



# PROYECTO DE BASE DE DATOS

Bases de Datos

Joaquin Barrero  
[joabarrod529@g.educaand.es](mailto:joabarrod529@g.educaand.es)

# Proyecto de base de Datos

Modelo Entidad Relación .....	3
MODELO RELACIONAL.....	5
PASO A TABLAS.....	7
SCRIPT CREACION BBDD.....	8
CONCLUSIONES .....	16

# Proyecto de base de Datos

Ilustración 1 Modelo E/R elaborado en Diagrams .....	4
Ilustración 2 diseño de tablas en Workbench.....	7

# Proyecto de base de Datos

## Modelo Entidad Relación

Se pretende analizar una pequeña empresa familiar la cual se encarga de tanto de la representación como la venta de productos alimenticios en su mayor parte de tipo “gourmet”.

Para ello la empresa en cuestión desea almacenar de sus clientes el nombre, domicilio, teléfono, el sector al que se dedica, la zona en la que se encuentra, el CIF, la persona de contacto, un apartado con ciertas observaciones que se puedan tener en cuenta. Además, referente a la información contable se debe almacenar, la tarifa aplicable a dicho cliente así como el límite de crédito establecido para el mismo, la forma de pago por defecto y la cuenta contable. Cada cliente será identificado con un numero correlativo.

De los distintos proveedores que sirven los productos se desea guardar, el código de proveedor, su CIF, el teléfono y su correo electrónico, la dirección y el País y la actividad a la que se dedica el mismo.

De los distintos productos que puede surtir un proveedor se debe almacenar el nombre del producto, el precio de compra y el precio de venta, así como el tipo de IVA aplicable.

Los productos pertenecen a distintas familias, de las cuales se guardará el nombre y una breve descripción de la misma.

De los empleados se desea guardar el identificador único de empleado, su nombre y apellidos, el domicilio, su DNI, los datos de contacto (teléfono y email). Hay dos tipos de empleados, comerciales o administrativos.

Los comerciales negocian con los clientes para las distintas operaciones (un cliente puede estar relacionado con distintos empleados y viceversa). De cada negociación se debe almacenar el tipo de comisión aplicable a la misma y el porcentaje de dicha comisión.

Los pedidos se van insertando en la BBDD con un nº correlativo y se debe guardar tanto la fecha de creación, la fecha en la que se debe servir, el importe y el estado del mismo. Los pedidos podrán ser solicitados tanto por el cliente como por los comerciales.

Cada pedido contiene una serie de productos, que tendrán un precio establecido en dicho pedido, la cantidad de productos solicitados y las unidades servidas de los mismos. Los productos pueden llevar un descuento si el comercial lo considera apropiado.

Cada pedido tiene un pago con un numero de factura, una fecha de emisión, la fecha de vencimiento y el importe del mismo.

Los productos pueden ser surtidos de distintos almacenes, propios o de los distintos proveedores (los cuales pueden tener varios almacenes). De los almacenes tiene se desea almacenar el nombre (que es único por almacén), la dirección y una breve descripción del mismo. Por su puesto se debe guardar un registro del stock de los productos que se almacenan en cada almacén.

# Proyecto de base de Datos

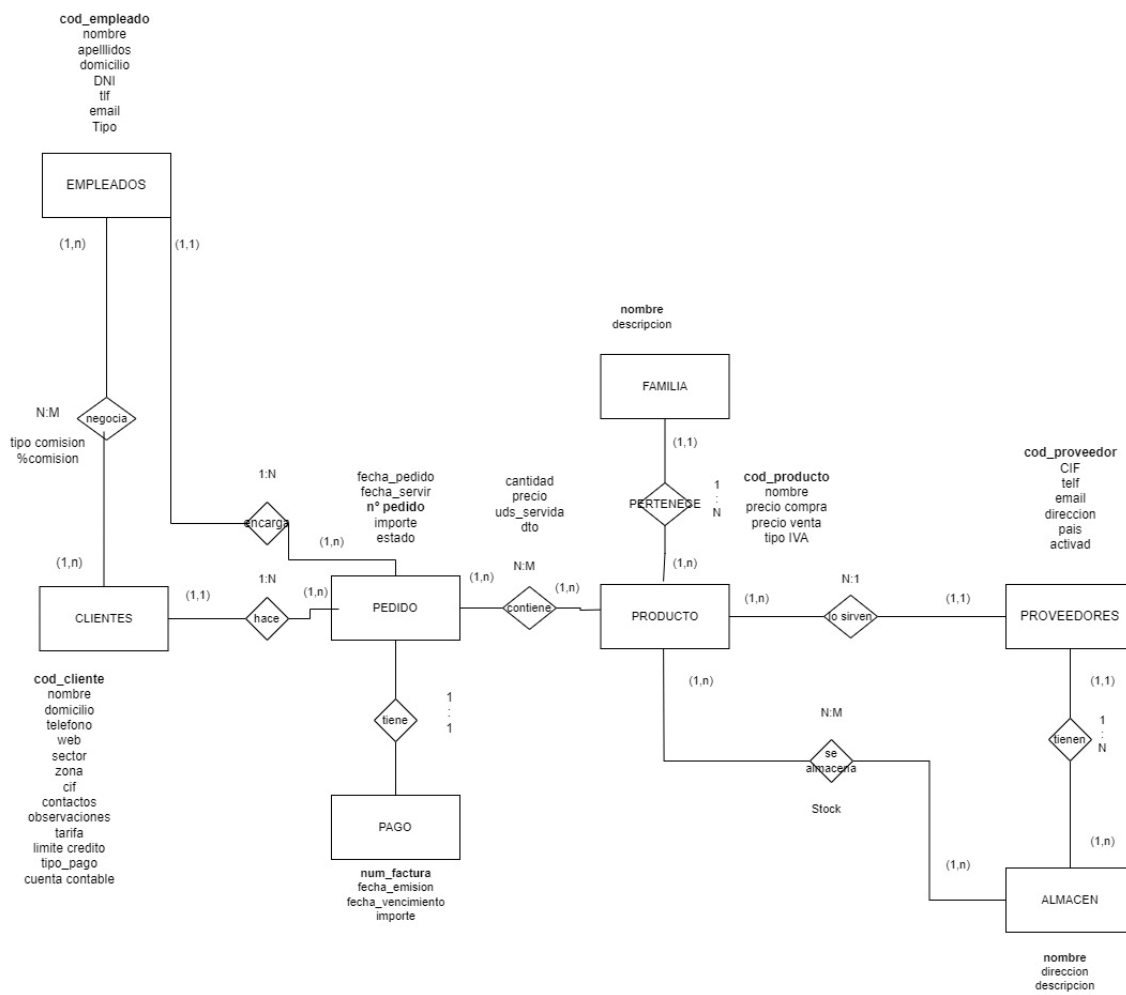


Ilustración 1 Modelo E/R elaborado en Diagrams

# Proyecto de base de Datos

## MODELO RELACIONAL

**EMPLEADOS** (cod\_empleado, Nombre, Apellidos, Domicilio, DNI, Telefono, Email, Tipo)

**PK:** cod\_empleado

**CLIENTES** (cod\_cliente, Nombre, Domicilio, Telefono, Sector, Zona, CIF, Persona\_Contacto, Observaciones, Tarifa\_aplicada, limite\_credito, Tipo\_pago, Cuenta\_Contable)

**PK:** cod\_cliente

**PEDIDO** (cod\_pedido, Fecha\_pedido, Fecha\_servir, Importe, Estado, cod\_cliente, cod\_empleado)

**PK:** cod\_pedido

**FK:** cod\_cliente referenciado a CLIENTES(cod\_cliente)

**FK:** cod\_empleado referenciado a EMPLEADOS (cod\_empleado).

**PAGO** (num\_factura, Fecha\_emision, Fecha\_vencimiento, Importe, cod\_pedido)

**PK:** num\_factura

**FK:** cod\_pedido referenciado a PEDIDO(cod\_pedido)

**FAMILIA**(Nombre, Descripción)

**PK:** Nombre

**PROVEEDORES**(cod\_proveedor, CIF, telefono, email, direccion, pais, actividad)

**PK:** cod\_proveedor

**PRODUCTO**(cod\_producto, Descripcion, Precio\_compra, Precio\_venta, Tipo\_IVA, nombre\_familia, cod\_proveedor)

**PK:** cod\_producto

**FK:** nombre\_familia referenciado a Familia(nombre)

**FK:** cod\_proveedor referenciado a PROVEEDOR(cod\_proveedor)

## Proyecto de base de Datos

**ALMACEN**(nombre, direccion, descripcion, cod\_proveedor)

**PK:** nombre

**FK:** cod\_proveedor referenciado a PROVEEDOR(cod\_proveedor)

**EMPLEADOS\_NEGOCIA\_CLIENTES**(cod\_empleado, cod\_cliente, tipo\_comision, porcentaje\_comision)

**PK:** cod\_empleado, cod\_cliente

**FK:** cod\_empleado referenciado a EMPLEADO(cod\_empleado)

**FK:** cod\_cliente referenciado a CLIENTE(cod\_cliente)

**DETALLE\_DE\_PEDIDO**(cod\_producto, cod\_pedido, cantidad, precio, unidades\_servidas, descuento)

**PK:** cod\_empleado, cod\_pedido

**FK:** cod\_empleado referenciado a EMPLEADO(cod\_empleado)

**FK:** cod\_pedido referenciado a PEDIDO(cod\_pedido)

**PRODUCTO\_ES\_ALMACENADO**(cod\_producto, nombre\_almacen, stock)

**PK:** cod\_producto, nombre\_almacen

**FK:** cod\_producto referenciado a PRODUCTO(cod\_producto)

**FK:** nombre\_almacen referenciado a ALMACEN(nombre)

# Proyecto de base de Datos

## PASO A TABLAS

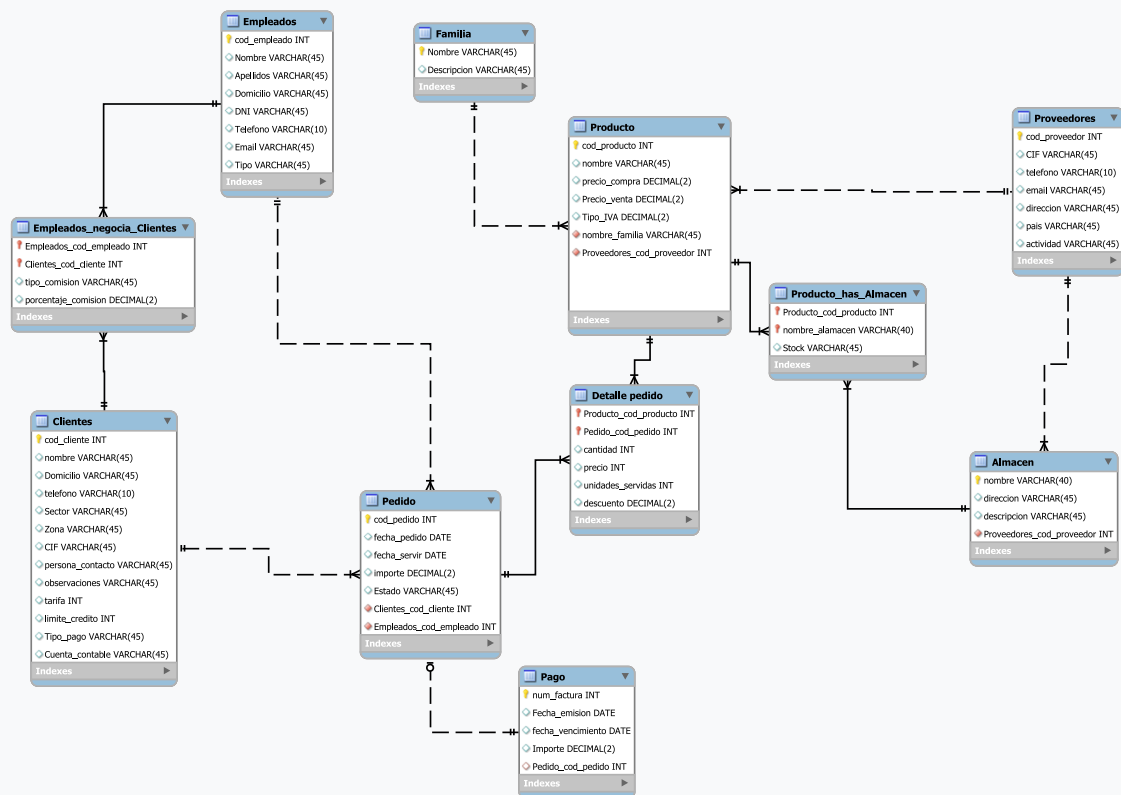


Ilustración 2 diseño de tablas en Workbench



# Proyecto de base de Datos

## SCRIPT CREACION BBDD

```
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_D
ATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
```

```
-----

-- Schema jb_exclusivas

-----
```

```
DROP SCHEMA IF EXISTS `jb_exclusivas` ;
```

```
-----

-- Schema jb_exclusivas

-----
```

```
CREATE SCHEMA IF NOT EXISTS `jb_exclusivas` DEFAULT CHARACTER SET utf8 ;
USE `jb_exclusivas` ;
```

```
-----

-- Table `jb_exclusivas`.`Clientes`

-----
```

```
DROP TABLE IF EXISTS `jb_exclusivas`.`Clientes` ;
```

```
CREATE TABLE IF NOT EXISTS `jb_exclusivas`.`Clientes` (
  `cod_cliente` INT NOT NULL,
  `nombre` VARCHAR(45) NULL,
  `domicilio` VARCHAR(45) NULL,
  `telefono` VARCHAR(10) NULL,
  `sector` VARCHAR(45) NULL,
  `zona` VARCHAR(45) NULL,
  `CIF` VARCHAR(45) NULL,
  `persona_contacto` VARCHAR(45) NULL,
```

## Proyecto de base de Datos

```
`observaciones` VARCHAR(45) NULL,
`tarifa` INT NULL,
`limite_credito` INT NULL,
`tipo_pago` VARCHAR(45) NULL,
`cuenta_contable` VARCHAR(45) NULL,
PRIMARY KEY (`cod_cliente`))
ENGINE = InnoDB;
```

```
-----
-- Table `jb_exclusivas`.`Empleados`
-----
```

```
DROP TABLE IF EXISTS `jb_exclusivas`.`Empleados` ;
```

```
CREATE TABLE IF NOT EXISTS `jb_exclusivas`.`Empleados` (
  `cod_empleado` INT NOT NULL,
  `Nombre` VARCHAR(45) NULL,
  `apellidos` VARCHAR(45) NULL,
  `domicilio` VARCHAR(45) NULL,
  `DNI` VARCHAR(45) NULL,
  `telefono` VARCHAR(10) NULL,
  `eEmail` VARCHAR(45) NULL,
  `tipo` VARCHAR(45) NULL,
  PRIMARY KEY (`cod_empleado`))
ENGINE = InnoDB;
```

```
-----
-- Table `jb_exclusivas`.`Pedido`
-----
```

```
DROP TABLE IF EXISTS `jb_exclusivas`.`Pedido` ;
```

```
CREATE TABLE IF NOT EXISTS `jb_exclusivas`.`Pedido` (
```

## Proyecto de base de Datos

```

`cod_pedido` INT NULL,
`fecha_pedido` DATE NULL,
`fecha_servir` DATE NULL,
`importe` DECIMAL(2) NULL,
`Estado` VARCHAR(45) NULL,
`Clientes_cod_cliente` INT NOT NULL,
`Empleados_cod_empleado` INT NOT NULL,
PRIMARY KEY (`cod_pedido`),
INDEX `fk_Pedido_Clientes1_idx` (`Clientes_cod_cliente` ASC) VISIBLE,
INDEX `fk_Pedido_Empleados1_idx` (`Empleados_cod_empleado` ASC) VISIBLE,
CONSTRAINT `fk_Pedido_Clientes1`
  FOREIGN KEY (`Clientes_cod_cliente`)
    REFERENCES `jb_exclusivas`.`Clientes` (`cod_cliente`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
CONSTRAINT `fk_Pedido_Empleados1`
  FOREIGN KEY (`Empleados_cod_empleado`)
    REFERENCES `jb_exclusivas`.`Empleados` (`cod_empleado`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `jb_exclusivas`.`Pago`
-----

```

```
DROP TABLE IF EXISTS `jb_exclusivas`.`Pago` ;
```

```
CREATE TABLE IF NOT EXISTS `jb_exclusivas`.`Pago` (
```

```

  `num_factura` INT NOT NULL,
  `Fecha_emision` DATE NULL,
  `fecha_vencimiento` DATE NULL,

```

## Proyecto de base de Datos

```
`Importe` DECIMAL(2) NULL,
`Pedido_cod_pedido` INT NULL,
PRIMARY KEY (`num_factura`),
INDEX `fk_Pago_Pedido1_idx` (`Pedido_cod_pedido` ASC) VISIBLE,
CONSTRAINT `fk_Pago_Pedido1`
FOREIGN KEY (`Pedido_cod_pedido`)
REFERENCES `jb_exclusivas`.`Pedido` (`cod_pedido`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
-----
-- Table `jb_exclusivas`.`Familia`
-----
```

```
DROP TABLE IF EXISTS `jb_exclusivas`.`Familia` ;
```

```
CREATE TABLE IF NOT EXISTS `jb_exclusivas`.`Familia` (
  `Nombre` VARCHAR(45) NOT NULL,
  `Descripcion` VARCHAR(45) NULL,
  PRIMARY KEY (`Nombre`))
ENGINE = InnoDB
COMMENT = ' ';
```

```
-----
-- Table `jb_exclusivas`.`Proveedores`
-----
```

```
DROP TABLE IF EXISTS `jb_exclusivas`.`Proveedores` ;
```

```
CREATE TABLE IF NOT EXISTS `jb_exclusivas`.`Proveedores` (
  `cod_proveedor` INT NOT NULL,
  `CIF` VARCHAR(45) NULL,
```

## Proyecto de base de Datos

```
`telefono` VARCHAR(10) NULL,
`email` VARCHAR(45) NULL,
`direccion` VARCHAR(45) NULL,
`pais` VARCHAR(45) NULL,
`actividad` VARCHAR(45) NULL,
PRIMARY KEY (`cod_proveedor`))
ENGINE = InnoDB
COMMENT = ' ';
```

```
-----
-- Table `jb_exclusivas`.`Producto`
-----
```

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

```
CREATE TABLE IF NOT EXISTS `jb_exclusivas`.`Producto` (
  `cod_producto` INT NOT NULL,
  `nombre` VARCHAR(45) NULL,
  `precio_compra` DECIMAL(2) NULL,
  `Precio_venta` DECIMAL(2) NULL,
  `Tipo_IVA` DECIMAL(2) NULL,
  `nombre_familia` VARCHAR(45) NOT NULL,
  `Proveedores_cod_proveedor` INT NOT NULL,
  PRIMARY KEY (`cod_producto`),
  INDEX `fk_Producto_Familia_idx` (`nombre_familia` ASC) VISIBLE,
  INDEX `fk_Producto_Proveedores1_idx` (`Proveedores_cod_proveedor` ASC) VISIBLE,
  CONSTRAINT `fk_Producto_Familia`
    FOREIGN KEY (`nombre_familia`)
      REFERENCES `jb_exclusivas`.`Familia` (`Nombre`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Producto_Proveedores1`
```

## Proyecto de base de Datos

```

FOREIGN KEY (`Proveedores_cod_proveedor`)
REFERENCES `jb_exclusivas`.`Proveedores` (`cod_proveedor`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB
COMMENT = ' ';

-----
-- Table `jb_exclusivas`.`Detalle pedido`
-----

DROP TABLE IF EXISTS `jb_exclusivas`.`Detalle pedido` ;

CREATE TABLE IF NOT EXISTS `jb_exclusivas`.`Detalle pedido` (
  `Producto_cod_producto` INT NOT NULL,
  `Pedido_cod_pedido` INT NOT NULL,
  `cantidad` INT NULL,
  `precio` INT NULL,
  `unidades_servidas` INT NULL,
  `descuento` DECIMAL(2) NULL,
  PRIMARY KEY (`Producto_cod_producto`, `Pedido_cod_pedido`),
  INDEX `fk_Producto_has_Pedido_Pedido1_idx` (`Pedido_cod_pedido` ASC) VISIBLE,
  INDEX `fk_Producto_has_Pedido_Producto1_idx` (`Producto_cod_producto` ASC) VISIBLE,
  CONSTRAINT `fk_Producto_has_Pedido_Producto1`
    FOREIGN KEY (`Producto_cod_producto`)
      REFERENCES `jb_exclusivas`.`Producto` (`cod_producto`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  CONSTRAINT `fk_Producto_has_Pedido_Pedido1`
    FOREIGN KEY (`Pedido_cod_pedido`)
      REFERENCES `jb_exclusivas`.`Pedido` (`cod_pedido`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION)

```

# Proyecto de base de Datos

ENGINE = InnoDB;

```
-----
-- Table `jb_exclusivas`.`Empleados_negocia_Clientes`
-----
```

DROP TABLE IF EXISTS `jb\_exclusivas`.`Empleados\_negocia\_Clientes` ;

```
CREATE TABLE IF NOT EXISTS `jb_exclusivas`.`Empleados_negocia_Clientes` (
  `Empleados_cod_empleado` INT NOT NULL,
  `Clientes_cod_cliente` INT NOT NULL,
  `tipo_comision` VARCHAR(45) NULL,
  `porcentaje_comision` DECIMAL(2) NULL,
  PRIMARY KEY (`Empleados_cod_empleado`, `Clientes_cod_cliente`),
  INDEX `fk_Empleados_has_Clientes_Clientes1_idx` (`Clientes_cod_cliente` ASC) VISIBLE,
  INDEX `fk_Empleados_has_Clientes_Empleados1_idx` (`Empleados_cod_empleado` ASC) VISIBLE,
  CONSTRAINT `fk_Empleados_has_Clientes_Empleados1`
    FOREIGN KEY (`Empleados_cod_empleado`)
      REFERENCES `jb_exclusivas`.`Empleados` (`cod_empleado`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Empleados_has_Clientes_Clientes1`
    FOREIGN KEY (`Clientes_cod_cliente`)
      REFERENCES `jb_exclusivas`.`Clientes` (`cod_cliente`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
-----
-- Table `jb_exclusivas`.`Almacen`
```

## Proyecto de base de Datos

```
-----
DROP TABLE IF EXISTS `jb_exclusivas`.`Almacen` ;
```

```
CREATE TABLE IF NOT EXISTS `jb_exclusivas`.`Almacen` (
  `nombre` VARCHAR(40) NOT NULL,
  `direccion` VARCHAR(45) NULL,
  `descripcion` VARCHAR(45) NULL,
  `Proveedores_cod_proveedor` INT NOT NULL,
  PRIMARY KEY (`nombre`),
  INDEX `fk_Almacen_Proveedores1_idx` (`Proveedores_cod_proveedor` ASC) VISIBLE,
  CONSTRAINT `fk_Almacen_Proveedores1`
    FOREIGN KEY (`Proveedores_cod_proveedor`)
      REFERENCES `jb_exclusivas`.`Proveedores` (`cod_proveedor`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
-----
-- Table `jb_exclusivas`.`Producto_has_Almacen`
-----
```

```
DROP TABLE IF EXISTS `jb_exclusivas`.`Producto_has_Almacen` ;
```

```
CREATE TABLE IF NOT EXISTS `jb_exclusivas`.`Producto_has_Almacen` (
  `Producto_cod_producto` INT NOT NULL,
  `nombre_alamacen` VARCHAR(40) NOT NULL,
  `Stock` VARCHAR(45) NULL,
  PRIMARY KEY (`Producto_cod_producto`, `nombre_alamacen`),
  INDEX `fk_Producto_has_Almacen_Almacen1_idx` (`nombre_alamacen` ASC) VISIBLE,
  INDEX `fk_Producto_has_Almacen_Producto1_idx` (`Producto_cod_producto` ASC) VISIBLE,
  CONSTRAINT `fk_Producto_has_Almacen_Producto1`
    FOREIGN KEY (`Producto_cod_producto`)
      REFERENCES `jb_exclusivas`.`Producto` (`cod_producto`)
```



# Proyecto de base de Datos

```
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `fk_Producto_has_Almacen_Almacen1`
FOREIGN KEY (`nombre_almacen`)
REFERENCES `jb_exclusivas`.`Almacen` (`nombre`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

Enlace a Github:

[https://github.com/jbarrod529/Proyecto\\_BBDD.git](https://github.com/jbarrod529/Proyecto_BBDD.git)

## CONCLUSIONES

Estos primeros pasos en la creación de este proyecto de base de datos me han ayudado a:

- Distinguir entre atributos propios de la una entidad y los de una relación (por ejemplo, el stock).
- La relevancia de la correcta asignación de la cardinalidad de las relaciones para su posterior paso a tablas.
- La instalación y creación de un servidor local en mi equipo personal para posteriormente poder poblar la BBDD y poder lanzar consultas.