**ANALISIS Y DESARROLLO DE SOFTWARE.**

**❖ Evidencia de producto: GA6-220501096-AA2-EV03 Script bases de datos del proyecto**

**Nancy Claritza jurado**

**CC :31569535**

**Ficha: 2721479**

**Profesor: Marín Emilio Ordonez**

**CENTRO PARA EL DESARROLLO TECNOLÓGICO DE LA CONSTRUCCIÓN Y LA INDUSTRIA**

**REGIONAL QUINDIO**

****

**Introducción**

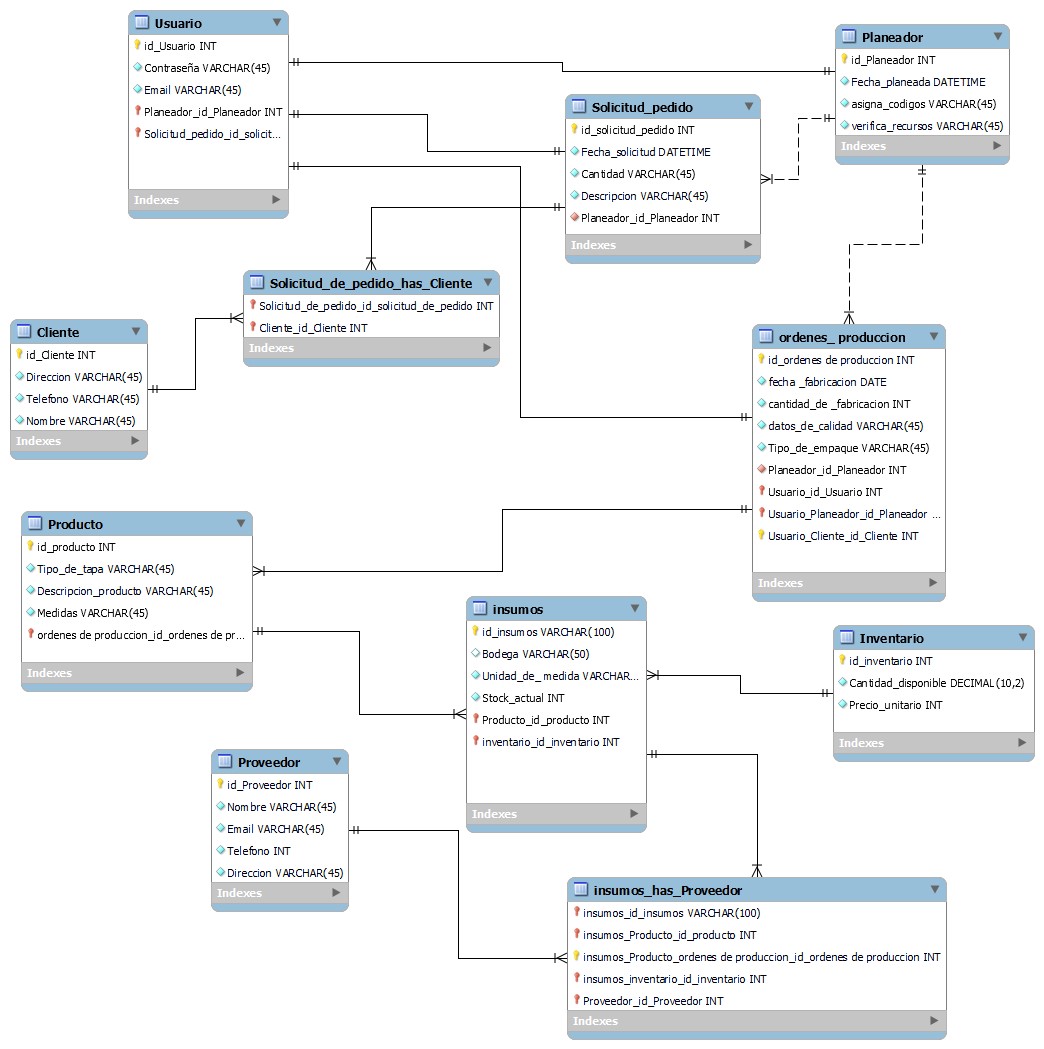
Los scripts de MySQL son un conjunto de instrucciones escritas en lenguaje SQL (Structured Query Language) que se ejecutan secuencialmente para realizar tareas específicas en una base de datos MySQL. Estos scripts pueden ser utilizados para diversas operaciones, como:

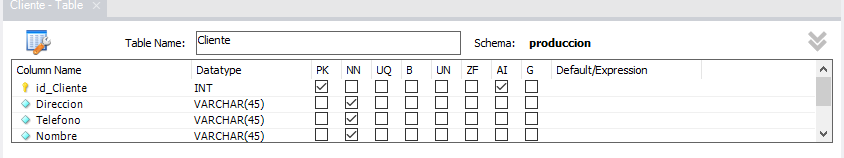
* **Crear y administrar bases de datos:** Puedes crear nuevas bases de datos, eliminar bases de datos existentes, modificar la configuración de las bases de datos, etc.
* **Crear y administrar tablas:** Puedes crear nuevas tablas en una base de datos, eliminar tablas existentes, modificar la estructura de las tablas (agregar o eliminar columnas), etc.
* **Insertar, actualizar y eliminar datos:** Puedes insertar nuevos registros en las tablas, actualizar datos existentes en las tablas y eliminar registros de las tablas.
* **Realizar consultas sobre los datos:** Puedes realizar consultas SQL complejas para extraer información específica de las tablas, filtrar datos, ordenar datos, realizar cálculos, etc.
* **Automatizar tareas:** Puedes automatizar tareas repetitivas de administración de bases de datos, como la creación de copias de seguridad, la restauración de datos, la migración de datos, etc.

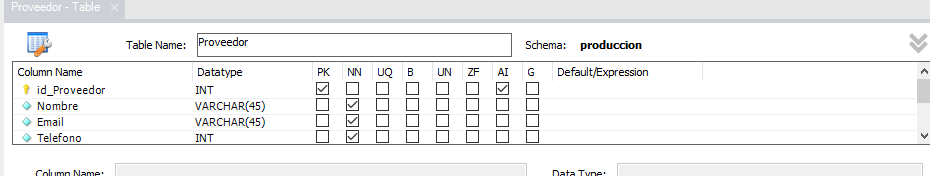
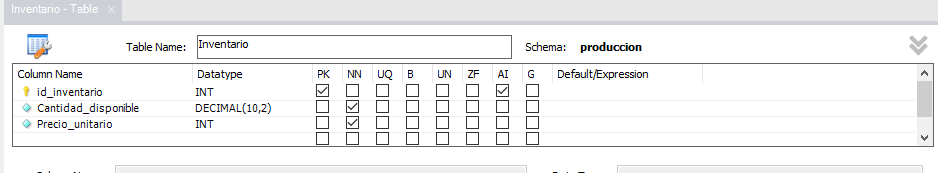
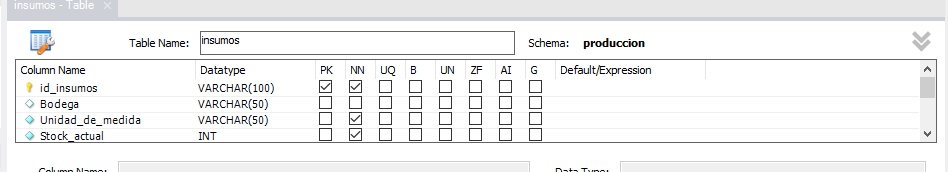
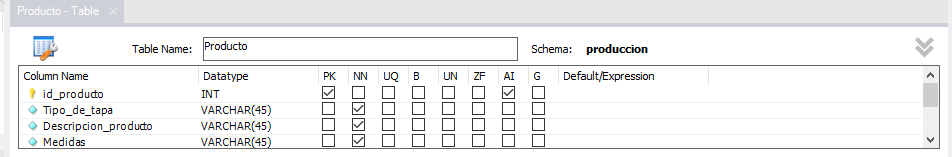
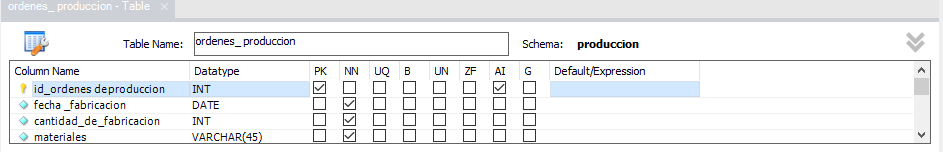
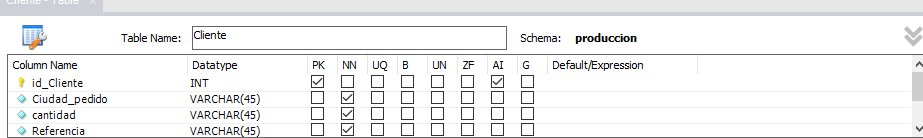
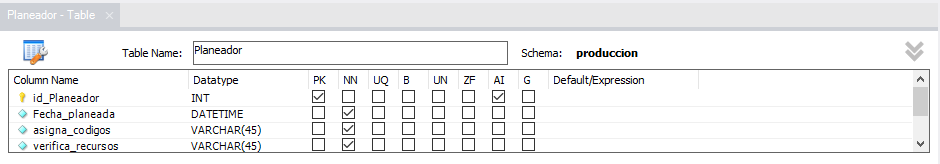
## Los scripts de MySQL ofrecen diversas ventajas, como:

* **Eficiencia:** Te permiten realizar tareas de manera más rápida y eficiente que usando la interfaz gráfica de usuario (GUI) de MySQL.
* **Precisión:** Te permiten ejecutar tareas con mayor precisión y menos errores que usando la GUI.
* **Automatización:** Te permiten automatizar tareas repetitivas, ahorrando tiempo y esfuerzo.
* **Reproducibilidad:** Te permiten reproducir fácilmente las mismas tareas en diferentes entornos.
* **Documentación:** Te permiten documentar tus tareas de administración de bases de datos, lo que facilita su comprensión y mantenimiento.

**Evidencia de producto: GA6-220501096-AA2-EV03 Script bases de datos del proyecto Teniendo en cuenta las características del software a desarrollar en su proyecto formativo y lo realizado en la evidencia EV02, cree el script de la base de datos.**







**Scrip**

-- MySQL Script generated by MySQL Workbench

-- Tue Jun 18 20:47:38 2024

-- Model: New Model Version: 1.0

-- 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 produccion

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

DROP SCHEMA IF EXISTS `produccion` ;

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

-- Schema produccion

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

CREATE SCHEMA IF NOT EXISTS `produccion` DEFAULT CHARACTER SET utf8 ;

SHOW WARNINGS;

USE `produccion` ;

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

-- Table `produccion`.`Cliente`

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

DROP TABLE IF EXISTS `produccion`.`Cliente` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`Cliente` (

`id\_Cliente` INT NULL AUTO\_INCREMENT,

`Direccion` VARCHAR(45) NOT NULL,

`Telefono` VARCHAR(45) NOT NULL,

`Nombre` VARCHAR(45) NOT NULL,

PRIMARY KEY (`id\_Cliente`))

ENGINE = InnoDB;

SHOW WARNINGS;

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

-- Table `produccion`.`Inventario`

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

DROP TABLE IF EXISTS `produccion`.`Inventario` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`Inventario` (

`id\_inventario` INT NULL AUTO\_INCREMENT,

`Cantidad\_disponible` DECIMAL(10,2) NOT NULL,

`Precio\_unitario` INT NOT NULL,

PRIMARY KEY (`id\_inventario`));

SHOW WARNINGS;

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

-- Table `produccion`.`Planeador`

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

DROP TABLE IF EXISTS `produccion`.`Planeador` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`Planeador` (

`id\_Planeador` INT NULL AUTO\_INCREMENT,

`Fecha\_planeada` DATETIME NOT NULL,

`asigna\_codigos` VARCHAR(45) NOT NULL,

`verifica\_recursos` VARCHAR(45) NOT NULL,

PRIMARY KEY (`id\_Planeador`))

ENGINE = InnoDB;

SHOW WARNINGS;

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

-- Table `produccion`.`Producto`

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

DROP TABLE IF EXISTS `produccion`.`Producto` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`Producto` (

`id\_producto` INT NULL AUTO\_INCREMENT,

`Tipo\_de\_tapa` VARCHAR(45) NOT NULL,

`Descripcion\_producto` VARCHAR(45) NOT NULL,

`Medidas` VARCHAR(45) NOT NULL,

`ordenes de produccion\_id\_ordenes de produccion` INT NOT NULL,

PRIMARY KEY (`id\_producto`, `ordenes de produccion\_id\_ordenes de produccion`));

SHOW WARNINGS;

CREATE INDEX `fk\_Producto\_ordenes de produccion1\_idx` ON `produccion`.`Producto` (`ordenes de produccion\_id\_ordenes de produccion` ASC) VISIBLE;

SHOW WARNINGS;

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

-- Table `produccion`.`Proveedor`

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

DROP TABLE IF EXISTS `produccion`.`Proveedor` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`Proveedor` (

`id\_Proveedor` INT NULL AUTO\_INCREMENT,

`Nombre` VARCHAR(45) NOT NULL,

`Email` VARCHAR(45) NOT NULL,

`Telefono` INT NOT NULL,

`Direccion` VARCHAR(45) NOT NULL,

PRIMARY KEY (`id\_Proveedor`))

ENGINE = InnoDB;

SHOW WARNINGS;

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

-- Table `produccion`.`Solicitud\_de\_pedido\_has\_Cliente`

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

DROP TABLE IF EXISTS `produccion`.`Solicitud\_de\_pedido\_has\_Cliente` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`Solicitud\_de\_pedido\_has\_Cliente` (

`Solicitud\_de\_pedido\_id\_solicitud\_de\_pedido` INT NOT NULL,

`Cliente\_id\_Cliente` INT NOT NULL,

PRIMARY KEY (`Solicitud\_de\_pedido\_id\_solicitud\_de\_pedido`, `Cliente\_id\_Cliente`))

ENGINE = InnoDB;

SHOW WARNINGS;

CREATE INDEX `fk\_Solicitud\_de\_pedido\_has\_Cliente\_Cliente1\_idx` ON `produccion`.`Solicitud\_de\_pedido\_has\_Cliente` (`Cliente\_id\_Cliente` ASC) VISIBLE;

SHOW WARNINGS;

CREATE INDEX `fk\_Solicitud\_de\_pedido\_has\_Cliente\_Solicitud\_de\_pedido1\_idx` ON `produccion`.`Solicitud\_de\_pedido\_has\_Cliente` (`Solicitud\_de\_pedido\_id\_solicitud\_de\_pedido` ASC) VISIBLE;

SHOW WARNINGS;

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

-- Table `produccion`.`Solicitud\_pedido`

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

DROP TABLE IF EXISTS `produccion`.`Solicitud\_pedido` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`Solicitud\_pedido` (

`id\_solicitud\_pedido` INT NULL AUTO\_INCREMENT,

`Fecha\_solicitud` DATETIME NOT NULL,

`Cantidad` VARCHAR(45) NOT NULL,

`Descripcion` VARCHAR(45) NOT NULL,

`Planeador\_id\_Planeador` INT NOT NULL,

PRIMARY KEY (`id\_solicitud\_pedido`))

ENGINE = InnoDB;

SHOW WARNINGS;

CREATE INDEX `fk\_Solicitud\_de\_pedido\_Planeador1\_idx` ON `produccion`.`Solicitud\_pedido` (`Planeador\_id\_Planeador` ASC) VISIBLE;

SHOW WARNINGS;

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

-- Table `produccion`.`Usuario`

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

DROP TABLE IF EXISTS `produccion`.`Usuario` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`Usuario` (

`id\_Usuario` INT NULL AUTO\_INCREMENT,

`Contraseña` VARCHAR(45) NOT NULL,

`Email` VARCHAR(45) NOT NULL,

`Planeador\_id\_Planeador` INT NOT NULL,

`Solicitud\_pedido\_id\_solicitud\_pedido` INT NOT NULL,

PRIMARY KEY (`id\_Usuario`, `Planeador\_id\_Planeador`, `Solicitud\_pedido\_id\_solicitud\_pedido`))

ENGINE = InnoDB;

SHOW WARNINGS;

CREATE INDEX `fk\_Usuario\_Planeador1\_idx` ON `produccion`.`Usuario` (`Planeador\_id\_Planeador` ASC) VISIBLE;

SHOW WARNINGS;

CREATE INDEX `fk\_Usuario\_Solicitud\_pedido1\_idx` ON `produccion`.`Usuario` (`Solicitud\_pedido\_id\_solicitud\_pedido` ASC) VISIBLE;

SHOW WARNINGS;

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

-- Table `produccion`.`insumos`

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

DROP TABLE IF EXISTS `produccion`.`insumos` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`insumos` (

`id\_insumos` VARCHAR(100) NOT NULL,

`Bodega` VARCHAR(50) NULL,

`Unidad\_de\_ medida` VARCHAR(50) NOT NULL,

`Stock\_actual` INT NOT NULL,

`Producto\_id\_producto` INT NOT NULL,

`inventario\_id\_inventario` INT NOT NULL,

PRIMARY KEY (`id\_insumos`, `Producto\_id\_producto`, `inventario\_id\_inventario`));

SHOW WARNINGS;

CREATE INDEX `fk\_insumos\_Producto1\_idx` ON `produccion`.`insumos` (`Producto\_id\_producto` ASC) VISIBLE;

SHOW WARNINGS;

CREATE INDEX `fk\_insumos\_inventario1\_idx` ON `produccion`.`insumos` (`inventario\_id\_inventario` ASC) VISIBLE;

SHOW WARNINGS;

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

-- Table `produccion`.`insumos\_has\_Proveedor`

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

DROP TABLE IF EXISTS `produccion`.`insumos\_has\_Proveedor` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`insumos\_has\_Proveedor` (

`insumos\_id\_insumos` VARCHAR(100) NOT NULL,

`insumos\_Producto\_id\_producto` INT NOT NULL,

`insumos\_Producto\_ordenes de produccion\_id\_ordenes de produccion` INT NOT NULL,

`insumos\_inventario\_id\_inventario` INT NOT NULL,

`Proveedor\_id\_Proveedor` INT NOT NULL,

PRIMARY KEY (`insumos\_id\_insumos`, `insumos\_Producto\_id\_producto`, `insumos\_Producto\_ordenes de produccion\_id\_ordenes de produccion`, `insumos\_inventario\_id\_inventario`, `Proveedor\_id\_Proveedor`));

SHOW WARNINGS;

CREATE INDEX `fk\_insumos\_has\_Proveedor\_Proveedor1\_idx` ON `produccion`.`insumos\_has\_Proveedor` (`Proveedor\_id\_Proveedor` ASC) VISIBLE;

SHOW WARNINGS;

CREATE INDEX `fk\_insumos\_has\_Proveedor\_insumos1\_idx` ON `produccion`.`insumos\_has\_Proveedor` (`insumos\_id\_insumos` ASC, `insumos\_Producto\_id\_producto` ASC, `insumos\_Producto\_ordenes de produccion\_id\_ordenes de produccion` ASC, `insumos\_inventario\_id\_inventario` ASC) VISIBLE;

SHOW WARNINGS;

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

-- Table `produccion`.`ordenes\_ produccion`

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

DROP TABLE IF EXISTS `produccion`.`ordenes\_ produccion` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`ordenes\_ produccion` (

`id\_ordenes de produccion` INT NULL AUTO\_INCREMENT,

`fecha \_fabricacion` DATE NOT NULL,

`cantidad\_de \_fabricacion` INT NOT NULL,

`datos\_de\_calidad` VARCHAR(45) NOT NULL,

`Tipo\_de\_empaque` VARCHAR(45) NOT NULL,

`Planeador\_id\_Planeador` INT NOT NULL,

`Usuario\_id\_Usuario` INT NOT NULL,

`Usuario\_Planeador\_id\_Planeador` INT NOT NULL,

`Usuario\_Cliente\_id\_Cliente` INT NOT NULL,

PRIMARY KEY (`id\_ordenes de produccion`, `Usuario\_id\_Usuario`, `Usuario\_Planeador\_id\_Planeador`, `Usuario\_Cliente\_id\_Cliente`));

SHOW WARNINGS;

CREATE INDEX `fk\_ordenes de produccion\_Planeador1\_idx` ON `produccion`.`ordenes\_ produccion` (`Planeador\_id\_Planeador` ASC) VISIBLE;

SHOW WARNINGS;

CREATE INDEX `fk\_ordenes de produccion\_Usuario1\_idx` ON `produccion`.`ordenes\_ produccion` (`Usuario\_id\_Usuario` ASC, `Usuario\_Planeador\_id\_Planeador` ASC, `Usuario\_Cliente\_id\_Cliente` ASC) VISIBLE;

SHOW WARNINGS;

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

-- Table `produccion`.`produccionordenes\_has\_operador`

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

DROP TABLE IF EXISTS `produccion`.`produccionordenes\_has\_operador` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`produccionordenes\_has\_operador` (

`produccionordenes\_id\_produccion` INT NOT NULL,

`operador\_id\_operador` VARCHAR(45) NOT NULL,

PRIMARY KEY (`produccionordenes\_id\_produccion`, `operador\_id\_operador`));

SHOW WARNINGS;

CREATE INDEX `fk\_produccionordenes\_has\_operador\_produccionordenes1\_idx` ON `produccion`.`produccionordenes\_has\_operador` (`produccionordenes\_id\_produccion` ASC) VISIBLE;

SHOW WARNINGS;

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

-- Table `produccion`.`table1`

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

DROP TABLE IF EXISTS `produccion`.`table1` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `produccion`.`table1` (

)

ENGINE = InnoDB;

SHOW WARNINGS;

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

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

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

--