

Lanchonete BD

Claudio Magno
Diogo Tomas





Motivação

- Buscar fonte de renda durante o período de pandemia de maneira segura.
- Ser destaque no quesito de organização do trabalho por entender-se como um diferencial.
- Promover um destaque inicial que desperte a curiosidade dos clientes.
- Implementar um sistema de qualidade e confiança buscando a continuidade do negócio após a pandemia.

Protótipo e tabela de dados



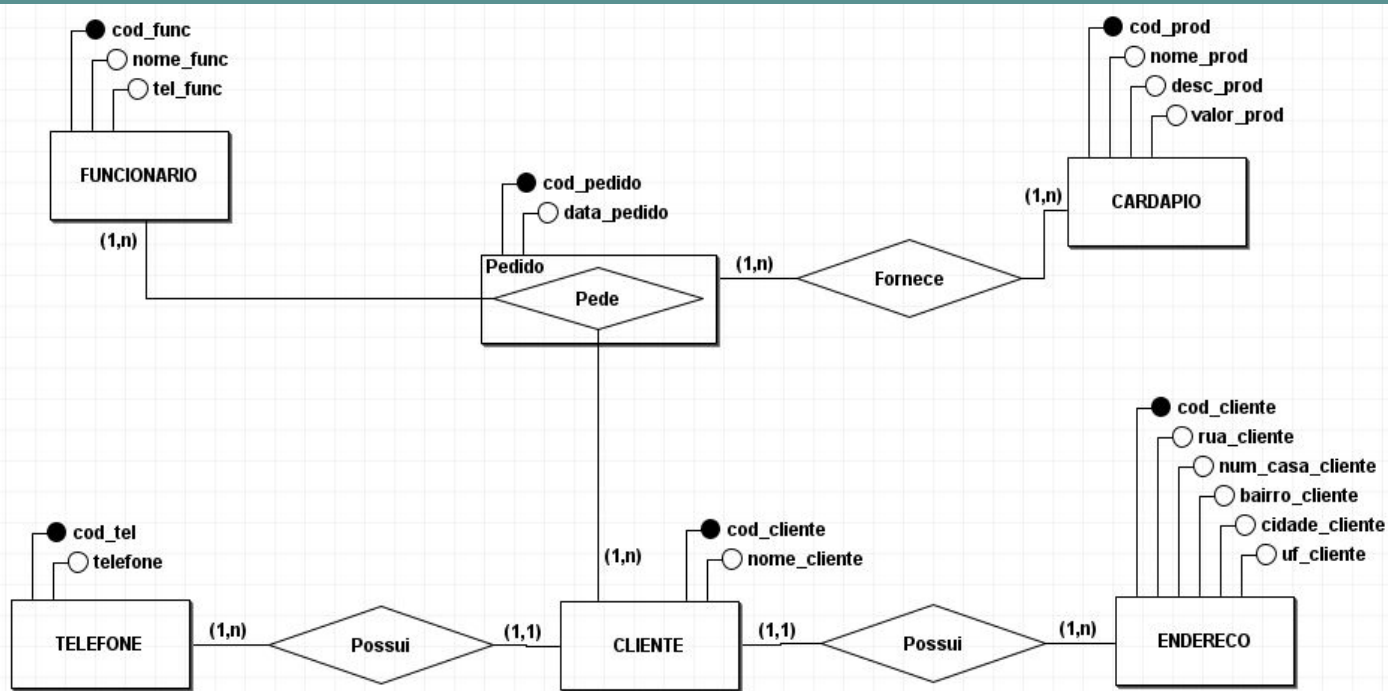
FUNCIONARIO			PEDIDO		
COD_FUNC	NOME_FUNC	TEL_FUNC	COD_PEDIDO	DATA_PEDIDO	COD_CLIENTE_PEDIDO
1	Jorge	(27) 99958-8954	1	12/02/2021 19:44	1
2	Luiz	(27) 99528-8986	2	12/02/2021 19:55	3
3	Henrique	(27) 99452-8524	3	12/02/2021 20:12	5
4	Julia	(27) 94521-5558	4	12/02/2021 20:33	4
5	Isadora	(27) 99784-7849	5	12/02/2021 20:35	2

TELEFONE		CARDAPIO_PEDIDO		CLIENTE	
CODIGO	TELEFONE	COD_ITEM	QUANT_PEDIDO	COD_CLIENTE	NOME_CLIENTE
1	(27) 99912-5687	1	2	1	Claudio
2	(27) 99562-8745	2	1	2	Isabelly
3	(27) 94588-8895	3	2	3	Gabriel
4	(27) 99945-4568	4	2	4	Julio
5	(27) 99985-9985	5	1	5	Augusto

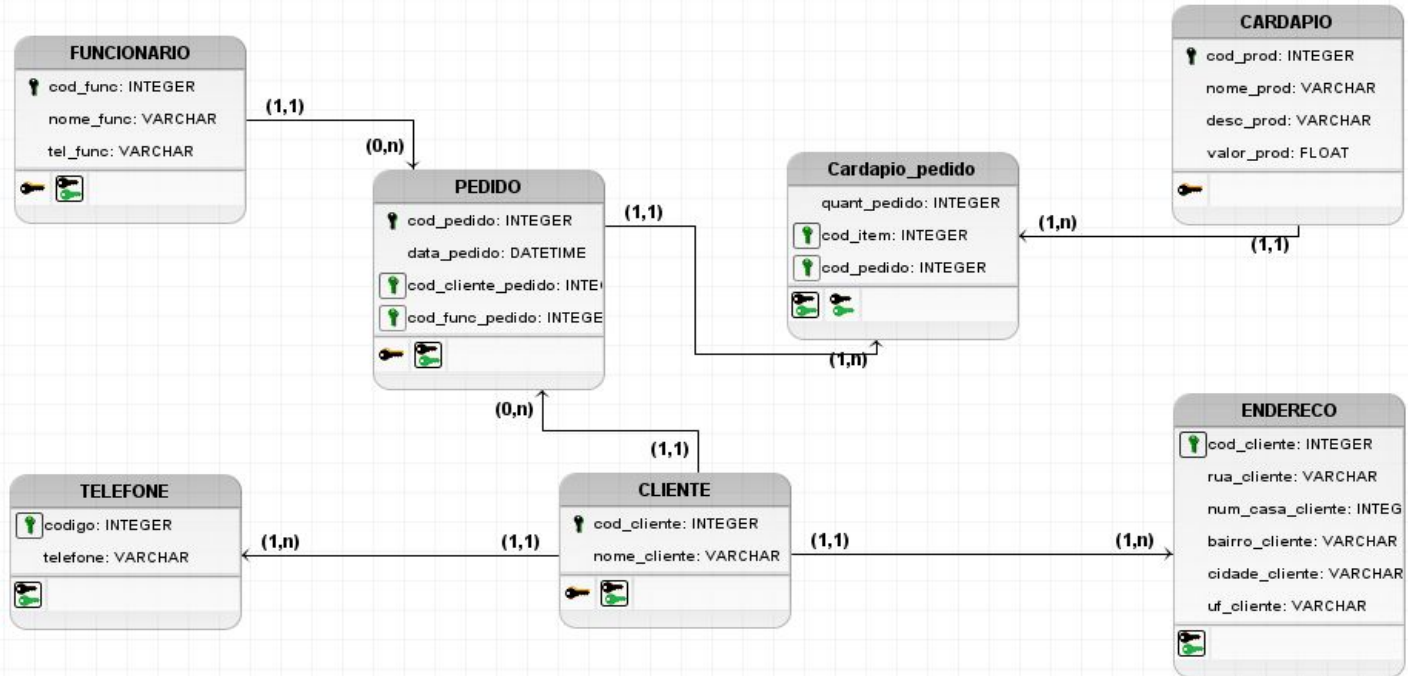
ENDERECO					
COD_CLIENTE	RUA_CLIENTE	NUM_CASA_CLIENTE	BAIRRO_CLIENTE	CIDADE_CLIENTE	UF_CLIENTE
1	Mossoró	12	Barcelona	Serra	ES
2	Andorinhas	597	Jacaraípe	Serra	ES
3	Cascardo	608	Barcelona	Serra	ES
4	João JJ	154	Taquara	Serra	ES
5	Senhorio	77	Laranjeiras	Serra	ES

CARDAPIO			
COD_PROD	NOME_PROD	DESC_PROD	VALOR_PROD
1	X-Bacon	x carne, bacon, alface e	R\$ 14,00
2	X-Tudo	, bacon, ovo, milho, alfac	R\$ 15,00
3	X-Egg	ie, 2x ovo, bacon, alface	R\$ 14,00
4	X-Egg Bacon	me, 2x ovo, bacon, alfac	R\$ 18,00
5	Misto	Pão, presunto e queijo.	R\$ 12,00

Modelo conceitual



Modelo lógico





Inserts

```
insert into cardapio values ((1,'X-Bacon', 'Pão, 2x carne, bacon, alface e salada.', 14.00), (2,'X-Tudo','Pão, carne, bacon, ovo, milho, alface e salada.', 15.00), (3,'X-Egg','Pão, carne, 2x ovo, bacon, alface e salada.', 14.00), (4,'X-Egg Bacon', 'Pão, 2x carne, 2x ovo, bacon, alface e salada.', 16.00), (5,'Misto', 'Pão, presuto e queijo.', 12.00));
```

```
insert into cliente values ((1, 'Claudio'), (2, 'Isabelly'), (3, 'Gabriel'), (4, 'Julio'), (5, 'Augusto'));
```

```
insert into endereco values ((1, 'Mossoró', 590, 'Barcelona', 'Serra', 'ES'),(3, 'Andorinhas', 12, 'Jacaraípe', 'Serra', 'ES'), (2, 'Cascardo', 175, 'Barcelona', 'Serra', 'ES'), (5, 'Alfredo Cunha', 154, 'Taquara', 'Serra', 'ES'), (4, 'Senhorio', 77, 'Barcelona', 'Serra', 'ES'));
```

```
insert into telefone values ((1, '(27) 99645-1234'), (3, '(27) 94521-8524'), (2, '(27) 99998-8524'), (4, '(27) 999645-12345'), (5, '(27) 99698-4785'));
```

```
insert into funcionario values ((1, 'Jorge', '(27) 99958-8954'), (2, 'Luiz', '(27) 99528-8966'), (3, 'Henrique', '(27) 99452-8524'), (4, 'Julia', '(27) 94521-5558'), (5, 'Isadora', '(27) 99784-7849'));
```

```
insert into pedido values ((1, '2020-02-12 19:44:21', 2, 1), (2, '2020-02-12 19:55:01', 1, 3), (3, '2020-02-12 20:12:36', 3, 3),(4, '2020-02-12 20:33:55', 4, 5), (5, '2020-02-12 20:55:12', 5, 4));
```

```
insert into cardapio_pedido values ((5, 1, 2), (4, 5, 1), (2, 3, 1), (3, 2, 3), (2, 4, 1));
```

Consultas com Inner Join e Order By

```
select cliente.nome_cliente as nome,
telefone.telefone as celular, endereco.bairro_cliente
as bairro, pedido.data_pedido as data,
cardapio_pedido.quant_pedido as quantidade,
cardapio.nome_prod as pedido,
funcionario.nome_func as atendente from cliente
inner join telefone on (cliente.cod_cliente =
telefone.codigo) inner join endereco on
(cliente.cod_cliente = endereco.cod_cliente) inner join
pedido on (cliente.cod_cliente =
pedido.cod_cliente_pedido) inner join
cardapio_pedido on (pedido.cod_pedido =
cardapio_pedido.cod_pedido) inner join cardapio on
(cardapio_pedido.cod_item = cardapio.cod_prod)
inner join funcionario on (pedido.cod_func_pedido =
funcionario.cod_func) order by data
```

```
[202] res = pd.read_sql_query('' select cliente.nome_cliente as nome, telefone.telefone as celular,
endereco.bairro_cliente as bairro, pedido.data_pedido as data,
cardapio_pedido.quant_pedido as quantidade, cardapio.nome_prod as pedido,
funcionario.nome_func as atendente from cliente inner join telefone
on (cliente.cod_cliente = telefone.codigo) inner join endereco on
(cliente.cod_cliente = endereco.cod_cliente) inner join pedido on
(cliente.cod_cliente = pedido.cod_cliente_pedido) inner join
cardapio_pedido on (pedido.cod_pedido = cardapio_pedido.cod_pedido)
inner join cardapio on (cardapio_pedido.cod_item = cardapio.cod_prod)
inner join funcionario on (pedido.cod_func_pedido = funcionario.cod_func) order by data '',conn)
res
```

	nome	celular	bairro	data	quantidade	pedido	atendente
0	Gabriel	(27) 94521-8524	Jacaraípe	20:12:36	1	X-Tudo	Henrique
1	Augusto	(27) 99698-4785	Taquara	20:55:12	1	X-Egg Bacon	Julia

Consultas com Inner Join e Order By

```
select pedido.data_pedido as  
pedidos_atrasados,  
cliente.nome_cliente,  
cardapio_pedido.quant_pedido  
from cliente inner join pedido on  
(cliente.cod_cliente =  
pedido.cod_cliente_pedido) inner  
join cardapio_pedido on  
(pedido.cod_pedido =  
cardapio_pedido.cod_pedido)  
order by pedidos_atrasados desc
```

```
[205] res = pd.read_sql_query('''      select pedido.data_pedido as pedidos_atrasados,  
      cliente.nome_cliente, cardapio_pedido.quant_pedido from cliente inner join pedido  
      on (cliente.cod_cliente = pedido.cod_cliente_pedido) inner join cardapio_pedido on  
      (pedido.cod_pedido = cardapio_pedido.cod_pedido) order by pedidos_atrasados desc  
      ''',conn)  
res
```

	pedidos_atrasados	nome_cliente	quant_pedido
0	20:55:12	Augusto	1
1	20:33:55	Julio	1
2	20:12:36	Gabriel	1
3	19:44:21	Isabelly	2

Consultas com Inner Join e Order By

```
select cardapio.nome_prod as  
pedido,  
cardapio_pedido.quant_pedido as  
quantidade from cardapio inner  
join cardapio_pedido on  
(cardapio.cod_prod =  
cardapio_pedido.cod_item) order  
by quantidade
```

```
[207] res = pd.read_sql_query(''' select cardapio.nome_prod as pedido,  
cardapio_pedido.quant_pedido as quantidade from cardapio inner join  
cardapio_pedido on (cardapio.cod_prod = cardapio_pedido.cod_item) order by quantidade  
''',conn)  
res
```

	pedido	quantidade
0	X-Egg Bacon	1
1	X-Tudo	1
2	X-Tudo	1
3	Misto	2

Consultas com Group By e Funções de Agrupamento

```
select cliente.nome_cliente as  
nome,  
count(pedido.cod_cliente_pedido)  
from cliente inner join pedido on  
(cliente.cod_cliente =  
pedido.cod_cliente_pedido) group  
by nome
```

```
[208] res = pd.read_sql_query(''' select cliente.nome_cliente as nome,  
count(pedido.cod_cliente_pedido) from cliente inner join pedido on  
(cliente.cod_cliente = pedido.cod_cliente_pedido) group by nome  
''',conn)  
res
```

	nome	count
0	Julio	1
1	Claudio Magno	1
2	Gabriel	1
3	Isabelly	1
4	Augusto	1

Consultas com Group By e Funções de Agrupamento

```
select cardapio_pedido.cod_item  
as codigo_lanche,  
sum(cardapio_pedido.quant_pedi  
do) as quantidade_total from  
cardapio_pedido group by  
codigo_lanche
```

```
[210] res = pd.read_sql_query(''' select cardapio_pedido.cod_item as codigo_lanche,  
sum(cardapio_pedido.quant_pedido) as quantidade_total from cardapio_pedido group by codigo_lanche  
''', conn)  
res
```

	codigo_lanche	quantidade_total
0	5	2
1	4	1
2	2	2

Consultas com Group By e Funções de Agrupamento

```
select cardapio.nome_prod,  
avg(cardapio.valor_prod) from  
cardapio where  
(cardapio.valor_prod > 14) group  
by cardapio.nome_prod
```

```
[212] res = pd.read_sql_query(''' select cardapio.nome_prod, avg(cardapio.valor_prod)  
      from cardapio where (cardapio.valor_prod > 14) group by cardapio.nome_prod  
      ''', conn)  
res
```

	nome_prod	avg
0	X-Tudo	15.0
1	X-Egg Bacon	15.0

Consultas que utilizem Left ou Right ou Self ou Full Join

```
select endereco.cod_cliente,  
endereco.rua_cliente,  
cliente.cod_cliente  
from endereco  
left outer join cliente  
on (cliente.cod_cliente =  
endereco.cod_cliente)  
order by cliente.cod_cliente;
```

```
res = pd.read_sql_query(''' select endereco.cod_cliente,  
endereco.rua_cliente, cliente.cod_cliente  
from endereco  
left outer join cliente  
on (cliente.cod_cliente = endereco.cod_cliente)  
order by cliente.cod_cliente ''',conn)  
res
```

	cod_cliente	rua_cliente	cod_cliente
0	3	Andorinhas	3
1	5	Alfredo Cunha	5

Consultas que utilizem View com Subconsultas

```
create view ordens as  
select cod_pedido as cod,  
data_pedido as hora,  
cod_cliente_pedido as clienteN,  
cod_func_pedido as funcN  
from pedido  
select * from ordens;
```

```
[222] res = pd.read_sql_query(''' create view ordens as  
    select cod_pedido as cod,  
    data_pedido as hora,  
    cod_cliente_pedido as clienteN,  
    cod_func_pedido as funcN  
    from pedido;  
    select * from ordens; ''',conn)  
res
```

	cod	hora	clienten	funcn
0	1	19:44:21	2	1
1	2	19:55:01	1	3
2	3	20:12:36	3	3
3	4	20:33:55	4	5
4	5	20:55:12	5	4

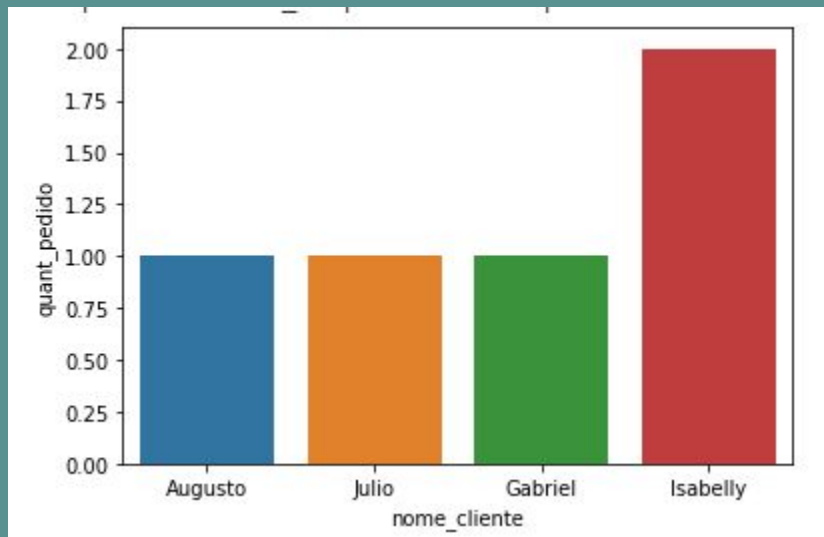


Perguntas a serem respondidas

- Relatório trazendo os pedidos feitos pelos clientes.
- Desempenho dos funcionários.
- Relatório referente a organização de entrega dos pedidos
- Variação dos pedidos com relação ao preço dos produtos.

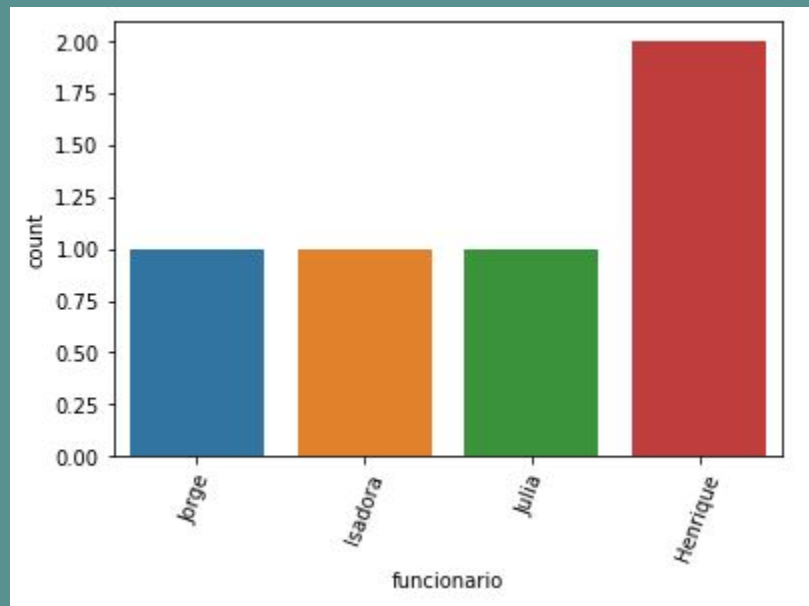
Relatórios

Obter relatório que mostre os pedidos realizados por cada cliente com a hora e a quantidade.



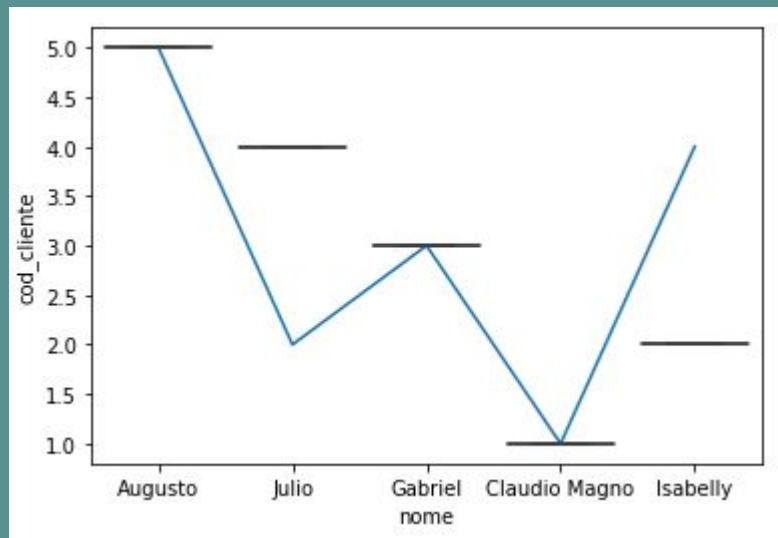
Relatório

Relatório referente a participação dos funcionários trazendo a quantidade de pedidos atendidos por cada.



Relatório

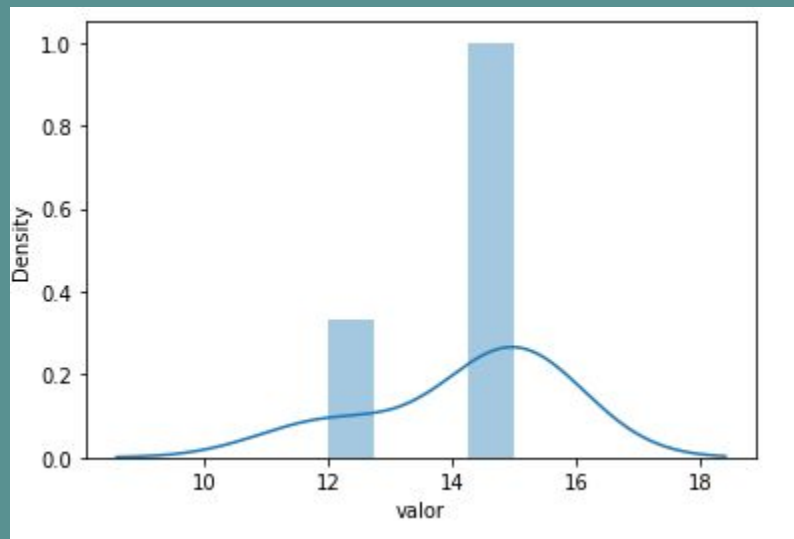
Relatório referente a hora em que os pedidos foram realizados, informações como nome do cliente e código do pedido e do funcionário atendente com o gráfico demonstrando a ordem em que os pedidos foram realizados ordenados pelos clientes mais recentes cadastrados.





Relatório

Obter Relatório que mostre todos os produtos vendidos pela lanchonete trazendo as informações de valores e quantidades vendidos por cada.



Relatório

Relatório demonstrando a ordem dos pedidos realizados e o valor total de cada pedido feito demonstrando o nome do cliente e código do pedido.

