

# **Princípios de banco de dados com MySQL 5.7**

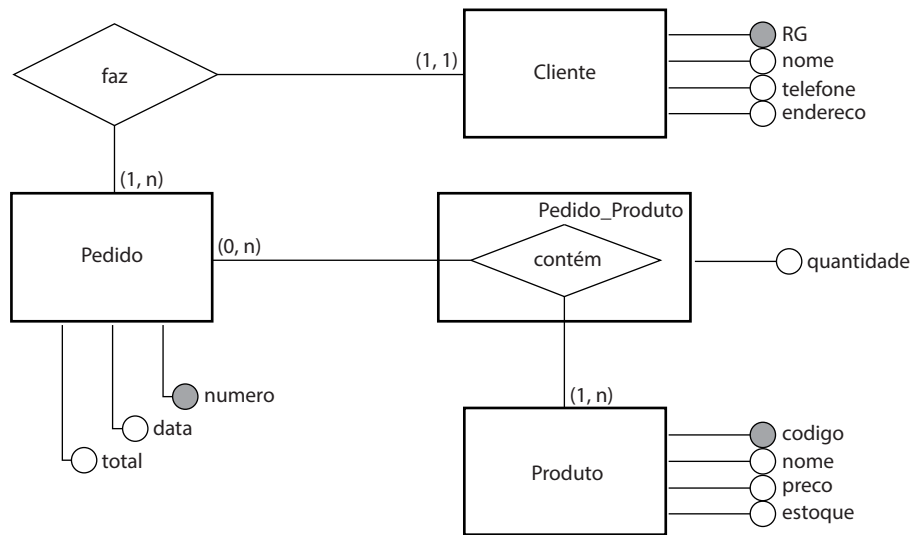
Respostas dos exercícios



## Capítulo 1 - Modelagem de dados

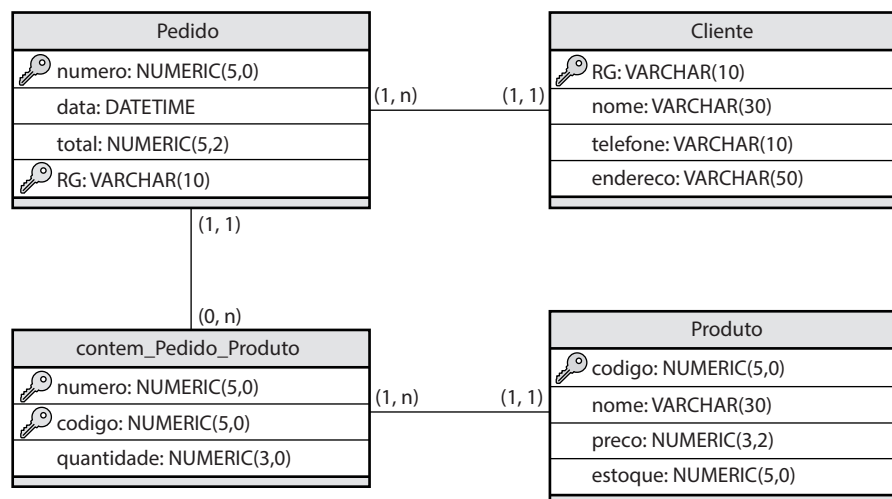
1.

### Modelo conceitual



Ivan Sardella

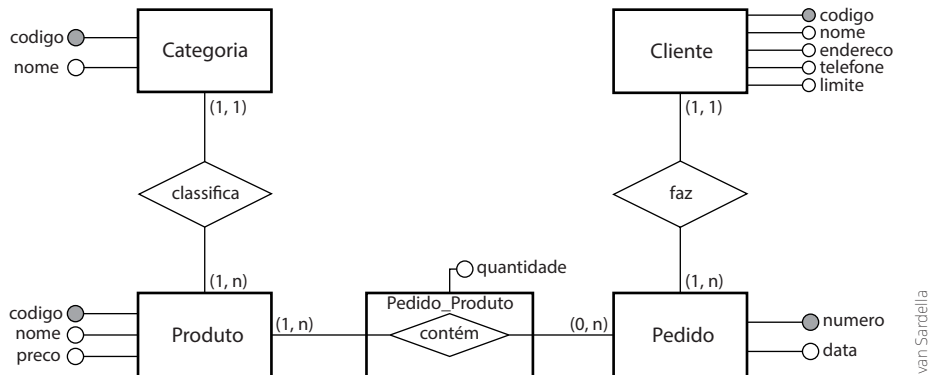
### Modelo lógico



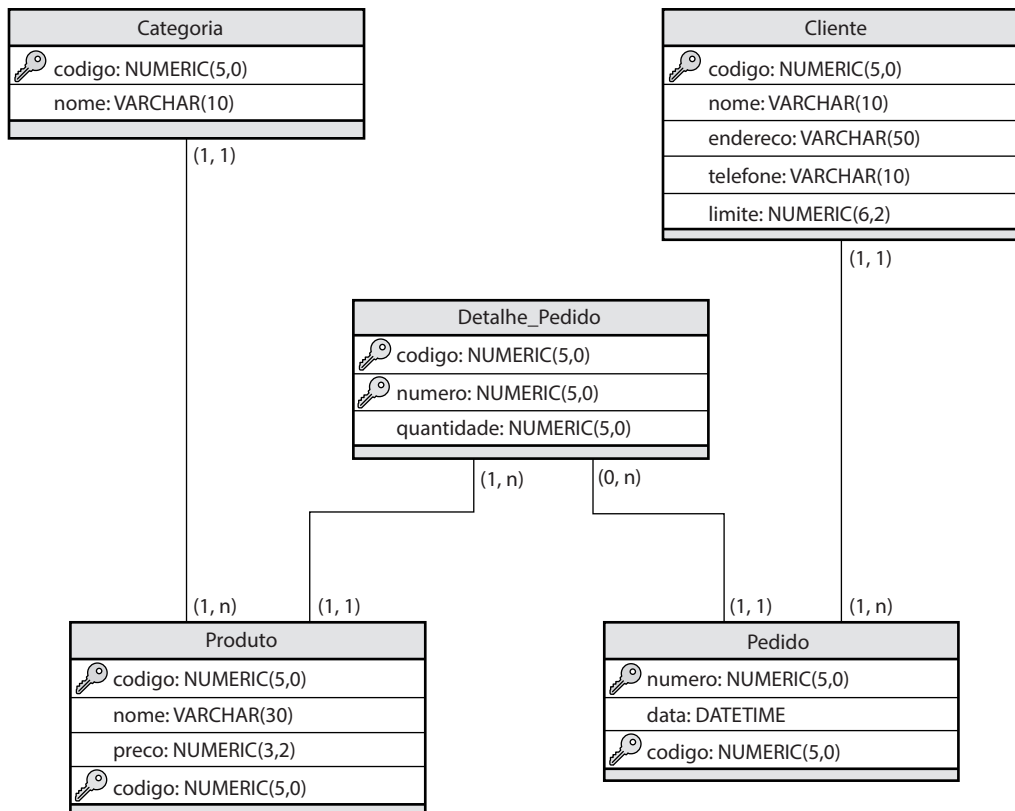
Ivan Sardella

## 2.

## Modelo conceitual

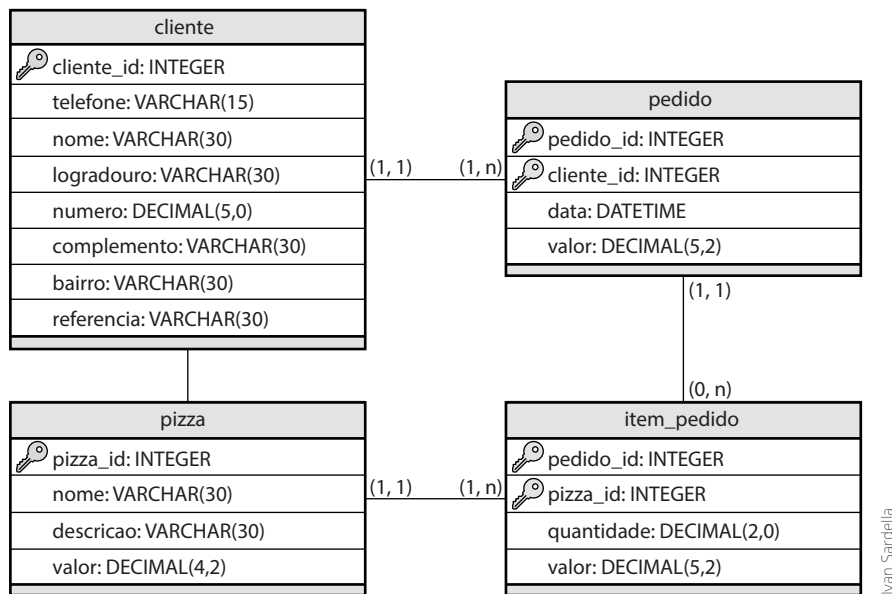


## Modelo lógico



## Capítulo 2 - Linguagem de Definição de Dados (DDL)

1.



```
-- Remover as tabelas, caso existam
DROP TABLE IF EXISTS item_pedido;
DROP TABLE IF EXISTS pedido;
DROP TABLE IF EXISTS cliente;
DROP TABLE IF EXISTS pizza;
```

-- Tabela CLIENTE

```
CREATE TABLE Cliente (
  cliente_id DECIMAL(5,0) AUTO_INCREMENT,
  telefone VARCHAR(10),
  nome VARCHAR(30),
  logradouro VARCHAR(30),
  numero DECIMAL(5,0),
  complemento VARCHAR(30),
  bairro VARCHAR(30),
  referencia VARCHAR(30),
  PRIMARY KEY (cliente_id)
);
```

**-- Tabela PIZZA**

```
CREATE TABLE pizza (  
    pizza_id INTEGER AUTO_INCREMENT,  
    nome VARCHAR(30),  
    descricao VARCHAR(30),  
    valor DECIMAL(4 , 2 ),  
    PRIMARY KEY (pizza_id)  
);
```

**-- Tabela PEDIDO**

```
CREATE TABLE pedido (  
    pedido_id INTEGER AUTO_INCREMENT,  
    cliente_id INTEGER,  
    data DATETIME,  
    valor DECIMAL(4 , 2 ),  
    PRIMARY KEY (pedido_id),  
    FOREIGN KEY (cliente_id)  
        REFERENCES Cliente (cliente_id)  
);
```

**-- Tabela ITEM\_PEDIDO**

```
CREATE TABLE item_pedido (  
    pedido_id INTEGER,  
    pizza_id INTEGER,  
    quantidade DECIMAL(2 , 0 ),  
    valor DECIMAL(5 , 2 ),  
    FOREIGN KEY (pizza_id)  
        REFERENCES Pizza (pizza_id),  
    FOREIGN KEY (pedido_id)  
        REFERENCES Pedido (pedido_id)  
);
```

## Capítulo 3 - Linguagem de Manipulação de Dados (DML)

1.

Tabela de clientes					
Nome	Telefone	Endereço	Número	Bairro	Referência
Bruna Dantas	(22) 2222-2222	Rua das Rosas	222	Cantareira	
Bruno Vieira	(33) 3333-3333	Rua das Avencas	333	Bela Vista	
Giulia Silva	(44) 4444-4444	Rua dos Cravos	444	Cantareira	Esquina do mercado
José Silva	(55) 5555-5555	Rua das Acácias	555	Bela Vista	
Laura Madureira	(66) 6666-6666	Rua das Gardêneas	666	Cantareira	

```
INSERT INTO 'cliente' VALUES (2, '(22) 2222-2222', 'Bruna Dantas',
    'Rua das Rosas', 222, NULL, 'Cantareira', NULL);
INSERT INTO 'cliente' VALUES (3, '(33) 3333-3333', 'Bruno Vieira',
    'Rua das Avencas', 333, NULL, 'Bela Vista', NULL);
INSERT INTO 'cliente' VALUES (4, '(44) 4444-4444', 'Giulia Silva',
    'Rua dos Cravos', 444, NULL, 'Cantareira', 'Esquina do mercado');
INSERT INTO 'cliente' VALUES (5, '(55) 5555-5555', 'José Silva',
    'Rua das Acácias', 555, NULL, 'Bela Vista', NULL);
INSERT INTO 'cliente' VALUES (6, '(66) 6666-6666', 'Laura Madureira',
    'Rua das Gardêneas', 666, NULL, 'Cantareira', NULL);
```

Tabela de pedidos			
Número	Cliente	Data e hora	Valor (R\$)
1	1	15/12/2016 20:30	32.00
2	2	15/12/2016 20:38	40.00
3	3	15/12/2016 20:59	22.00
4	1	17/12/2016 22:00	42.00
5	2	18/12/2016 19:00	45.00
6	3	18/12/2016 21:12	44.00
7	4	19/12/2016 22:22	72.00
8	6	19/12/2016 22:26	34.00

```
INSERT INTO 'pedido' VALUES (1, 1, '2016-12-15 20:30:00', 32.00);
INSERT INTO 'pedido' VALUES (2, 2, '2016-12-15 20:38:00', 40.00);
INSERT INTO 'pedido' VALUES (3, 3, '2016-12-15 20:59:00', 22.00);
INSERT INTO 'pedido' VALUES (4, 1, '2016-12-17 22:00:00', 42.00);
INSERT INTO 'pedido' VALUES (5, 2, '2016-12-18 19:00:00', 45.00);
INSERT INTO 'pedido' VALUES (6, 3, '2016-12-18 21:12:00', 44.00);
INSERT INTO 'pedido' VALUES (7, 4, '2016-12-19 22:22:00', 72.00);
INSERT INTO 'pedido' VALUES (8, 6, '2016-12-19 22:26:00', 34.00);
```

Tabela de itens de pedido			
Número	Pizza	Quantidade	Valor total
1	1	1	15.00
1	4	1	17.00
2	3	2	40.00
3	5	1	22.00
4	3	1	20.00
4	5	1	22.00
5	1	3	45.00
6	5	2	44.00
7	1	2	30.00
7	3	1	20.00
7	5	1	22.00
8	4	2	34.00

```

INSERT INTO 'item_pedido' VALUES (1, 1, 1, 15.00);
INSERT INTO 'item_pedido' VALUES (1, 4, 1, 17.00);
INSERT INTO 'item_pedido' VALUES (2, 3, 2, 40.00);
INSERT INTO 'item_pedido' VALUES (3, 5, 1, 22.00);
INSERT INTO 'item_pedido' VALUES (4, 3, 1, 20.00);
INSERT INTO 'item_pedido' VALUES (4, 5, 1, 22.00);
INSERT INTO 'item_pedido' VALUES (5, 1, 3, 45.00);
INSERT INTO 'item_pedido' VALUES (6, 5, 2, 44.00);
INSERT INTO 'item_pedido' VALUES (7, 1, 2, 30.00);
INSERT INTO 'item_pedido' VALUES (7, 3, 1, 20.00);
INSERT INTO 'item_pedido' VALUES (7, 5, 1, 22.00);
INSERT INTO 'item_pedido' VALUES (8, 4, 2, 34.00);

```

## Capítulo 4 - Funções

1.

- a) `SELECT MIN(data) FROM pedido;`
- b) `SELECT MAX(data) FROM pedido;`
- c) `SELECT MAX(valor) FROM pedido;`
- d) `SELECT MIN(valor) FROM pedido;`



- e) `SELECT MIN(nome) FROM cliente;`
- f) `SELECT MAX(nome) FROM cliente;`
- g) `SELECT UPPER(nome) FROM cliente;`
- h) `SELECT SUM(valor) FROM pedido WHERE DATE(data) = '2016-12-18';`
- i) `SELECT DATEDIFF('2016-12-19', '2016-12-15') AS "Dias";`  
Mas podemos utilizar sub consultas:  
`SELECT DATEDIFF(  
    (SELECT MAX(data) FROM PEDIDO),  
    (SELECT MIN(data) FROM PEDIDO)  
) AS "Dias";`

## Capítulo 5 - Consultas com informações agrupadas

1.

- a) `SELECT  
    pizza_id AS "Código da Pizza",  
    SUM(quantidade) AS "Quantidade"  
    FROM item_pedido  
    GROUP BY pizza_id  
    ORDER BY "Quantidade" desc;`
- b) `SELECT  
    cliente_id AS "Código do Cliente",  
    SUM(quantidade) AS "Total"  
    FROM pedido  
    GROUP BY cliente_id  
    ORDER BY "Total" desc;`

## Capítulo 6 - Junção de tabelas

- 1. `select p.pedido_id as "Número", date(p.data) as "Data",  
    c.nome as "Cliente"  
    from pedido p inner join cliente c  
    on p.cliente_id = c.cliente_id  
    order by p.pedido_id;`

## Capítulo 7 - Índices e visões

1. `create unique index idx_telefone on cliente(telefone);`
2. `create view abc_de_venda  
as  
select p.nome as "Pizza", sum(i.quantidade) as "Qtde"  
from item_pedido i inner join pizza p  
on i.pizza_id = p.pizza_id  
group by pizza  
order by qtde desc;`

## Capítulo 8 - Funções, procedimentos e gatilhos

1. `delimiter //  
create function ultimoPedido(cod_cliente integer) returns integer  
begin  
return  
(select pedido_id from pedido where data =  
(select max(data) from pedido where cliente_id = cod_cliente)  
);  
end //  
delimiter ;`
2. `delimiter //  
create procedure pizzasDoUltimoPedido(in cod_cliente integer)  
begin  
select nome from pizza p inner join item_pedido i  
on p.pizza_id = i.pizza_id  
where pedido_id = ultimoPedido(cod_cliente);  
end //  
delimiter ;`