DnaSequenceRepository cuando se procesan las solicitudes de ADN.
3. **Cálculo de estadísticas**:
   - En el método getStats(), se cuentan las secuencias de ADN mutantes y humanas usando el repositorio.
   - Se calcula la razón entre el número de secuencias mutantes y humanas.
   - Se devuelve un objeto Map<String, Object> con los conteos y la razón, junto con una respuesta HTTP 200 OK.

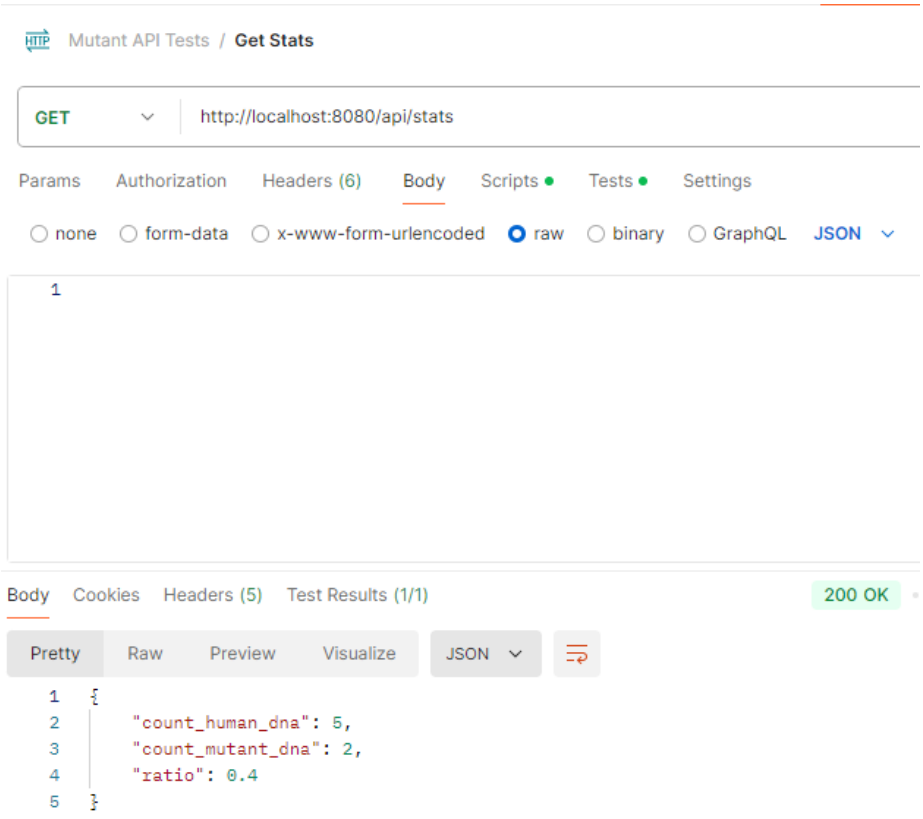
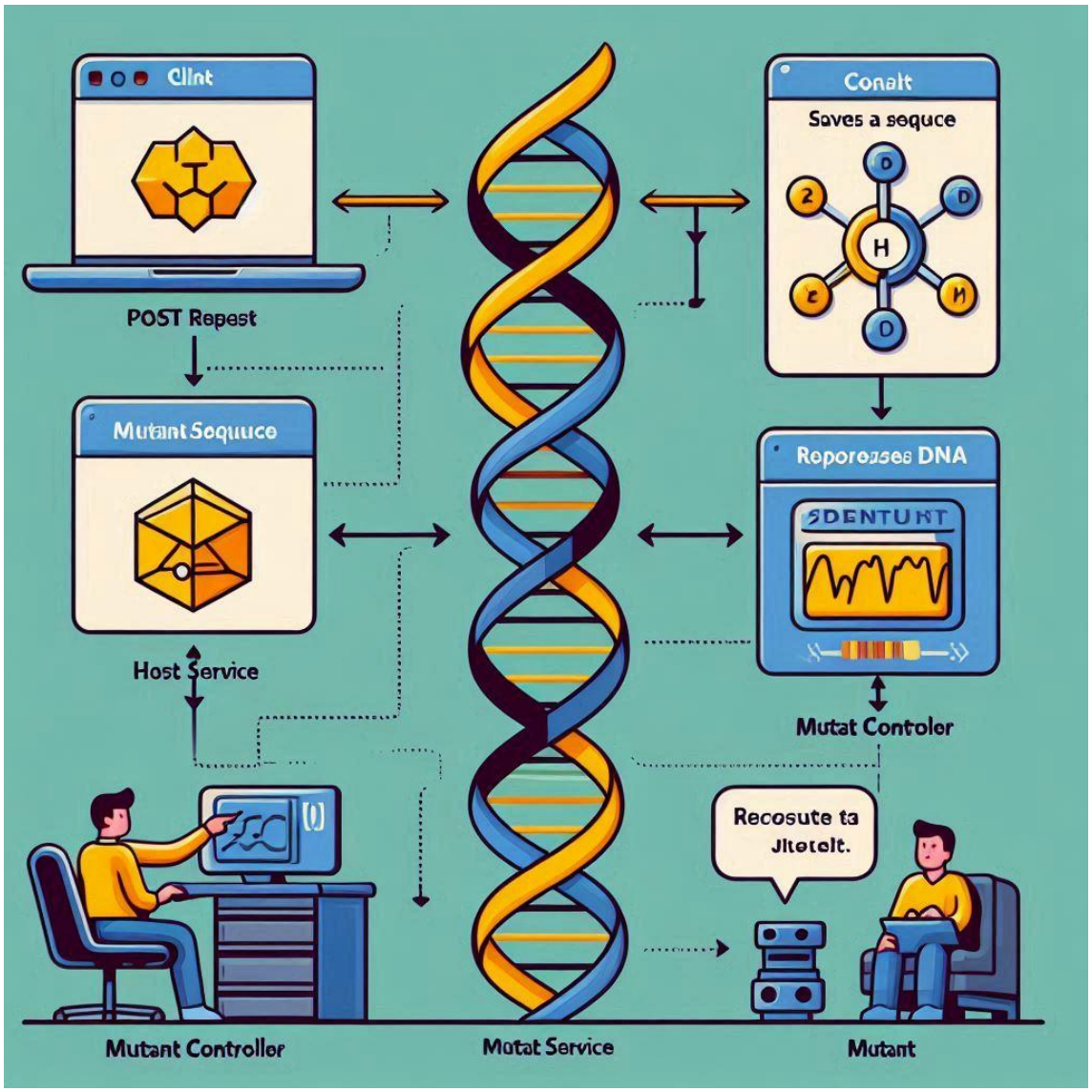


Diagrama de secuencia



Resultados de los Test

MagnetodnaApplicationTests

todos los > com.example.magnetodna > MagnetodnaApplicationTests

5	0	0	2.032s
Pruebas	Fallas	Ignorado	duración

100% Exitoso

Pruebas | Salida estándar

Prueba	Nombre del método	Duración	Resultado
testIsMutant()	testIsMutant()	1.997s	Pasado
[1] dnaString=ATGCGA,CAGTGC,TTATGT,AGAAGG,CCCCTA,TCACTG, esperado=verdadero	testIsMutant(Cadena, booleano)[1]	0.020s	Pasado
[2] dnaString=ATGCGA,CAGTGC,TTATTT,AGACGG,GCGTCA,TCACTG, esperado=falso	testIsMutant(Cadena, booleano)[2]	0.004s	Pasado
[3] dnaString=AAAA,AAAA,AAAA,AAAA, esperado=verdadero	testIsMutant(Cadena, booleano)[3]	0.006s	Pasado
[4] dnaString=AAAA, esperado=verdadero	testIsMutant(Cadena, booleano)[4]	0.005s	Pasado

Líneas ☐ de ajuste
Generado por Gradle 8.9-rc-2 el 14 oct. 2024 00:08:13

MagnetodnaApplicationTests

todos los > com.example.magnetodna > MagnetodnaApplicationTests

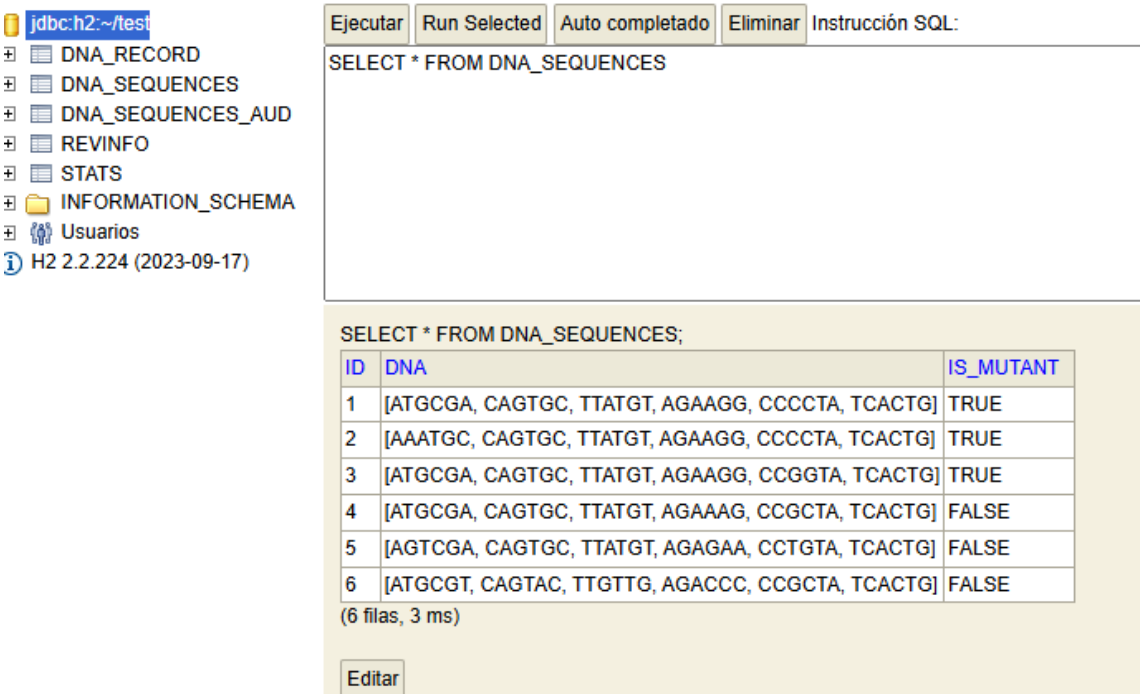
8	0	0	1.249s
Pruebas	Fallas	Ignorado	duración

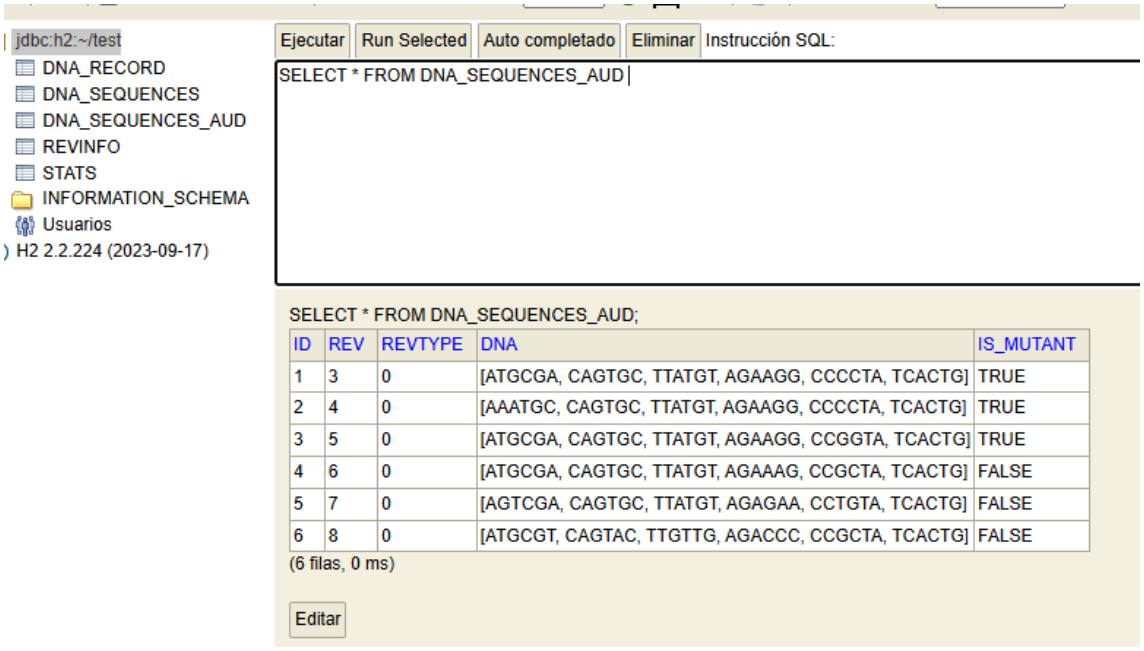
100% Exitoso

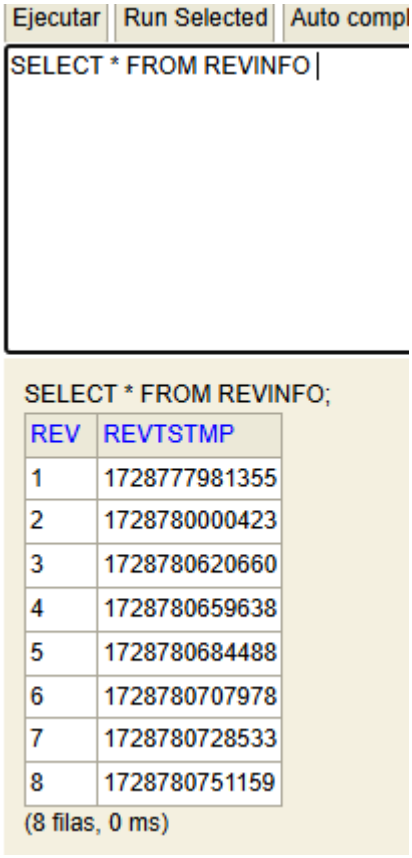
Pruebas | Salida estándar

Prueba	Nombre del método	Duración	Resultado
testEmptyDna()	testEmptyDna()	0.005s	Pasado
testInvalidDnaCharacters()	testInvalidDnaCharacters()	0.005s	Pasado
testIrregularDnaMatrix()	testIrregularDnaMatrix()	0.860s	Pasado
[1] dnaString=ATGCGA,CAGTGC,TTATGT,AGAAGG,CCCCTA,TCACTG, esperado=verdadero	testIsMutant(Cadena, booleano)[1]	0.059s	Pasado
[2] dnaString=ATGCGA,CAGTGC,TTATTT,AGACGG,GCGTCA,TCACTG, esperado=falso	testIsMutant(Cadena, booleano)[2]	0.015s	Pasado
[3] dnaString=AAAA,AAAA,AAAA,AAAA, esperado=verdadero	testIsMutant(Cadena, booleano)[3]	0.007s	Pasado
[4] dnaString=AAAA, esperado=verdadero	testIsMutant(Cadena, booleano)[4]	0.006s	Pasado
testStatsCalculation()	testStatsCalculation()	0.292s	Pasado

DataBase en H2




jdbc:h2:~/test
DNA_RECORD
DNA_SEQUENCES
DNA_SEQUENCES_AUD
REVINFO
STATS
INFORMATION_SCHEMA
Usuarios
H2 2.2.224 (2023-09-17)
Ejecutar
Run Selected
Auto completado
Eliminar
Instrucción SQL:
SELECT * FROM DNA_SEQUENCES_AUD
SELECT * FROM DNA_SEQUENCES_AUD;
ID REV REVTYPE DNA IS_MUTANT
1 3 0 [ATGCGA, CAGTGC, TTATGT, AGAAGG, CCCCTA, TCACTG] TRUE
2 4 0 [AAATGC, CAGTGC, TTATGT, AGAAGG, CCCCTA, TCACTG] TRUE
3 5 0 [ATGCGA, CAGTGC, TTATGT, AGAAGG, CCGGTA, TCACTG] TRUE
4 6 0 [ATGCGA, CAGTGC, TTATGT, AGAAAG, CCGCTA, TCACTG] FALSE
5 7 0 [AGTCGA, CAGTGC, TTATGT, AGAGAA, CCTGTA, TCACTG] FALSE
6 8 0 [ATGCGT, CAGTAC, TTGTTG, AGACCC, CCGCTA, TCACTG] FALSE
(6 filas, 0 ms)
Editar


Ejecutar
Run Selected
Auto compl
SELECT * FROM REVINFO
SELECT * FROM REVINFO;
REV REVTSTMP
1 1728777981355
2 1728780000423
3 1728780620660
4 1728780659638
5 1728780684488
6 1728780707978
7 1728780728533
8 1728780751159
(8 filas, 0 ms)