**EJERCICIOS DE NIVELACIÓN SQL**

**Indicaciones**

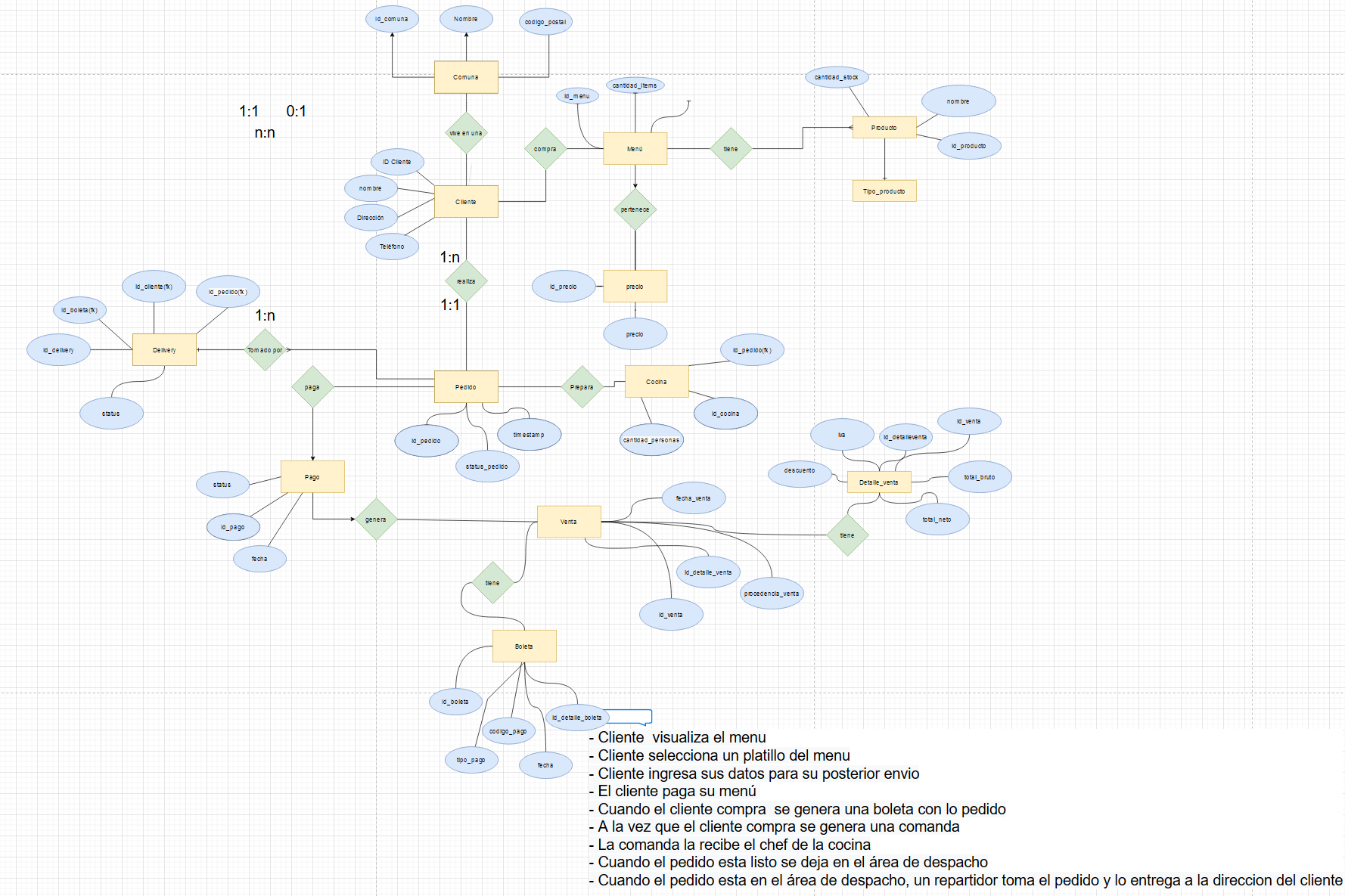
* Modo de entrega: Individual, en aula de MIND HUB, Entrega clase 8.1
* Plazo: Hasta las 9:00 del 12 de abril
* Adjuntos: Este documento, más archivos de diagrama & schema.sql

**Grupo N°: 3**

**Integrantes:**

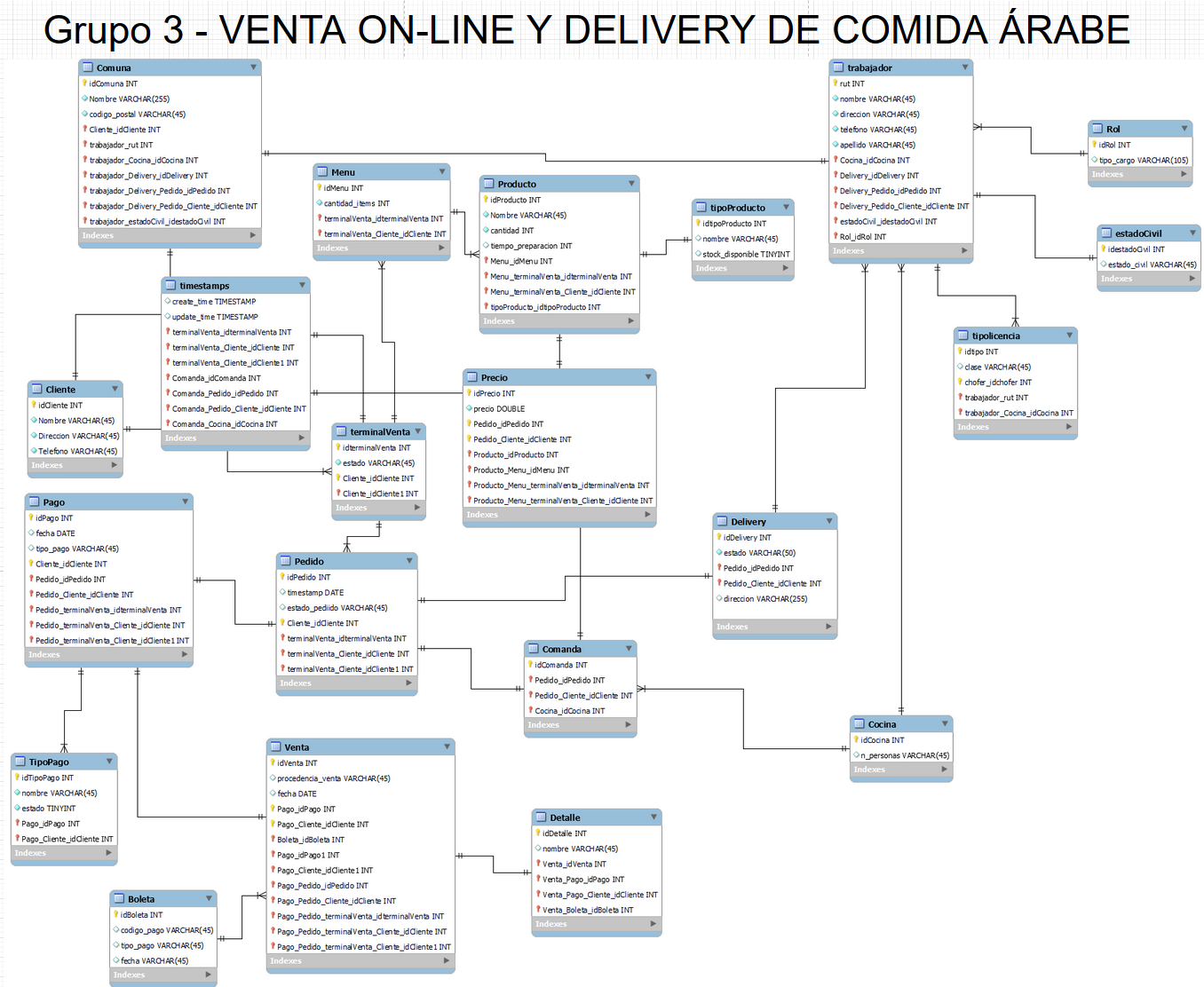
* Felipe bravo
* Carlos Loyola
* Carlo Schoenfeldt
* Ma. Natalia Guajardo

**Modelo Entidad Relación:**



URL https://app.diagrams.net/#G18hE6CAHNmjGlcd-x0wUzq6-Y3K3yoWjF

**Modelo Relacional**



URL https://app.diagrams.net/#G18hE6CAHNmjGlcd-x0wUzq6-Y3K3yoWjF

**Código Schema .SQL**

-- MySQL Script generated by MySQL Workbench

-- Mon Apr 11 14:41:25 2022

-- 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 mydb

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

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

-- Schema mydb

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

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

USE `mydb` ;

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

-- Table `mydb`.`Cliente`

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

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

  `idCliente` INT NOT NULL AUTO\_INCREMENT,

  `Nombre` VARCHAR(45) NOT NULL,

  `Direccion` VARCHAR(45) NOT NULL,

  `Telefono` VARCHAR(45) NOT NULL,

  PRIMARY KEY (`idCliente`))

ENGINE = InnoDB;

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

-- Table `mydb`.`Cocina`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Cocina` (

  `idCocina` INT NOT NULL,

  `n\_personas` VARCHAR(45) NULL,

  PRIMARY KEY (`idCocina`))

ENGINE = InnoDB;

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

-- Table `mydb`.`terminalVenta`

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

CREATE TABLE IF NOT EXISTS `mydb`.`terminalVenta` (

  `idterminalVenta` INT NOT NULL,

  `estado` VARCHAR(45) NOT NULL,

  `Cliente\_idCliente` INT NOT NULL,

  `Cliente\_idCliente1` INT NOT NULL,

  PRIMARY KEY (`idterminalVenta`, `Cliente\_idCliente`, `Cliente\_idCliente1`),

  INDEX `fk\_terminalVenta\_Cliente1\_idx` (`Cliente\_idCliente1` ASC) VISIBLE,

  CONSTRAINT `fk\_terminalVenta\_Cliente1`

    FOREIGN KEY (`Cliente\_idCliente1`)

    REFERENCES `mydb`.`Cliente` (`idCliente`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`Pedido`

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

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

  `idPedido` INT NOT NULL,

  `timestamp` DATE NULL,

  `estado\_pediido` VARCHAR(45) NULL,

  `Cliente\_idCliente` INT NOT NULL,

  `terminalVenta\_idterminalVenta` INT NOT NULL,

  `terminalVenta\_Cliente\_idCliente` INT NOT NULL,

  `terminalVenta\_Cliente\_idCliente1` INT NOT NULL,

  PRIMARY KEY (`idPedido`, `Cliente\_idCliente`, `terminalVenta\_idterminalVenta`, `terminalVenta\_Cliente\_idCliente`, `terminalVenta\_Cliente\_idCliente1`),

  INDEX `fk\_Pedido\_terminalVenta1\_idx` (`terminalVenta\_idterminalVenta` ASC, `terminalVenta\_Cliente\_idCliente` ASC, `terminalVenta\_Cliente\_idCliente1` ASC) VISIBLE,

  CONSTRAINT `fk\_Pedido\_terminalVenta1`

    FOREIGN KEY (`terminalVenta\_idterminalVenta` , `terminalVenta\_Cliente\_idCliente` , `terminalVenta\_Cliente\_idCliente1`)

    REFERENCES `mydb`.`terminalVenta` (`idterminalVenta` , `Cliente\_idCliente` , `Cliente\_idCliente1`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`Delivery`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Delivery` (

  `idDelivery` INT NOT NULL,

  `estado` VARCHAR(50) NOT NULL,

  `Pedido\_idPedido` INT NOT NULL,

  `Pedido\_Cliente\_idCliente` INT NOT NULL,

  `direccion` VARCHAR(255) NULL,

  PRIMARY KEY (`idDelivery`, `Pedido\_idPedido`, `Pedido\_Cliente\_idCliente`),

  INDEX `fk\_Delivery\_Pedido1\_idx` (`Pedido\_idPedido` ASC, `Pedido\_Cliente\_idCliente` ASC) VISIBLE,

  CONSTRAINT `fk\_Delivery\_Pedido1`

    FOREIGN KEY (`Pedido\_idPedido` , `Pedido\_Cliente\_idCliente`)

    REFERENCES `mydb`.`Pedido` (`idPedido` , `Cliente\_idCliente`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`estadoCivil`

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

CREATE TABLE IF NOT EXISTS `mydb`.`estadoCivil` (

  `idestadoCivil` INT NOT NULL,

  `estado\_civil` VARCHAR(45) NULL,

  PRIMARY KEY (`idestadoCivil`))

ENGINE = InnoDB;

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

-- Table `mydb`.`Rol`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Rol` (

  `idRol` INT NOT NULL,

  `tipo\_cargo` VARCHAR(105) NULL,

  PRIMARY KEY (`idRol`))

ENGINE = InnoDB;

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

-- Table `mydb`.`trabajador`

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

CREATE TABLE IF NOT EXISTS `mydb`.`trabajador` (

  `rut` INT NOT NULL,

  `nombre` VARCHAR(45) NOT NULL,

  `direccion` VARCHAR(45) NOT NULL,

  `telefono` VARCHAR(45) NOT NULL,

  `apellido` VARCHAR(45) NOT NULL,

  `Cocina\_idCocina` INT NOT NULL,

  `Delivery\_idDelivery` INT NOT NULL,

  `Delivery\_Pedido\_idPedido` INT NOT NULL,

  `Delivery\_Pedido\_Cliente\_idCliente` INT NOT NULL,

  `estadoCivil\_idestadoCivil` INT NOT NULL,

  `Rol\_idRol` INT NOT NULL,

  PRIMARY KEY (`rut`, `Cocina\_idCocina`, `Delivery\_idDelivery`, `Delivery\_Pedido\_idPedido`, `Delivery\_Pedido\_Cliente\_idCliente`, `estadoCivil\_idestadoCivil`, `Rol\_idRol`),

  INDEX `fk\_trabajador\_Cocina1\_idx` (`Cocina\_idCocina` ASC) VISIBLE,

  INDEX `fk\_trabajador\_Delivery1\_idx` (`Delivery\_idDelivery` ASC, `Delivery\_Pedido\_idPedido` ASC, `Delivery\_Pedido\_Cliente\_idCliente` ASC) VISIBLE,

  INDEX `fk\_trabajador\_estadoCivil1\_idx` (`estadoCivil\_idestadoCivil` ASC) VISIBLE,

  INDEX `fk\_trabajador\_Rol1\_idx` (`Rol\_idRol` ASC) VISIBLE,

  CONSTRAINT `fk\_trabajador\_Cocina1`

    FOREIGN KEY (`Cocina\_idCocina`)

    REFERENCES `mydb`.`Cocina` (`idCocina`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION,

  CONSTRAINT `fk\_trabajador\_Delivery1`

    FOREIGN KEY (`Delivery\_idDelivery` , `Delivery\_Pedido\_idPedido` , `Delivery\_Pedido\_Cliente\_idCliente`)

    REFERENCES `mydb`.`Delivery` (`idDelivery` , `Pedido\_idPedido` , `Pedido\_Cliente\_idCliente`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION,

  CONSTRAINT `fk\_trabajador\_estadoCivil1`

    FOREIGN KEY (`estadoCivil\_idestadoCivil`)

    REFERENCES `mydb`.`estadoCivil` (`idestadoCivil`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION,

  CONSTRAINT `fk\_trabajador\_Rol1`

    FOREIGN KEY (`Rol\_idRol`)

    REFERENCES `mydb`.`Rol` (`idRol`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`Comuna`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Comuna` (

  `idComuna` INT NOT NULL AUTO\_INCREMENT,

  `Nombre` VARCHAR(255) NOT NULL,

  `codigo\_postal` VARCHAR(45) NOT NULL,

  `Cliente\_idCliente` INT NOT NULL,

  `trabajador\_rut` INT NOT NULL,

  `trabajador\_Cocina\_idCocina` INT NOT NULL,

  `trabajador\_Delivery\_idDelivery` INT NOT NULL,

  `trabajador\_Delivery\_Pedido\_idPedido` INT NOT NULL,

  `trabajador\_Delivery\_Pedido\_Cliente\_idCliente` INT NOT NULL,

  `trabajador\_estadoCivil\_idestadoCivil` INT NOT NULL,

  PRIMARY KEY (`idComuna`, `Cliente\_idCliente`, `trabajador\_rut`, `trabajador\_Cocina\_idCocina`, `trabajador\_Delivery\_idDelivery`, `trabajador\_Delivery\_Pedido\_idPedido`, `trabajador\_Delivery\_Pedido\_Cliente\_idCliente`, `trabajador\_estadoCivil\_idestadoCivil`),

  INDEX `fk\_Comuna\_Cliente1\_idx` (`Cliente\_idCliente` ASC) VISIBLE,

  INDEX `fk\_Comuna\_trabajador1\_idx` (`trabajador\_rut` ASC, `trabajador\_Cocina\_idCocina` ASC, `trabajador\_Delivery\_idDelivery` ASC, `trabajador\_Delivery\_Pedido\_idPedido` ASC, `trabajador\_Delivery\_Pedido\_Cliente\_idCliente` ASC, `trabajador\_estadoCivil\_idestadoCivil` ASC) VISIBLE,

  CONSTRAINT `fk\_Comuna\_Cliente1`

    FOREIGN KEY (`Cliente\_idCliente`)

    REFERENCES `mydb`.`Cliente` (`idCliente`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION,

  CONSTRAINT `fk\_Comuna\_trabajador1`

    FOREIGN KEY (`trabajador\_rut` , `trabajador\_Cocina\_idCocina` , `trabajador\_Delivery\_idDelivery` , `trabajador\_Delivery\_Pedido\_idPedido` , `trabajador\_Delivery\_Pedido\_Cliente\_idCliente` , `trabajador\_estadoCivil\_idestadoCivil`)

    REFERENCES `mydb`.`trabajador` (`rut` , `Cocina\_idCocina` , `Delivery\_idDelivery` , `Delivery\_Pedido\_idPedido` , `Delivery\_Pedido\_Cliente\_idCliente` , `estadoCivil\_idestadoCivil`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`Menu`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Menu` (

  `idMenu` INT NOT NULL AUTO\_INCREMENT,

  `cantidad\_items` INT NOT NULL,

  `terminalVenta\_idterminalVenta` INT NOT NULL,

  `terminalVenta\_Cliente\_idCliente` INT NOT NULL,

  PRIMARY KEY (`idMenu`, `terminalVenta\_idterminalVenta`, `terminalVenta\_Cliente\_idCliente`),

  INDEX `fk\_Menu\_terminalVenta1\_idx` (`terminalVenta\_idterminalVenta` ASC, `terminalVenta\_Cliente\_idCliente` ASC) VISIBLE,

  CONSTRAINT `fk\_Menu\_terminalVenta1`

    FOREIGN KEY (`terminalVenta\_idterminalVenta` , `terminalVenta\_Cliente\_idCliente`)

    REFERENCES `mydb`.`terminalVenta` (`idterminalVenta` , `Cliente\_idCliente`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`tipoProducto`

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

CREATE TABLE IF NOT EXISTS `mydb`.`tipoProducto` (

  `idtipoProducto` INT NOT NULL,

  `nombre` VARCHAR(45) NULL,

  `stock\_disponible` TINYINT NULL,

  PRIMARY KEY (`idtipoProducto`))

ENGINE = InnoDB;

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

-- Table `mydb`.`Producto`

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

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

  `idProducto` INT NOT NULL,

  `Nombre` VARCHAR(45) NOT NULL,

  `cantidad` INT NOT NULL,

  `tiempo\_preparacion` INT NULL,

  `Menu\_idMenu` INT NOT NULL,

  `Menu\_terminalVenta\_idterminalVenta` INT NOT NULL,

  `Menu\_terminalVenta\_Cliente\_idCliente` INT NOT NULL,

  `tipoProducto\_idtipoProducto` INT NOT NULL,

  PRIMARY KEY (`idProducto`, `Menu\_idMenu`, `Menu\_terminalVenta\_idterminalVenta`, `Menu\_terminalVenta\_Cliente\_idCliente`, `tipoProducto\_idtipoProducto`),

  INDEX `fk\_Producto\_Menu1\_idx` (`Menu\_idMenu` ASC, `Menu\_terminalVenta\_idterminalVenta` ASC, `Menu\_terminalVenta\_Cliente\_idCliente` ASC) VISIBLE,

  INDEX `fk\_Producto\_tipoProducto1\_idx` (`tipoProducto\_idtipoProducto` ASC) VISIBLE,

  CONSTRAINT `fk\_Producto\_Menu1`

    FOREIGN KEY (`Menu\_idMenu` , `Menu\_terminalVenta\_idterminalVenta` , `Menu\_terminalVenta\_Cliente\_idCliente`)

    REFERENCES `mydb`.`Menu` (`idMenu` , `terminalVenta\_idterminalVenta` , `terminalVenta\_Cliente\_idCliente`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION,

  CONSTRAINT `fk\_Producto\_tipoProducto1`

    FOREIGN KEY (`tipoProducto\_idtipoProducto`)

    REFERENCES `mydb`.`tipoProducto` (`idtipoProducto`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`Precio`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Precio` (

  `idPrecio` INT NOT NULL,

  `precio` DOUBLE NOT NULL,

  `Pedido\_idPedido` INT NOT NULL,

  `Pedido\_Cliente\_idCliente` INT NOT NULL,

  `Producto\_idProducto` INT NOT NULL,

  `Producto\_Menu\_idMenu` INT NOT NULL,

  `Producto\_Menu\_terminalVenta\_idterminalVenta` INT NOT NULL,

  `Producto\_Menu\_terminalVenta\_Cliente\_idCliente` INT NOT NULL,

  PRIMARY KEY (`idPrecio`, `Pedido\_idPedido`, `Pedido\_Cliente\_idCliente`, `Producto\_idProducto`, `Producto\_Menu\_idMenu`, `Producto\_Menu\_terminalVenta\_idterminalVenta`, `Producto\_Menu\_terminalVenta\_Cliente\_idCliente`),

  INDEX `fk\_Precio\_Producto1\_idx` (`Producto\_idProducto` ASC, `Producto\_Menu\_idMenu` ASC, `Producto\_Menu\_terminalVenta\_idterminalVenta` ASC, `Producto\_Menu\_terminalVenta\_Cliente\_idCliente` ASC) VISIBLE,

  CONSTRAINT `fk\_Precio\_Producto1`

    FOREIGN KEY (`Producto\_idProducto` , `Producto\_Menu\_idMenu` , `Producto\_Menu\_terminalVenta\_idterminalVenta` , `Producto\_Menu\_terminalVenta\_Cliente\_idCliente`)

    REFERENCES `mydb`.`Producto` (`idProducto` , `Menu\_idMenu` , `Menu\_terminalVenta\_idterminalVenta` , `Menu\_terminalVenta\_Cliente\_idCliente`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`Pago`

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

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

  `idPago` INT NOT NULL,

  `fecha` DATE NULL,

  `tipo\_pago` VARCHAR(45) NULL,

  `Cliente\_idCliente` INT NOT NULL,

  `Pedido\_idPedido` INT NOT NULL,

  `Pedido\_Cliente\_idCliente` INT NOT NULL,

  `Pedido\_terminalVenta\_idterminalVenta` INT NOT NULL,

  `Pedido\_terminalVenta\_Cliente\_idCliente` INT NOT NULL,

  `Pedido\_terminalVenta\_Cliente\_idCliente1` INT NOT NULL,

  PRIMARY KEY (`idPago`, `Cliente\_idCliente`, `Pedido\_idPedido`, `Pedido\_Cliente\_idCliente`, `Pedido\_terminalVenta\_idterminalVenta`, `Pedido\_terminalVenta\_Cliente\_idCliente`, `Pedido\_terminalVenta\_Cliente\_idCliente1`),

  INDEX `fk\_Pago\_Pedido1\_idx` (`Pedido\_idPedido` ASC, `Pedido\_Cliente\_idCliente` ASC, `Pedido\_terminalVenta\_idterminalVenta` ASC, `Pedido\_terminalVenta\_Cliente\_idCliente` ASC, `Pedido\_terminalVenta\_Cliente\_idCliente1` ASC) VISIBLE,

  CONSTRAINT `fk\_Pago\_Pedido1`

    FOREIGN KEY (`Pedido\_idPedido` , `Pedido\_Cliente\_idCliente` , `Pedido\_terminalVenta\_idterminalVenta` , `Pedido\_terminalVenta\_Cliente\_idCliente` , `Pedido\_terminalVenta\_Cliente\_idCliente1`)

    REFERENCES `mydb`.`Pedido` (`idPedido` , `Cliente\_idCliente` , `terminalVenta\_idterminalVenta` , `terminalVenta\_Cliente\_idCliente` , `terminalVenta\_Cliente\_idCliente1`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`Boleta`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Boleta` (

  `idBoleta` INT NOT NULL,

  `codigo\_pago` VARCHAR(45) NULL,

  `tipo\_pago` VARCHAR(45) NULL,

  `fecha` VARCHAR(45) NULL,

  PRIMARY KEY (`idBoleta`))

ENGINE = InnoDB;

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

-- Table `mydb`.`Comanda`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Comanda` (

  `idComanda` INT NOT NULL,

  `Pedido\_idPedido` INT NOT NULL,

  `Pedido\_Cliente\_idCliente` INT NOT NULL,

  `Cocina\_idCocina` INT NOT NULL,

  PRIMARY KEY (`idComanda`, `Pedido\_idPedido`, `Pedido\_Cliente\_idCliente`, `Cocina\_idCocina`),

  INDEX `fk\_Comanda\_Pedido1\_idx` (`Pedido\_idPedido` ASC, `Pedido\_Cliente\_idCliente` ASC) VISIBLE,

  INDEX `fk\_Comanda\_Cocina1\_idx` (`Cocina\_idCocina` ASC) VISIBLE,

  CONSTRAINT `fk\_Comanda\_Pedido1`

    FOREIGN KEY (`Pedido\_idPedido` , `Pedido\_Cliente\_idCliente`)

    REFERENCES `mydb`.`Pedido` (`idPedido` , `Cliente\_idCliente`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION,

  CONSTRAINT `fk\_Comanda\_Cocina1`

    FOREIGN KEY (`Cocina\_idCocina`)

    REFERENCES `mydb`.`Cocina` (`idCocina`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`timestamps`

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

CREATE TABLE IF NOT EXISTS `mydb`.`timestamps` (

  `create\_time` TIMESTAMP NULL DEFAULT CURRENT\_TIMESTAMP,

  `update\_time` TIMESTAMP NULL,

  `terminalVenta\_idterminalVenta` INT NOT NULL,

  `terminalVenta\_Cliente\_idCliente` INT NOT NULL,

  `terminalVenta\_Cliente\_idCliente1` INT NOT NULL,

  `Comanda\_idComanda` INT NOT NULL,

  `Comanda\_Pedido\_idPedido` INT NOT NULL,

  `Comanda\_Pedido\_Cliente\_idCliente` INT NOT NULL,

  `Comanda\_Cocina\_idCocina` INT NOT NULL,

  PRIMARY KEY (`terminalVenta\_idterminalVenta`, `terminalVenta\_Cliente\_idCliente`, `terminalVenta\_Cliente\_idCliente1`, `Comanda\_idComanda`, `Comanda\_Pedido\_idPedido`, `Comanda\_Pedido\_Cliente\_idCliente`, `Comanda\_Cocina\_idCocina`),

  INDEX `fk\_timestamps\_Comanda1\_idx` (`Comanda\_idComanda` ASC, `Comanda\_Pedido\_idPedido` ASC, `Comanda\_Pedido\_Cliente\_idCliente` ASC, `Comanda\_Cocina\_idCocina` ASC) VISIBLE,

  CONSTRAINT `fk\_timestamps\_terminalVenta1`

    FOREIGN KEY (`terminalVenta\_idterminalVenta` , `terminalVenta\_Cliente\_idCliente` , `terminalVenta\_Cliente\_idCliente1`)

    REFERENCES `mydb`.`terminalVenta` (`idterminalVenta` , `Cliente\_idCliente` , `Cliente\_idCliente1`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION,

  CONSTRAINT `fk\_timestamps\_Comanda1`

    FOREIGN KEY (`Comanda\_idComanda` , `Comanda\_Pedido\_idPedido` , `Comanda\_Pedido\_Cliente\_idCliente` , `Comanda\_Cocina\_idCocina`)

    REFERENCES `mydb`.`Comanda` (`idComanda` , `Pedido\_idPedido` , `Pedido\_Cliente\_idCliente` , `Cocina\_idCocina`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION);

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

-- Table `mydb`.`Venta`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Venta` (

  `idVenta` INT NOT NULL,

  `procedencia\_venta` VARCHAR(45) NULL,

  `Pago\_idPago` INT NOT NULL,

  `Pago\_Cliente\_idCliente` INT NOT NULL,

  `Boleta\_idBoleta` INT NOT NULL,

  `Pago\_idPago1` INT NOT NULL,

  `Pago\_Cliente\_idCliente1` INT NOT NULL,

  `Pago\_Pedido\_idPedido` INT NOT NULL,

  `Pago\_Pedido\_Cliente\_idCliente` INT NOT NULL,

  `Pago\_Pedido\_terminalVenta\_idterminalVenta` INT NOT NULL,

  `Pago\_Pedido\_terminalVenta\_Cliente\_idCliente` INT NOT NULL,

  `Pago\_Pedido\_terminalVenta\_Cliente\_idCliente1` INT NOT NULL,

  `timestamps\_terminalVenta\_idterminalVenta` INT NOT NULL,

  `timestamps\_terminalVenta\_Cliente\_idCliente` INT NOT NULL,

  `timestamps\_terminalVenta\_Cliente\_idCliente1` INT NOT NULL,

  `timestamps\_Comanda\_idComanda` INT NOT NULL,

  `timestamps\_Comanda\_Pedido\_idPedido` INT NOT NULL,

  `timestamps\_Comanda\_Pedido\_Cliente\_idCliente` INT NOT NULL,

  `timestamps\_Comanda\_Cocina\_idCocina` INT NOT NULL,

  PRIMARY KEY (`idVenta`, `Pago\_idPago`, `Pago\_Cliente\_idCliente`, `Boleta\_idBoleta`, `Pago\_idPago1`, `Pago\_Cliente\_idCliente1`, `Pago\_Pedido\_idPedido`, `Pago\_Pedido\_Cliente\_idCliente`, `Pago\_Pedido\_terminalVenta\_idterminalVenta`, `Pago\_Pedido\_terminalVenta\_Cliente\_idCliente`, `Pago\_Pedido\_terminalVenta\_Cliente\_idCliente1`, `timestamps\_terminalVenta\_idterminalVenta`, `timestamps\_terminalVenta\_Cliente\_idCliente`, `timestamps\_terminalVenta\_Cliente\_idCliente1`, `timestamps\_Comanda\_idComanda`, `timestamps\_Comanda\_Pedido\_idPedido`, `timestamps\_Comanda\_Pedido\_Cliente\_idCliente`, `timestamps\_Comanda\_Cocina\_idCocina`),

  INDEX `fk\_Venta\_Boleta1\_idx` (`Boleta\_idBoleta` ASC) VISIBLE,

  INDEX `fk\_Venta\_Pago1\_idx` (`Pago\_idPago1` ASC, `Pago\_Cliente\_idCliente1` ASC, `Pago\_Pedido\_idPedido` ASC, `Pago\_Pedido\_Cliente\_idCliente` ASC, `Pago\_Pedido\_terminalVenta\_idterminalVenta` ASC, `Pago\_Pedido\_terminalVenta\_Cliente\_idCliente` ASC, `Pago\_Pedido\_terminalVenta\_Cliente\_idCliente1` ASC) VISIBLE,

  INDEX `fk\_Venta\_timestamps1\_idx` (`timestamps\_terminalVenta\_idterminalVenta` ASC, `timestamps\_terminalVenta\_Cliente\_idCliente` ASC, `timestamps\_terminalVenta\_Cliente\_idCliente1` ASC, `timestamps\_Comanda\_idComanda` ASC, `timestamps\_Comanda\_Pedido\_idPedido` ASC, `timestamps\_Comanda\_Pedido\_Cliente\_idCliente` ASC, `timestamps\_Comanda\_Cocina\_idCocina` ASC) VISIBLE,

  CONSTRAINT `fk\_Venta\_Boleta1`

    FOREIGN KEY (`Boleta\_idBoleta`)

    REFERENCES `mydb`.`Boleta` (`idBoleta`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION,

  CONSTRAINT `fk\_Venta\_Pago1`

    FOREIGN KEY (`Pago\_idPago1` , `Pago\_Cliente\_idCliente1` , `Pago\_Pedido\_idPedido` , `Pago\_Pedido\_Cliente\_idCliente` , `Pago\_Pedido\_terminalVenta\_idterminalVenta` , `Pago\_Pedido\_terminalVenta\_Cliente\_idCliente` , `Pago\_Pedido\_terminalVenta\_Cliente\_idCliente1`)

    REFERENCES `mydb`.`Pago` (`idPago` , `Cliente\_idCliente` , `Pedido\_idPedido` , `Pedido\_Cliente\_idCliente` , `Pedido\_terminalVenta\_idterminalVenta` , `Pedido\_terminalVenta\_Cliente\_idCliente` , `Pedido\_terminalVenta\_Cliente\_idCliente1`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION,

  CONSTRAINT `fk\_Venta\_timestamps1`

    FOREIGN KEY (`timestamps\_terminalVenta\_idterminalVenta` , `timestamps\_terminalVenta\_Cliente\_idCliente` , `timestamps\_terminalVenta\_Cliente\_idCliente1` , `timestamps\_Comanda\_idComanda` , `timestamps\_Comanda\_Pedido\_idPedido` , `timestamps\_Comanda\_Pedido\_Cliente\_idCliente` , `timestamps\_Comanda\_Cocina\_idCocina`)

    REFERENCES `mydb`.`timestamps` (`terminalVenta\_idterminalVenta` , `terminalVenta\_Cliente\_idCliente` , `terminalVenta\_Cliente\_idCliente1` , `Comanda\_idComanda` , `Comanda\_Pedido\_idPedido` , `Comanda\_Pedido\_Cliente\_idCliente` , `Comanda\_Cocina\_idCocina`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`Detalle`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Detalle` (

  `idDetalle` INT NOT NULL,

  `nombre` VARCHAR(45) NULL,

  `Venta\_idVenta` INT NOT NULL,

  `Venta\_Pago\_idPago` INT NOT NULL,

  `Venta\_Pago\_Cliente\_idCliente` INT NOT NULL,

  `Venta\_Boleta\_idBoleta` INT NOT NULL,

  PRIMARY KEY (`idDetalle`, `Venta\_idVenta`, `Venta\_Pago\_idPago`, `Venta\_Pago\_Cliente\_idCliente`, `Venta\_Boleta\_idBoleta`),

  INDEX `fk\_Detalle\_Venta1\_idx` (`Venta\_idVenta` ASC, `Venta\_Pago\_idPago` ASC, `Venta\_Pago\_Cliente\_idCliente` ASC, `Venta\_Boleta\_idBoleta` ASC) VISIBLE,

  CONSTRAINT `fk\_Detalle\_Venta1`

    FOREIGN KEY (`Venta\_idVenta` , `Venta\_Pago\_idPago` , `Venta\_Pago\_Cliente\_idCliente` , `Venta\_Boleta\_idBoleta`)

    REFERENCES `mydb`.`Venta` (`idVenta` , `Pago\_idPago` , `Pago\_Cliente\_idCliente` , `Boleta\_idBoleta`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`tipolicencia`

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

CREATE TABLE IF NOT EXISTS `mydb`.`tipolicencia` (

  `idtipo` INT NOT NULL,

  `clase` VARCHAR(45) NULL,

  `chofer\_idchofer` INT NOT NULL,

  `trabajador\_rut` INT NOT NULL,

  `trabajador\_Cocina\_idCocina` INT NOT NULL,

  PRIMARY KEY (`idtipo`, `chofer\_idchofer`, `trabajador\_rut`, `trabajador\_Cocina\_idCocina`),

  INDEX `fk\_tipolicencia\_trabajador1\_idx` (`trabajador\_rut` ASC, `trabajador\_Cocina\_idCocina` ASC) VISIBLE,

  CONSTRAINT `fk\_tipolicencia\_trabajador1`

    FOREIGN KEY (`trabajador\_rut` , `trabajador\_Cocina\_idCocina`)

    REFERENCES `mydb`.`trabajador` (`rut` , `Cocina\_idCocina`)

    ON DELETE NO ACTION

    ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`TipoPago`

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

CREATE TABLE IF NOT EXISTS `mydb`.`TipoPago` (

  `idTipoPago` INT NOT NULL,

  `nombre` VARCHAR(45) NOT NULL,

  `estado` TINYINT NOT NULL,

  `Pago\_idPago` INT NOT NULL,

  `Pago\_Cliente\_idCliente` INT NOT NULL,

  PRIMARY KEY (`idTipoPago`, `Pago\_idPago`, `Pago\_Cliente\_idCliente`),

  INDEX `fk\_TipoPago\_Pago1\_idx` (`Pago\_idPago` ASC, `Pago\_Cliente\_idCliente` ASC) VISIBLE,

  CONSTRAINT `fk\_TipoPago\_Pago1`

    FOREIGN KEY (`Pago\_idPago` , `Pago\_Cliente\_idCliente`)

    REFERENCES `mydb`.`Pago` (`idPago` , `Cliente\_idCliente`)

    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;

    © 2022 GitHub, Inc.

    Terms

    Privacy

    Security

    Status

    Docs

    Contact GitHub

    Pricing

    API

    Training

    Blog

    About

Loading complete

**Guía 8.1:**

Con la base de datos del aula (sakila.db), responda a las siguientes consultas en SQL.

Usaremos https://sqliteonline.com/

1. De la tabla film traer solo las columnas title, release\_year, special\_features y rating.

select title, release\_year, special\_features, rating from film;

2. De la tabla actor traer solo el first\_name y el last\_name.

select first\_name, last\_name from actor;

3. De la tabla customer traer las columnas customer\_id, first\_name, email.

select customer\_id, first\_name, email from customer;

4. ¿Qué consulta tengo que ejecutar para traer todas las columnas de la tabla staff?

select \* from staff;

5. Ver toda la tabla film\_list completa.

select \* from film\_list;

6. Filtrar los registros de la tabla payment que hayan pagado exactamente $0.99.

select \* from payment where amount = '0.99'

7. Idem a 6) pero traer solo las columnas customer\_id, rental\_id y amount para verificar que estén bien filtrados.

select customer\_id, rental\_id,  amount from payment where amount = '0.99';

8. De la tabla payment traer todos los alquileres del cliente cuyo número de cliente sea 386 y el monto pagado sea mayor a 2.

select rental\_id from payment where customer\_id = '386' and amount > 2;

9. ¿Cuáles son todas las películas que pertenece al género Horror? Usar la tabla film\_list.

SELECT \* from film\_list WHERE category='Horror'

10. ¿Qué películas pertenecen al género Foreign y duran más de 2 horas? Mostrar los títulos, la duración y el rating.

SELECT title,length,rating from film\_list WHERE category='Foreign' AND length>120

11. Seleccionar de la tabla actor aquellos registros cuyo apellido sea Kilmer, Bloom o Wayne.

SELECT \* from actor where last\_name in ('Kilmer', 'Bloom', 'Wayne');

12. ¿Qué películas de la tabla film aceptan un período de alquiler entre 4 y 6 días? Mostrar solo las columnas título, duración del alquiler y duración de la película.

SELECT title, length, rental\_duration from film where rental\_duration>4 AND rental\_duration<6

13. Un cliente de la tienda 2 efectuó un pago por un alquiler pero el personal se olvidó de registrarlo y ahora no recuerda bien el nombre, solo sabe que el nombre empezaba con "CH". Listar los posibles clientes para ver si el empleado recuerda el nombre. Ordernarlos por nombre. Recordar que el símbolo de porcentaje% se utiliza como comodín.

SELECT first\_name from customer where first\_name LIKE 'CH%' and store\_id=2 order by first\_name ;

14. ¿Cuáles son los distintos géneros de películas disponibles en el videoclub?

SELECT DISTINCT category from film\_list

15. Tabla film. ¿Qué películas cuestan menos de 4 dólares, duran entre 80 y 110 minutos y su rating es "PG-13"?

SELECT \* from film\_list where length BETWEEN 80 AND 110 AND price<4 AND rating='PG-13

16. ¿Cuáles son los distintos valores posibles para el rating de las películas?

SELECT DISTINCT rating from film\_list

17. De la tabla customer\_list filtrar los registros de los clientes que sean de Argentina o de Bolivia. Ordenar por país y por nombre.

SELECT \* from customer\_list where country in ('Bolivia','Argentina') order by country, name

18. ¿Cuántos clientes inactivos hay? Ver tabla customer.

SELECT count(\*) from customer where active=0

19. ¿Cuál es el máximo y el mínimo monto pagos en el video club?, ¿y el promedio?

SELECT max(amount), min(amount), avg(amount)  from payment

20. ¿Cuántas películas aptas para todo público hay?, ¿cuál es la duración promedio? Redondear el promedio.

SELECT COUNT(film\_id), ROUND(AVG(length),2) from film WHERE rating='G'

21. ¿Cuántos clientes hay de Argentina, Brazil, Bolivia o Chile? ¿Hay algún otro país de latinoamérica?

CREATE TEMP TABLE lista(pais text);

insert into lista values ('Colombia'),('Costa Rica'),('Cuba'),('Ecuador'),('El Salvador'),('Guatemala'),('Guayana Francesa'),('Granada'),('Guayana'),('Haití'),('Honduras'),('Jamaica'),('México'),('Nicaragua'),('Panamá'),('Paraguay'),('Perú'),('Puerto Rico'),('República Dominicana'),('Surinam'),('Uruguay'),('Venezuela');

SELECT count(customer\_id) as cantidad\_cliente, country from city

INNER JOIN country ON city.country\_id = country.country\_id

INNER JOIN address ON address.city\_id = city.city\_id

INNER JOIN customer ON customer.address\_id = address.address\_id

WHERE country in (SELECT \* from lista)

group by country