



INSTITUTO FEDERAL DA PARAÍBA CAMPUS CAMPINA GRANDE
BACHAREL EM ENGENHARIA DE COMPUTAÇÃO

DISCIPLINA: BANCO DE DADOS

PERÍODO: 2024.2

ALUNO: Guilherme José Mendes Jerônimo

MATRÍCULA: 202321250017

ALUNO: João Gabriel T. de Vasconcelos Souza

MATRÍCULA: 202321250011

Projeto Final de Banco de Dados

1A) - Introdução:

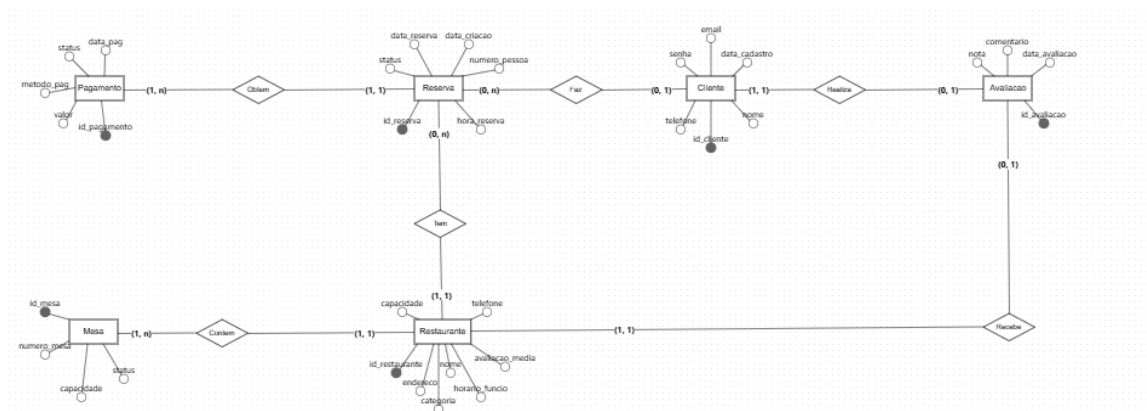
Nosso projeto veio para ajudar a mitigar as dificuldades na organização das reservas por parte dos restaurantes, resolvendo, por exemplo, a falta de controle sobre a capacidade do local e dificuldades na alocação de mesas. Para resolver essas questões, será desenvolvido um sistema de gerenciamento de reservas, permitindo que clientes realizem agendamentos de mesas de forma eficiente e que os restaurantes possam gerenciar sua capacidade com maior precisão.

O sistema contará com funcionalidades para que os clientes possam visualizar os horários disponíveis e realizar reservas antecipadas. Cada reserva será vinculada a um restaurante específico e poderá envolver uma ou mais mesas, dependendo do número de pessoas informadas no momento da solicitação. O restaurante poderá confirmar, modificar ou cancelar reservas conforme necessário. Além disso, poderá ser implementada uma funcionalidade opcional para pagamento antecipado das reservas.

1B) - Regra De Negócio:

- Cada reserva deve estar associada a um cliente e a um restaurante.
- Um cliente pode realizar múltiplas reservas, mas não pode reservar mesas em horários já ocupados.
- Um cliente não pode fazer mais de uma reserva no mesmo horário.
- O sistema deve verificar a disponibilidade de mesas antes de confirmar uma reserva.
- Restaurantes podem definir a capacidade máxima e as mesas disponíveis para reservas.
- A avaliação pode ou não ser feita pelo cliente.

2) - Modelo Relacional:



3A) - Modelo Lógico:

CARDINALIDADES (1:1 E 1:N):

pagamento(data_pag, status, metodo_pag, valor,*id_pagamento, id_reserva_fk)

Reserva(*id_reserva, hora_reserva, data_reserva, data_criacao, numero_pessoa,status, id_cliente_fk,id_Restaurante_fk)

Cliente(*id_cliente, senha, data_cadastro, nome)

Avaliacao(id_cliente_fk, nota, *id_avaliacao, comentario, data_avaliacao, id_restaurante_fk)

Restaurante(*id_restaurante , rua, numero, bairro, cidade, estado, pais, categoria, nome, horario_funcio, avaliacao_media, capacidade)

Mesa(id_restaurante_fk, *id_mesa, numero_mesa, capacidade, status)

ATRIBUTOS MULTIVALORADOS:

e-mail(*e-mail,*id_cliente)

telefone(*telefone,*id_cliente)

telefone_rest(*telefone,*id_restaurante)

3B) - Modelo Lógico (Dicionário de dados):

- Cliente:

Atributo	Tipo	Nulo	Descrição	Domínio	PRI	Chave EST	CAN
id_cliente	Inteiro	não	Código do cliente	Valor inteiro positivo	X		
nome	Texto(100)	não	Nome do cliente	Nome completo			
senha	Texto(255)	não	Senha do cliente	Contém espaços e frases			
data_cadastro	Texto(10)	não	Data do cadastro no aplicativo	String com traços com Dia,Mês e Ano			

● Pagamento:

Atributo	Tipo	Nulo	Descrição	Domínio	PRI	Chave EST	CAN
valor	Real	não	Valor do pagamento	Valor inteiro fracionário			
data_pag	Texto (10)	não	Data do pagamento	Contém dia, mês e ano			
metodo_pag	Enum	não	Método de pagamento	Cartão, dinheiro, Pix, Crédito...			
status	Enum	não	Estado do pagamento	Aprovado,Pendente,Cancelado			
id_pagamento	Inteiro	não	Código do pagamento	Valor inteiro	X		
id_reserva_fk	Inteiro	não	Código da reserva	Valor inteiro		X	

● Restaurante:

Atributo	Tipo	Nulo	Descrição	Domínio	PRI	Chave EST	CAN
id_restaurante	Inteiro	não	Código do restaurante	Valor inteiro positivo	X		
rua	Texto(100)	não	Nome da rua do restaurante	Rua do restaurante			
numero	Inteiro	não	Número da rua do restaurante	Número do mrestaurante			
bairro	Texto(100)	não	Nome do bairro do restaurante	Bairro da localização do restaurante			
cidade	Texto(100)	não	Nome da cidade do restaurante	Cidade do restaurante			
estado	Texto(100)	não	Nome do estado do restaurante	Estado do restaurante			
pais	Ttexto(100)	não	Nome do país do restaurante	País do restaurante			
categoria	Texto(50)	não	Tipo do restaurante	Chinês,Italiano,nordestino,japonês			
nome	Texto(100)	não	Nome do restaurante	String com nome do restaurante			
horario_funcio	Texto(100)	não	Horário do funcionamento do restaurante	String com horario			
avaliacao_media	Real	não	Avaliação do restaurante	Valor fracionário da avaliação			
capacidade	Inteiro	não	Capacidade do restaurante	Valor da capacidade do restaurante			

● Reserva:

Atributo	Tipo	Nulo	Descrição	Domínio	PRI	Chave EST	CAN
id_reserva	Inteiro	Não	Número da reserva	Valor inteiro positivo	X		
hora_reserva	Texto(10)	Não	Hora da reserva	String com horario de reserva			
data_reserva	Texto(10)	Não	Data da reserva	String com data de reserva			
data_criacao	Texto(10)	Não	Data da criação da reserva	String com data de criação da reserva			
numero_pessoa	Inteiro	Não	Quantidade de pessoas da reserva	Valor inteiro positivo			
status	Enum	Não	Estado da reserva	String de estados			
id_cliente_fk	Inteiro	Não	Número identificador do cliente	Valor inteiro positivo		X	
id_restaurante_fk	Inteiro	Não	Número identificador do restaurante	Valor inteiro positivo		X	

● Mesa:

Atributo	Tipo	Nulo	Descrição	Domínio	PRI	Chave EST	CAN
id_mesa	Inteiro	Não	Código da mesa	Valor inteiro positivo	X		
id_restaurante_fk	Inteiro	Não	Código do restaurante da mesa	Valor inteiro positivo		X	
numero_mesa	Inteiro	Não	Número da mesa	Valor inteiro positivo			
capacidade	Inteiro	Não	Capacidade da mesa	Valor inteiro positivo			
status	Enum	Não	Estatos de ocupação da mesa	Ocupada, Disponível, Reservada			

● Avaliação:

Atributo	Tipo	Nulo	Descrição	Domínio	PRI	Chave EST	CAN
id_avaliacao	Inteiro	Não	Código da avaliação	Valor inteiro positivo	X		
nota	Inteiro	Não	Número da nota da avaliação	Valor inteiro positivo			
comentario	Texto(50)	Não	Texto com comentário	String com comentário			
data_avaliacao	Texto(10)	Não	String com data da avaliação	String com a data			
id_restaurante_fk	Inteiro	Não	Código do restaurante	Valor inteiro positivo		X	
id_cliente_fk	Inteiro	Sim	Código do Cliente	Valor inteiro positivo		X	

● E-mail:

Atributo	Tipo	Nulo	Descrição	Domínio	PRI	Chave EST	CAN
e-mail	Texto(100)	Não	Email do cliente	String com o email	X		
id_cliente	Inteiro	Não	Código do cliente	Valor inteiro positivo	X		

● Telefone_cli:

Atributo	Tipo	Nulo	Descrição	Domínio	PRI	Chave EST	CAN
telefone	Texto(20)	Não	Telefone do cliente	String do telefone do cliente	X		
id_cliente	Inteiro	Não	Código do cliente	Valor inteiro positivo	X		

● Telefone_rest:

Atributo	Tipo	Nulo	Descrição	Domínio	PRI	Chave EST	CAN
telefone	Texto(20)	Não	Telefone do restaurante	String telefone do restaurante	X		
id_restaurante	Inteiro	Não	Código do restaurante	Valor inteiro positivo	X		

4A) - Modelo Físico (Scripts de Criação):

```
create table cliente (
    id_cliente int primary key auto_increment,
    nome varchar(100) not null,
    email varchar(100) unique not null,
    telefone varchar(20),
    senha_hash varchar(255) not null,
```

```
data_cadastro timestamp default current_timestamp,  
);
```

```
create table restaurante (  
    id_restaurante int primary key auto_increment,  
    nome varchar(100) not null,  
    endereco text not null,  
    telefone varchar(20),  
    categoria varchar(50),  
    horario_funcionamento varchar(50),  
    capacidade_total int,  
    avaliacao_media decimal(3,2) default 0.0  
);
```

```
create table reserva (  
    id_reserva int primary key auto_increment,  
    id_cliente int,  
    id_restaurante int,  
    data_reserva date not null,  
    hora_reserva time not null,  
    numero_pessoas int not null,  
    status enum('pendente', 'confirmada', 'cancelada', 'concluída') default 'pendente',  
    data_criacao timestamp default current_timestamp,  
    foreign key (id_cliente) references cliente(id_cliente),  
    foreign key (id_restaurante) references restaurante(id_restaurante)  
);
```

```
create table mesa (  
    id_mesa int primary key auto_increment,  
    id_restaurante int,  
    numero_mesa int,  
    capacidade int,  
    status enum('disponível', 'reservada', 'ocupada') default 'disponível',  
    foreign key (id_restaurante) references restaurante(id_restaurante)  
);
```

```
create table pagamento (  
    id_pagamento int primary key auto_increment,  
    id_reserva int unique,  
    metodo_pagamento enum('cartão', 'pix', 'dinheiro', 'crédito restaurante'),  
    valor decimal(10,2) not null,  
    status enum('aprovado', 'pendente', 'cancelado') default 'pendente',  
    data_pagamento timestamp default current_timestamp,  
    foreign key (id_reserva) references reserva(id_reserva)
```

);

```
create table avaliacao (  
    id_avaliacao int primary key auto_increment,  
    id_restaurante int,  
    id_cliente int null,  
    nota int check (nota between 1 and 5),  
    comentario text,  
    data_avaliacao timestamp default current_timestamp,  
    foreign key (id_restaurante) references restaurante(id_restaurante),  
    foreign key (id_cliente) references cliente(id_cliente)  
);
```

4B) - Alimentação inicial do banco de dados:

```
INSERT INTO restaurante (nome, endereco, telefone, categoria, horario_funcionamento,  
capacidade_total, avaliacao_media) VALUES  
(('Sushi House', 'Rua Aristides Lobo, 123, João Pessoa', '(83) 4002-8922', 'Japonês', '12:00 -  
22:00', 50, 4.5),  
(('Charque Show', 'Av. Joaquim Caroca, 456, Campina Grande', '(83) 3003-1234', 'Italiano',  
'11:00 - 23:00', 80, 4.7),  
(('Churrascaria Gaúcha', 'Rua Antônio Bernadino de Sena, 789, Patos', '(83) 3456-7890',  
'Churrasco', '12:00 - 23:00', 100, 4.8),  
(('Fast Burgers', 'Av. Evaldo Braga, 321, Guarabira', '(83) 4009-1122', 'Fast Food', '10:00 - 22:00',  
60, 4.3),  
(('Vegano Delícia', 'Rua Francisco, 654, Sousa', '(83) 3654-7891', 'Vegano', '11:00 - 21:00', 40,  
4.6),  
(('Pizzaria Napoli', 'Rua das Flores, 100, João Pessoa', '(83) 3322-1100', 'Italiana', '18:00 -  
23:30', 70, 4.6),  
(('Casa do Peixe', 'Av. Beira Rio, 200, Campina Grande', '(83) 3201-2202', 'Frutos do Mar', '12:00  
- 22:00', 50, 4.8),  
(('Steak House Grill', 'Rua da Estação, 300, Patos', '(83) 3355-3311', 'Churrasco', '11:30 - 23:00',  
90, 4.7),  
(('Pastel & Cia', 'Av. Central, 400, Guarabira', '(83) 3211-5544', 'Fast Food', '09:00 - 22:00', 40,  
4.2),  
(('Restaurante Imperial', 'Rua da Paz, 500, Sousa', '(83) 3566-7788', 'Gourmet', '12:00 - 23:00',  
100, 4.9);
```

```
INSERT INTO cliente (nome, email, telefone, senha_hash) VALUES  
(('João Silva', 'joao.silva@email.com', '(11) 98765-4321', 'hashsenha123'),  
(('Maria Oliveira', 'maria.oliveira@email.com', '(21) 91234-5678', 'senhaSegura456'),  
(('Carlos Souza', 'carlos.souza@email.com', '(31) 99876-5432', 'hashsenha789'),  
(('Ana Santos', 'ana.santos@email.com', '(41) 93456-7890', 'senha123abc'),
```

```
('Fernanda Lima', 'fernanda.lima@email.com', '(51) 97654-3210', 'hash456xyz'),  
('Roberto Figueiredo', 'roberto.figueiredo@email.com', '(11) 98888-1111', 'senha123!'),  
('Luciana Mendes', 'luciana.mendes@email.com', '(21) 97777-2222', 'segura456$'),  
('Fernando Costa', 'fernando.costa@email.com', '(31) 96666-3333', 'hashSenha789@'),  
('Paula Ribeiro', 'paula.ribeiro@email.com', '(41) 95555-4444', 'senhaForte!'),  
('Mariana Alves', 'mariana.alves@email.com', '(51) 94444-5555', 'segredoXYZ#');
```

```
INSERT INTO reserva (id_cliente, id_restaurante, data_reserva, hora_reserva,  
numero_pessoas, status) VALUES  
(1, 1, '2025-02-15', '19:00', 2, 'Confirmada'),  
(2, 2, '2025-02-16', '20:00', 4, 'Pendente'),  
(3, 3, '2025-02-17', '21:00', 6, 'Confirmada'),  
(4, 4, '2025-02-18', '18:30', 3, 'Cancelada'),  
(5, 5, '2025-02-19', '19:45', 5, 'Concluída'),  
(6, 6, '2025-02-20', '19:00', 2, 'Confirmada'),  
(7, 7, '2025-02-21', '20:30', 4, 'Pendente'),  
(8, 8, '2025-02-22', '21:15', 3, 'Confirmada'),  
(9, 9, '2025-02-23', '18:45', 5, 'Cancelada'),  
(10, 10, '2025-02-24', '19:30', 6, 'Concluída');
```

```
INSERT INTO mesa (id_restaurante, numero_mesa, capacidade, status) VALUES  
(1, 1, 4, 'Disponível'),  
(1, 2, 6, 'Reservada'),  
(2, 3, 2, 'Ocupada'),  
(3, 4, 8, 'Disponível'),  
(4, 5, 6, 'Disponível'),  
(6, 6, 4, 'Disponível'),  
(6, 7, 6, 'Reservada'),  
(7, 8, 2, 'Ocupada'),  
(8, 9, 8, 'Disponível'),  
(9, 10, 6, 'Disponível');
```

```
INSERT INTO pagamento (id_reserva, metodo_pagamento, valor, status) VALUES  
(1, 'Cartão', 150.00, 'Aprovado'),  
(2, 'Pix', 200.00, 'Pendente'),  
(3, 'Dinheiro', 300.00, 'Aprovado'),  
(4, 'Crédito Restaurante', 120.00, 'Cancelado'),  
(5, 'Cartão', 250.00, 'Aprovado'),  
(6, 'Cartão', 180.00, 'Aprovado'),  
(7, 'Pix', 250.00, 'Pendente'),  
(8, 'Dinheiro', 270.00, 'Aprovado'),  
(9, 'Crédito Restaurante', 100.00, 'Cancelado'),  
(10, 'Cartão', 320.00, 'Aprovado');
```

```
INSERT INTO avaliacao (id_cliente, id_restaurante, nota, comentario) VALUES
(1, 1, 5, 'Comida excelente e atendimento impecável!'),
(3, 3, 5, 'Melhor churrasco da minha vida!'),
(4, 4, 3, 'Hambúrguer bom, mas atendimento poderia melhorar.'),
(5, 5, 4, 'Ótima opção vegana, recomendo muito!'),
(6, 6, 5, 'Melhor pizza que já comi!'),
(7, 7, 4, 'Peixe delicioso, mas demorou um pouco.'),
(8, 8, 5, 'Churrasco no ponto perfeito, recomendo!'),
(10, 10, 5, 'Melhor experiência gastronômica da região!');
```

4C) - Atualização do banco de dados:

Inserção

```
insert into mesa (id_restaurante,numero_mesa,capacidade,status)VALUES
(4,3,4,'Disponivel');
```

Remoções

```
delete from pagamento WHERE id_reserva = 4;
delete from mesa WHERE id_restaurante =1;
```

```
start transation;
```

```
delete from pagamento WHERE id_reserva = 4;
```

```
delete from reserva WHERE id_reserva = 4;
```

```
commit;
```

Atualização de tabela

```
update cliente
set nome = 'Ana'
where id_cliente=1;
```

```
update mesa
set status = 'Reservado'
where id_restaurante = 1;
```

```
update restaurante r
join (
  select id_restaurante, avg(nota) Media
  from avaliacao
```



```

group by id_restaurante
) a on r.id_restaurante = a.id_restaurante
set r.avaliacao_media = a.media;

```

4D) - Consultas:

1. Listar todas as reservas, incluindo informações do cliente e do restaurante

```

select r.id_reserva, c.nome Cliente, rest.nome Restaurante, r.hora_reserva HorarioDaReserva,
r.numero_pessoas NumeroDePessoas, r.status Status from reserva r join cliente c on
r.id_cliente = c.id_cliente join restaurante rest on r.id_restaurante = rest.id_restaurante;

```

id_reserva	Cliente	Restaurante	HorarioDaReserva	NumeroDePessoas	Status
1	Ana	Sushi House	19:00:00	2	Confirmada
2	Maria Oliveira	Charque Show	20:00:00	4	Pendente
3	Carlos Souza	Churrascaria Ga?cha	21:00:00	6	Confirmada
5	Fernanda Lima	Vegano Del?cia	19:45:00	5	Concluída
6	Roberto Figueiredo	Pizzaria Napoli	19:00:00	2	Confirmada
7	Luciana Mendes	Casa do Peixe	20:30:00	4	Pendente
8	Fernando Costa	Steak House Grill	21:15:00	3	Confirmada
9	Paula Ribeiro	Pastel & Cia	18:45:00	5	Cancelada
10	Mariana Alves	Restaurante Imperial	19:30:00	6	Concluída

2. Listar todas as avaliações feitas por clientes, incluindo o nome do restaurante

```

select a.id_avaliacao, c.nome cliente, rest.nome restaurante, a.nota, a.comentario from
avaliacao a join cliente c on a.id_cliente = c.id_cliente join restaurante rest on a.id_restaurante
= rest.id_restaurante;

```

id_avaliacao	cliente	restaurante	nota	comentario
1	Ana	Sushi House	5	Comida excelente e atendimento impec?vel!
2	Maria Oliveira	Charque Show	4	?tima comida, mas demorou um pouco.
3	Carlos Souza	Churrascaria Ga?cha	5	Melhor churrasco da minha vida!
4	Ana Santos	Fast Burgers	3	Hamb?rguer bom, mas atendimento poderia melhorar.
5	Fernanda Lima	Vegano Del?cia	4	?tima op??o vegana, recomendo muito!
6	Roberto Figueiredo	Pizzaria Napoli	5	Melhor pizza que j? comi!
7	Luciana Mendes	Casa do Peixe	4	Peixe delicioso, mas demorou um pouco.
8	Fernando Costa	Steak House Grill	5	Churrasco no ponto perfeito, recomendo!
9	Paula Ribeiro	Pastel & Cia	3	Pastel gostoso, mas atendimento m?dio.
10	Mariana Alves	Restaurante Imperial	5	Melhor experi?ncia gastron?mica da regi?o!

3. Mostrar todas as reservas confirmadas com o método De pagamento e status Do pagamento

```

select r.id_reserva, c.nome cliente, rest.nome restaurante, r.data_reserva, r.hora_reserva,
p.metodo_pagamento, p.status statusPagamento from reserva r join cliente c on r.id_cliente =

```

c.id_cliente join restaurante rest on r.id_restaurante = rest.id_restaurante left join pagamento p on r.id_reserva = p.id_reserva where r.status = 'Confirmada';

id_reserva	cliente	restaurante	data_reserva	hora_reserva	metodo_pagamento	statusPagamento
1	Ana	Sushi House	2025-02-15	19:00:00	Cartão	Aprovado
3	Carlos Souza	Churrascaria Ga?cha	2025-02-17	21:00:00	Dinheiro	Aprovado
6	Roberto Figueiredo	Pizzaria Napoli	2025-02-20	19:00:00	Cartão	Aprovado
8	Fernando Costa	Steak House Grill	2025-02-22	21:15:00	Dinheiro	Aprovado

4. Listar todas as mesas disponíveis em cada restaurante

select m.numero_mesa, rest.nome restaurante, m.capacidade from mesa m join restaurante rest on m.id_restaurante = rest.id_restaurante where m.status = 'Disponível';

numero_mesa	restaurante	capacidade
4	Churrascaria Ga?cha	8
5	Fast Burgers	6
6	Pizzaria Napoli	4
9	Steak House Grill	8
10	Pastel & Cia	6

5. Listar clientes que já fizeram reservas, incluindo o nome Do restaurante e a quantidade De reservas feitas

select c.nome cliente, rest.nome restaurante, count(r.id_reserva) totalReservas from cliente c join reserva r on c.id_cliente = r.id_cliente join restaurante rest on r.id_restaurante = rest.id_restaurante group by c.nome, rest.nome;

cliente	restaurante	totalReservas
Ana	Sushi House	1
Carlos Souza	Churrascaria Ga?cha	1
Fernanda Lima	Vegano Del?cia	1
Fernando Costa	Steak House Grill	1
Luciana Mendes	Casa do Peixe	1
Maria Oliveira	Charque Show	1
Mariana Alves	Restaurante Imperial	1
Paula Ribeiro	Pastel & Cia	1
Roberto Figueiredo	Pizzaria Napoli	1

6. Listar todas as avaliações feitas e mostrar também clientes que nunca avaliaram:

```
select c.nome cliente, rest.nome restaurante, a.nota, a.comentario from cliente c left join
avaliacao a on c.id_cliente = a.id_cliente left join restaurante rest on a.id_restaurante =
rest.id_restaurante;
```

cliente	restaurante	nota	comentario
Ana	Sushi House	5	Comida excelente e atendimento impecável!
Maria Oliveira	NULL	NULL	NULL
Carlos Souza	Churrascaria Gaúcha	5	Melhor churrasco da minha vida!
Ana Santos	Fast Burgers	3	Hambúrguer bom, mas atendimento poderia melhorar.
Fernanda Lima	Vegano Delícia	4	Ótima opção vegana, recomendo muito!
Roberto Figueiredo	Pizzaria Napoli	5	Melhor pizza que já comi!
Luciana Mendes	Casa do Peixe	4	Peixe delicioso, mas demorou um pouco.
Fernando Costa	Steak House Grill	5	Churrasco no ponto perfeito, recomendo!
Paula Ribeiro	NULL	NULL	NULL
Mariana Alves	Restaurante Imperial	5	Melhor experiência gastronômica da região!

7. Listar todas as reservas e, se houver pagamento, exibir o valor pago:

```
select r.id_reserva, c.nome cliente, rest.nome restaurante, r.data_reserva, p.valor valorPago
from reserva r left join cliente c on r.id_cliente = c.id_cliente left join restaurante rest on
r.id_restaurante = rest.id_restaurante left join pagamento p on r.id_reserva = p.id_reserva;
```

id_reserva	cliente	restaurante	data_reserva	valorPago
1	Ana	Sushi House	2025-02-15	150.00
2	Maria Oliveira	Charque Show	2025-02-16	200.00
3	Carlos Souza	Churrascaria Gaúcha	2025-02-17	300.00
5	Fernanda Lima	Vegano Delícia	2025-02-19	250.00
6	Roberto Figueiredo	Pizzaria Napoli	2025-02-20	180.00
7	Luciana Mendes	Casa do Peixe	2025-02-21	250.00
8	Fernando Costa	Steak House Grill	2025-02-22	270.00
9	Paula Ribeiro	Pastel & Cia	2025-02-23	100.00
10	Mariana Alves	Restaurante Imperial	2025-02-24	320.00

8. Contar o número total De reservas feitas por restaurante:

```
select rest.nome restaurante, count(r.id_reserva) totalReservas from reserva r join restaurante
rest on r.id_restaurante = rest.id_restaurante group by rest.nome;
```

restaurante	totalReservas
Casa do Peixe	1
Charque Show	1
Churrascaria Ga?cha	1
Pastel & Cia	1
Pizzaria Napoli	1
Restaurante Imperial	1
Steak House Grill	1
Sushi House	1
Vegano Del?cia	1

9. Calcular a média Das avaliações De cada restaurante:

```
select rest.nome restaurante, round(avg(a.nota), 2) mediaAvaliacao from avaliacao a join
restaurante rest on a.id_restaurante = rest.id_restaurante group by rest.nome;
```

restaurante	mediaAvaliacao
Casa do Peixe	4.00
Churrascaria Ga?cha	5.00
Fast Burgers	3.00
Pizzaria Napoli	5.00
Restaurante Imperial	5.00
Steak House Grill	5.00
Sushi House	5.00
Vegano Del?cia	4.00

10. Quantidade De mesas por restaurante:

```
select rest.nome restaurante, sum(m.capacidade) totalCapacidade from mesa m join
restaurante rest on m.id_restaurante = rest.id_restaurante group by rest.nome;
```

restaurante	totalCapacidade
Casa do Peixe	2
Charque Show	2
Churrascaria Ga?cha	8
Fast Burgers	10
Pastel & Cia	6
Pizzaria Napoli	10
Steak House Grill	8

11. Valor total arrecadado por restaurante em pagamentos confirmados:

```
select rest.nome restaurante, sum(p.valor) totalArrecadado from pagamento p join reserva r on
p.id_reserva = r.id_reserva join restaurante rest on r.id_restaurante = rest.id_restaurante where
p.status = 'Aprovado' group by rest.nome;
```

restaurante	totalArrecadado
Churrascaria Ga?cha	300.00
Pizzaria Napoli	180.00
Restaurante Imperial	320.00
Steak House Grill	270.00
Sushi House	150.00
Vegano Del?cia	250.00

12. Listar restaurantes que receberam menos de 3 reservas:

```
select rest.nome restaurante, count(r.id_reserva) totalReservas from reserva r join restaurante
rest on r.id_restaurante = rest.id_restaurante group by rest.nome having count(r.id_reserva) < 3;
```

restaurante	totalReservas
Casa do Peixe	1
Charque Show	1
Churrascaria Ga?cha	1
Pastel & Cia	1
Pizzaria Napoli	1
Restaurante Imperial	1
Steak House Grill	1
Sushi House	1
Vegano Del?cia	1

13. Mostrar restaurantes com avaliação média acima De 4.5:

```
select rest.nome restaurante, avg(a.nota) mediaAvaliacao from avaliacao a join restaurante rest
on a.id_restaurante = rest.id_restaurante group by rest.nome having avg(a.nota) > 4.5;
```

restaurante	mediaAvaliacao
Churrascaria Ga?cha	5.0000
Pizzaria Napoli	5.0000
Restaurante Imperial	5.0000
Steak House Grill	5.0000
Sushi House	5.0000

14. Listar restaurantes que faturaram mais De R\$ 200 em pagamentos aprovados:

```
select rest.nome restaurante, sum(p.valor) totalFaturamento from pagamento p join reserva r on
p.id_reserva = r.id_reserva join restaurante rest on r.id_restaurante = rest.id_restaurante where
p.status = 'Aprovado' group by rest.nome having sum(p.valor) > 200;
```

restaurante	totalFaturamento
Churrascaria Ga?cha	300.00
Restaurante Imperial	320.00
Steak House Grill	270.00
Vegano Del?cia	250.00

15. Listar clientes que fizeram menos de 3 reservas:

```
select c.nome cliente, count(r.id_reserva) totalReservas from reserva r join cliente c on
r.id_cliente = c.id_cliente group by c.nome having count(r.id_reserva) < 3;
```

cliente	totalReservas
Ana	1
Carlos Souza	1
Fernanda Lima	1
Fernando Costa	1
Luciana Mendes	1
Maria Oliveira	1
Mariana Alves	1
Paula Ribeiro	1
Roberto Figueiredo	1