



	<b>CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN (CSI)</b>	<b>BASES DE DATOS</b>
 <b>Escuela Superior de Ingeniería y Tecnología</b> Universidad de La Laguna	<b>PROYECTO: Concesionario</b>	Generación de código
	<b>Autor:</b> Cristian Vilanova González Carlos García Lezcano Javier Herrera Serpa Aram Pérez Dios	<b>ALU:</b> <b>alu0101014968</b> <b>alu0101208268</b> <b>alu0100830972</b> <b>alu0101244488</b>
Versión: 1	Tiempo invertido: 4 horas	Fecha : 06/02/22

**Entregar un enlace a github al producto desarrollado en cada una de las fases del proyecto, y un archivo zip con el proyecto git.**

**El Readme de Github debe incluir una descripción de la GENERACIÓN DE CÓDIGO DE COMPONENTES Y PROCEDIMIENTOS (CSI-2)**

<b>CÓDIGO DE COMPONENTES</b>
<p>Script SQL de generación de tablas (tanto para BDR como para BDROO) y objetos ( BDROO)</p> <p>DROP TYPE IF EXISTS Direccion CASCADE;  CREATE TYPE Direccion AS (  Nombre_via VarChar(100),  Poblacion VarChar(50),  Provincia VarChar(50));</p> <p>DROP TYPE IF EXISTS Nombre_completo CASCADE;  CREATE TYPE Nombre_completo AS (  Nombre VarChar(30),  Apellidos VarChar(80));</p> <p>DROP TABLE IF EXISTS Empleado CASCADE;  CREATE TABLE Empleado (  DNI VarChar(20) PRIMARY KEY,  Edad INT,  Estado_civil VarChar(20),  Direccion Direccion,  Nombre_completo Nombre_completo);</p> <p>DROP TABLE IF EXISTS Cliente CASCADE;  CREATE TABLE Cliente (  DNI VarChar(45) PRIMARY KEY,  Edad INT,  Direccion Direccion,  Nombre_completo Nombre_completo);</p> <p>DROP TABLE IF EXISTS Concesionario CASCADE;  CREATE TABLE Concesionario (  Credenciales VarChar(20) PRIMARY KEY,  Nombre VarChar(30),  Direccion Direccion,</p>

	CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN (CSI)	BASES DE DATOS
 <b>Escuela Superior de Ingeniería y Tecnología</b> Universidad de La Laguna	<b>PROYECTO: Concesionario</b>	Generación de código
	<b>Autor:</b> Cristian Vilanova González Carlos García Lezcano Javier Herrera Serpa Aram Pérez Dios	<b>ALU:</b> <b>alu0101014968</b> <b>alu0101208268</b> <b>alu0100830972</b> <b>alu0101244488</b>
Versión: 1	Tiempo invertido: 4 horas	Fecha : 06/02/22

```

DNI_empleado_responsable  VarChar(20) REFERENCES Empleado(DNI));

DROP TABLE IF EXISTS Coche CASCADE;
CREATE TABLE Coche (
Numero_de_bastidor      VarChar(30) PRIMARY KEY,
Matricula              VarChar(10),
Tipo                   VarChar(10),
Color                  VarChar(10),
Modelo                 VarChar(15),
Marca                  VarChar(15),
Consumo                DOUBLE PRECISION,
Indice_de_contaminacion  DOUBLE PRECISION,
Capacidad_baterias     DOUBLE PRECISION,
Rango_estimado         DOUBLE PRECISION,
Credenciales_concesionario  VarChar(20) REFERENCES Concesionario(Credenciales));


DROP TABLE IF EXISTS Reparaciones CASCADE;
CREATE TABLE Reparaciones (
Codigo                 INT PRIMARY KEY,
Fecha                 DATE,
Descripcion            VarChar(200),
Numero_de_bastidor     VarChar(30) REFERENCES Coche(Numero_de_bastidor));

DROP TABLE IF EXISTS Piezas CASCADE;
CREATE TABLE Piezas (
Codigo                INT PRIMARY KEY,
Nombre                VarChar(20),
Codigo_reparacion     INT REFERENCES Reparaciones(Codigo));

DROP TABLE IF EXISTS Trabaja CASCADE;
CREATE TABLE Trabaja (
Codigo                INT PRIMARY KEY,
DNI                   VarChar(45) REFERENCES Empleado(DNI),
Credenciales          VarChar(20) REFERENCES Concesionario(Credenciales));

DROP TABLE IF EXISTS Venta CASCADE;
CREATE TABLE Venta (
Codigo                INT PRIMARY KEY,
Numero_de_bastidor    VARCHAR(30) REFERENCES Coche(Numero_de_bastidor),
DNI_cliente           VARCHAR(45) REFERENCES Cliente(DNI),

```

	CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN (CSI)	BASES DE DATOS
 <b>Escuela Superior de Ingeniería y Tecnología</b> Universidad de La Laguna	<b>PROYECTO: Concesionario</b>	Generación de código
	<b>Autor:</b> Cristian Vilanova González Carlos García Lezcano Javier Herrera Serpa Aram Pérez Dios	<b>ALU:</b> <b>alu0101014968</b> <b>alu0101208268</b> <b>alu0100830972</b> <b>alu0101244488</b>
Versión: 1		Tiempo invertido: 4 horas
		Fecha : 06/02/22


```

DNI_empleado    VARCHAR(45) REFERENCES Empleado(DNI),
Precio          INT,
Fecha           DATE);

DROP TABLE IF EXISTS Punto_de_recarga CASCADE;
CREATE TABLE Punto_de_recarga (
Numero_de_serie  VARCHAR(20) PRIMARY KEY,
Direccion       Direccion);

DROP TABLE IF EXISTS Provee CASCADE;
CREATE TABLE Provee (
Codigo          INT PRIMARY KEY,
Codigo_coche_electrico VARCHAR(45) REFERENCES Coche(Numero_de_bastidor),
Numero_de_serie  VARCHAR(45) REFERENCES Punto_de_recarga(Numero_de_serie));


```

	CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN (CSI)	BASES DE DATOS
 <b>Escuela Superior de Ingeniería y Tecnología</b> Universidad de La Laguna	<b>PROYECTO: Concesionario</b>	Generación de código
	<b>Autor:</b> Cristian Vilanova González Carlos García Lezcano Javier Herrera Serpa Aram Pérez Dios	<b>ALU:</b> <b>alu0101014968</b> <b>alu0101208268</b> <b>alu0100830972</b> <b>alu0101244488</b>
Versión: 1	Tiempo invertido: 4 horas	Fecha : 06/02/22

```

psql:script-final.sql:1: NOTICE: drop cascades to 4 other objects
DETAIL: drop cascades to column direccion of table empleado
drop cascades to column direccion of table cliente
drop cascades to column direccion of table concesionario
drop cascades to column direccion of table punto_de_recarga
DROP TYPE
CREATE TYPE
psql:script-final.sql:7: NOTICE: drop cascades to 2 other objects
DETAIL: drop cascades to column nombre_completo of table empleado
drop cascades to column nombre_completo of table cliente
DROP TYPE
CREATE TYPE
psql:script-final.sql:12: NOTICE: drop cascades to 3 other objects
DETAIL: drop cascades to constraint concesionario_dni_empleado_responsable_fkey on table concesionario
drop cascades to constraint trabaja_dni_fkey on table trabaja
drop cascades to constraint venta_dni_empleado_fkey on table venta
DROP TABLE
CREATE TABLE
psql:script-final.sql:20: NOTICE: drop cascades to constraint venta_dni_cliente_fkey on table venta
DROP TABLE
CREATE TABLE
psql:script-final.sql:27: NOTICE: drop cascades to 2 other objects
DETAIL: drop cascades to constraint coche_credenciales_concesionario_fkey on table coche
drop cascades to constraint trabaja_credenciales_fkey on table trabaja
DROP TABLE
CREATE TABLE
psql:script-final.sql:34: NOTICE: drop cascades to 3 other objects
DETAIL: drop cascades to constraint reparaciones_numero_de_bastidor_fkey on table reparaciones
drop cascades to constraint venta_numero_de_bastidor_fkey on table venta
drop cascades to constraint provee_codigo_coche_electrico_fkey on table provee
DROP TABLE
CREATE TABLE
psql:script-final.sql:48: NOTICE: drop cascades to constraint piezas_codigo_reparacion_fkey on table piezas
DROP TABLE
CREATE TABLE
DROP TABLE
CREATE TABLE
DROP TABLE
CREATE TABLE
DROP TABLE
CREATE TABLE
psql:script-final.sql:76: NOTICE: drop cascades to constraint provee_numero_de_serie_fkey on table provee
DROP TABLE
CREATE TABLE
DROP TABLE
CREATE TABLE


```

	CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN (CSI)	BASES DE DATOS
 <b>Escuela Superior de Ingeniería y Tecnología</b> Universidad de La Laguna	<b>PROYECTO: Concesionario</b>	Generación de código
	<b>Autor:</b> Cristian Vilanova González Carlos García Lezcano Javier Herrera Serpa Aram Pérez Dios	<b>ALU:</b> <b>alu0101014968</b> <b>alu0101208268</b> <b>alu0100830972</b> <b>alu0101244488</b>
Versión: 1		Tiempo invertido: 4 horas
		Fecha : 06/02/22

```

psql:script-final.sql:76: NOTICE: drop cascades to constraint provee_numero_de_serie_fkey on table provee
DROP TABLE
CREATE TABLE
DROP TABLE
CREATE TABLE
CREATE FUNCTION
psql:script-final.sql:104: NOTICE: trigger "trigger_solo_coche_electrico" for relation "provee" does not exist, skipping
DROP TRIGGER
CREATE TRIGGER
CREATE FUNCTION
psql:script-final.sql:122: NOTICE: trigger "trigger_coche_solo_un_tipo" for relation "coche" does not exist, skipping
DROP TRIGGER
CREATE TRIGGER
CREATE FUNCTION
psql:script-final.sql:139: NOTICE: trigger "trigger_trabaja_comprobar_pk" for relation "trabaja" does not exist, skipping
DROP TRIGGER
CREATE TRIGGER
CREATE FUNCTION
psql:script-final.sql:156: NOTICE: trigger "trigger_provee_comprobar_pk" for relation "provee" does not exist, skipping
DROP TRIGGER
CREATE TRIGGER
CREATE FUNCTION
psql:script-final.sql:173: NOTICE: trigger "trigger_venta_comprobar_pk" for relation "venta" does not exist, skipping
DROP TRIGGER
CREATE TRIGGER
START TRANSACTION
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
COMMIT
START TRANSACTION
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
COMMIT
START TRANSACTION
INSERT 0 1
INSERT 0 1
COMMIT
START TRANSACTION

```

 <b>Escuela Superior de Ingeniería y Tecnología</b> Universidad de La Laguna	<b>CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN (CSI)</b>		<b>BASES DE DATOS</b>
	<b>PROYECTO: Concesionario</b>		Generación de código
	<b>Autor:</b> Cristian Vilanova González Carlos García Lezcano Javier Herrera Serpa Aram Pérez Dios		<b>ALU:</b> <b>alu0101014968</b> <b>alu0101208268</b> <b>alu0100830972</b> <b>alu0101244488</b>
	Versión: 1	Tiempo invertido: 4 horas	Fecha : 06/02/22

postgres=# \dt

List of relations			
Schema	Name	Type	Owner
public	cliente	table	postgres
public	coche	table	postgres
public	concesionario	table	postgres
public	empleado	table	postgres
public	piezas	table	postgres
public	provee	table	postgres
public	punto_de_recarga	table	postgres
public	reparaciones	table	postgres
public	trabaja	table	postgres
public	venta	table	postgres

(10 rows)

postgres=# SELECT \* FROM Coche;

numero_de_bastidor	matricula	tipo	color	modelo	marca	consumo	indice_de_contaminacion	capacidad_baterias	rango_estimado	credenciales_concesionario
1v	111aaa	Tipo1	Color1	Modelo1.1	Marca1	4.5	2.3			Credenciales1
2v	222aaa	Tipo2	Color2	Modelo1.2	Marca1			200	200	Credenciales1
3v	333aaa	Tipo2	Color3	Modelo1.3	Marca1	4.5		20		Credenciales1
4v	444aaa	Tipo1	Color1	Modelo1.1	Marca2	4.5	2.3			Credenciales2

(4 rows)

postgres=# SELECT \* FROM Concesionario;

credenciales	nombre	direccion			dni_empleado_responsable
Credenciales1	Nombre1	(calle1.3,población1.3,provincia1.3)			1111111A
Credenciales2	Nombre2	(calle2.3,población2.3,provincia2.3)			2222222A


(2 rows)

postgres=# SELECT \* FROM Venta;

codigo	numero_de_bastidor	dni_cliente	dni_empleado	precio	fecha
1	1v	1111111B	1111111A	10000	2022-02-05

(1 row)

<b>CÓDIGO DE PROCEDIMIENTOS DE OPERACIÓN Y SEGURIDAD</b>
Código SQL de los asertos y disparadores presentes en la base de datos
CREATE OR REPLACE FUNCTION solo_coche_electrico() RETURNS TRIGGER AS

	CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN (CSI)	BASES DE DATOS
 <b>Escuela Superior de Ingeniería y Tecnología</b> Universidad de La Laguna	<b>PROYECTO: Concesionario</b>	Generación de código
	<b>Autor:</b> Cristian Vilanova González Carlos García Lezcano Javier Herrera Serpa Aram Pérez Dios	<b>ALU:</b> <b>alu0101014968</b> <b>alu0101208268</b> <b>alu0100830972</b> <b>alu0101244488</b>
Versión: 1	Tiempo invertido: 4 horas	Fecha : 06/02/22

```


$solo_coche_electrico$
DECLARE
    cap    double precision;
    rang   double precision;
    ind    double precision;
    con    double precision;

BEGIN
    SELECT  Capacidad_baterias,Rango_estimado,Indice_de_contaminacion,Consumo
INTO      cap,rang,ind,con    FROM      Coche      WHERE      Numero_de_bastidor      =
NEW.Codigo_coche_electrico;
    IF cap IS NOT NULL AND rang IS NOT NULL AND ind IS NULL AND con IS NULL
THEN
        RETURN NEW;
    ELSE
        RAISE WARNING 'El coche no es electrico';
        RETURN NULL;
    END IF;
END;

$solo_coche_electrico$ LANGUAGE plpgsql;
DROP TRIGGER IF EXISTS trigger_solo_coche_electrico ON Provee;
CREATE TRIGGER trigger_solo_coche_electrico BEFORE INSERT ON Provee
FOR EACH ROW EXECUTE PROCEDURE solo_coche_electrico();

CREATE OR REPLACE FUNCTION  coche_solo_un_tipo() RETURNS TRIGGER AS
$coche_solo_un_tipo$
BEGIN
    IF NEW.Capacidad_baterias IS NOT NULL AND NEW.Rango_estimado IS NOT NULL
AND NEW.Indice_de_contaminacion IS NULL AND NEW.Consumo IS NULL THEN
        RETURN NEW;
    ELSIF NEW.Capacidad_baterias IS NOT NULL AND NEW.Rango_estimado IS NULL
AND NEW.Indice_de_contaminacion IS NULL AND NEW.Consumo IS NOT NULL THEN
        RETURN NEW;
    ELSIF NEW.Capacidad_baterias IS NULL AND NEW.Rango_estimado IS NULL AND
NEW.Indice_de_contaminacion IS NOT NULL AND NEW.Consumo IS NOT NULL THEN
        RETURN NEW;
    ELSE
        RAISE WARNING 'El tipo de coche no es correcto';
        RETURN NULL;
    END IF;

```

	<b>CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN (CSI)</b>	<b>BASES DE DATOS</b>
 <b>Escuela Superior de Ingeniería y Tecnología</b> Universidad de La Laguna	<b>PROYECTO: Concesionario</b>	Generación de código
	<b>Autor:</b> Cristian Vilanova González Carlos García Lezcano Javier Herrera Serpa Aram Pérez Dios	<b>ALU:</b> <b>alu0101014968</b> <b>alu0101208268</b> <b>alu0100830972</b> <b>alu0101244488</b>
Versión: 1	Tiempo invertido: 4 horas	Fecha : 06/02/22

```


END;
$coche_solo_un_tipo$ LANGUAGE plpgsql;
DROP TRIGGER IF EXISTS trigger_coche_solo_un_tipo ON Coche;
CREATE TRIGGER trigger_coche_solo_un_tipo BEFORE INSERT ON Coche
FOR EACH ROW EXECUTE PROCEDURE coche_solo_un_tipo();

CREATE OR REPLACE FUNCTION trabaja_comprobar_pk() RETURNS TRIGGER AS
$trabaja_comprobar_pk$
DECLARE
    result INT;
BEGIN
    SELECT Codigo INTO result FROM Trabaja WHERE DNI = NEW.DNI AND
Credenciales = NEW.Credenciales;
    IF result IS NULL THEN
        RETURN NEW;
    ELSE
        RAISE WARNING 'El empleado ya trabaja en ese concesionario';
        RETURN NULL;
    END IF;
END;
$trabaja_comprobar_pk$ LANGUAGE plpgsql;
DROP TRIGGER IF EXISTS trigger_trabaja_comprobar_pk ON Trabaja;
CREATE TRIGGER trigger_trabaja_comprobar_pk BEFORE INSERT ON Trabaja
FOR EACH ROW EXECUTE PROCEDURE trabaja_comprobar_pk();

CREATE OR REPLACE FUNCTION provee_comprobar_pk() RETURNS TRIGGER AS
$provee_comprobar_pk$
DECLARE
    result INT;
BEGIN
    SELECT Codigo INTO result FROM Provee WHERE Codigo_coche_electrico =
NEW.Codigo_coche_electrico AND Numero_de_serie = NEW.Numero_de_serie;
    IF result IS NULL THEN
        RETURN NEW;
    ELSE
        RAISE WARNING 'Esa entrada ya existe';
        RETURN NULL;
    END IF;
END;
$provee_comprobar_pk$ LANGUAGE plpgsql;

```



	<b>CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN (CSI)</b>	<b>BASES DE DATOS</b>
 <b>Escuela Superior de Ingeniería y Tecnología</b> Universidad de La Laguna	<b>PROYECTO: Concesionario</b>	Generación de código
	<b>Autor:</b> Cristian Vilanova González Carlos García Lezcano Javier Herrera Serpa Aram Pérez Dios	<b>ALU:</b> <b>alu0101014968</b> <b>alu0101208268</b> <b>alu0100830972</b> <b>alu0101244488</b>
Versión: 1	Tiempo invertido: 4 horas	Fecha : 06/02/22

```

DROP TRIGGER IF EXISTS trigger_provee_comprobar_pk ON Provee;
CREATE TRIGGER trigger_provee_comprobar_pk BEFORE INSERT ON Provee
FOR EACH ROW EXECUTE PROCEDURE provee_comprobar_pk();

CREATE OR REPLACE FUNCTION venta_comprobar_pk() RETURNS TRIGGER AS
$venta_comprobar_pk$
  DECLARE
    result INT;
  BEGIN
    SELECT Codigo INTO result FROM Venta WHERE Numero_de_bastidor =
NEW.Numero_de_bastidor;
    IF result IS NULL THEN
      RETURN NEW;
    ELSE
      RAISE WARNING 'Este coche ya ha sido vendido';
      RETURN NULL;
    END IF;
  END;
$venta_comprobar_pk$ LANGUAGE plpgsql;
DROP TRIGGER IF EXISTS trigger_venta_comprobar_pk ON Venta;
CREATE TRIGGER trigger_venta_comprobar_pk BEFORE INSERT ON Venta
FOR EACH ROW EXECUTE PROCEDURE venta_comprobar_pk();

```