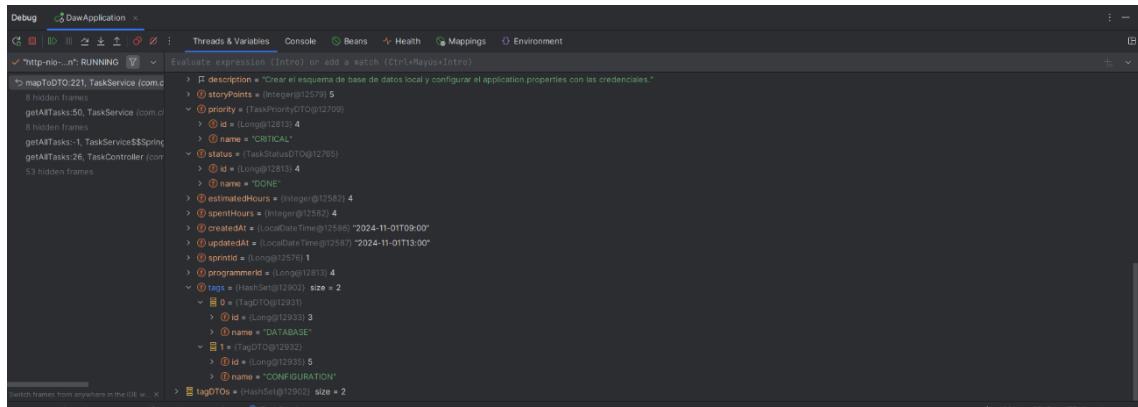


## EJERCICIOS SPRING BOOT.

**Subir las evidencias necesarias para demostrar que se ha realizado las siguientes actividades**

### Actividad 1

Realizar la relación many to many de etiquetas con tareas.



### Actividad 2

Introducir datos de prueba 10 tareas, 5 programadores, 2 sprints y 10 etiquetas. Los datos de prueba deben ser reales

#### -- 1. LIMPIEZA DE DATOS

-- Elimino datos de todas las tablas en orden para evitar errores de Foreign Key.  
-- RESTART IDENTITY reinicia los contadores auto-incrementales a 1.

TRUNCATE TABLE

```
task_tag,  
task,  
tag,  
sprint,  
programmer,  
rol_programmer,  
sprint_status,  
task_priority,  
task_status,  
usuario  
RESTART IDENTITY CASCADE;
```

#### -- 2. INSERCIÓN DE DATOS MAESTROS / AUXILIARES

-- Roles de Programador (Necesarios para crear Programadores)

```
INSERT INTO rol_programmer (name) VALUES  
    ('Junior Developer'),  
    ('Senior Developer'),  
    ('Tech Lead'),
```

```

('QA Engineer'),
('DevOps Engineer');

-- Estados de Sprint (Necesarios para crear Sprints)
INSERT INTO sprint_status (name) VALUES
    ('Planning'),
    ('In Progress'),
    ('Finished'),
    ('Review');

-- Prioridades de Tarea
INSERT INTO task_priority (name) VALUES
    ('Low'),
    ('Medium'),
    ('High'),
    ('Critical');

-- Estados de Tarea
INSERT INTO task_status (name) VALUES
    ('To Do'),
    ('In Progress'),
    ('Blocked'),
    ('Done');

-- 3. INSERCIÓN DE DATOS PRINCIPALES

-- Programadores
INSERT INTO programmer (name, email, capacity, rol_id) VALUES
    ('Ana García', 'ana.garcia@devteam.com', 5, 1), -- Junior
    ('Carlos Ruiz', 'carlos.ruiz@devteam.com', 8, 2), -- Senior
    ('Elena Martín', 'elena.martin@devteam.com', 10, 3), -- Lead
    ('David López', 'david.lopez@devteam.com', 6, 4), -- QA
    ('Sofía Hernández', 'sofia.hernandez@devteam.com', 7, 5); -- DevOps

-- Sprints
INSERT INTO sprint (name, start_date, end_date, goal, status_id) VALUES
    ('Sprint 1: Core', '2023-10-01', '2023-10-14', 'Implementación del núcleo de la aplicación y base de datos', 3), -- Finished
    ('Sprint 2: UI & Features', '2023-10-15', '2023-10-29', 'Desarrollo de interfaz de usuario y funcionalidades principales', 2); -- In Progress

-- Etiquetas / Tags
INSERT INTO tag (name) VALUES
    ('Backend'),
    ('Frontend'),
    ('Database'),
    ('API REST'),
    ('Security'),
    ('UI/UX'),
    ('Bug'),
    ('Refactoring'),
    ('Documentation'),
    ('Testing');

-- Tareas
-- Relacionamos con Sprints (1 y 2), Programadores (1-5), Prioridades y Estados.
INSERT INTO task (title, description, story_points, estimated_hours, spent_hours, created_at, updated_at, priority_id, status_id, sprint_id, programmer_id) VALUES
    ('Diseño de Base de Datos', 'Crear diagrama ER y scripts SQL iniciales', 5, 8, 8, NOW(), NOW(), 3, 4, 1, 3), -- Done, Lead
    ('Configuración Inicial Spring', 'Setup del proyecto, dependencias y estructura de carpetas', 3, 4, 4, NOW(), NOW(), 2, 4, 1, 5), --

```

#### Done, DevOps

('Implementar Entidades JPA', 'Crear clases @Entity para el modelo de datos', 5, 10, 10, NOW(), NOW(), 3, 4, 1, 1), -- Done, Junior  
('Configurar Seguridad', 'Implementar Spring Security básico', 8, 16, 16, NOW(), NOW(), 4, 4, 1, 2), -- Done, Senior  
('API: CRUD Usuarios', 'Endpoints para gestión de usuarios', 3, 6, 6, NOW(), NOW(), 2, 4, 1, 1), -- Done, Junior

#### -- Tareas del Sprint 2

('Maquetación Dashboard', 'Diseñar vista principal con HTML/CSS', 5, 10, 5, NOW(), NOW(), 2, 2, 2, 1), -- In Progress, Junior  
('Integración API Tareas', 'Conectar frontend con endpoint de tareas', 5, 8, 2, NOW(), NOW(), 3, 2, 2, 2), -- In Progress, Senior  
('Corrección Login Bug', 'Arreglar error al redirigir tras login fallido', 2, 4, 0, NOW(), NOW(), 4, 1, 2, 4), -- To Do, QA  
('Optimización Consultas', 'Mejorar performance en listado de sprints', 3, 5, 0, NOW(), NOW(), 1, 1, 2, 3), -- To Do, Lead  
('Escribir Tests Unitarios', 'Cubrir servicios principales con JUnit', 8, 12, 0, NOW(), NOW(), 2, 1, 2, 4); -- To Do, QA

#### -- 4. RELACIONES MUCHOS A MUCHOS

INSERT INTO task\_tag (task\_id, tag\_id) VALUES

(1, 3), (1, 1), -- Tarea 1: Database, Backend  
(2, 1), (2, 5), -- Tarea 2: Backend, Security  
(3, 1), (3, 3), -- Tarea 3: Backend, Database  
(4, 5), (4, 1), -- Tarea 4: Security, Backend  
(5, 4), (5, 1), -- Tarea 5: API REST, Backend  
(6, 2), (6, 6), -- Tarea 6: Frontend, UI/UX  
(7, 2), (7, 4), -- Tarea 7: Frontend, API REST  
(8, 7), (8, 6), -- Tarea 8: Bug, UI/UX  
(9, 8), (9, 3), -- Tarea 9: Refactoring, Database  
(10, 10), (10, 1); -- Tarea 10: Testing, Backend

```
[2025-12-10 20:43:50] 10 rows affected in 5 ms
postgres> INSERT INTO task (title, description, story_points, estimated_hours, spent_hours, created_at, updated_at, priority_id, status_id, sprint_id, programmer_id) VALUES
-- Tareas del Sprint 1
('Diseño de Base de Datos', 'Crear diagrama ER y scripts SQL iniciales', 5, 8, 8, NOW(), NOW(), 3, 4, 1, 3), -- Done, Lead
('Configuración Inicial Spring', 'Setup del proyecto, dependencias y estructura de carpetas', 3, 4, 4, NOW(), NOW(), 2, 4, 1, 5), -- Done, DevOps
('Implementar Entidades JPA', 'Crear clases @Entity para el modelo de datos', 5, 10, 10, NOW(), NOW(), 3, 4, 1, 1), -- Done, Junior
('Configurar Seguridad', 'Implementar Spring Security básico', 8, 16, 16, NOW(), NOW(), 4, 4, 1, 2), -- Done, Senior
('API: CRUD Usuarios', 'Endpoints para gestión de usuarios', 3, 6, 6, NOW(), NOW(), 2, 4, 1, 1), -- Done, Junior

-- Tareas del Sprint 2
('Maquetación Dashboard', 'Diseñar vista principal con HTML/CSS', 5, 10, 5, NOW(), NOW(), 2, 2, 2, 1), -- In Progress, Junior
('Integración API Tareas', 'Conectar frontend con endpoint de tareas', 5, 8, 2, NOW(), NOW(), 3, 2, 2, 2), -- In Progress, Senior
('Corrección Login Bug', 'Arreglar error al redirigir tras login fallido', 2, 4, 0, NOW(), NOW(), 4, 1, 2, 4), -- To Do, QA
('Optimización Consultas', 'Mejorar performance en listado de sprints', 3, 5, 0, NOW(), NOW(), 1, 1, 2, 3), -- To Do, Lead
('Escribir Tests Unitarios', 'Cubrir servicios principales con JUnit', 8, 12, 0, NOW(), NOW(), 2, 1, 2, 4)
[2025-12-10 20:43:50] 10 rows affected in 7 ms
postgres> INSERT INTO task_tag (task_id, tag_id) VALUES
(1, 3), (1, 1), -- Tarea 1: Database, Backend
(2, 1), (2, 5), -- Tarea 2: Backend, Security
(3, 1), (3, 3), -- Tarea 3: Backend, Database
(4, 5), (4, 1), -- Tarea 4: Security, Backend
(5, 4), (5, 1), -- Tarea 5: API REST, Backend
(6, 2), (6, 6), -- Tarea 6: Frontend, UI/UX
(7, 2), (7, 4), -- Tarea 7: Frontend, API REST
(8, 7), (8, 6), -- Tarea 8: Bug, UI/UX
(9, 8), (9, 3), -- Tarea 9: Refactoring, Database
(10, 10), (10, 1)

[2025-12-10 20:43:50] 20 rows affected in 6 ms
```

## Actividad 3

Realizar una consulta nativa que busque tareas que busque por estimación de horas y horas invertidas

```
List<Task> listTasksByEstimatedHoursAndSpentHours(@Param("estimatedHours") Integer estimatedHours, @Param("spentHours") Integer spentHours);
```

```
//consulta nativa que busque tareas que busque por estimación de horas y horas invertidas
@Query(value = "SELECT * FROM task WHERE estimated_hours = :estimatedHours AND spent_hours = :spentHours", nativeQuery = true) no usages new *
List<Task> listTasksByEstimatedHoursAndSpentHours(@Param("estimatedHours") Integer estimatedHours, @Param("spentHours") Integer spentHours);
```

Prueba de que funciona:

The screenshot shows a database interface with a code editor at the top containing Java code for a repository. Below it is a results grid showing a single row of data from a table named 'task'. The columns include 'id', 'created\_at', 'description', 'estimated\_hours', 'spent\_hours', 'story\_points', 'title', and 'updated\_at'. The data is as follows:

	id	created_at	description	estimated_hours	spent_hours	story_points	title	updated_at
1	1	2025-12-10 19:45:50.737685	Crear diagrama ER y scripts SQL iniciales	8	8	5	Diseño de Base de Datos	2025-12-10 19:45:50.737685

## Actividad 4

Realizar una consulta nativa que busque por objetivo y lista de sprints.

```
//Actividad 4 Realizar una consulta nativa que busque por objetivo y lista de sprints.
@Query(value = "SELECT * FROM sprint WHERE goal LIKE '%' || :goal || '%' AND id IN :ids", nativeQuery = true) no usages new *
List<Sprint> findByGoalContainingAndIdIn(@Param("goal") String goal, @Param("ids") List<Long> ids);
```

Prueba de que funciona:

The screenshot shows a database interface with a code editor at the top containing Java code for a repository. Below it is a results grid showing a single row of data from a table named 'sprint'. The columns include 'id', 'end\_date', 'goal', 'name', 'start\_date', and 'status\_id'. The data is as follows:

	id	end_date	goal	name	start_date	status_id
1	2	2025-10-29	Desarrollo de interfaz de usuario y funcionalidades principales	Sprint 2: UI & Features	2023-10-15	2

## EJERCICIO EXTRA

Realizar una query nativa que reciba una lista de ids de tareas y busque mediante “IN” esa lista de tareas.

```

43
44 //Realizar una query nativa que reciba una lista de ids de tareas y busque mediante "IN" esa lista de tareas.
45     @Query(value = "SELECT * FROM task WHERE id IN :taskIds", nativeQuery = true) no usages new *
46     List<Task> listTasksByTaskIds(@Param("taskIds") List<Long> taskIds);
47

```

## Prueba de que funciona:

The screenshot shows a PostgreSQL database interface. On the left, there's a sidebar with navigation links like 'Story', 'StoryRepository', 'Story', 'TaskRepository', 'Task', 'Task', 'TaskStatus', 'TaskTag', 'User', 'Sequences', 'Database Objects', and 'Server Objects'. The main area has tabs for 'Output', 'postgres.public.task' (which is selected), and 'Tx'. The 'postgres.public.task' tab displays a table with the following data:

	id	created_at	description	estimated_hours	spent_hours	story_points	title
1	1	2025-12-10 19:43:50.737685	Crear diagrama ER y scripts SQL iniciales	8	8	5	Diseño de Base de Datos
2	2	2025-12-10 19:43:50.737685	Setup del proyecto, dependencias y estructura de carpetas	4	4	3	Configuración Inicial Sp
3	3	2025-12-10 19:43:50.737685	Crear clases Entity para el modelo de datos	10	10	5	Implementar Entidades JP
4	4	2025-12-10 19:43:50.737685	Implementar Spring Security básico	16	16	8	Configurar Seguridad
5	5	2025-12-10 19:43:50.737685	Endpoints para gestión de usuarios	6	6	3	API: CRUD Usuarios
6	6	2025-12-10 19:43:50.737685	Arreglar error al redirigir tras login fallido	4	0	2	Corrección Login Bug
7	7	2025-12-10 19:43:50.737685	Mejorar performance en listado de sprints	5	0	3	Optimización Consultas