7) Entity Relationship Diagram - Simple Order Manager

Design the model of a simple Order Manager System. The system consists of:
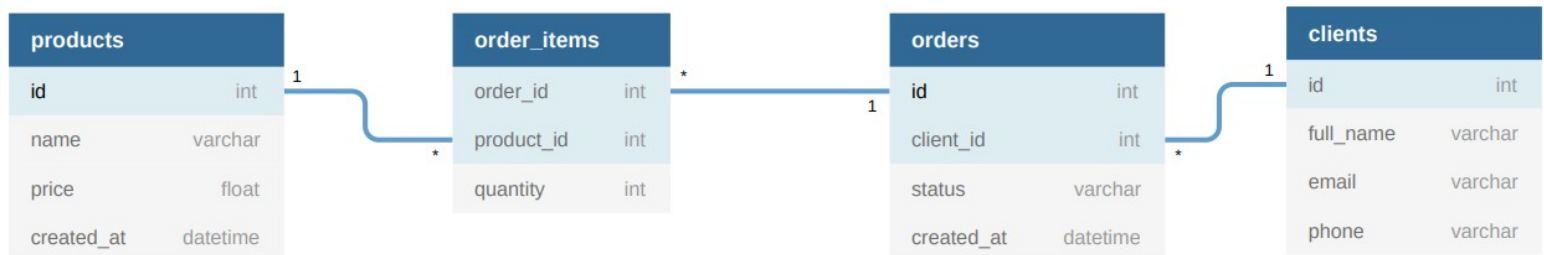 • Clients
  • Products
  • Orders

You can draw, describe, or list the tables as SQL.

Extras:
  • SQL: list ORDERS with number of items
  • Which indexes should be created in this model?

Attention: this exercise is documentation only - there's no executable to run in this case.

Diagram

SQL

```
CREATE TABLE `clients`(
`id` int AUTO_INCREMENT PRIMARY KEY,
`full_name` varchar(255),
`email` varchar(255),
`phone` varchar(255)
);

CREATE TABLE `order_items`(
`order_id` int AUTO_INCREMENT,
`product_id` int,
`quantity` int DEFAULT 1
);
```

```
CREATE TABLE `orders`(
`id` int PRIMARY KEY,
`client_id` int UNIQUE NOT NULL,
`status` varchar(255),
`created_at` datetime COMMENT 'When order created'
);

CREATE TABLE `products`(
`id` int UNIQUE PRIMARY KEY,
`name` varchar(255),
`price` float,
`created_at` datetime DEFAULT (now())
);

ALTER TABLE `order_items` ADD FOREIGN KEY (`order_id`) REFERENCES
`orders` (`id`);
ALTER TABLE `order_items` ADD FOREIGN KEY (`product_id`) REFERENCES
`products` (`id`);
ALTER TABLE `orders` ADD FOREIGN KEY (`client_id`) REFERENCES
`clients` (`id`);
```

Extras

Query:

```
SELECT o.id, o.client_id o.status, COUNT(i.quantity)
FROM orders as o, order_items as i
WHERE o.id = i.order_id
GROUP BY i.order_id
```

Should be created in this model the following indexes:

- clients.id → Primary Key
- orders.id → Primary Key
- products.id → Primary Key, UNIQUE