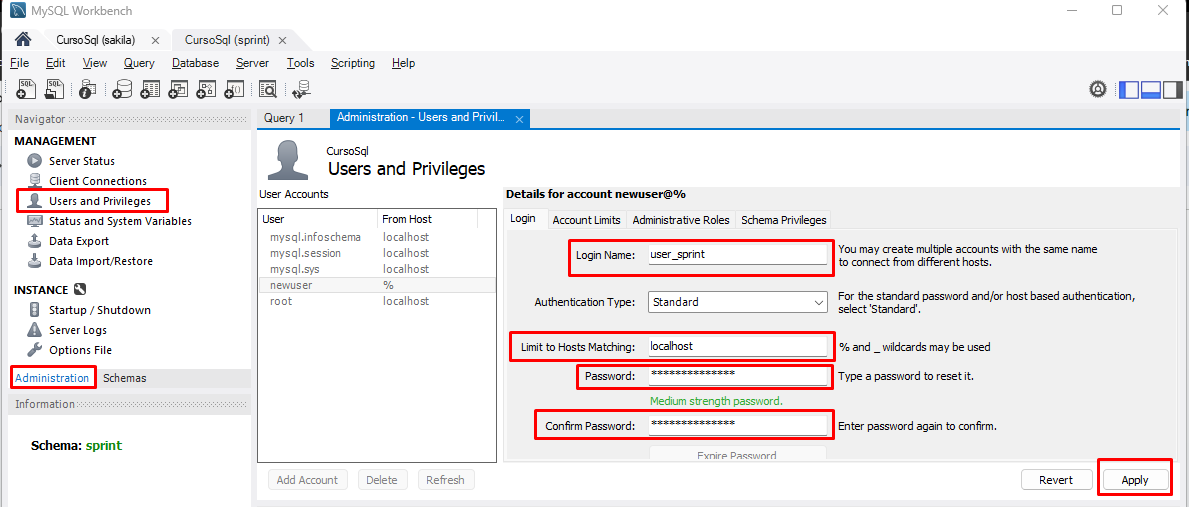
**Evaluación Final Módulo 5 Base Datos**

**Sprint**

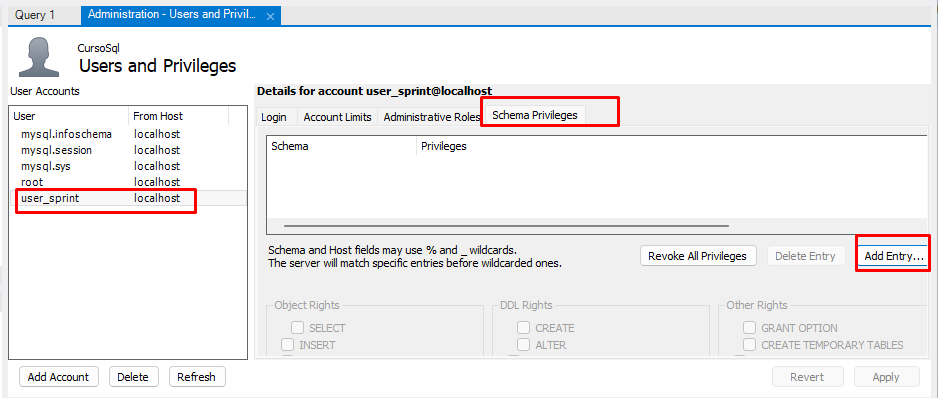
**Integrantes: Bastián Poloni, Paula Rauld, Boris Guíñez, Nibaldo Quezada y Paula Macaya**.

**Repositorio GitHub:** [**https://github.com/nquezada1981/SPRINT-5**](https://github.com/nquezada1981/SPRINT-5)

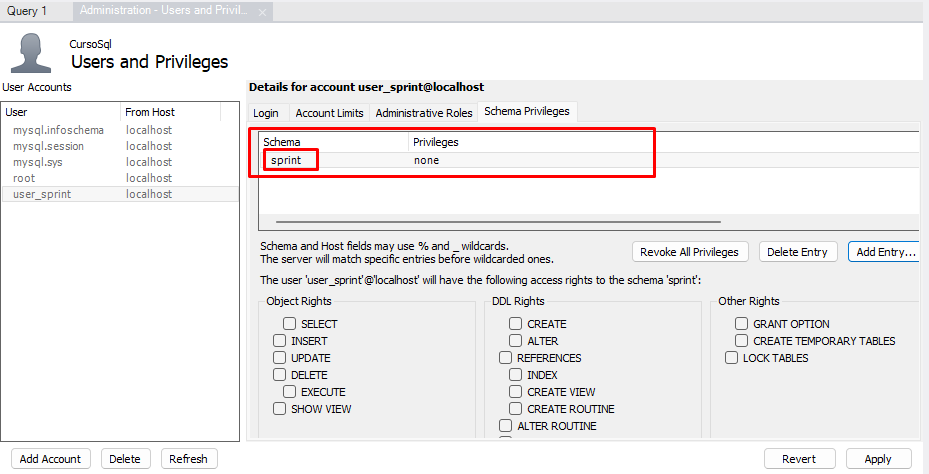
* **Crear base de datos y usuarios con privilegios pertinentes para la realización del ejercicio**
* Entrar a workbench con la única conexión existente y su usuario root.
* Una vez dentro entra a administración, luego user add privileges y se abrirá una nueva ventana.
* En el input Login Name: Se crea el usuario a utilizar en el sprint llamado user\_sprint.
* En Limit to Hosts Matching: Se ingresa localhost porque será el espacio que aloja las base de datos localmente.
* Se crea un password para este usuario.



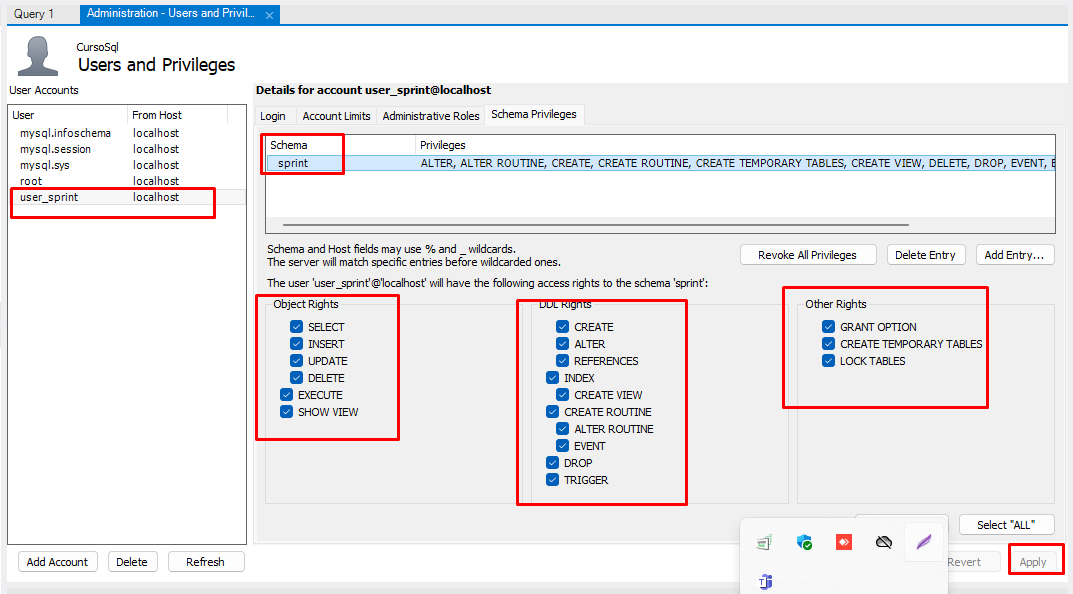
* Una vez creado el usuario entrar a opción schema privileges.
* pinchar Add Entry



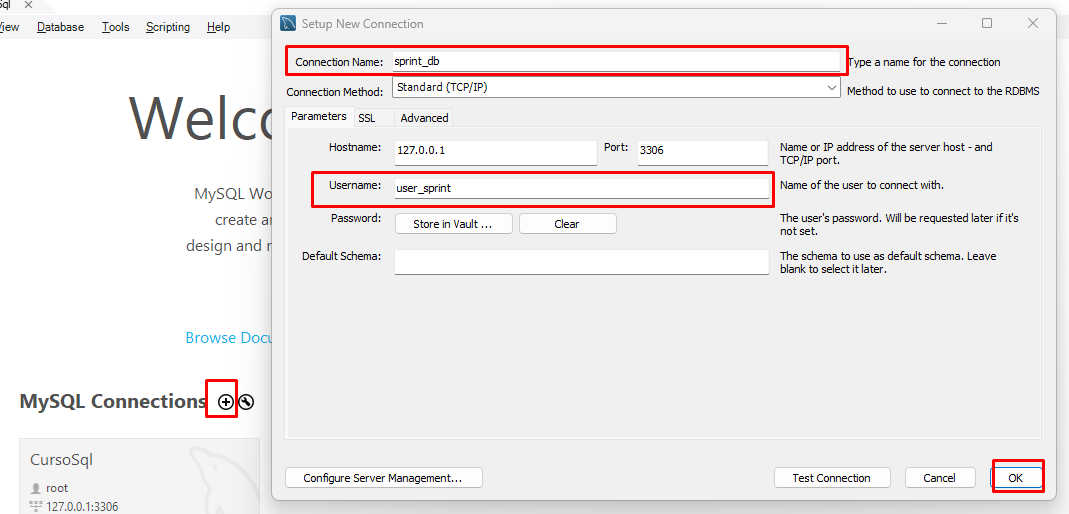
* Se desplegará otra ventana para escoger cuál o cuáles serán las bases de datos a la cual el usuario creado tendrá accesos.
* Ahora vemos que el usuario tiene acceso a la base de datos sprint, en la cual se le asignarán los privilegios para que pueda desarrollar el sprint..



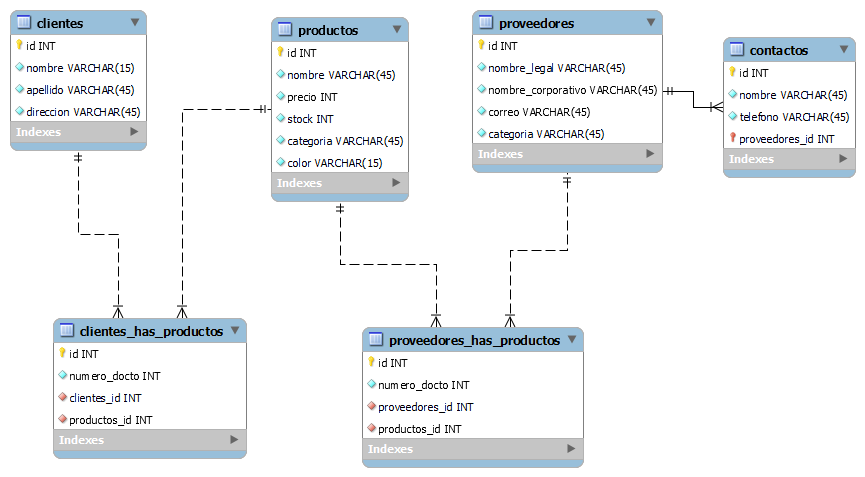
* Le daremos todos los permisos, ya que, tiene que crear tablas e insertar y modificar.

.

* Finalmente creamos una nueva conexión con el usuario creado.



* **Elaborar Modelo Entidad Relación coherentes y de acuerdo a los estándares.**



* **Creación de Tablas**

-- MySQL Workbench Forward Engineering

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

-- -----------------------------------------------------

-- Schema mydb

-- -----------------------------------------------------

-- -----------------------------------------------------

-- Schema sprint

-- -----------------------------------------------------

-- -----------------------------------------------------

-- Creación de base sprint

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `sprint` DEFAULT CHARACTER SET utf8mb3 ;

USE `sprint` ;

-- -----------------------------------------------------

-- Creacion Tabla Clientes `sprint`.`clientes`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `sprint`.`clientes` (

`id` INT NOT NULL AUTO\_INCREMENT,

`nombre` VARCHAR(15) NOT NULL,

`apellido` VARCHAR(45) NOT NULL,

`direccion` VARCHAR(45) NOT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

-- -----------------------------------------------------

-- Creación Tabla Proveedores `sprint`.`proveedores`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `sprint`.`proveedores` (

`id` INT NOT NULL AUTO\_INCREMENT,

`nombre\_legal` VARCHAR(45) NOT NULL,

`nombre\_corporativo` VARCHAR(45) NOT NULL,

`correo` VARCHAR(45) NOT NULL,

`categoria` VARCHAR(45) NOT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

-- -----------------------------------------------------

-- Creación Tabla Contactos `sprint`.`contactos`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `sprint`.`contactos` (

`id` INT NOT NULL AUTO\_INCREMENT,

`nombre` VARCHAR(45) NOT NULL,

`telefono` VARCHAR(45) NOT NULL,

`proveedores\_id` INT NOT NULL,

PRIMARY KEY (`id`, `proveedores\_id`),

INDEX `fk\_contactos\_proveedores\_idx` (`proveedores\_id` ASC) VISIBLE,

CONSTRAINT `fk\_contactos\_proveedores`

FOREIGN KEY (`proveedores\_id`)

REFERENCES `sprint`.`proveedores` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

-- -----------------------------------------------------

-- Creación Tabla Productos `sprint`.`productos`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `sprint`.`productos` (

`id` INT NOT NULL AUTO\_INCREMENT,

`nombre` VARCHAR(45) NOT NULL,

`precio` INT NOT NULL,

`stock` INT NOT NULL,

`categoria` VARCHAR(45) NOT NULL,

`color` VARCHAR(15) NOT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

-- -----------------------------------------------------

-- Creación Tabla relación `sprint`.`proveedores\_has\_productos`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `sprint`.`proveedores\_has\_productos` (

`id` INT NOT NULL AUTO\_INCREMENT,

`numero\_docto` INT NOT NULL,

`proveedores\_id` INT NOT NULL,

`productos\_id` INT NOT NULL,

PRIMARY KEY (`id`),

INDEX `fk\_proveedores\_has\_productos\_productos1\_idx` (`productos\_id` ASC) VISIBLE,

INDEX `fk\_proveedores\_has\_productos\_proveedores1\_idx` (`proveedores\_id` ASC) VISIBLE,

CONSTRAINT `fk\_proveedores\_has\_productos\_productos1`

FOREIGN KEY (`productos\_id`)

REFERENCES `sprint`.`productos` (`id`),

CONSTRAINT `fk\_proveedores\_has\_productos\_proveedores1`

FOREIGN KEY (`proveedores\_id`)

REFERENCES `sprint`.`proveedores` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

-- -----------------------------------------------------

-- Creación Tabla relación `sprint`.`clientes\_has\_productos`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `sprint`.`clientes\_has\_productos` (

`id` INT NOT NULL AUTO\_INCREMENT,

`numero\_docto` INT NOT NULL,

`clientes\_id` INT NOT NULL,

`productos\_id` INT NOT NULL,

INDEX `fk\_clientes\_has\_productos\_productos1\_idx` (`productos\_id` ASC) VISIBLE,

INDEX `fk\_clientes\_has\_productos\_clientes1\_idx` (`clientes\_id` ASC) VISIBLE,

PRIMARY KEY (`id`),

CONSTRAINT `fk\_clientes\_has\_productos\_clientes1`

FOREIGN KEY (`clientes\_id`)

REFERENCES `sprint`.`clientes` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_clientes\_has\_productos\_productos1`

FOREIGN KEY (`productos\_id`)

REFERENCES `sprint`.`productos` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

* **Inserta 5 Proveedores**

**-- Inserción de proveedores**

INSERT INTO proveedores (nombre\_legal, nombre\_corporativo, correo, categoria)

VALUES

('Juan Pérez', 'ElectroCorp', 'juanperez@example.com', 'Electrónica'),

('María Gómez', 'FashionStyle', 'mariagomez@example.com', 'Moda'),

('Carlos Rodríguez', 'HogarIdeal', 'carlosrodriguez@example.com', 'Hogar y Decoración'),

('Ana Martínez', 'SportFit', 'anamartinez@example.com', 'Deportes y Fitness'),

('Luis Sánchez', 'AlimentosDeliciosos', 'luissanchez@example.com', 'Alimentación');

**-- Inserción de contactos**

INSERT INTO `sprint`.`contactos` (`nombre`, `telefono`, `proveedores\_id`) VALUES

('Juan Pérez', '+56912345678', 1),

('María Gómez', '+56923456789', 1),

('Pedro Rodríguez', '+56934567890', 2),

('Carolina López', '+56945678901', 3),

('Luis Torres', '+56956789012', 4),

('Ana Martínez', '+56967890123', 4),

('Diego Herrera', '+56978901234', 4),

('Valentina Silva', '+56989012345', 5);

* **Inserta 5 Clientes**

INSERT INTO `sprint`.`clientes` (`nombre`, `apellido`, `direccion`) VALUES

('Alejandra', 'González', 'Calle 123'),

('Juan', 'López', 'Avenida Principal 456'),

('María', 'Rodríguez', 'Calle Secundaria 789'),

('Pedro', 'Fernández', 'Avenida Central 101'),

('Carolina', 'Torres', 'Calle Principal 234');

* **Inserción de datos en la tabla "proveedores\_has\_productos"**

INSERT INTO `sprint`.`proveedores\_has\_productos` (`numero\_docto`, `proveedores\_id`, `productos\_id`) VALUES

(12345, 1, 1),

(23456, 2, 2),

(34567, 3, 3),

(45678, 4, 4),

(56789, 5, 5),

(67890, 1, 6),

(78901, 2, 7),

(89012, 3, 8),

(90123, 4, 9),

(12345, 5, 10);

* **Inserción de datos en la tabla "clientes\_has\_productos"**

INSERT INTO `sprint`.`clientes\_has\_productos` (`numero\_docto`, `clientes\_id`, `productos\_id`) VALUES

(54321, 1, 1),

(65432, 2, 2),

(76543, 3, 3),

(87654, 4, 4),

(98765, 5, 5),

(09876, 1, 6),

(10987, 2, 7),

(21098, 3, 8),

(32109, 4, 9),

(43210, 5, 10);

* **SQL Querys**
* **Cuál es la categoría de productos que más se repite.**

SELECT categoria, COUNT(\*) AS repeticiones

FROM productos

GROUP BY categoria

ORDER BY repeticiones DESC

LIMIT 1;



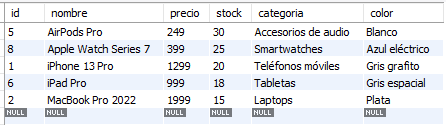
* **Cuáles son los productos con mayor stock**

SELECT \*

FROM productos

ORDER BY stock DESC

LIMIT 5;



* **Qué color de producto es más común en nuestra tienda.**

SELECT color, COUNT(\*) AS cantidad

FROM productos

GROUP BY color

ORDER BY cantidad DESC

LIMIT 1;



* **Cual o cuales son los proveedores con menor stock de productos.**

SELECT p.id, p.nombre\_corporativo, (

SELECT SUM(stock)

FROM productos

WHERE id IN (

SELECT productos\_id

FROM proveedores\_has\_productos

WHERE proveedores\_id = p.id

)

) AS total\_stock

FROM proveedores p

ORDER BY total\_stock ASC

LIMIT 1;

****

* **Cambien la categoría de productos más popular por ‘Electrónica y computación’.**

UPDATE productos

SET categoria = 'Electrónica y computación'

WHERE id = (

SELECT productos\_id

FROM (

SELECT productos\_id, COUNT(\*) AS cantidad

FROM clientes\_has\_productos

GROUP BY productos\_id

ORDER BY cantidad DESC

LIMIT 1

) AS subquery

);



