

PediidON

BD1 | Eduarda e Thiago

Window Name

Central de Pedidos

1

2

3

4

Mesas

Cliente: Daenerys

CPF: 11111111111

Código	Quantidade	Produto	Observação	Horário do Pedido
0001	1	Reta da Penha		21:52
0002	1	Parque Moscoso	Sem legumes	21:52
0003	1	Ilha de		21:52
0004	1	Barlo		21:52

Solicitação de Alteração

⌚ 22:00

Colocar os legumes

Aceitar

Negar

Versão funcionário

09:52 AM

Comanda

Mesa 1

Reio da Penha	x1
Porque Morcego	x1
Ihe de Saco	x1
Borlante	x1

Alto o preço

Encerar

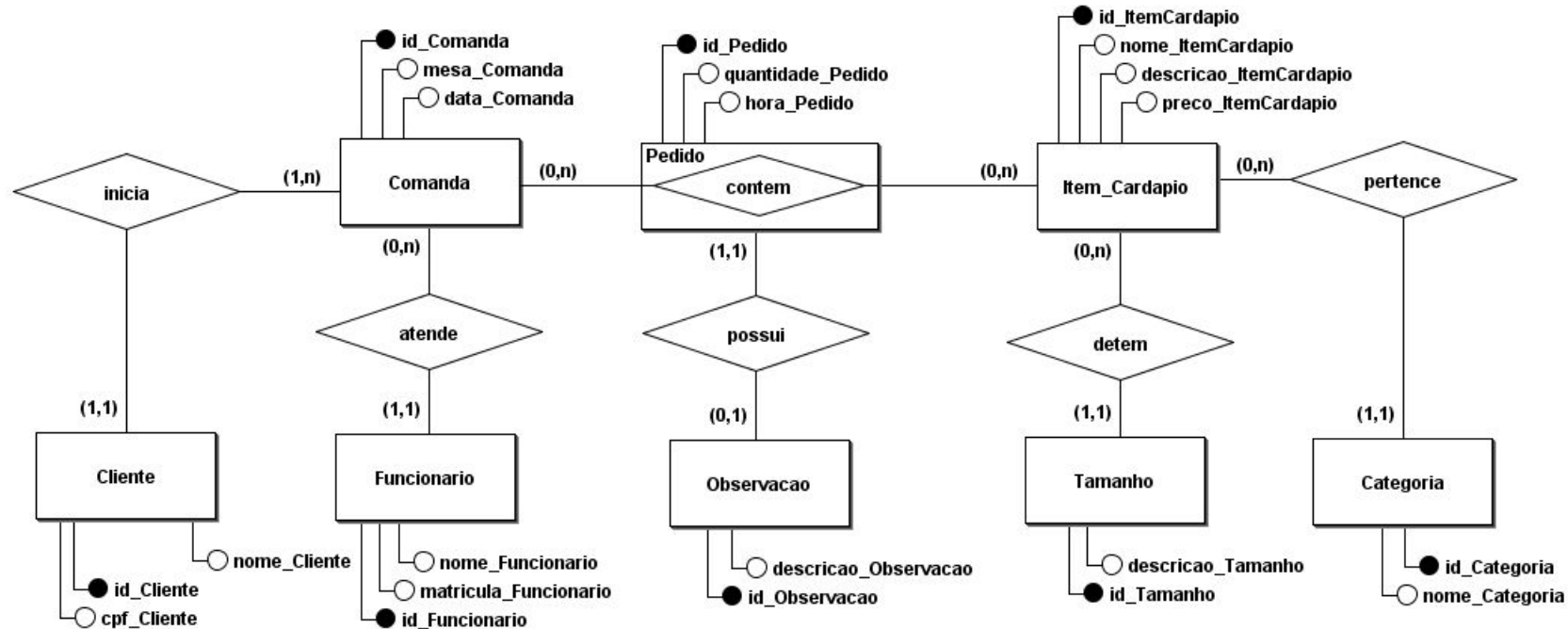
Cardápio

Versão cliente

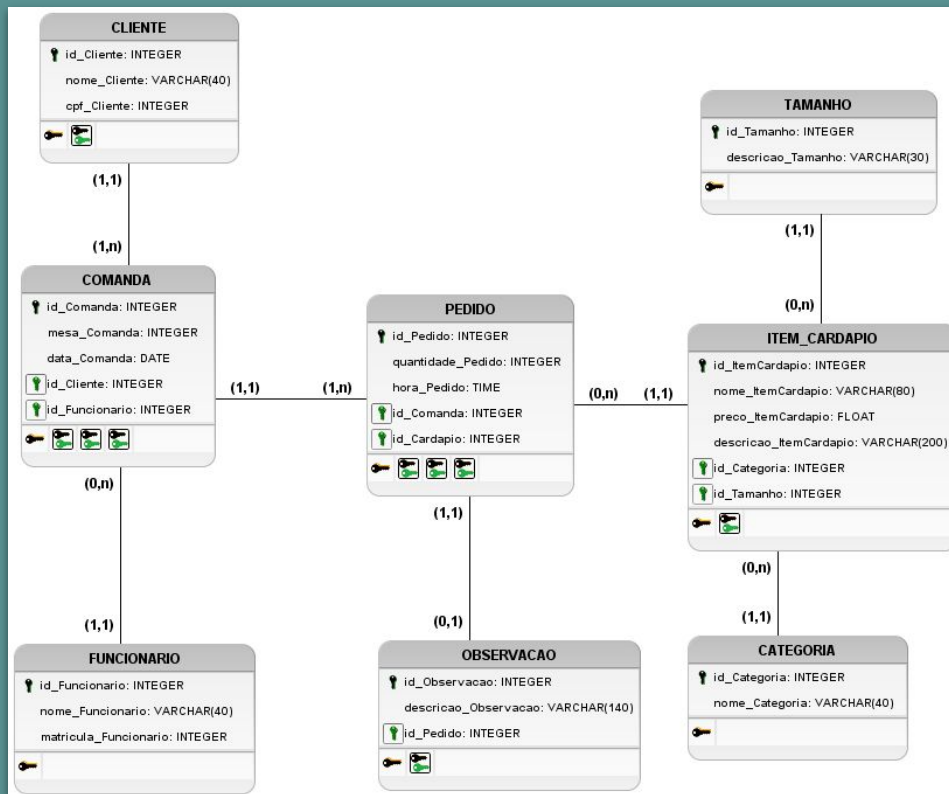
[illegible]

Tabela de dados

MODELO CONCEITUAL



MODELO LÓGICO





MODELO FÍSICO

```
CREATE TABLE CLIENTE (  
  id_Cliente SERIAL PRIMARY KEY,  
  nome_Cliente VARCHAR(40),  
  cpf_Cliente INTEGER);
```

```
CREATE TABLE FUNCIONARIO (  
  id_Funcionario SERIAL PRIMARY KEY,  
  nome_Funcionario VARCHAR(40),  
  matricula_Funcionario INTEGER);
```

```
CREATE TABLE COMANDA (  
  id_Comanda SERIAL PRIMARY KEY,  
  mesa INTEGER,  
  data DATE,  
  fk_id_Cliente INTEGER,  
  fk_id_Funcionario INTEGER,  
  FOREIGN KEY (fk_id_Cliente)  
  REFERENCES CLIENTE(id_Cliente),  
  FOREIGN KEY (fk_id_Funcionario)  
  REFERENCES  
  FUNCIONARIO(id_Funcionario));
```

```
CREATE TABLE TAMANHO (  
  id_Tamanho SERIAL PRIMARY KEY,  
  descricao_tamanho VARCHAR(30));
```

```
CREATE TABLE CATEGORIA (  
  id_Categoria SERIAL PRIMARY KEY,  
  nome_Categoria VARCHAR(40));
```

```
CREATE TABLE PEDIDO (  
  id_Pedido SERIAL PRIMARY KEY,  
  quantidade INTEGER,  
  hora TIME,  
  fk_id_Comanda INTEGER,  
  fk_id_ItemCardapio INTEGER,  
  FOREIGN KEY (fk_id_Comanda)  
  REFERENCES  
  COMANDA(id_Comanda));
```

```
CREATE TABLE OBSERVACAO (  
  id_Obs SERIAL PRIMARY KEY,  
  descricao_obs VARCHAR(140),  
  fk_id_Pedido INTEGER,  
  FOREIGN KEY (fk_id_Pedido)  
  REFERENCES PEDIDO(id_Pedido));
```

```
CREATE TABLE ITEM_CARDAPIO (  
  id_ItemCardapio SERIAL PRIMARY KEY,  
  nome_Item VARCHAR(80),  
  preco FLOAT,  
  descricao_item VARCHAR(200),  
  fk_id_Categoria INTEGER,  
  fk_id_Tamanho INTEGER,  
  FOREIGN KEY (fk_id_Categoria)  
  REFERENCES  
  CATEGORIA(id_Categoria),  
  FOREIGN KEY (fk_id_Tamanho)  
  REFERENCES  
  TAMANHO(id_Tamanho));
```

```
ALTER TABLE PEDIDO ADD  
  FOREIGN KEY (fk_id_ItemCardapio)  
  REFERENCES  
  ITEM_CARDAPIO(id_ItemCardapio);
```



INSERT: Exemplos

```
insert into cliente(nome_cliente,  
cpf_cliente) values('Lara', 10582),  
('Jorge', 11736), ('Clara', 12890), ('Davi',  
14045), ('Milton', 15199), ('Luiza', 18662),  
('Ana', 16354), ('Tais', 17508), ('Isabella',  
19817), ('Henrique', 22010);
```

```
insert into funcionario(nome_funcionario,  
matricula_funcionario) values('Bernardo',  
2210921), ('Matheus', 3874638), ('Ismael',  
4733307);
```

```
insert into comanda(mesa, data,  
fk_id_cliente, fk_id_funcionario)  
values(13, '2020-07-30', 1, 1), (1,  
'2020-07-30', 2, 1), (4, '2020-07-30', 3, 2),  
(12, '2020-07-30', 4, 3), (3, '2020-07-30',  
5, 2), (1, '2020-07-31', 2, 3), (13,  
'2020-07-31', 6, 1), (2, '2020-07-31', 7, 2),  
(12, '2020-07-31', 8, 3), (4, '2020-08-01',  
4, 1), (13, '2020-08-01', 2, 2), (12,  
'2020-08-01', 9, 3), (13, '2020-08-01', 10,  
2);
```

```
insert into tamanho(descricao) values('2  
pessoas'), ('1 pessoa'), ('infantil'), ('long  
neck'), ('lata');
```

```
insert into categoria(nome_categoria)  
values('entrada'), ('drink'), ('principal'),  
('sobremesa');
```

```
insert into item_cardapio(nome_item,  
descricao, preco, fk_id_categoria,  
fk_id_tamanho) values('Reta da Penha',  
'Filé Mignon marinado na mostarda  
antiga, servido com pães da casa.', 56.90,  
1, 2), ('Rua Sete', 'Filé Mignon ao  
conhaque, servido com pães da casa', 55,  
1, 2), ('Parque Moscoso', 'Polvo à galega -  
Polvo cozido servido com sal e azeite,  
acompanhado de batatas e legumes  
tostados', 145, 3, 1), (('Barlavento', 'Gin,  
xarope de morango, maracujá, suco de  
laranja, espuma de gengibre e angostura  
bitter', 25.9, 2, 2);
```

```
insert into pedido(quantidade, hora, fk_id_comanda, fk  
_id_itemcardapio) values(2, '12:58:00', 1,  
1), (1, '13:19:00', 1, 2), (1, '12:34:00', 2,  
3), (3, '12:45:00', 3, 4), (2, '12:21:00', 7,  
2), (3, '13:15:00', 8, 3), (1, '12:57:00', 6,  
1), (3, '13:38:00', 8, 4), (1, '12:43:00', 12,  
1), (2, '13:09:00', 12, 2), (3, '13:38:00', 13,  
3), (4, '13:51:00', 13, 4);
```

```
insert into observacao(descricao,  
fk_id_pedido) values('Sem sal', 1), ('Um  
prato e talher extra', 4), ('Sem nozes', 5),
```

CONSULTA COM INNER JOIN E ORDER BY

```
select nome_funcionario, id_comanda, mesa  
from comanda  
inner join funcionario on  
comanda.fk_id_funcionario = funcionario.id_funcionario  
order by nome_funcionario;
```

	nome_funcionario	id_comanda	mesa
0	Bernardo	1	13
1	Bernardo	2	1
2	Bernardo	7	13
3	Bernardo	10	4
4	Ismael	4	12
5	Ismael	6	1
6	Ismael	12	12
7	Ismael	9	12
8	Matheus	5	3
9	Matheus	13	13
10	Matheus	11	13
11	Matheus	3	4
12	Matheus	8	2

CONSULTA COM INNER JOIN E ORDER BY

```
select nome_cliente, id_comanda, mesa  
from comanda  
inner join cliente on  
comanda.fk_id_cliente = cliente.id_cliente;
```

	nome_cliente	id_comanda	mesa
0	Lara	1	13
1	Jorge	2	1
2	Clara	3	4
3	Davi	4	12
4	Milton	5	3
5	Jorge	6	1
6	Luiza	7	13
7	Ana	8	2
8	Tais	9	12
9	Davi	10	4
10	Jorge	11	13
11	Isabella	12	12
12	Henrique	13	13

CONSULTA COM INNER JOIN E ORDER BY

```
select nome_cliente, nome_funcionario  
from cliente  
inner join comanda on  
cliente.id_cliente = comanda.fk_id_cliente  
inner join funcionario on  
funcionario.id_funcionario =  
comanda.fk_id_funcionario  
order by nome_cliente;
```

	nome_cliente	nome_funcionario
0	Ana	Matheus
1	Ana	Ismael
2	Ana	Adrian
3	Antonella	Maria
4	Antonella	Juliana
5	Antonella	Martin
6	Bento	Martin
7	Bento	Maria
8	Bento	Matheus
9	Clara	Juliana
10	Clara	Matheus
11	Clara	Maria
12	Clarice	Fernanda

CONSULTA COM GROUP BY E FUNÇÕES DE AGRUPAMENTO

```
select nome_funcionario,  
count(nome_funcionario) as  
comandas_atendidas  
from comanda as c  
inner join pedido as p on  
(c.id_comanda = p.fk_id_comanda)  
inner join funcionario as f on  
(c.fk_id_funcionario = f.id_funcionario)  
group by nome_funcionario;
```

	nome_funcionario	comandas_atendidas
0	Marcos	1
1	Adrian	3
2	Fernanda	3
3	Lucio	1
4	Aline	3
5	Rivaldo	1
6	Eloisa	3
7	Gabriel	3
8	Maria	3
9	Ronaldo	1
10	Martin	3
11	Roberto Carlos	1
12	Juliana	3
13	Ana Clara	3

CONSULTA COM GROUP BY E FUNÇÕES DE AGRUPAMENTO

```
select id_comanda, min(preco) as  
menor_valor_pedido  
from comanda as c  
inner join pedido as p on  
(c.id_comanda = p.fk_id_comanda)  
inner join item_cardapio as ic on  
(p.fk_id_itemcardapio = ic.id_itemcardapio)  
group by id_comanda  
order by id_comanda;
```

	id_comanda	menor_valor_pedido
0	1	55.00
1	2	145.00
2	3	22.00
3	4	5.99
4	5	6.90
5	6	29.90
6	7	55.00
7	8	115.00
8	9	22.00
9	10	5.99
10	11	6.90
11	12	55.00
12	13	22.00
13	14	17.99

CONSULTA COM GROUP BY E FUNÇÕES DE AGRUPAMENTO

```
select id_comanda, sum(preco) as  
valor_total  
from comanda as c  
inner join pedido as p on  
(c.id_comanda = p.fk_id_comanda)  
inner join item_cardapio as ic on  
(p.fk_id_itemcardapio = ic.id_itemcardapio)  
group by id_comanda  
order by valor_total desc;
```

	id_comanda	valor_total
0	21	1890.00
1	22	559.00
2	39	529.00
3	45	359.00
4	13	307.90
5	70	299.00
6	20	299.00
7	26	298.99
8	8	260.00
9	40	249.00
10	27	217.99
11	28	177.99
12	3	162.90
13	2	145.00

CONSULTA COM LEFT JOIN

```
select id_pedido, descricao_item,  
descricao_obs  
from pedido as p  
inner join item_cardapio as ic on  
(p.fk_id_itemcardapio = ic.id_itemcardapio)  
left join observacao as o on  
(o.fk_id_pedido = p.id_pedido)  
limit 23;
```

	id_pedido	descricao_item	descricao_obs
0	1	Filé Mignon marinado na mostarda antiga, servi...	Sem sal
1	4	Peixe do dia servido inteiro e assado com limã...	Um prato e talher extra
2	5	Chocolate em mousse fria e semi congelada com ...	Sem nozes
3	7	Gin, xarope de morango, maracujá, suco de lara...	Sem espuma
4	9	Refrigerante	Dois copos com gelo e limão
5	12	Filé Mignon ao conhaque, servido com pães da casa	Sem os pães da casa
6	10	Whiskey Jack Daniels, maracujá, mix de cítrico...	Sem maracujá
7	8	Stella Artois Long Neck	Servir no copo
8	11	Filé Mignon marinado na mostarda antiga, servi...	Adicionar pimenta
9	14	Peixe do dia servido inteiro e assado com limã...	Porção extra de arroz
10	27	Massa ao molho de queijo e iscas de filé migno...	Só filé mignon
11	30	Princesa Isabel	Copo com gelo
12	35	Uva: Pinot Grigio. País: Itália	Azeltonas no copo
13	40	Entrecôte, salsa chermoula e legumes tostados	Legumes no ponto
14	41	Acerola, Laranja, Maracujá ou Limonada	Limonada
15	50	Uva: Sauvignon Blanc. País: Chile	Abrir na minha frete
16	54	com água de coco separado	Água de coco junto
17	55	Indica	Coloca limão
18	66	Ostras frescas de Santa Catarina	Ostras de Vitória mesmo. De Santa Catarina vai...
19	72	com 03 águas de coco	2 águas de coco
20	20	Filé Mignon marinado na mostarda antiga, servi...	None
21	25	Gin, redução de flor de sabugueiro, maracujá, ...	None
22	26	Ostras frescas de Santa Catarina	None

VIEW COM SUBCONSULTA

create or replace view

itens_pedidos_que_possuem_observacao as

select nome_item **as** itens_pedidos

from pedido **as** p

inner join item_cardapio **as** ic **on**

(p.fk_id_itemcardapio = ic.id_itemcardapio)

where id_pedido **in** (**select** fk_id_pedido

from observacao)

group by nome_item;

select * from itens_pedidos_que

possuem_observacao;

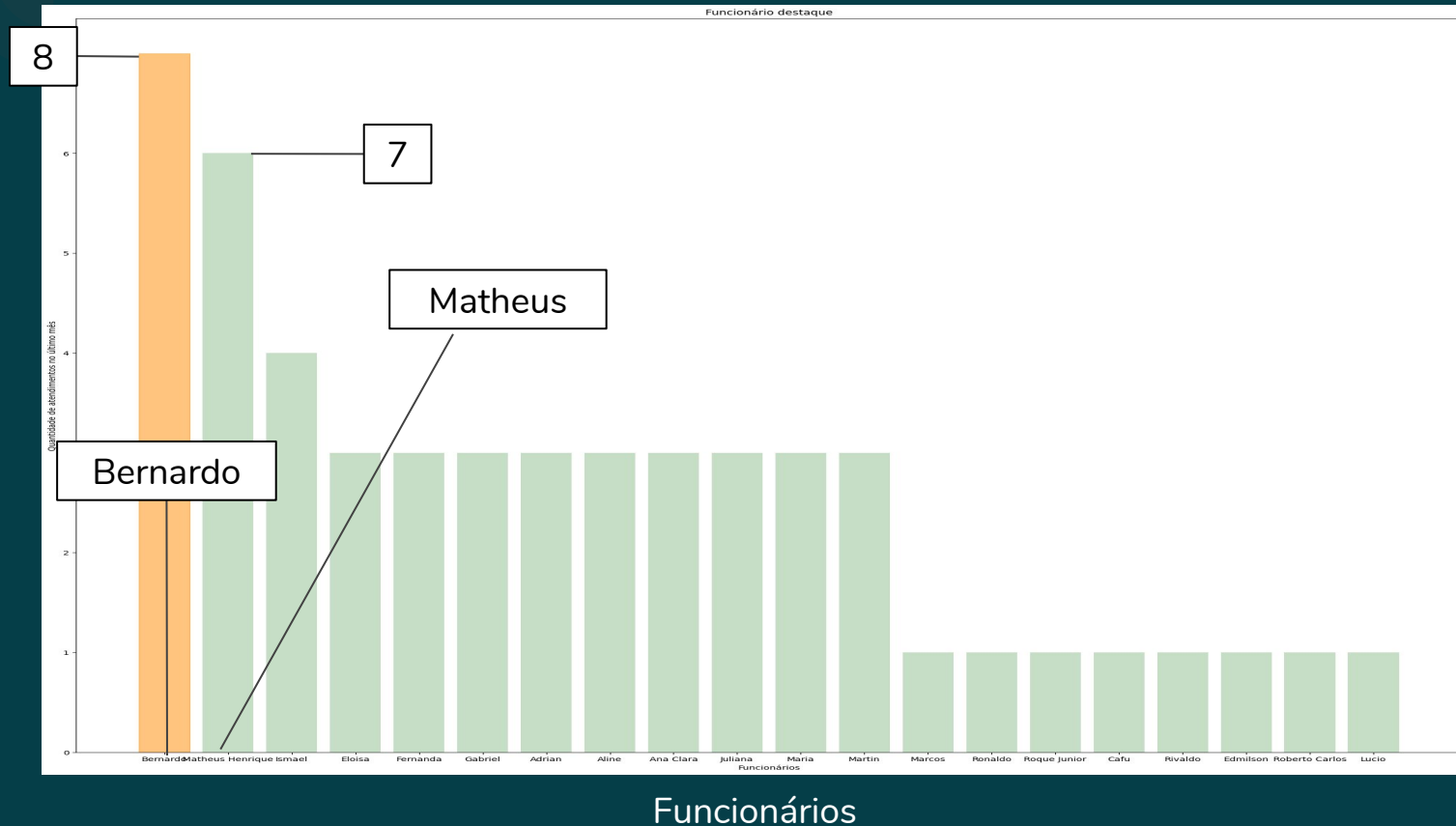
itens_pedidos	
0	Colorado
1	Jack Daniels
2	Reta da Penha
3	Caiaque
4	Gruta da onça
5	Borgo Magredo
6	Pepsi
7	Convento da Penha
8	Ilha de Socó
9	Stella Artois
10	Cidade Sol
11	Cachaça
12	Beira mar

PERGUNTAS A SEREM RESPONDIDAS

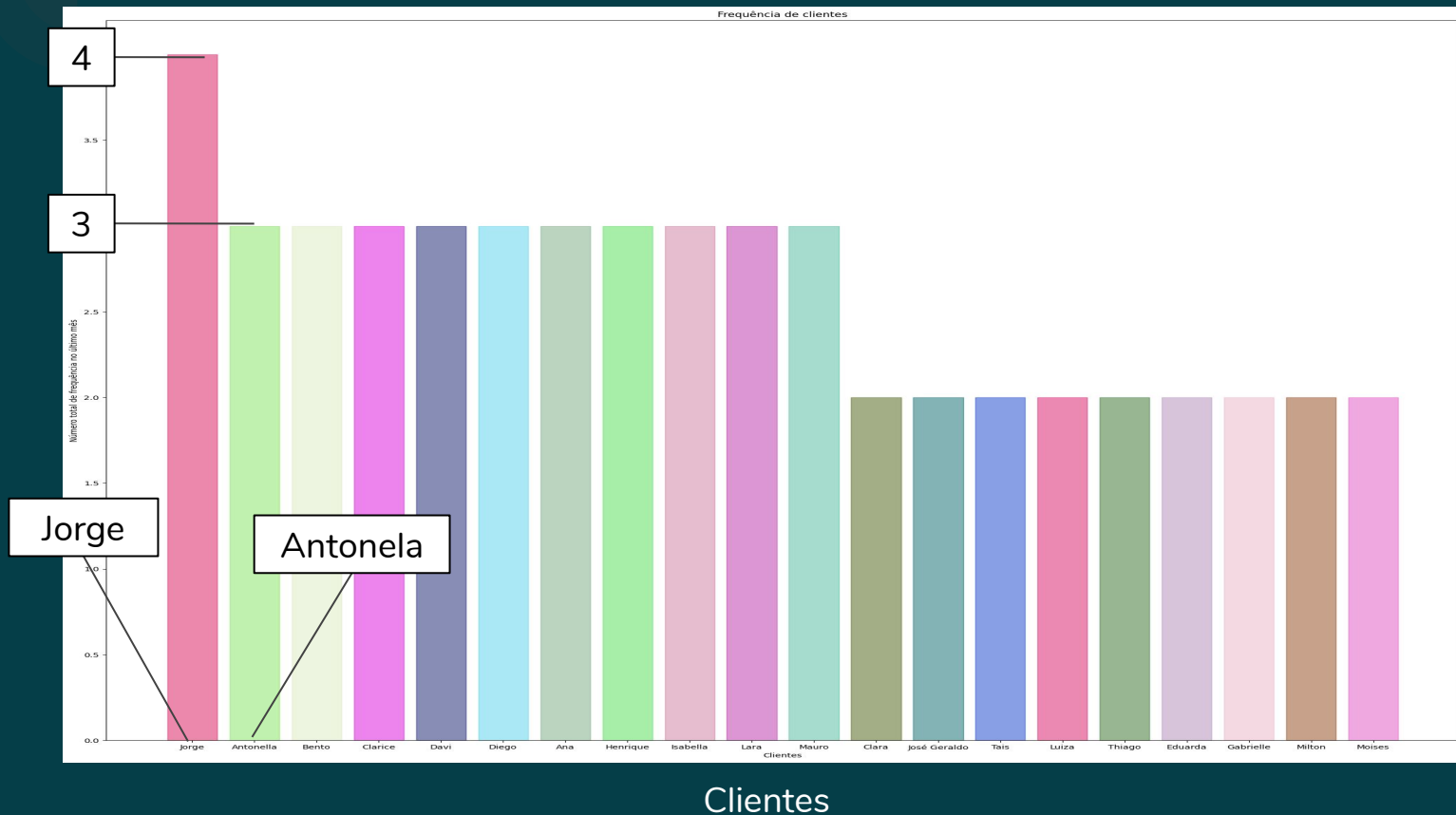
O restaurante em questão precisa inicialmente dos seguintes relatórios:

- **Relatório Funcionário destaque:** nome e a quantidade de comandas atendidas pelo funcionário que mais realizou atendimentos no mês.
- **Relatório Frequência de clientes:** nome e a frequência do cliente no mês.
- **Relatório Cardápio Top 3:** nome e valor dos itens de cardápio mais pedidos no mês.
- **Relatório Faturamento diário:** valor total faturado no dia.
- **Relatório Gasto por clientes:** nomes e o valor total gasto de cada cliente no mês.

RELATÓRIO 01 - FUNCIONÁRIO DESTAQUE



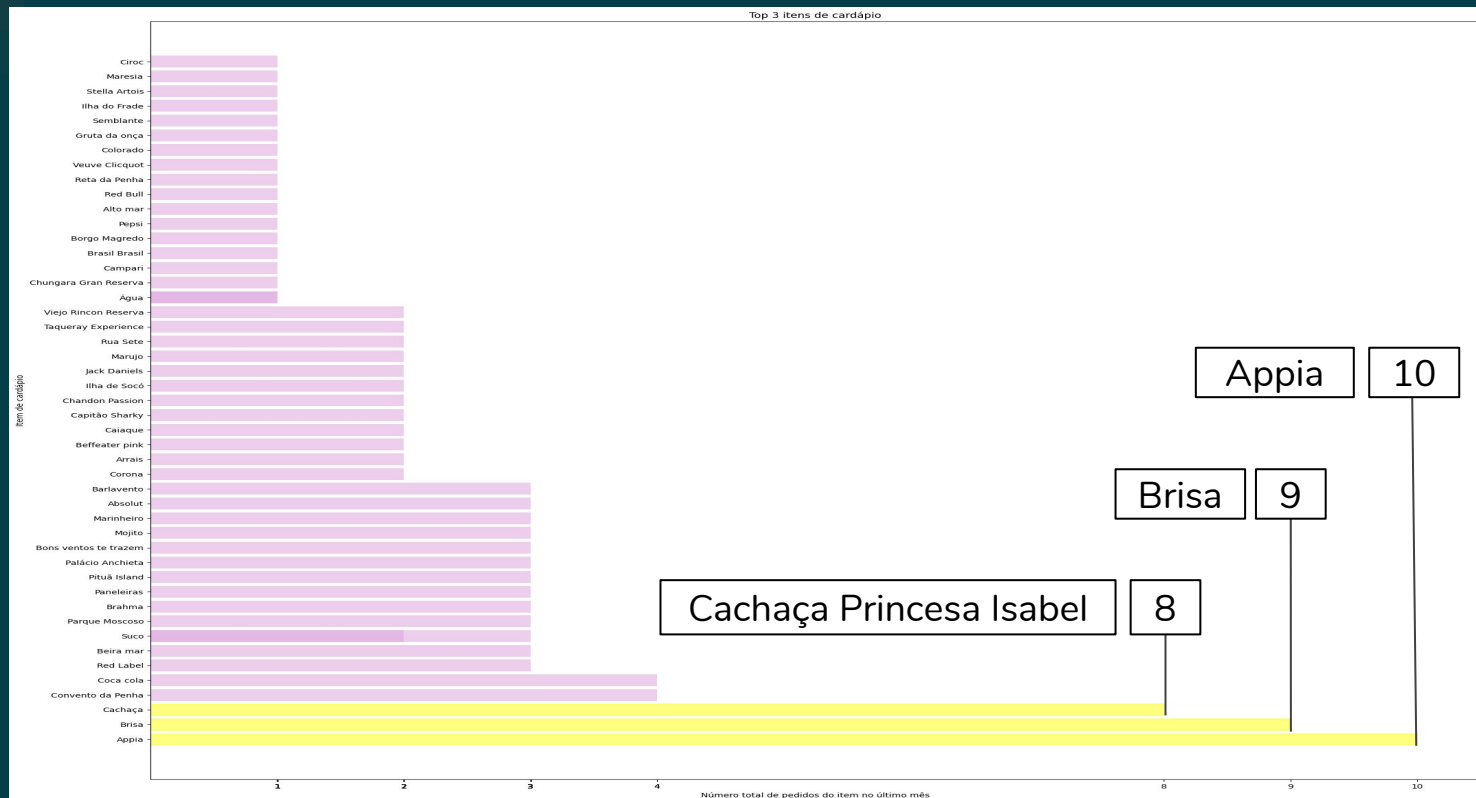
RELATÓRIO 02 - FREQUÊNCIA CLIENTES



Frequência

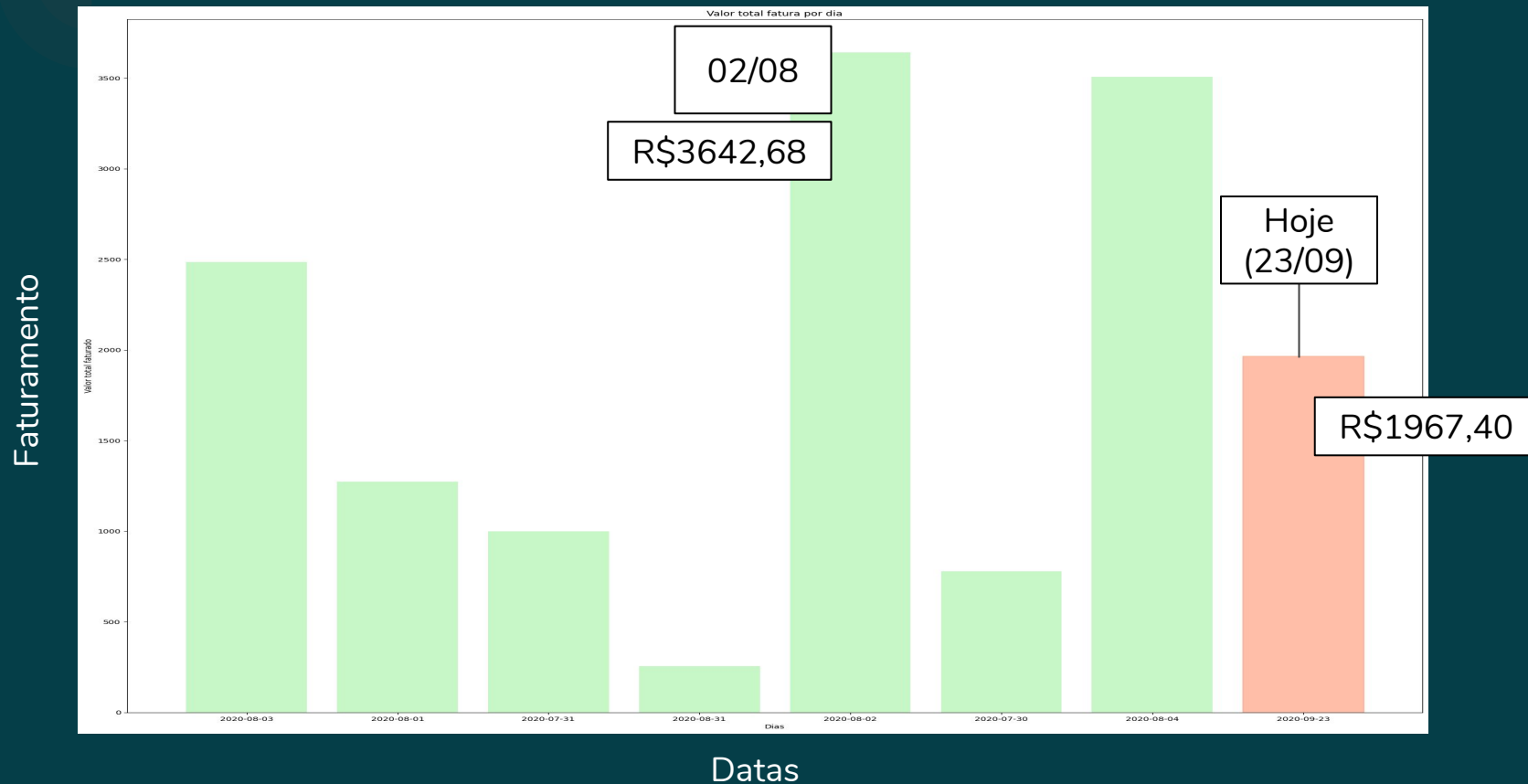
RELATÓRIO 03 - CARDÁPIO TOP 3

Itens cardápio

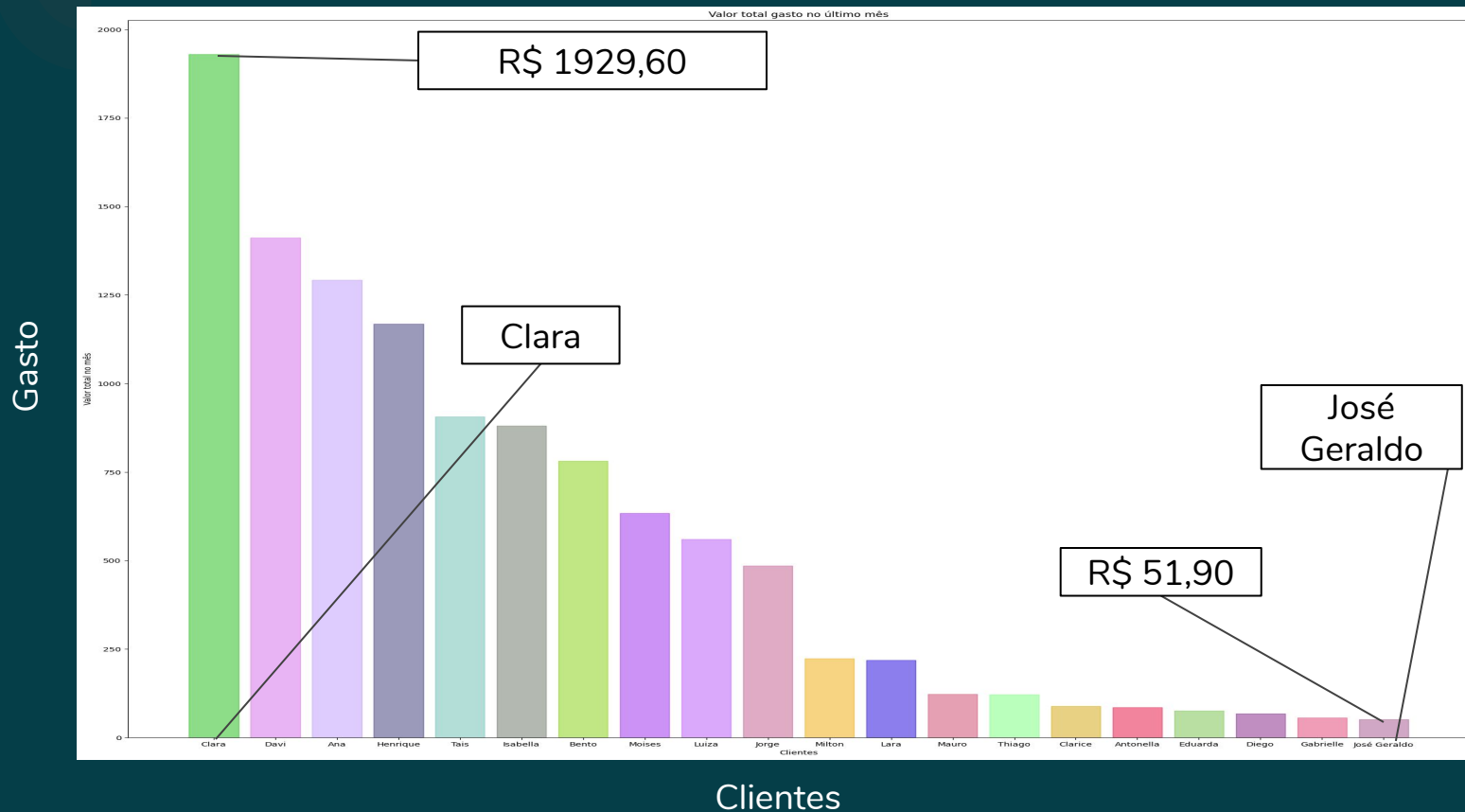


Pedidos

RELATÓRIO 04 - FATURAMENTO DIÁRIO

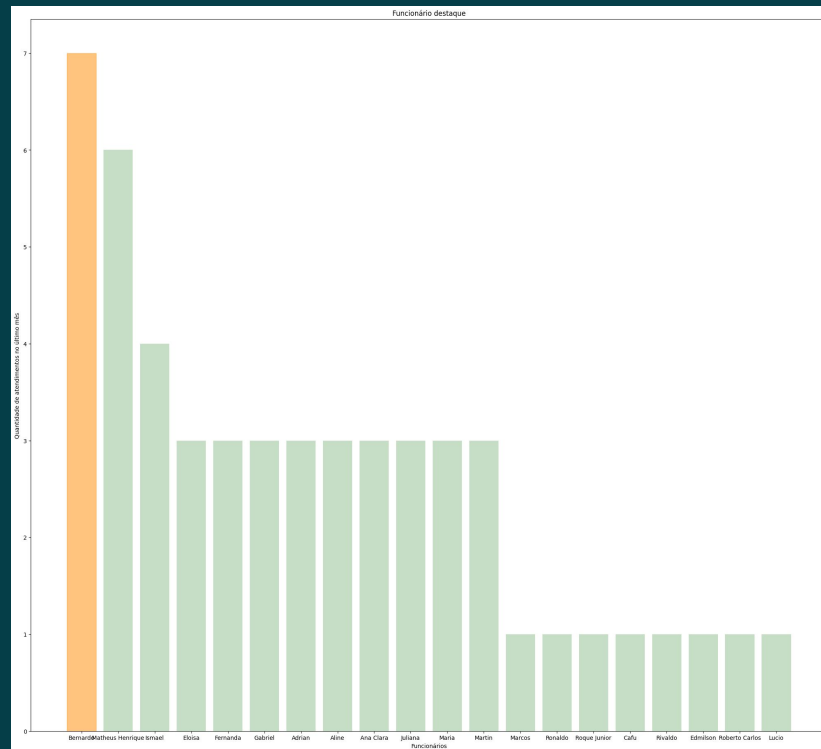


RELATÓRIO 05 - GASTO POR CLIENTE



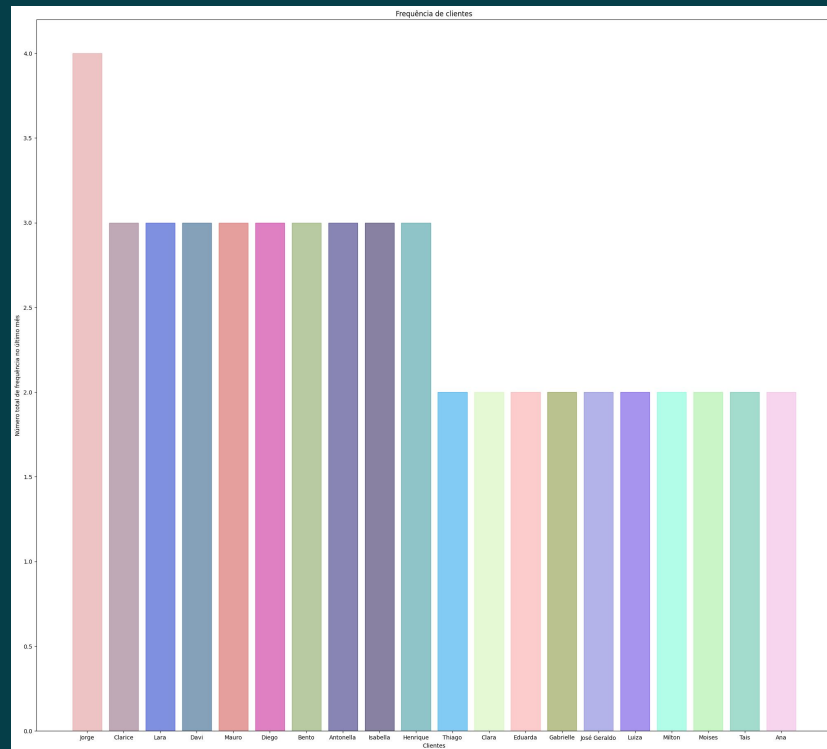
RELATÓRIO 01 - FUNCIONÁRIO DESTAQUE

```
select nome_funcionario,  
count(Fk_id_Funcionario) as atendimentos  
from comanda as c  
inner join funcionario as f on  
(f.id_Funcionario = c.Fk_id_Funcionario)  
where date_part('month',data) =  
date_part('month',(current_date +  
INTERVAL '-1 month')) and  
date_part('year',data) =  
date_part('year',(current_date + INTERVAL  
'-1 month'))  
group by nome_funcionario  
order by atendimentos desc
```

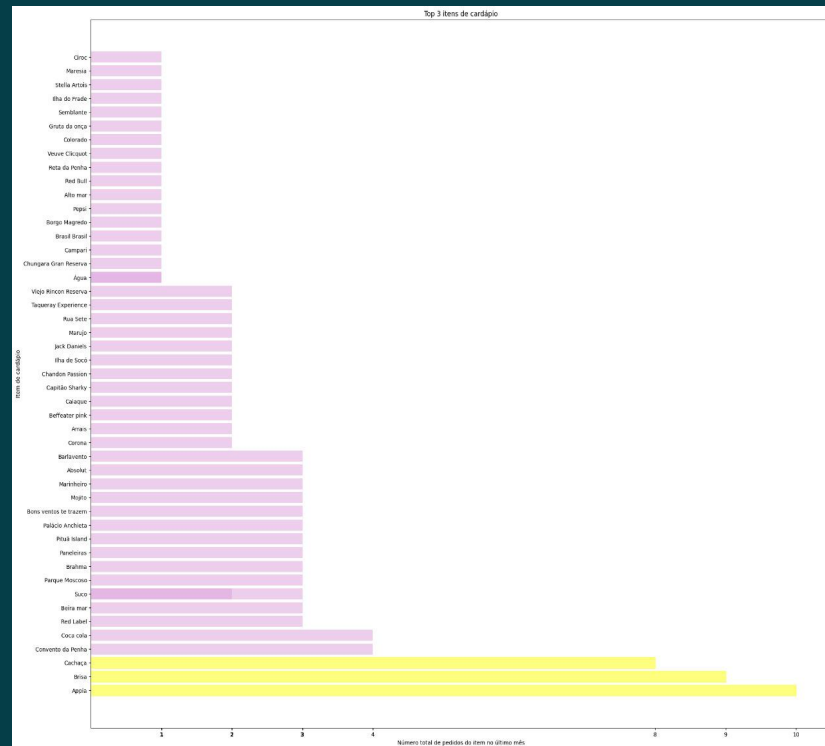


RELATÓRIO 02 - FREQUÊNCIA CLIENTES

```
select nome_cliente, count(distinct data) as  
frequencia  
from cliente as c  
inner join comanda as cd on  
(c.id_cliente = cd.fk_id_cliente)  
where date_part('month',data) =  
date_part('month',(current_date + INTERVAL  
'-1 month')) and date_part('year',data) =  
date_part('year',(current_date + INTERVAL  
'-1 month'))  
group by nome_cliente  
order by frequencia desc;
```



```
select ic.nome_item, ic.preco as valor,  
sum(quantidade) as total_pedidos  
from pedido as p  
inner join item_cardapio as ic on  
(ic.id_itemcardapio = p.fk_id_itemcardapio)  
inner join comanda as cd on  
(p.fk_id_comanda = cd.id_comanda)  
where date_part('month',data) =  
date_part('month',(current_date + INTERVAL  
'-1 month')) and date_part('year',data) =  
date_part('year',(current_date + INTERVAL '-1  
month'))  
group by ic.nome_item, ic.preco  
order by total_pedidos desc
```



RELATÓRIO 04 - FATURAMENTO DIÁRIO

```
select sum(preco * quantidade) as  
valor_total_dia, data  
from item_cardapio as cp  
inner join pedido as p on  
(p.fk_id_itemcardapio = cp.id_itemcardapio)  
inner join comanda as cd on  
(p.fk_id_comanda = cd.id_comanda)  
group by data
```



RELATÓRIO 05 - GASTO POR CLIENTE

```
select nome_cliente, sum(quantidade * preco) as
valor_total from comanda as cd
inner join pedido as p on
(cd.id_comanda = p.fk_id_comanda)
inner join item_cardapio as ic on(
p.fk_id_itemcardapio = ic.id_itemcardapio)
inner join cliente as c on
(cd.fk_id_cliente = c.id_cliente)
where date_part('month',data) =
date_part('month',(current_date + INTERVAL '-1
month')) and date_part('year',data) =
date_part('year',(current_date + INTERVAL '-1
month')) group by nome_cliente
order by valor_total desc;
```

