PASOS PARA HACER FUNCIONAR EL SISTEMA:

1. DESCARGAR EL PROYECTO
2. UNA VEZ DESCARGADO EL PROYECTO, SE DEBE EJCUTAR EL SIGUIENTE SCRIP EN EL WORKBENCH DE MYSQL:

```
CREATE DATABASE db_proyectokodigo;

USE db_proyectokodigo;
```

3. VERIFICAR QUE EN INTELLIJIDEA EL ARCHIVO: application.yml se vea así: “create”. Además en el caso del PORT se usa el 3306, USER se designó que fuese: “root”, y la PASSWORD es vacía: “ “. Por lo tanto, las varibales de entorno ya están programadas para manejar esos datos.

```yaml
spring:
  application:
    name: proyecto_Inventario
  jackson:
    default-property-inclusion: non_null
    property-naming-strategy: SNAKE_CASE
  datasource:
    url: ${DB_HOST}
    username: ${DB_USER}
    password: ${DB_PASSWORD}
    driver-class-name: com.mysql.cj.jdbc.Driver
  jpa:
    hibernate:
      ddl-auto: create
    show-sql: true

logging:
  level:
    org.springframework: DEBUG
    com.proyectoInventario_Kodigo.proyecto_Inventario: DEBUG
```

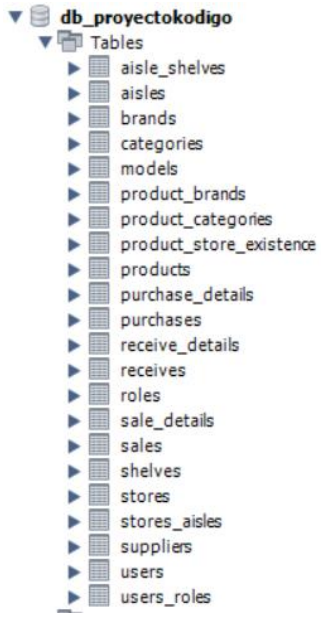
4. Ejecutar spring y esperar ya que por el DEBUG aparecerán muchos logs, hasta lograr esto:

```
: Started ProyectoInventarioApplication in 11.299 seconds (process running for 12.009)
: Application availability state LivenessState changed to CORRECT
: Application availability state ReadinessState changed to ACCEPTING_TRAFFIC
```

5. Luego colocar en el application.yml la siguiente palabra: “update”:

```yaml
jpa:
  hibernate:
    ddl-auto: update
  show-sql: true
```

6. Luego es necesario que en la base de datos se vea así:



7. Luego es necesario hacer estos insert:

```
INSERT INTO roles (name) VALUES ("ROLE_ADMIN");

INSERT INTO roles (name) VALUES ("ROLE_USER");

SELECT * FROM roles;
```

8. Luego es necesario ir a postman y colocar la siguiente URL:
   POST: http://localhost:8080/users
   Crea cualquier usuario:

```
{

"username": "admin",

"email": "admin@gmail.com",

"password": "12345",

"admin": true

}
```

9. Luego hace logín con la siguiente URL:
   POST: http://localhost:8080/login

```
{

  "username": "admin",

  "password": "12345"

}
```

10. Cada vez que haga “login” necesita ver lo siguiente:

Ese token debe de usarlo en cada endoint que desee realizar. Para eso se explica en el video y en la documentación. **Video en minuto 19:50** se empiezan a utilizar tanto en SWAGGER como en POSTMAN.