

UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA

FACULTAD DE INGENIERÍA EN SISTEMAS DE INFORMACIÓN Y CIENCIAS DE LA  
COMPUTACIÓN

PROGRAMACION III

ING. CARLOS JOSÉ MARTÍNEZ BOLAÑOS



## Tarea 11- Hashing SHA-256

	≡ Nombre	≡ Carné
1	Cruz Francisco Estrada Gregorio	7690-23-5339

Plan sábado Virtual 3

10/05/2025

# Explicación

## Qué es SHA-256

Yo lo entiendo de esta manera:

Es como tener una **máquina mágica**. ala cual se le da una palabra o una frase, y ella te devuelve un resultado extraño, una especie de código con letras y números (una encriptación). Ese resultado **siempre tendrá el mismo tamaño** (64 caracteres), **no importa qué tan largo o corto sea lo que se le de**. Esa máquina mágica es lo que llamamos una **función hash**, y en este caso, **SHA-256** es una de las más conocidas y seguras.

## Qué significa SHA-256

- **SHA** = *Secure Hash Algorithm* (Algoritmo Seguro de Hash).
- **256** = El resultado tiene 256 bits (o 64 caracteres en hexadecimal).

Es una **forma matemática de transformar datos en algo que parece aleatorio**, pero que en realidad **siempre produce el mismo resultado para el mismo dato de entrada**

## Para qué se usa

SHA-256 **no sirve para encriptar** cosas que luego quieras recuperar.

**Sirve para verificar cosas**, especialmente:

- Contraseñas (guardarlas de forma segura).
- Archivos (para saber si se han alterado).
- Firmas digitales y criptomonedas (como Bitcoin).

## Por qué es útil para contraseñas

En ejemplo si alguien se mete a una base de datos y las contraseñas están en texto plano como "1234" seria facilísimo, pero si solo tienes el hash: 1234 → 03ac674216f3e15c761ee1a5e255f067953623c8b388b4459e13f978d7c846f4, entonces no puede saber cuál era la contraseña original, solo quien tenga el hash original generado.

SHA-256 es como una licuadora de datos, se mete un texto y obtienes una mezcla irrepetible de números y letras que **no se puede deshacer**.

La usamos para guardar contraseñas de forma segura, porque aunque no podemos leer la contraseña original desde el hash, sí podemos **comparar el hash de lo que escribe un usuario con el hash guardado**, y así saber si es correcta.

Tanto que un solo pequeño cambio de letra o numero afecta completamente el hash devuelto.

	≡ Entrada (contraseña)	≡ SHA-256 Hash
1	hola123	cfb82bba2295ac3a2c...
2	HOLA123	bd9f7769b3a8a2f3a...
3	hola1234	4c3450cb3e950b331...

## Programa

Este programa sirve para **registrar un usuario o iniciar sesión**, pero lo hace de forma muy simple y segura. En lugar de guardar la contraseña tal cual, la convierte en un código especial (un "hash") para que nadie pueda ver la contraseña real, ni siquiera si acceden a la base de datos. todo compuesto por las clases descritas:

Clase Main – donde todo comienza Esta clase es como el menú principal del programa. Al ejecutarlo, me pregunta si quiero: Registrar un nuevo usuario o iniciar sesión.

2. Clase UsuarioDAO – la que habla con la base de datos, Esta clase se encarga de guardar los usuarios o verificarlos al iniciar sesión. Usa una base de datos **MySQL**, y ahí guarda el nombre del usuario y el **hash de la contraseña**, no la contraseña real.

Tiene dos funciones importantes:

**registrarUsuario(nombre, contraseña)**

- Toma el nombre de usuario y la contraseña, llama a HashUtil para convertir la contraseña en un hash y guarda el usuario y el hash en la base de datos.

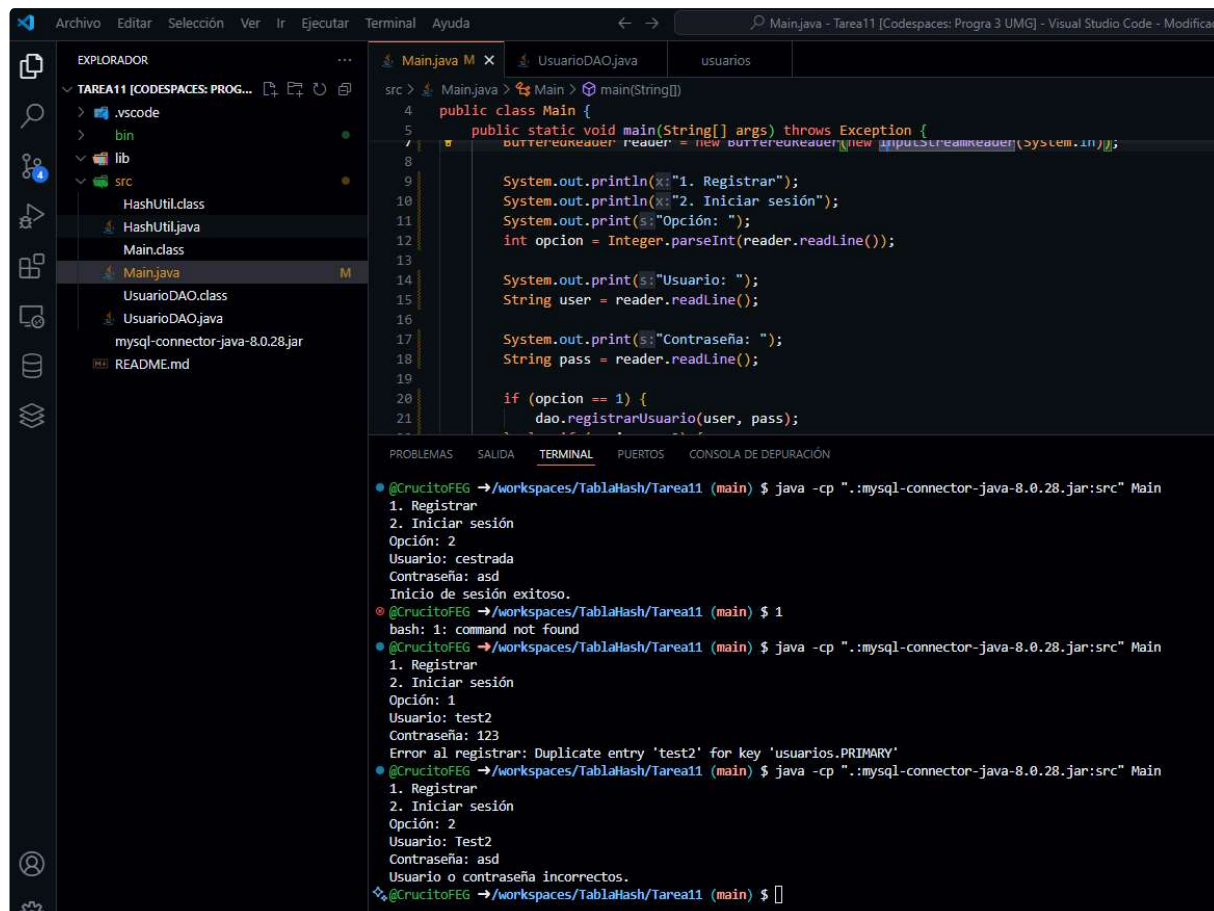
**iniciarSesion(nombre, contraseña)**

- También convierte la contraseña a hash, compara ese hash con el que está en la base de datos y si coinciden, el inicio es exitoso. Si no, dice que es incorrecto. **Esta clase no guarda contraseñas reales, solo códigos hash seguros. Eso protege a los usuarios.**

Para esto vale la pena mencionar que se utilizan las siguientes librerías.

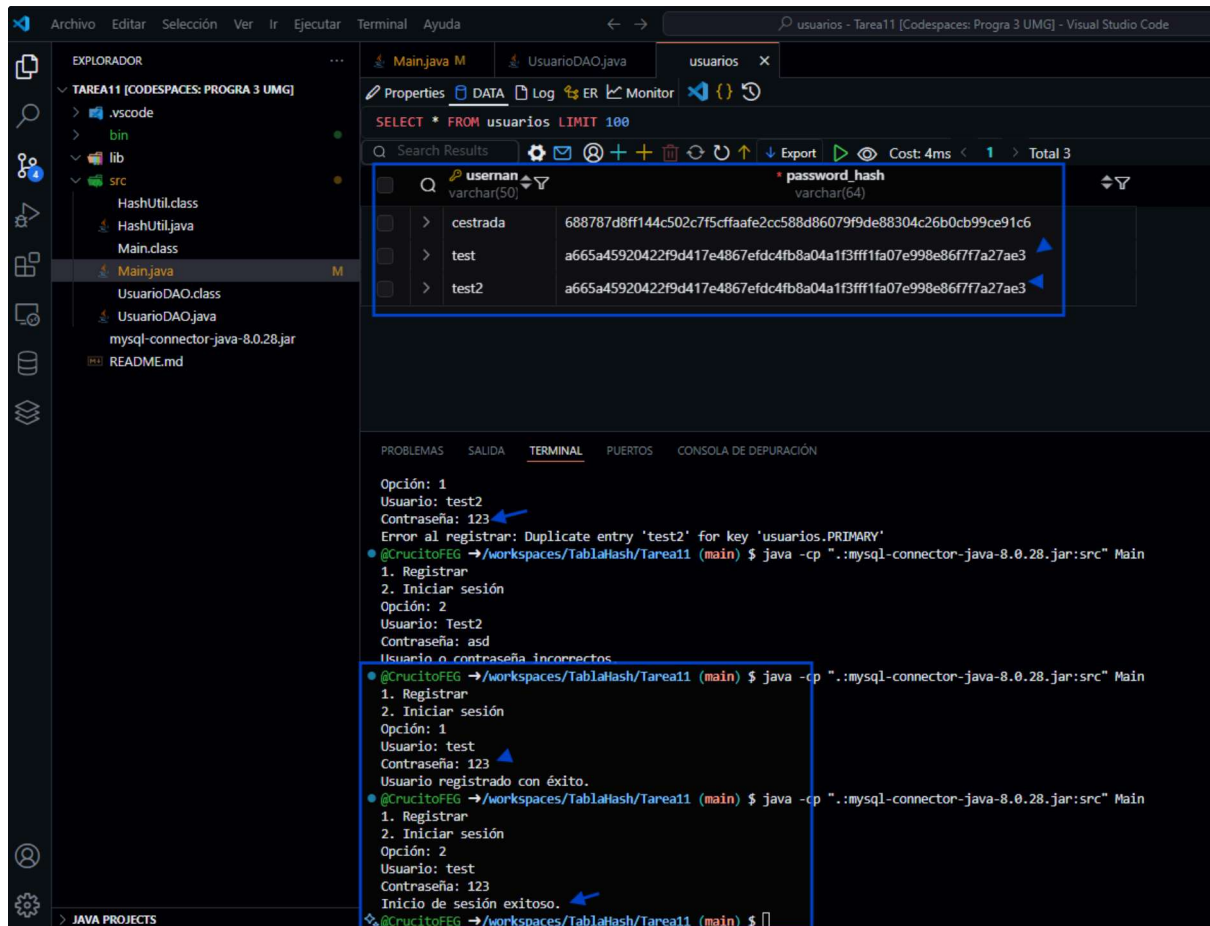
- import java.security.MessageDigest; - que es para el tema de la encriptación
- import java.io.BufferedReader; import java.io.InputStreamReader; - y aquí es para el menú del main.

## Funcionamiento



```
src > Main.java > Main > main(String[])
4 public class Main {
5     public static void main(String[] args) throws Exception {
6         BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));
7
8
9         System.out.println("1. Registrar");
10        System.out.println("2. Iniciar sesión");
11        System.out.print("Opción: ");
12        int opcion = Integer.parseInt(reader.readLine());
13
14        System.out.print("Usuario: ");
15        String user = reader.readLine();
16
17        System.out.print("Contraseña: ");
18        String pass = reader.readLine();
19
20        if (opcion == 1) {
21            dao.registrarUsuario(user, pass);
22        }
23    }
24 }
```

```
@CrucitoFEG →/workspaces/TablaHash/Tarea11 (main) $ java -cp ".:mysql-connector-java-8.0.28.jar:src" Main
1. Registrar
2. Iniciar sesión
Opción: 2
Usuario: cestrada
Contraseña: asd
Inicio de sesión exitoso.
@CrucitoFEG →/workspaces/TablaHash/Tarea11 (main) $ 1
bash: 1: command not found
@CrucitoFEG →/workspaces/TablaHash/Tarea11 (main) $ java -cp ".:mysql-connector-java-8.0.28.jar:src" Main
1. Registrar
2. Iniciar sesión
Opción: 1
Usuario: test2
Contraseña: 123
Error al registrar: Duplicate entry 'test2' for key 'usuarios.PRIMARY'
@CrucitoFEG →/workspaces/TablaHash/Tarea11 (main) $ java -cp ".:mysql-connector-java-8.0.28.jar:src" Main
1. Registrar
2. Iniciar sesión
Opción: 2
Usuario: Test2
Contraseña: asd
Usuario o contraseña incorrectos.
@CrucitoFEG →/workspaces/TablaHash/Tarea11 (main) $
```



## Código Completo

```

1 import java.io.BufferedReader;
2 import java.io.InputStreamReader;
3
4 public class Main {
5     public static void main(String[] args) throws Exception {
6         UsuarioDAO dao = new UsuarioDAO();
7         BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));
8
9         System.out.println("1. Registrar");
10        System.out.println("2. Iniciar sesión");
11        System.out.print("Opción: ");
12        int opcion = Integer.parseInt(reader.readLine());
13
14        System.out.print("Usuario: ");
15        String user = reader.readLine();
16
17        System.out.print("Contraseña: ");
18        String pass = reader.readLine();
19
20        if (opcion == 1) {
21            dao.registrarUsuario(user, pass);
22        } else if (opcion == 2) {
23            dao.iniciarSesion(user, pass);
24        } else {
25            System.out.println("Opción inválida");
26        }
27    }
28 }
29
30 import java.sql.*;

```

```

31
32 public class UsuarioDAO {
33     private static final String URL = "jdbc:mysql://localhost:3306/usuarios_db";
34     private static final String USUARIO = "usuario"; // Cambia si usas otro usuario
35     private static final String PASSWORD = "abc123"; // Cambia si tienes contraseña
36
37     public void registrarUsuario(String username, String passwordPlano) {
38         String passwordHash = HashUtil.hashSHA256(passwordPlano);
39
40         try (Connection conn = DriverManager.getConnection(URL, USUARIO, PASSWORD)) {
41             String sql = "INSERT INTO usuarios (username, password_hash) VALUES (?, ?)";
42             PreparedStatement stmt = conn.prepareStatement(sql);
43             stmt.setString(1, username);
44             stmt.setString(2, passwordHash);
45             stmt.executeUpdate();
46             System.out.println("Usuario registrado con éxito.");
47         } catch (SQLException e) {
48             System.out.println("Error al registrar: " + e.getMessage());
49         }
50     }
51
52     public void iniciarSesion(String username, String passwordPlano) {
53         String passwordHash = HashUtil.hashSHA256(passwordPlano);
54
55         try (Connection conn = DriverManager.getConnection(URL, USUARIO, PASSWORD)) {
56             String sql = "SELECT * FROM usuarios WHERE username = ? AND password_hash = ?";
57             PreparedStatement stmt = conn.prepareStatement(sql);
58             stmt.setString(1, username);
59             stmt.setString(2, passwordHash);
60
61             ResultSet rs = stmt.executeQuery();
62             if (rs.next()) {
63                 System.out.println("Inicio de sesión exitoso.");
64             } else {
65                 System.out.println("Usuario o contraseña incorrectos.");
66             }
67         } catch (SQLException e) {
68             System.out.println("Error al iniciar sesión: " + e.getMessage());
69         }
70     }
71 }
72
73 import java.security.MessageDigest;
74
75 public class HashUtil {
76     public static String hashSHA256(String input) {
77         try {
78             MessageDigest digest = MessageDigest.getInstance("SHA-256");
79             byte[] hashBytes = digest.digest(input.getBytes("UTF-8"));
80             StringBuilder hexString = new StringBuilder();
81             for (byte b : hashBytes) {
82                 hexString.append(String.format("%02x", b));
83             }
84             return hexString.toString();
85         } catch (Exception e) {
86             System.out.println("Error al hashear: " + e.getMessage());
87             return null;
88         }
89     }
90 }

```