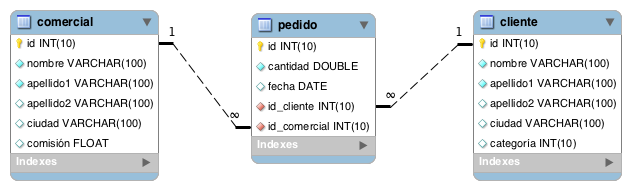
**Gestión de ventas**

**Modelo entidad/relación**



**Base de datos para MySQL**

**DROP** **DATABASE** **IF** **EXISTS** ventas;

**CREATE** **DATABASE** ventas CHARACTER **SET** utf8mb4;

**USE** ventas;

**CREATE** **TABLE** cliente (

**id** INT UNSIGNED AUTO\_INCREMENT **PRIMARY** **KEY**,

nombre VARCHAR(100) **NOT** **NULL**,

apellido1 VARCHAR(100) **NOT** **NULL**,

apellido2 VARCHAR(100),

ciudad VARCHAR(100),

categoria INT UNSIGNED

);

**CREATE** **TABLE** comercial (

**id** INT UNSIGNED AUTO\_INCREMENT **PRIMARY** **KEY**,

nombre VARCHAR(100) **NOT** **NULL**,

apellido1 VARCHAR(100) **NOT** **NULL**,

apellido2 VARCHAR(100),

comision FLOAT

);

**CREATE** **TABLE** pedido (

**id** INT UNSIGNED AUTO\_INCREMENT **PRIMARY** **KEY**,

total DOUBLE **NOT** **NULL**,

fecha DATE,

id\_cliente INT UNSIGNED **NOT** **NULL**,

id\_comercial INT UNSIGNED **NOT** **NULL**,

**FOREIGN** **KEY** (id\_cliente) **REFERENCES** cliente(**id**),

**FOREIGN** **KEY** (id\_comercial) **REFERENCES** comercial(**id**)

);

**INSERT** **INTO** cliente **VALUES**(1, 'Aarón', 'Rivero', 'Gómez', 'Almería', 100);

**INSERT** **INTO** cliente **VALUES**(2, 'Adela', 'Salas', 'Díaz', 'Granada', 200);

**INSERT** **INTO** cliente **VALUES**(3, 'Adolfo', 'Rubio', 'Flores', 'Sevilla', **NULL**);

**INSERT** **INTO** cliente **VALUES**(4, 'Adrián', 'Suárez', **NULL**, 'Jaén', 300);

**INSERT** **INTO** cliente **VALUES**(5, 'Marcos', 'Loyola', 'Méndez', 'Almería', 200);

**INSERT** **INTO** cliente **VALUES**(6, 'María', 'Santana', 'Moreno', 'Cádiz', 100);

**INSERT** **INTO** cliente **VALUES**(7, 'Pilar', 'Ruiz', **NULL**, 'Sevilla', 300);

**INSERT** **INTO** cliente **VALUES**(8, 'Pepe', 'Ruiz', 'Santana', 'Huelva', 200);

**INSERT** **INTO** cliente **VALUES**(9, 'Guillermo', 'López', 'Gómez', 'Granada', 225);

**INSERT** **INTO** cliente **VALUES**(10, 'Daniel', 'Santana', 'Loyola', 'Sevilla', 125);

**INSERT** **INTO** comercial **VALUES**(1, 'Daniel', 'Sáez', 'Vega', 0.15);

**INSERT** **INTO** comercial **VALUES**(2, 'Juan', 'Gómez', 'López', 0.13);

**INSERT** **INTO** comercial **VALUES**(3, 'Diego','Flores', 'Salas', 0.11);

**INSERT** **INTO** comercial **VALUES**(4, 'Marta','Herrera', 'Gil', 0.14);

**INSERT** **INTO** comercial **VALUES**(5, 'Antonio','Carretero', 'Ortega', 0.12);

**INSERT** **INTO** comercial **VALUES**(6, 'Manuel','Domínguez', 'Hernández', 0.13);

**INSERT** **INTO** comercial **VALUES**(7, 'Antonio','Vega', 'Hernández', 0.11);

**INSERT** **INTO** comercial **VALUES**(8, 'Alfredo','Ruiz', 'Flores', 0.05);

**INSERT** **INTO** pedido **VALUES**(1, 150.5, '2017-10-05', 5, 2);

**INSERT** **INTO** pedido **VALUES**(2, 270.65, '2016-09-10', 1, 5);

**INSERT** **INTO** pedido **VALUES**(3, 65.26, '2017-10-05', 2, 1);

**INSERT** **INTO** pedido **VALUES**(4, 110.5, '2016-08-17', 8, 3);

**INSERT** **INTO** pedido **VALUES**(5, 948.5, '2017-09-10', 5, 2);

**INSERT** **INTO** pedido **VALUES**(6, 2400.6, '2016-07-27', 7, 1);

**INSERT** **INTO** pedido **VALUES**(7, 5760, '2015-09-10', 2, 1);

**INSERT** **INTO** pedido **VALUES**(8, 1983.43, '2017-10-10', 4, 6);

**INSERT** **INTO** pedido **VALUES**(9, 2480.4, '2016-10-10', 8, 3);

**INSERT** **INTO** pedido **VALUES**(10, 250.45, '2015-06-27', 8, 2);

**INSERT** **INTO** pedido **VALUES**(11, 75.29, '2016-08-17', 3, 7);

**INSERT** **INTO** pedido **VALUES**(12, 3045.6, '2017-04-25', 2, 1);

**INSERT** **INTO** pedido **VALUES**(13, 545.75, '2019-01-25', 6, 1);

**INSERT** **INTO** pedido **VALUES**(14, 145.82, '2017-02-02', 6, 1);

**INSERT** **INTO** pedido **VALUES**(15, 370.85, '2019-03-11', 1, 5);

**INSERT** **INTO** pedido **VALUES**(16, 2389.23, '2019-03-11', 1, 5);

db.cliente.insertMany([ { \_id: 1, nombre: 'Aarón', apellido1: 'Rivero', apellido2: 'Gómez', ciudad: 'Almería', saldo: 100 }, { \_id: 2, nombre: 'Adela', apellido1: 'Salas', apellido2: 'Díaz', ciudad: 'Granada', saldo: 200 }, { \_id: 3, nombre: 'Adolfo', apellido1: 'Rubio', apellido2: 'Flores', ciudad: 'Sevilla', saldo: null }, { \_id: 4, nombre: 'Adrián', apellido1: 'Suárez', apellido2: null, ciudad: 'Jaén', saldo: 300 }, { \_id: 5, nombre: 'Marcos', apellido1: 'Loyola', apellido2: 'Méndez', ciudad: 'Almería', saldo: 200 }, { \_id: 6, nombre: 'María', apellido1: 'Santana', apellido2: 'Moreno', ciudad: 'Cádiz', saldo: 100 }, { \_id: 7, nombre: 'Pilar', apellido1: 'Ruiz', apellido2: null, ciudad: 'Sevilla', saldo: 300 }, { \_id: 8, nombre: 'Pepe', apellido1: 'Ruiz', apellido2: 'Santana', ciudad: 'Huelva', saldo: 200 }, { \_id: 9, nombre: 'Guillermo', apellido1: 'López', apellido2: 'Gómez', ciudad: 'Granada', saldo: 225 }, { \_id: 10, nombre: 'Daniel', apellido1: 'Santana', apellido2: 'Loyola', ciudad: 'Sevilla', saldo: 125 } ]);

db.comercial.insertMany([ { \_id: 1, nombre: 'Daniel', apellido1: 'Sáez', apellido2: 'Vega', comision: 0.15 }, { \_id: 2, nombre: 'Juan', apellido1: 'Gómez', apellido2: 'López', comision: 0.13 }, { \_id: 3, nombre: 'Diego', apellido1: 'Flores', apellido2: 'Salas', comision: 0.11 }, { \_id: 4, nombre: 'Marta', apellido1: 'Herrera', apellido2: 'Gil', comision: 0.14 }, { \_id: 5, nombre: 'Antonio', apellido1: 'Carretero', apellido2: 'Ortega', comision: 0.12 }, { \_id: 6, nombre: 'Manuel', apellido1: 'Domínguez', apellido2: 'Hernández', comision: 0.13 }, { \_id: 7, nombre: 'Antonio', apellido1: 'Vega', apellido2: 'Hernández', comision: 0.11 }, { \_id: 8, nombre: 'Alfredo', apellido1: 'Ruiz', apellido2: 'Flores', comision: 0.05 } ]);

db.pedido.insertMany([ { \_id: 1, total: 150.5, fecha: ISODate("2017-10-05T00:00:00Z"), cliente\_id: 5, comercial\_id: 2 }, { \_id: 2, total: 270.65, fecha: ISODate("2016-09-10T00:00:00Z"), cliente\_id: 1, comercial\_id: 5 }, { \_id: 3, total: 65.26, fecha: ISODate("2017-10-05T00:00:00Z"), cliente\_id: 2, comercial\_id: 1 }, { \_id: 4, total: 110.5, fecha: ISODate("2016-08-17T00:00:00Z"), cliente\_id: 8, comercial\_id: 3 }, { \_id: 5, total: 948.5, fecha: ISODate("2017-09-10T00:00:00Z"), cliente\_id: 5, comercial\_id: 2 }, { \_id: 6, total: 2400.6, fecha: ISODate("2016-07-27T00:00:00Z"), cliente\_id: 7, comercial\_id: 1 }, { \_id: 7, total: 5760, fecha: ISODate("2015-09-10T00:00:00Z"), cliente\_id: 2, comercial\_id: 1 }, { \_id: 8, total: 1983.43, fecha: ISODate("2017-10-10T00:00:00Z"), cliente\_id: 4, comercial\_id: 6 }, { \_id: 9, total: 2480.4, fecha: ISODate("2016-10-10T00:00:00Z"), cliente\_id: 8, comercial\_id: 3 }, { \_id: 10, total: 250.45, fecha: ISODate("2015-06-27T00:00:00Z"), cliente\_id: 8, comercial\_id: 2 }, { \_id: 11, total: 75.29, fecha: ISODate("2016-08-17T00:00:00Z"), cliente\_id: 3, comercial\_id: 7 }, { \_id: 12, total: 3045.6, fecha: ISODate("2017-04-25T00:00:00Z"), cliente\_id: 2, comercial\_id: 1 }, { \_id: 13, total: 545.75, fecha: ISODate("2019-01-25T00:00:00Z"), cliente\_id: 6, comercial\_id: 1 }, { \_id: 14, total: 145.82, fecha: ISODate("2017-02-02T00:00:00Z"), cliente\_id: 6, comercial\_id: 1 }, { \_id: 15, total: 370.85, fecha: ISODate("2019-03-11T00:00:00Z"), cliente\_id: 1, comercial\_id: 5 }, { \_id: 16, total: 2389.23, fecha: ISODate("2019-03-11T00:00:00Z"), cliente\_id: 1, comercial\_id: 5 } ]);

**Consultas sobre una tabla**

1. Devuelve un listado con todos los pedidos que se han realizado. Los pedidos deben estar ordenados por la fecha de realización, mostrando en primer lugar los pedidos más recientes.

**db.pedido.find().sort({ fecha: -1})**

1. Devuelve todos los datos de los dos pedidos de mayor valor.

**db.pedido.find().sort({ total: -1}).limit(2)**

1. Devuelve un listado con los identificadores de los clientes que han realizado algún pedido. Tenga en cuenta que no debe mostrar identificadores que estén repetidos.

**db.pedido.distinct("cliente\_id")**

1. Devuelve un listado de todos los pedidos que se realizaron durante el año 2017, cuya cantidad total sea superior a 500€.

**db.pedido.find({ fecha: { $gte: ISODate("2017-01-01"), $lt: ISODate("2018-01-01") }, total: { $gt: 500}})**

1. Devuelve un listado con el nombre y los apellidos de los comerciales que tienen una comisión entre 0.05 y 0.11.

**db.comercial.find({ comision: { $gte: 0.05, $lte: 0.11 }}, { nombre:1, apellido1: 1, apellido2: 1})**

1. Devuelve el valor de la comisión de mayor valor que existe en la tabla comercial.

**db.comercial.find().sort({ comision: -1}).limit(1)**

1. Devuelve el identificador, nombre y primer apellido de aquellos clientes cuyo segundo apellido **no** es NULL. El listado deberá estar ordenado alfabéticamente por apellidos y nombre.

**db.cliente.find({ apellido2: {$ne: null}}, {nombre:1, apellido1:1}).sort({apellido1: 1, nombre: 1})**

1. Devuelve un listado de los nombres de los clientes que empiezan por A y terminan por n y también los nombres que empiezan por P. El listado deberá estar ordenado alfabéticamente.

**db.cliente. find({ $or: [ {nombre: /^A.\*n$/}, {nombre: /^P/}]}, {\_id:0, nombre:1}).sort({ nombre:1})**

1. Devuelve un listado de los nombres de los clientes que **no** empiezan por A. El listado deberá estar ordenado alfabéticamente.

**db.cliente. find({ nombre: {$not: { $regex:/^A/ }}}, {\_id:0, nombre:1}).sort({ nombre:1})**

**gestion\_ventas> db.cliente. find({ nombre: {$not:/^A/}}, {\_id:0, nombre:1}).sort({ nombre:1})**

1. Devuelve un listado con los nombres de los comerciales que terminan por el o o. Tenga en cuenta que se deberán eliminar los nombres repetidos.

**db.comercial.distinct("nombre", { nombre: {$regex: /o$/ , $options: "i"}})**

**Consultas multitabla (Composición interna)**

Resuelva todas las consultas utilizando la sintaxis de SQL1 y SQL2.

1. Devuelve un listado con el identificador, nombre y los apellidos de todos los clientes que han realizado algún pedido. El listado debe estar ordenado alfabéticamente y se deben eliminar los elementos repetidos.

|  |
| --- |
| db.getCollection(**'pedido'**).aggregate( |
| [ |
| { |
| $lookup: { |
| from: **'cliente'**, |
| localField: **'cliente\_id'**, |
| foreignField: **'\_id'**, |
| as: **'cliente\_info'** |
| } |
| }, |
| { $unwind: **'$cliente\_info'** }, |
| { |
| $group: { |
| \_id: **'$cliente\_info.\_id'**, |
| nombre: { |
| $first: **'$cliente\_info.nombre'** |
| }, |
| apellido1: { |
| $first: **'$cliente\_info.apellido1'** |
| }, |
| apellido2: { |
| $first: **'$cliente\_info.apellido2'** |
| } |
| } |
| }, |
| { $sort: { apellido1: 1, nombre: 1 } } |
| ], |
| { maxTimeMS: 60000, allowDiskUse: true } |
| ); |

1. Devuelve un listado que muestre todos los pedidos que ha realizado cada cliente. El resultado debe mostrar todos los datos de los pedidos y del cliente. El listado debe mostrar los datos de los clientes ordenados alfabéticamente.

|  |
| --- |
| db.getCollection(**'pedido'**).aggregate( |
| [ |
| { |
| $lookup: { |
| from: **'cliente'**, |
| localField: **'cliente\_id'**, |
| foreignField: **'\_id'**, |
| as: **'cliente\_info'** |
| } |
| }, |
| { $unwind: **'$cliente\_info'** }, |
| { |
| $project: { |
| pedido\_id: 1, |
| total: 1, |
| fecha: 1, |
| cliente\_id: 1, |
| **'Nombre de Cliente'**: |
| **'$cliente\_info.nombre'**, |
| Apellido1: **'$cliente\_info.apellido1'**, |
| Apellido2: **'$cliente\_info.apellido2'** |
| } |
| }, |
| { $sort: { Apellido1: 1 } } |
| ] |
|  |
| ); |

1. Devuelve un listado que muestre todos los pedidos en los que ha participado un comercial. El resultado debe mostrar todos los datos de los pedidos y de los comerciales. El listado debe mostrar los datos de los comerciales ordenados alfabéticamente.

|  |
| --- |
| db.getCollection(**'pedido'**).aggregate( |
| [ |
| { |
| $lookup: { |
| from: **'comercial'**, |
| localField: **'comercial\_id'**, |
| foreignField: **'\_id'**, |
| as: **'comercial\_info'** |
| } |
| }, |
| { $unwind: **'$comercial\_info'** }, |
| { |
| $project: { |
| pedido\_id: 1, |
| total: 1, |
| fecha: 1, |
| comercial\_id: 1, |
| **'Nombre de Comercial'**: |
| **'$comercial\_info.nombre'**, |
| Apellido1: **'$comercial\_info.apellido1'**, |
| Apellido2: **'$comercial\_info.apellido2'** |
| } |
| }, |
| { $sort: { Apellido1: 1 } } |
| ], |
| { maxTimeMS: 60000, allowDiskUse: true } |
| ); |

1. Devuelve un listado que muestre todos los clientes, con todos los pedidos que han realizado y con los datos de los comerciales asociados a cada pedido.

|  |
| --- |
| db.getCollection(**'pedido'**).aggregate( |
| [ |
| { |
| $lookup: { |
| from: **'comercial'**, |
| localField: **'comercial\_id'**, |
| foreignField: **'\_id'**, |
| as: **'comercial\_info'** |
| } |
| }, |
| { $unwind: **'$comercial\_info'** }, |
| { |
| $lookup: { |
| from: **'cliente'**, |
| localField: **'cliente\_id'**, |
| foreignField: **'\_id'**, |
| as: **'cliente\_info'** |
| } |
| }, |
| { $unwind: **'$cliente\_info'** }, |
| { |
| $project: { |
| pedido\_id: 1, |
| total: 1, |
| fecha: 1, |
| cliente\_id: 1, |
| **'Nombre de Cliente'**: |
| **'$cliente\_info.nombre'**, |
| Apellido1: **'$cliente\_info.apellido1'**, |
| Apellido2: **'$cliente\_info.apellido2'**, |
| comercial\_id: 1, |
| **'Nombre de Comercial'**: |
| **'$comercial\_info.nombre'**, |
| **'Apellido Comercial 1'**: |
| **'$comercial\_info.apellido1'**, |
| **'Apellido Comercial 2'**: |
| **'$comercial\_info.apellido2'** |
| } |
| } |
| ], |
| { maxTimeMS: 60000, allowDiskUse: true } |
| ); |

1. Devuelve un listado de todos los clientes que realizaron un pedido durante el año 2017, cuya cantidad esté entre 300 € y 1000 €.

|  |
| --- |
| db.getCollection(**'pedido'**).aggregate( |
| [ |
| { |
| $lookup: { |
| from: **'cliente'**, |
| localField: **'cliente\_id'**, |
| foreignField: **'\_id'**, |
| as: **'cliente\_info'** |
| } |
| }, |
| { $unwind: **'$cliente\_info'** }, |
| { |
| $match: { |
| fecha: { |
| $gte: ISODate( |
| **'2017-01-01T00:00:00.000Z'** |
| ), |
| $lt: ISODate(**'2018-01-01T00:00:00.000Z'**) |
| }, |
| total: { $gte: 300, $lte: 1000 } |
| } |
| }, |
| { |
| $project: { |
| cliente\_id: 1, |
| **'Nombre de Cliente'**: |
| **'$cliente\_info.nombre'**, |
| Apellido1: **'$cliente\_info.apellido1'**, |
| Apellido2: **'$cliente\_info.apellido2'** |
| } |
| } |
| ], |
| { maxTimeMS: 60000, allowDiskUse: true } |
| ); |

1. Devuelve el nombre y los apellidos de todos los comerciales que ha participado en algún pedido realizado por María Santana Moreno.

|  |
| --- |
| db.getCollection(**'pedido'**).aggregate( |
| [ |
| { |
| $lookup: { |
| from: **'comercial'**, |
| localField: **'comercial\_id'**, |
| foreignField: **'\_id'**, |
| as: **'comercial\_info'** |
| } |
| }, |
| { $unwind: **'$comercial\_info'** }, |
| { |
| $lookup: { |
| from: **'cliente'**, |
| localField: **'cliente\_id'**, |
| foreignField: **'\_id'**, |
| as: **'cliente\_info'** |
| } |
| }, |
| { $unwind: **'$cliente\_info'** }, |
| { |
| $match: { |
| **'cliente\_info.nombre'**: **'María'**, |
| **'cliente\_info.apellido1'**: **'Santana'**, |
| **'cliente\_info.apellido2'**: **'Moreno'** |
| } |
| }, |
| { |
| $project: { |
| comercial\_id: 1, |
| **'Nombre de Comercial'**: |
| **'$comercial\_info.nombre'**, |
| **'Apellido Comercial 1'**: |
| **'$comercial\_info.apellido1'**, |
| **'Apellido Comercial 2'**: |
| **'$comercial\_info.apellido2'** |
| } |
| } |
| ], |
| { maxTimeMS: 60000, allowDiskUse: true } |
| ); |

1. Devuelve el nombre de todos los clientes que han realizado algún pedido con el comercial Daniel Sáez Vega.

|  |
| --- |
| db.getCollection(**'pedido'**).aggregate( |
| [ |
| { |
| $lookup: { |
| from: **'comercial'**, |
| localField: **'comercial\_id'**, |
| foreignField: **'\_id'**, |
| as: **'comercial\_info'** |
| } |
| }, |
| { $unwind: **'$comercial\_info'** }, |
| { |
| $lookup: { |
| from: **'cliente'**, |
| localField: **'cliente\_id'**, |
| foreignField: **'\_id'**, |
| as: **'cliente\_info'** |
| } |
| }, |
| { $unwind: **'$cliente\_info'** }, |
| { |
| $match: { |
| **'comercial\_info.nombre'**: **'Daniel'**, |
| **'comercial\_info.apellido1'**: **'Sáez'**, |
| **'comercial\_info.apellido2'**: **'Vega'** |
| } |
| }, |
| { |
| $project: { |
| cliente\_id: 1, |
| **'Nombre de Cliente'**: |
| **'$cliente\_info.nombre'**, |
| Apellido1: **'$cliente\_info.apellido1'**, |
| Apellido2: **'$cliente\_info.apellido2'** |
| } |
| } |
| ], |
| { maxTimeMS: 60000, allowDiskUse: true } |
| ); |