

```

-- Crear la base de datos si no existe y seleccionarla
CREATE DATABASE IF NOT EXISTS behome;
USE behome;

-- Desactivar las restricciones de claves foráneas
SET FOREIGN_KEY_CHECKS = 0;

-- Eliminar las tablas en el orden correcto
DROP TABLE IF EXISTS Evento;
DROP TABLE IF EXISTS Calendario;
DROP TABLE IF EXISTS Tarea;
DROP TABLE IF EXISTS ListaCompra_Producto;
DROP TABLE IF EXISTS Producto;
DROP TABLE IF EXISTS ListaCompra;
DROP TABLE IF EXISTS Lista;
DROP TABLE IF EXISTS Usuario;
DROP TABLE IF EXISTS Piso;

-- Volver a activar las restricciones de claves foráneas
SET FOREIGN_KEY_CHECKS = 1;

-- Crear las tablas en el orden correcto
CREATE TABLE Piso (
    id VARCHAR(6) PRIMARY KEY,
    nombre VARCHAR(50)
);

CREATE TABLE Usuario (
    id INT PRIMARY KEY AUTO_INCREMENT,
    nombre VARCHAR(50),
    email VARCHAR(100),
    contrasena VARCHAR(100),
    id_piso VARCHAR(6),
    FOREIGN KEY (id_piso) REFERENCES Piso(id)
);

CREATE TABLE Lista (
    id INT PRIMARY KEY AUTO_INCREMENT,
    nombre VARCHAR(50),
    id_piso VARCHAR(6),
    FOREIGN KEY (id_piso) REFERENCES Piso(id)
);

CREATE TABLE Producto (
    id INT PRIMARY KEY AUTO_INCREMENT,

```

```
    nombre VARCHAR(50)
);
```

```
CREATE TABLE Lista_Producto (
    id_lista INT,
    id_producto INT,
    FOREIGN KEY (id_lista) REFERENCES Lista(id),
    FOREIGN KEY (id_producto) REFERENCES Producto(id),
    PRIMARY KEY (id_lista, id_producto)
);
```

```
CREATE TABLE Tarea (
    id INT PRIMARY KEY AUTO_INCREMENT,
    nombre VARCHAR(100),
    id_usuario INT,
    id_piso VARCHAR(6),
    fecha_limite DATE,
    frecuencia ENUM('diario', 'semanal', 'mensual', 'ninguna'),
    completado BOOLEAN,
    FOREIGN KEY (id_usuario) REFERENCES Usuario(id),
    FOREIGN KEY (id_piso) REFERENCES Piso(id)
);
```

```
CREATE TABLE Calendario (
    id INT PRIMARY KEY AUTO_INCREMENT,
    id_piso VARCHAR(6),
    FOREIGN KEY (id_piso) REFERENCES Piso(id)
);
```

```
CREATE TABLE Evento (
    id INT PRIMARY KEY AUTO_INCREMENT,
    id_calendario INT,
    nombre VARCHAR(100),
    fecha DATE,
    FOREIGN KEY (id_calendario) REFERENCES Calendario(id)
);
```

```
-- Insertar datos de prueba en la tabla Piso
INSERT INTO Piso (id, nombre) VALUES
('1GD3S4', 'El Pisito'),
('9PE3SD', 'La Chozita');
```

```
-- Insertar datos de prueba en la tabla Usuario
```

```
INSERT INTO Usuario (nombre, email, contrasena, id_piso) VALUES
('JuanPe', 'juan@gmail.com', 'contrasena123', '1GD3S4'),
('mariaGarcia', 'maria@example.com', 'password456', '1GD3S4'),
('sara', 'sara@gmail.com', 'sara1234', '9PE3SD');
```

```
-- Insertar datos de prueba en la tabla ListaCompra
INSERT INTO Lista (nombre, id_piso) VALUES
('Compra', '1GD3S4'),
('Viaje', '1GD3S4');
```

```
-- Insertar datos de prueba en la tabla Producto
INSERT INTO Producto (nombre) VALUES
('Leche'),
('Pan'),
('Huevos');
```

```
-- Insertar datos de prueba en la tabla Lista_Producto
INSERT INTO Lista_Producto (id_lista, id_producto) VALUES
(1, 1),
(1, 2),
(2, 2),
(2, 3);
```

```
-- Insertar datos de prueba en la tabla Tarea
INSERT INTO Tarea (nombre, id_usuario, id_piso, fecha_limite, frecuencia, completado)
VALUES
('Limpiar cocina', 1, '1GD3S4', '2024-05-29', 'ninguna', 0),
('Limpiar arenero gato', 2, '1GD3S4', '2024-05-31', 'ninguna', 0),
('Pagar factura de agua', 3, '9PE3SD', '2024-06-10', 'ninguna', 0);
```

```
-- Insertar datos de prueba en la tabla Calendario
INSERT INTO Calendario (id_piso) VALUES
('1GD3S4'),
('9PE3SD');
```

```
-- Insertar datos de prueba en la tabla Evento
INSERT INTO Evento (id_calendario, nombre, fecha) VALUES
(1, 'Reunión de vecinos', '2024-05-05'),
(2, 'Cumpleaños de María', '2024-05-10');
```