

**UNIVERSIDADE FEDERAL DE UBERLÂNDIA**  
**FACULDADE DE COMPUTAÇÃO**

**P R O J E T O: PIZZA DELIVERY COM ENTRETENIMENTO**

**Curso: Bacharelado em Sistemas de Informação**

**Disciplina: Banco de Dados 1**

**Professora: Maria Camila Nardini Barioni**

**Alunos: Vitor Manoel Gonçalves Teixeira, Vinícius Rodrigues, Wallace Keller  
Silva Pinto**

**19 de Novembro de 2019**

## ÍNDICE

<b>1) REQUISITOS ADICIONAIS .....</b>	<b>3</b>
<b>2) ESQUEMA CONCEITUAL .....</b>	<b>3</b>
2.1) DIAGRAMA DO MODELO ENTIDADE-RELACIONAMENTO.	3
2.2) ESCOLHAS DE MODELAGEM E JUSTIFICATIVA.....	5
<b>3) ESQUEMA RELACIONAL.....</b>	<b>9</b>
3.1) MODELO RELACIONAL.....	9
3.2) ESCOLHAS DE MAPEAMENTO E JUSTIFICATIVAS.....	11
<b>4) CRIAÇÃO DE TABELAS EM SQL.....</b>	<b>12</b>
<b>5) ESPECIFICAÇÃO DE CONSULTAS EM SQL.....</b>	<b>20</b>
5.1) CONSULTAS.....	20
5.2) OPERAÇÕES DE INSERÇÃO.....	28

## **1) REQUISITOS ADICIONAIS**

Como requisitos extras, foram adicionados os requisitos de que o cliente possa fazer mais dois tipos de pedidos(que podem ser do mesmo pedido, ou seja, um pedido pode ser tanto um pedido padrão, quanto pedido de entretenimento e também desses outros dois tipos ao mesmo tempo). São eles: o pedido de retirada e o pedido de combos.

O pedido de retirada serve para que o usuário reserve seu pedido para ser retirado na pizzeria, e possui o atributo com o horário em que o cliente retirou o pedido na pizzeria. O pedido de combos permite que o cliente peça combos(promoções com mais de uma pizza e/ou acompanhamentos extras por preços mais baratos).

Com isso, a entidade de pedido\_combos se relaciona com a entidade de combos, com atributos com informações sobre o combo, e essa, por sua vez, se relaciona com a entidade de pizza. Um combo pode ter n pizzas, e uma pizza pode estar em n combos.

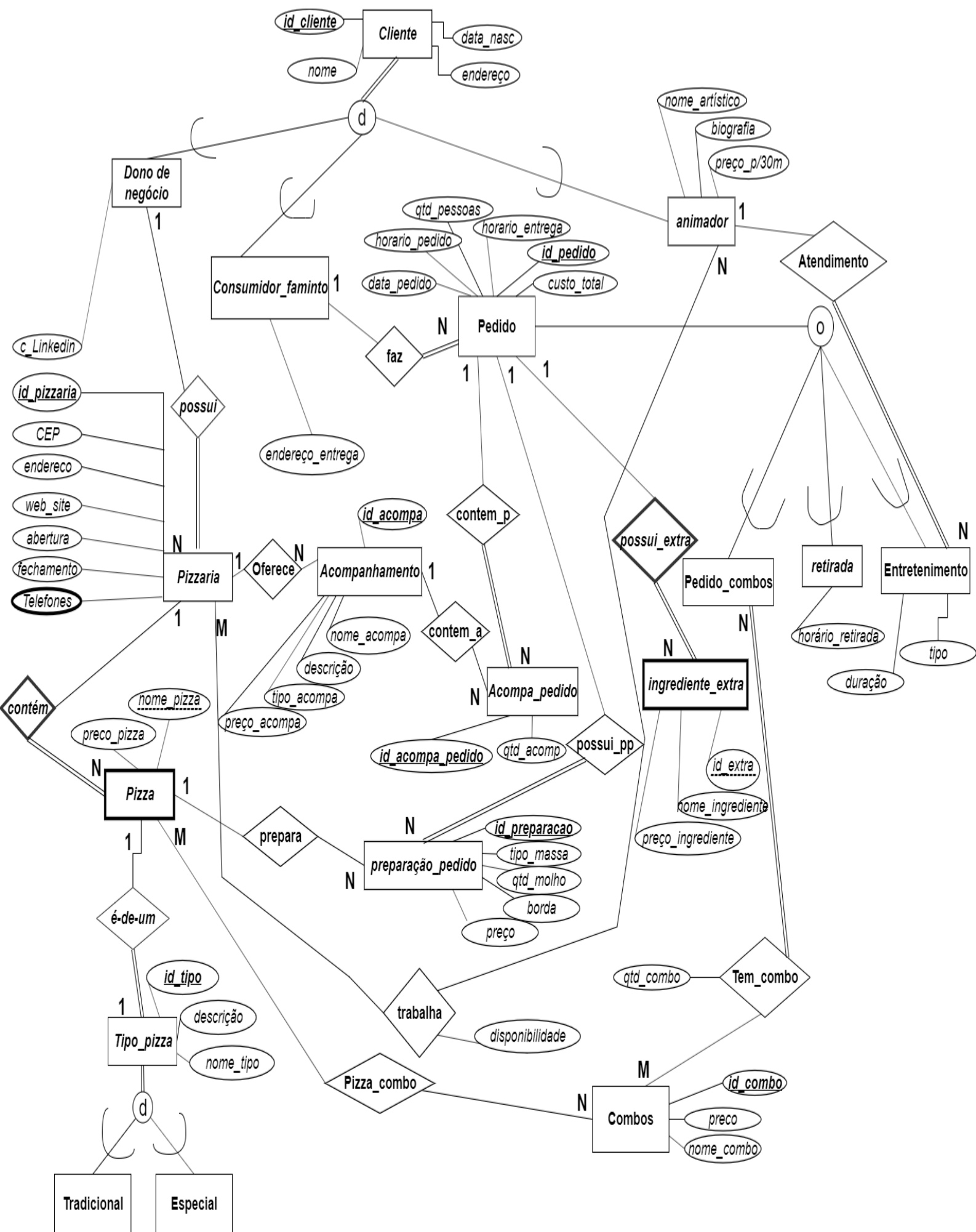
Tanto o pedido de combos, quando o de retirada e o de entretenimento, foram modelados como subclasses da entidade pedido, sob uma disjunção do tipo “o” parcial, para que o usuário possa fazer pedidos padrão(a superclasse pedido), de retirada, de combo e/ou de entretenimento.

## **2) ESQUEMA CONCEITUAL**

Aqui mostra-se o diagrama do modelo entidade-relacionamento, assim como as explicações e justificativas para as escolhas de modelagem de cada requisito exigido pelo trabalho em seguida.

### **2.1) DIAGRAMA DO MODELO ENTIDADE-RELACIONAMENTO**

A imagem que mostra o diagrama encontra-se na próxima página.



## 2.2) ESCOLHAS DE MODELAGEM E JUSTIFICATIVAS

Aqui são explicadas as escolhas de modelagem de cada requisito citado no trabalho, representado por sua letra no enunciado:

- a) Sobre o pedido de entretenimento, ele faz parte de uma disjunção do tipo “o” parcial e é uma subclasse da entidade “pedido”(a disjunção é parcial para que a entidade “pedido” possa representar o “pedido padrão” sem nenhuma outra subclasse). A entidade “entretenimento” possui o atributo “duração”, que representa por quanto tempo o animador ficou com o(s) cliente(s). Essa Entidade tem participação total com o relacionamento “atendimento”, que a relaciona com a entidade “animador”, sendo que 1 animador pode fazer N pedidos de entretenimento.
- b) Para esse requisito, foi criada a entidade “cliente”, que possui todos os atributos mencionados no requisito(nome, endereço, data de nascimento) e uma chave primária id\_cliente. Nota-se que “cliente” não se refere a “cliente de pizzeria”, mas sim, cliente do aplicativo, que pode ser um consumidor faminto, um dono de pizzeria ou um animador. Para isso, a entidade “cliente” faz uma disjunção do tipo “d” total, que cria 3 subclasses, sendo elas: consumidor\_faminto, dono\_de\_negócio e animador.
- c) Para o requisito do dono de negócio, existe a entidade já mencionada “dono\_de\_negócio”, que possui apenas um atributo, sendo ele o atributo “c\_linkedin”, que contém o código do perfil do linkedin do usuário. Este código, adicionado ao final da url “linkedin.com/in/”, se torna o link direto para o perfil.

- d) Foi feito da maneira requisitada. 1 dono de negócio possui N pizzarias.
- e) Foi feita a entidade “pizzaria”, e ela possui todos os atributos requisitados(cep, endereço, número de telefone, web site, horário de abertura e horário de fechamento). O atributo “telefones” foi feito como um atributo multivalorado, pro caso de uma pizzaria ter vários telefones(fixo, celular, whatsapp, etc). Também foi criada uma chave primária “id\_pizzaria”, que identifica cada uma de forma única.
- f) A cardinalidade entre pizzaria e pizza foi feita como requisitado: 1 pizzaria contém N pizzas. A entidade “pizza” foi feita como entidade fraca, por entender-se que ela depende da pizzaria para existir. Ela contém os atributos exigidos(nome e preço), e o atributo nome é a chave parcial dessa entidade fraca. Sendo assim, cada pizza será distinguível, por sua chave ser o id da pizzaria e o nome dela.
- g) Para esse requisito, foi entendido que cada pizza pode ter 1 tipo, e cada tipo está relacionado a uma pizza, assim, cardinalidade 1:1. Para isso, foi criada a entidade forte “tipo\_pizza”, que se liga de forma total ao relacionamento “é-de-um” com a entidade “pizza”. A entidade “tipo\_pizza”, por sua vez, possui os atributos “id\_tipo”, que identifica unicamente, “descrição”, que é a descrição do tipo, e “nome\_tipo”, que é o seu nome. Essa entidade também faz uma disjunção do tipo “d” total, e cria 2 subclasses: “especial” e “tradicional”.
- h) A cardinalidade entre pizzaria e acompanhamentos foi decidida logicamente, pois cada pizzaria oferece vários acompanhamentos, resultando numa cardinalidade de 1 pizzaria para N acompanhamentos, pela relação “oferece”.

A entidade “acompanhamento” possui todos os atributos requisitados(nome, descrição, tipo e preço), além de id, que é a chave primária. Essa entidade, por sua vez, se relaciona com a classe acompa\_pedido, por meio do relacionamento “contem\_a”, sob a cardinalidade de 1 acompanhamento para N acompa\_pedidos. A entidade “acompa\_pedido” serve para armazenar a quantidade de cada acompanhamento incluso num pedido, e também possui seu próprio id(chave primária). Essa entidade, por sua vez, se relaciona com a entidade “pedido” por meio do relacionamento “possui\_pp”, seguindo uma cardinalidade de N acompa\_pedidos para 1 pedido, e assim, cada pedido pode ter vários acompanhamentos, e será armazenada a quantidade requisitada de cada um deles pelo consumidor faminto.

- i) Foi criada a entidade “consumidor\_faminto” como requisitado, com o atributo correspondente ao endereço de entrega.
- j) A cardinalidade decidida entre consumidor faminto e pedido foi que 1 consumidor faminto faz N pedidos, logicamente. Para isso, foi criada a entidade “pedido”, que contém todos os atributos requisitados(id(chave primária), a data e horário de efetuação do pedido, o atributo opcional de horário posterior para entrega e a quantidade de pessoas que irão consumir o pedido).
- k) No início, a cardinalidade entre pedido e pizza foi determinada como requisitado: 1 pedido para N pizzas, com um relacionamento ligando pizza diretamente com pedido. Mas por serem necessários vários atributos de preparação da pizza, foi criada uma entidade intermediária “preparação\_pedido”, que possui um id próprio, os atributos requisitados(opções de massa, borda e quantidade de molho), e um atributo

adicional “preço”, que representa o preço final pela preparação da pizza, que aumentaria conforme as modalidades requisitadas.

- l) Esse requisito foi explicado durante a explicação do requisito “h”. Foi feita uma entidade intermediária entre acompanhamento e pedido para que seja armazenada a quantidade de cada acompanhamento.
- m) Para isso, foi criada uma entidade fraca “ingrediente\_extra”, que se relaciona com pedido no relacionamento “possui\_extra”, sob cardinalidade de N ingredientes extras para 1 pedido.
- n) Todos os atributos requisitados foram adicionados à entidade “ingrediente\_extra”.
- o) O requisito do pedido de entretenimento foi explicado na letra “a”. Para que pudessem ser resgatadas as informações do pedido padrão, ele foi modelado como uma subclasse de “pedido”.
- p) O modelo permite que esse cálculo seja feito.
- q) Foi criada a entidade “animador”, que é subclasse da entidade “cliente”, e ela possui os atributos requisitados (nome artístico, biografia e preço por 30 minutos);



- r) A cardinalidade entre a entidade de entretenimento e a de animador, sob o relacionamento “atendimento”, foi estabelecida como requisitado: 1 animador para N pedidos de entretenimento, pois cada animador pode participar de vários pedidos desse tipo, mas cada pedido só pode ter um animador.
- s) Foi criado um relacionamento “trabalha” entre animador e pizzaria, com cardinalidade N para M, já que uma pizzaria pode ter vários animadores, e cada animador pode trabalhar em várias pizzarias. Esse relacionamento possui o atributo “disponibilidade”, que representa os dias da semana em que o animador trabalhará nessa pizzaria.

### 3) ESQUEMA RELACIONAL

Aqui mostra-se o modelo relacional resultante do mapeamento do modelo entidade-relacionamento, além da explicação das escolhas de mapeamento em seguida.

#### 3.1) MODELO RELACIONAL

CLIENTE(id\_cliente, nome\_cliente, data\_nasc, endereco);

DONO\_DE\_NEGOCIO(id\_dono(CLIENTE.id\_cliente), linkedin);

PIZZARIA(id\_pizzaria, id\_dono(DONO\_DE\_NEGOCIO.id\_dono), nome, cep, endereco, web\_site, abertura, fechamento);

TELEFONE(pizzaria(PIZZARIA.id\_pizzaria), telefone);

PIZZA(pizzaria(PIZZARIA.id\_pizzaria), nome\_pizza, preco);

TIPO\_PIZZA(id\_tipo, pizzaria(PIZZA.pizzaria), nome\_pizza(PIZZA.nome\_pizza),  
nome\_tipo,descricao);

TRADICIONAL(tipo(TIPO\_PIZZA.id\_tipo));

ESPECIAL(tipo(TIPO\_PIZZA.id\_tipo));

ACOMPANHAMENTO(id\_acompa, pizzaria(PIZZARIA.id\_pizzaria), nome\_acompa,  
descricao, tipo\_acompa, preco\_acompa);

CONSUMIDOR\_FAMINTO(cliente(CLIENTE.id\_cliente), endereco\_entrega);

PEDIDO(id\_pedido, cliente(CONSUMIDOR\_FAMINTO.cliente), qtd\_pessoas,  
horario\_entrega, custo\_total, data\_pedido, horario\_pedido);

ACOMPA\_PEDIDO(id\_acompa\_pedido,  
id\_acompa(ACOMPANHAMENTO.id\_acompa), pedido(PEDIDO.id\_pedido),  
qtd\_acomp);

PREPARACAO\_PEDIDO(id\_preparacao, pedido(PEDIDO.id\_pedido),  
pizzaria(PIZZA.pizzaria), nome\_pizza(PIZZA.nome\_pizza ), tipo\_massa, borda,  
qtd\_molho,preco);

INGREDIENTE\_EXTRA(pedido(PEDIDO.id\_pedido), id\_extra, nome\_ingre,  
preco\_ingre);

PEDIDO\_COMBOS(pedido(PEDIDO.id\_pedido));

COMBOS(id\_combo, nome\_combo,preco)

TEM\_COMBO(combos(COMBOS.id\_combo), pedido(PEDIDO\_COMBOS.pedido),  
qtd\_combo)

ANIMADOR(id\_animador(CLIENTE.id\_cliente), nome\_artistico, biografia,  
preco\_30m);

TRABALHA(id\_animador(ANIMADOR.id\_animador), pizzaria(PIZZARIA.id\_pizzaria),  
disponibilidade)

PIZZA\_COMBO(combo(COMBOS.id\_combo), pizzaria(PIZZARIA.id\_pizzaria),  
nome\_pizza(PIZZA.nome\_pizza));

RETIRADA(pedido(PEDIDO.id\_pedido), horario\_retirada);

ENTRETENIMENTO(pedido(PEDIDO.id\_pedido),  
id\_animador(ANIMADOR.id\_animador), duracao, tipo\_entreter);

### 3.2) ESCOLHAS DE MAPEAMENTO E JUSTIFICATIVAS

Existe apenas um relacionamento de cardinalidade **1:1**, que é o relacionamento "**pizza é-de-um tipo\_pizza**". Optamos pela opção de repetir a chave primária de pizza na tabela tipo\_pizza (como chave estrangeira), por não haver necessidade de adicionar mais atributos na tabela de pizza.

Nos relacionamentos de cardinalidade **1:N**, em todos eles a abordagem foi a mesma: considerando cada relacionamento **E1(1) <-> E2(N)**, cada **E1** foi mapeada com os seus atributos normais apenas, e **E2** foi mapeada com uma ou mais chaves estrangeiras referenciando a chave primária de **E1**, além de seus atributos normais. Essa abordagem foi escolhida para evitar criação excessiva e desnecessária de tabelas (relações). Caso a abordagem que cria uma tabela extra para cada relacionamento fosse usada, seria necessário criar quase 10 tabelas a mais para o

modelo, o que aumentaria a complexidade de implementação e compreensão do projeto.

Existe apenas um atributo multivalorado, que é o atributo **telefones** da entidade **pizzaria**. Escolhemos fazer uma tabela própria para ele(que tem uma chave estrangeira referenciando o id da pizzaria e mais o telefone em si), pois entendemos que uma pizzaria pode ter vários números de telefone, e não quisemos impor um limite máximo de telefones com a outra opção.

Sobre o mapeamento das três disjunções do **MER**(CLIENTE, PEDIDO e TIPO\_PIZZA), preferiu-se usar a **opção “A”** de mapeamento, que faz tabelas para cada subclasse, e cada uma delas possui os atributos específicos da subclasse e a chave estrangeira que referencia a tabela da superclasse com os atributos restantes. Essa opção foi escolhida por ser a mais intuitiva e fácil de mapear, além de deixar o MR mais limpo. A **opção “B”** poderia ser usada para as disjunções de cliente e tipo\_pizza(por serem de participação total), mas preferimos mapear as três disjunções da mesma forma, pois, como já foi mencionado, é a mais intuitiva e fácil de mapear, e serve para todos os casos.

Por não haver tipo-relacionamento ternário e nem agregação no MER, não há mais nenhuma parte do mapeamento com mais de uma opção ensinada em aula. Esclarece-se que **todas as outras partes do modelo foram mapeadas adequadamente, seguindo o que foi ensinado na disciplina.**

#### 4) CRIAÇÃO DE TABELAS EM SQL

```
CREATE TABLE CLIENTE (  
    id_cliente smallint NOT NULL,  
    nome_cliente varchar(100) NOT NULL,  
    data_nasc date NOT NULL,  
    endereco varchar(200) NOT NULL,  
    CONSTRAINT clientepk PRIMARY KEY (id_cliente)
```

);

```
CREATE TABLE DONO_DE_NEGOCIO (  
    id_dono smallint NOT NULL,  
    linkedin varchar(100) NOT NULL,  
    CONSTRAINT donopk PRIMARY KEY (id_dono),  
    CONSTRAINT donofk FOREIGN KEY (id_dono) REFERENCES  
    CLIENTE(id_cliente)  
);
```

```
CREATE TABLE PIZZARIA (  
    id_pizzaria smallint NOT NULL,  
    id_dono smallint NOT NULL,  
    nome varchar(100) NOT NULL,  
    cep varchar(15) NOT NULL,  
    endereco varchar(200) NOT NULL,  
    web_site varchar(100) NOT NULL,  
    abertura time NOT NULL,  
    fechamento time NOT NULL,  
    CONSTRAINT pizzariapk PRIMARY KEY (id_pizzaria),  
    CONSTRAINT pizzariafk FOREIGN KEY (id_dono) REFERENCES  
    DONO_DE_NEGOCIO(id_dono)  
);
```

```
CREATE TABLE TELEFONE (  
    pizzaria smallint NOT NULL,  
    telefone varchar(15) NOT NULL,  
    CONSTRAINT telefonefk FOREIGN KEY (pizzaria) REFERENCES  
    PIZZARIA(id_pizzaria)
```

);

CREATE TABLE PIZZA (

pizzaria smallint NOT NULL,

nome\_pizza varchar(50) NOT NULL,

preco float NOT NULL,

CONSTRAINT pizzapk PRIMARY KEY (pizzaria,nome\_pizza),

CONSTRAINT pizzafk FOREIGN KEY (pizzaria) REFERENCES  
PIZZARIA(id\_pizzaria)

);

CREATE TABLE TIPO\_PIZZA (

id\_tipo smallint NOT NULL,

pizzaria smallint NOT NULL,

descricao varchar(100) NOT NULL,

nome\_pizza varchar(100) NOT NULL,

nome\_tipo varchar(100) NOT NULL,

CONSTRAINT tipo\_pizzapk PRIMARY KEY (id\_tipo),

CONSTRAINT tipo\_pizzafk FOREIGN KEY (pizzaria ,nome\_pizza)  
REFERENCES PIZZA(pizzaria,nome\_pizza)

);

CREATE TABLE TRADICIONAL (

tipo smallint NOT NULL,

CONSTRAINT tradiconalpk PRIMARY KEY (tipo),

CONSTRAINT tradiconalfk FOREIGN KEY (tipo) REFERENCES  
TIPO\_PIZZA(id\_tipo)

);

```
CREATE TABLE ESPECIAL (  
    tipo smallint NOT NULL,  
    CONSTRAINT especialpk PRIMARY KEY (tipo),  
    CONSTRAINT especialfk FOREIGN KEY (tipo) REFERENCES  
    TIPO_PIZZA(id_tipo)  
);
```

```
CREATE TABLE ACOMPANHAMENTO (  
    id_acompa smallint NOT NULL,  
    pizzaria smallint NOT NULL,  
    nome_acompa varchar(50) NOT NULL,  
    descricao varchar(200) NOT NULL,  
    tipo_acompa varchar(100) NOT NULL,  
    preco_acompa float NOT NULL,  
    CONSTRAINT acompanhamentopk PRIMARY KEY (id_acompa),  
    CONSTRAINT acompanhamentofk FOREIGN KEY (pizzaria) REFERENCES  
    PIZZARIA(id_pizzaria)  
);
```

```
CREATE TABLE CONSUMIDOR_FAMINTO (  
    cliente smallint NOT NULL,  
    endereco_entrega varchar(100),  
    CONSTRAINT consumidor_famintopk PRIMARY KEY (cliente),  
    CONSTRAINT consumidor_famintofk FOREIGN KEY (cliente) REFERENCES  
    CLIENTE(id_cliente)  
);
```

```
CREATE TABLE PEDIDO (  
    id_pedido smallint NOT NULL,
```

```
    cliente smallint NOT NULL,  
    qtd_pessoas int2 DEFAULT 1,  
    horario_entrega time,  
    custo_total float NOT NULL,  
    data_pedido date NOT NULL,  
    horario_pedido time NOT NULL,  
    CONSTRAINT pedidopk PRIMARY KEY (id_pedido),  
    CONSTRAINT pedidofk FOREIGN KEY (cliente) REFERENCES  
CONSUMIDOR_FAMINTO(cliente)  
);
```

```
CREATE TABLE ACOMP_ACOMP_PEDIDO (  
    id_acomp_pedido smallint NOT NULL,  
    id_acomp smallint NOT NULL,  
    pedido smallint NOT NULL,  
    qtd_acomp int2 NOT NULL,  
    CONSTRAINT acomp_pedidopk PRIMARY KEY (id_acomp_pedido),  
    CONSTRAINT acomp_pedidofk FOREIGN KEY (id_acomp)  
REFERENCES ACOMPANHAMENTO(id_acomp),  
    CONSTRAINT acomp_pedidofk2 FOREIGN KEY (pedido) REFERENCES  
PEDIDO(id_pedido)  
);
```

```
CREATE TABLE PREPARACAO_PEDIDO (  
    id_preparacao smallint NOT NULL,  
    pedido smallint NOT NULL,  
    pizzeria smallint NOT NULL,  
    nome_pizza varchar(50) NOT NULL,  
    tipo_massa varchar(50) NOT NULL,
```



```

        borda varchar(50) NOT NULL,
qtd_molho varchar(50) NOT NULL,
        preco float NOT NULL,
        CONSTRAINT preparacao_pedidopk PRIMARY KEY (id_preparacao),
        CONSTRAINT preparacao_pedidofk FOREIGN KEY (pedido) REFERENCES
PEDIDO(id_pedido),
        CONSTRAINT preparacao_pedidofk2 FOREIGN KEY (pizzaria ,nome_pizza)
REFERENCES PIZZA(pizzaria,nome_pizza)
);

```

```

CREATE TABLE INGREDIENTE_EXTRA (
        pedido smallint NOT NULL,
        nome_ingre varchar(50) NOT NULL,
        preco_ingre float NOT NULL,
        id_extra smallint NOT NULL,
        CONSTRAINT ingrediente_extrapk PRIMARY KEY (pedido,id_extra),
        CONSTRAINT ingrediente_extrafk FOREIGN KEY (pedido) REFERENCES
PEDIDO(id_pedido)
);

```

```

CREATE TABLE PEDIDO_COMBOS (
        pedido smallint NOT NULL,
        CONSTRAINT pedido_combospk PRIMARY KEY (pedido),
        CONSTRAINT pedido_combosfk FOREIGN KEY (pedido) REFERENCES
PEDIDO(id_pedido)
);

```

```

CREATE TABLE COMBOS (
        id_combo smallint NOT NULL,
        nome_combo varchar(50),

```

```
    preco float NOT NULL,  
    CONSTRAINT combospk PRIMARY KEY (id_combo)  
);
```

```
CREATE TABLE TEM_COMBO (  
    combos smallint NOT NULL,  
    pedido smallint NOT NULL,  
    qtd_combo int2 NOT NULL,  
    CONSTRAINT tem_combopk PRIMARY KEY (combos,pedido),  
    CONSTRAINT tem_combofk FOREIGN KEY (combos) REFERENCES  
COMBOS(id_combo),  
    CONSTRAINT tem_combofk2 FOREIGN KEY (pedido) REFERENCES  
PEDIDO_COMBOS(pedido)  
);
```

```
CREATE TABLE PIZZA_COMBO (  
    combo smallint NOT NULL,  
    pizzeria smallint NOT NULL,  
    nome_pizza varchar(50) NOT NULL,  
    CONSTRAINT pizza_combopk PRIMARY KEY (combo,pizzeria,nome_pizza),  
    CONSTRAINT pizza_combofk FOREIGN KEY (combo) REFERENCES  
COMBOS(id_combo),  
    CONSTRAINT pizza_combofk2 FOREIGN KEY (pizzeria, nome_pizza)  
REFERENCES PIZZA(pizzeria ,nome_pizza)  
);
```

```
CREATE TABLE ANIMADOR (  
    id_animador smallint NOT NULL,  
    nome_artistico varchar(50) NOT NULL,
```

```
    biografia varchar(100) NOT NULL,  
    preco_30m float NOT NULL,  
    CONSTRAINT animadorpk PRIMARY KEY (id_animador),  
    CONSTRAINT animadorfk FOREIGN KEY (id_animador) REFERENCES  
    CLIENTE(id_cliente)  
);
```

```
CREATE TABLE RETIRADA (  
    pedido smallint NOT NULL,  
    horario_retirada time NOT NULL,  
    CONSTRAINT retiradapk PRIMARY KEY (pedido),  
    CONSTRAINT retiradafk FOREIGN KEY (pedido) REFERENCES  
    PEDIDO(id_pedido)  
);
```

```
CREATE TABLE ENTRETENIMENTO (  
    pedido smallint NOT NULL,  
    id_animador smallint NOT NULL,  
    duracao int2 NOT NULL,  
    tipo_entreteri varchar(50) NOT NULL,  
    CONSTRAINT entretenimentopk PRIMARY KEY (pedido),  
    CONSTRAINT entretenimentofk FOREIGN KEY (pedido) REFERENCES  
    PEDIDO(id_pedido),  
    CONSTRAINT entretenimentofk2 FOREIGN KEY (id_animador)  
    REFERENCES ANIMADOR(id_animador)  
);
```

```
CREATE TABLE TRABALHA (  
    id_animador smallint NOT NULL,  
    pizzeria smallint NOT NULL,
```

```

    disponibilidade varchar(50),
    CONSTRAINT trabalhapk PRIMARY KEY (id_animador,pizzaria),
    CONSTRAINT trabalhafk FOREIGN KEY (id_animador) REFERENCES
    ANIMADOR(id_animador),
    CONSTRAINT trabalhafk2 FOREIGN KEY (pizzaria) REFERENCES
    PIZZARIA(id_pizzaria)
);

```

## 5) ESPECIFICAÇÃO DE CONSULTAS EM SQL

Aqui mostra-se as consultas complexas e as inserções de tuplas nas tabelas do banco de dados.

### 5.1) CONSULTAS

1)O dono do aplicativo quer fazer um estudo de mercado e valores, e quer saber quantos pedidos por cliente foram abaixo de 40 reais e qual o valor total desses pedidos por cliente.

```

SELECT cl.nome_cliente as cliente, count(pd.id_pedido), sum(pd.custo_total) FROM
cliente cl, pedido pd, consumidor_faminto cf

```

```

WHERE cl.id_cliente = pd.cliente and custo_total < 40.00 and cf.cliente =
cl.id_cliente

```

```

GROUP BY cl.nome_cliente

```

#### **Resultado:**

"Camila Antônia"	"2"	"62.4"
------------------	-----	--------

"Vitor Daniel"	"3"	"96.4"
----------------	-----	--------

"João Guedes"	"1"	"34.9"
---------------	-----	--------

"Ramon Orlinda"	"1"	"34.9"
"Raul Seixas"	"1"	"34.9"
"Camila Ribeiro"	"2"	"68.5"
"Carlos Flowers"	"1"	"34.9"
"Tietê Elise"	"1"	"34.9"
"Haroldo Felipe"	"2"	"69.8"
"Maria Bonita"	"2"	"69.8"

2) Um investidor precisa saber o nome de todas as pizzarias do dono de negocio 'Luiz Felipe' em ordem alfabética.

```
SELECT piz.nome FROM pizzeria piz , dono_de_negocio dn , cliente cl
WHERE cl.nome_cliente = 'Luiz Felipe' and cl.id_cliente = dn.id_dono and
dn.id_dono = piz.id_dono ORDER BY piz.nome
```

### **Resultado:**

"Pizzaria Sabor da itália"

"Pizzaria São Jorge"

3) O dono do aplicativo precisa saber os pedidos do cliente 'Vitor Daniel' e valores, entre os meses 11/2019 e 12/2019 em order decrescente.

```
SELECT pe.id_pedido , pe.custo_total FROM pedido pe, cliente cl ,
CONSUMIDOR_FAMINTO cf
```

```
WHERE pe.data_pedido between '2019-11-01' and '2019-12-31' and  
cl.nome_cliente='Vitor Daniel'
```

```
and cl.id_cliente = cf.cliente and cf.cliente = pe.cliente ORDER BY pe.id_pedido  
desc
```

**Resultado:**

18     "34.9"

7       "54.9"

4       "22.6"

4) Quais são os tipos de pizza e as pizzas existentes nas pizzarias de cada dono de negócio, sem repetições?

```
SELECT cl.nome_cliente as dono, string_agg(distinct ti.nome_tipo, ', ') as tipos,  
string_agg(distinct pi.nome_pizza, ', ') as pizzas
```

```
FROM pizzeria piz, pizza pi , tipo_pizza ti, cliente cl, dono_de_negocio dn
```

```
WHERE cl.id_cliente = dn.id_dono and dn.id_dono=piz.id_dono and piz.id_pizzaria =  
pi.pizzaria
```

```
and pi.pizzaria = ti.pizzaria and ti.nome_pizza = pi.nome_pizza
```

```
GROUP BY cl.nome_cliente
```

```
ORDER BY dono asc
```

**Resultado:**

"Amélia Ribeiro"     "especial-salgada, tradicional-doce, tradicional-salgada"

"bacon-m, espanhola-g, lombo-canadense-g, prestígio-m"

"Camila Cabello" "especial-salgada, tradicional-doce, tradicional-salgada"

"banana-g, brasileira-g, marguerita-g, mineira-g"

"João Foguetes" "especial-salgada, tradicional-salgada" "muçarela-g,  
presunto-p"

"Luiz Felipe" "especial-salgada, tradicional-salgada" "atum-g, calabresa-m,  
marguerita-g, quatro-queijos-g"

"Maria Felipe" "especial-salgada, tradicional-salgada" "marguerita-p,  
napolitana-g"

"Ramona Flowers" "especial-salgada, tradicional-salgada" "carne-seca-g,  
escandinava-m"

"Vitor Teixeira" "especial-salgada, tradicional-salgada" "alemã-g,  
frango&catupiry-g"

5)O aplicativo tem uma política que gera cupons de desconto para os clientes que mais fazem pedidos no aplicativo e que mais gastam, com prioridade em quem faz mais pedidos, e desde que o cliente tenha um gasto total de no mínimo 50 reais no aplicativo. Para isso, eles fazem a consulta a seguir periodicamente e enviam os cupons de vez em quando os clientes que seguirem os requisitos.

```
SELECT cl.nome_cliente as cliente, count(pe.id_pedido) as pedidos,  
sum(pe.custo_total) as gasto_total FROM pedido pe ,
```

```
cliente cl , consumidor_faminto cf
```

```
WHERE cl.id_cliente = cf.cliente and cf.cliente= pe.cliente
```

```
group by cl.nome_cliente having sum(pe.custo_total) > 50.00
```

```
order by pedidos desc, gasto_total desc
```

## Resultado

"Vitor Daniel"	"4"	"151.3"
"Carlos Flowers"	"3"	"159.6"
"Camila Antônia"	"3"	"106.4"
"Haroldo Felipe"	"2"	"69.8"
"Maria Bonita"	"2"	"69.8"
"Camila Ribeiro"	"2"	"68.5"

6) Quais pedidos tem acompanhamento da 'Pizzaria São Jorge', e quantos acompanhamentos foram pedidos em cada?

```
SELECT ap.pedido, sum(ap.qtd_acomp) as quantidade
```

```
FROM pizzaria pi , acompa_pedido ap , acompanhamento ac
```

```
WHERE pi.nome='Pizzaria São Jorge' and pi.id_pizzaria=ac.pizzaria and  
ap.id_acomp=ac.id_acomp
```

```
Group by ap.pedido
```

**Resultado:**

2	"4"
3	"4"
5	"2"
7	"6"
9	"2"
10	"2"



7) Quantos pedidos de entretenimento foram feitos por cada cliente(que ja fez pelo menos um pedido de entretenimento)?

```
SELECT cl.nome_cliente, count(ent.pedido) as pedidos from cliente cl,  
entretenimento ent, pedido pd, consumidor_faminto cf
```

```
where cl.id_cliente = cf.cliente and pd.cliente = cf.cliente and ent.pedido =  
pd.id_pedido
```

```
group by cl.nome_cliente
```

```
order by pedidos desc
```

**Resultado:**

"Camila Antônia" "3"

"Vitor Daniel" "2"

"Carlos Flowers" "2"

"Raul Seixas" "1"

"Tietê Elise" "1"

"Haroldo Felipe" "1"

8) Quais pizzarias oferecem acompanhamentos, quantos acompanhamentos e qual a média de preço deles?

```
SELECT pi.nome as pizzeria, count(ac.id_acompa) as acompanhamentos,  
avg(ac.preco_acompa) as media
```

FROM pizzeria pi, acompanhamento ac WHERE pi.id\_pizzaria = ac.pizzaria

group by pi.nome

**Resultado:**

"Pizzaria Sabor da Itália" "1" "3"

"Pizzaria Santa Luzia" "2" "2.25"

"Pizzaria Orlanda" "2" "5.2"

"Pizzaria São Jorge" "3" "2.1333333333333333"

"Pizzaria Parque das Américas" "2" "4.95"

9)Qual o horário médio que cada cliente faz pedidos no aplicativo.

SELECT cl.nome\_cliente as cliente, avg(pe.horario\_pedido) FROM cliente cl ,  
pedido pe , consumidor\_faminto cf

WHERE cl.id\_cliente = cf.cliente and cf.cliente = pe.cliente

group by cl.nome\_cliente

**Resultado:**

"Camila Antônia" "18:47:20"

"Vitor Daniel" "12:45:45"

"João Guedes" "06:30:00"

"Ramon Orlanda" "10:30:00"

"Raul Seixas" "19:00:00"

"Camila Ribeiro" "11:27:30"

"Carlos Flowers" "15:58:20"

"Tietê Elise" "18:00:00"

"Haroldo Felipe" "14:00:00"

"Maria Bonita" "14:30:00"

10) Quais pizzarias possuem animadores trabalhando nelas nas segundas, e quantos por pizzeria?

```
SELECT pi.nome, count(ani.id_animador) FROM animador ani , trabalha tr , pizzeria pi
```

```
WHERE ani.id_animador = tr.id_animador and disponibilidade like '%segunda%'
```

```
and tr.pizzaria=pi.id_pizzaria
```

```
GROUP BY pi.nome
```

### **Resultado:**

"Pizzaria do zé" "1"

"Pizzaria Jardim Ipanema" "1"

"Pizzaria orlanda" "1"

"Pizzaria Santa Luzia" "1"

"Pizzaria São Jorge" "2"

## 5.2) OPERAÇÕES DE INSERÇÃO

INSERT INTO CLIENTE VALUES (1, 'Vitor Teixeira', '03/07/2000', 'Rua Bergamo, 1840, Jardim Europa'),

(2, 'Luiz Felipe', '04/11/1987', 'Rua Manoel Machado Filho, 206, Chácaras Tubalina e Quartel'),

(3, 'Amélia Ribeiro', '09/12/1999', 'Rua Siomara Carla de Jesus, 764, Shopping Park'),

(4, 'Camila Cabello', '15/03/2003', 'Rua Bergamo, 190, Jardim Europa'),

(5, 'Bartolomeu Seixas', '23/12/2001', 'Praça dos Araújo, 2043, Mansões Aeroporto'),

(6, 'João Foguetes', '24/07/1995', 'Rua do Estivador, 203, Jardim das Palmeiras'),

(7, 'Maria Felipe', '12/12/2000', 'Rua Francino Vieira de Barros, 1530, Tocantins'),

(8, 'Ramon Seixas', '12/11/2001', 'Rua Bergamo, 1856, Jardim Europa'),

(9, 'Ramona Flowers', '03/04/2005', 'Praça dos Araújo, 1989, Mansões Aeroporto'),

(10, 'Madalena Elise', '07/08/1982', 'Rua Dirceu Alves Pereira, 1326, Chácaras Tubalina e Quartel'),

(11, 'Vitor Daniel', '03/07/2000', 'Rua Bergamo, 1832, Jardim Europa'),

(12, 'Haroldo Felipe', '04/11/1987', 'Rua Manoel Machado Filho, 236, Chácaras Tubalina e Quartel'),

(13, 'Camila Ribeiro', '09/12/1999', 'Rua Siomara Carla de Jesus, 564, Shopping Park'),

(14, 'Camila Antônia', '15/03/2003', 'Rua Bergamo, 208, Jardim Europa'),

(15, 'Raul Seixas', '23/12/2001', 'Praça dos Araújo, 243, Mansões Aeroporto'),

(16, 'João Guedes', '24/07/1995', 'Rua do Estivador, 303, Jardim das Palmeiras'),

(17, 'Maria Bonita', '12/12/2000', 'Rua Francino Vieira de Barros, 1670, Tocantins'),

(18, 'Ramon Orlinda', '12/11/2001', 'Rua Bergamo, 1976, Jardim Europa'),

(19, 'Carlos Flowers', '03/04/2005', 'Praça dos Araújo's, 2349, Mansões Aeroporto'),

(20, 'Tietê Elise', '07/08/1982', 'Rua Dirceu Alves Pereira, 1674, Chácara's Tubalina e Quartel'),

(21, 'Viktor Dmitri', '03/07/2000', 'Rua Bergamo, 2354, Jardim Europa'),

(22, 'Luiz Fernando', '04/11/1987', 'Rua Manoel Machado Filho, 386, Chácara's Tubalina e Quartel'),

(23, 'Armando Ribeiro', '09/12/1999', 'Rua Siomara Carla de Jesus, 984, Shopping Park'),

(24, 'Ortízio Cabello', '15/03/2003', 'Rua Bergamo, 540, Jardim Europa'),

(25, 'Eliseu Seixas', '23/12/2001', 'Praça dos Araújo's, 2865, Mansões Aeroporto'),

(26, 'João Menezes', '24/07/1995', 'Rua do Estivador, 1233, Jardim das Palmeiras'),

(27, 'Humberto Felipe', '12/12/2000', 'Rua Francino Vieira de Barros, 2430, Tocantins'),

(28, 'Gabriel Seixas', '12/11/2001', 'Rua Bergamo, 3256, Jardim Europa'),

(29, 'Ramona Elise', '03/04/2005', 'Praça dos Araújo's, 1629, Mansões Aeroporto'),

(30, 'Madalena Manoela', '07/08/1982', 'Rua Dirceu Alves Pereira, 2126, Chácara's Tubalina e Quartel');

INSERT INTO DONO\_DE\_NEGOCIO VALUES (1, 'teixeira-eof'),

(2, 'felipe-lu206'),

(3, 'rib-amelia99'),  
(4, 'hairr-camila'),  
(5, 'six-bart243'),  
(6, 'firejohn203'),  
(7, 'phil-mary'),  
(8, 'six-ramon001'),  
(9, 'flores-ram'),  
(10, 'madelyne-her');

INSERT INTO PIZZARIA VALUES (1, 2, 'Pizzaria São Jorge', '38410-190', 'Rua Francisco Basílio Neto, 206, São Jorge', 'batepapopizza.com', '18:00', '02:00'),

(2, 7, 'Pizzaria Santa Luzia', '38408-784', 'Rua Cornélio Arantes, 340, Santa Luzia', 'pizzaboa.com', '20:00', '00:00'),

(3, 4, 'Pizzaria Jardim Ipanema', '38406-265', 'Rua Adeostia Amélia Pereira, 1286, Jardim Ipanema', 'martepizza.com', '06:00', '00:00'),

(4, 3, 'Pizzaria orlanda', '38406-265', 'Rua Adeostia Amélia Pereira, 347, Jardim Ipanema', 'venuspizza.com', '06:00', '02:00'),

(5, 1, 'Pizzaria Portuguesa', '38406-784', 'Rua Cornélio Arantes, 293, Santa Luzia', 'pizzatop.com', '06:00', '00:00'),

(6, 2, 'Pizzaria Sabor da Itália', '38400-603', 'Travessa Professor Macedo, 874, Nossa Senhora Aparecida', 'batepapopizza.com', '18:00', '02:00'),

(7, 6, 'Pizzaria Bela Vista', '38412-506', 'Rua Nupotira, 113, Bela Vista', 'loucapizza.com', '20:00', '00:00'),

(8, 4, 'Pizzaria Parque das Américas', '38412-796', 'Rua Guatemala, 512, Parque das Américas', 'martepizza.com', '06:00', '00:00'),

(9, 3, 'Pizzaria Dona Florinda', '38412-796', 'Rua Guatemala, 528, Parque das Américas', 'venuspizza.com', '06:00', '02:00'),

(10, 9, 'Pizzaria do zé', '38410235', 'Rua Julieta Ferreira de Lima, 773, Laranjeiras', 'pizzamara.com', '06:00', '00:00');

INSERT INTO TELEFONE VALUES (1, '(34)3752-3638'),

(2, '(34)7849-0828'),  
(3, '(34)6805-5516'),  
(4, '(34)2730-5812'),  
(5, '(34)1059-5832'),  
(6, '(34)5248-8838'),  
(7, '(34)1062-0495'),  
(8, '(34)2737-3334'),  
(9, '(34)6060-1800'),  
(10, '(34)6519-2723');

INSERT INTO PIZZA VALUES (1, 'marguerita-g', 32.90),

(1, 'calabresa-m', 20.50),  
(2, 'napolitana-g', 38.99),  
(2, 'marguerita-p', 18.00),  
(3, 'brasileira-g', 28.90),  
(3, 'marguerita-g', 44.90),  
(4, 'espanhola-g', 42.90),  
(4, 'bacon-m', 29.90),  
(5, 'frango&catupiry-g', 32.90),  
(5, 'alemã-g', 32.90),  
(6, 'atum-g', 32.90),  
(6, 'quatro-queijos-g', 32.90),  
(7, 'muçarela-g', 32.90),  
(7, 'presunto-p', 32.90),  
(8, 'banana-g', 32.90),  
(8, 'mineira-g', 32.90),  
(9, 'prestigio-m', 32.90),  
(9, 'lombo-canadense-g', 32.90),  
(10, 'escandinava-m', 32.90),

(10, 'carne-seca-g', 32.90);

```
INSERT INTO TIPO_PIZZA VALUES (1, 1, 'pizza salgada tradicional', 'marguerita-g',
'tradicional-salgada'),
(2, 1, 'pizza salgada especial', 'calabresa-m', 'especial-salgada'),
(3, 2, 'pizza salgada tradicional', 'napolitana-g',
'tradicional-salgada'),
(4, 2, 'pizza salgada especial', 'marguerita-p', 'especial-salgada'),
(5, 3, 'pizza salgada tradicional', 'brasileira-g',
'tradicional-salgada'),
(6, 3, 'pizza salgada especial', 'marguerita-g', 'especial-salgada'),
(7, 4, 'pizza salgada tradicional', 'espanhola-g',
'tradicional-salgada'),
(8, 4, 'pizza salgada especial', 'bacon-m', 'especial-salgada'),
(9, 5, 'pizza salgada tradicional', 'frango&catupiry-g',
'tradicional-salgada'),
(10, 5, 'pizza salgada especial', 'alemã-g', 'especial-salgada'),
(11, 6, 'pizza salgada tradicional', 'atum-g', 'tradicional-salgada'),
(12, 6, 'pizza salgada especial', 'quatro-queijos-g',
'especial-salgada'),
(13, 7, 'pizza salgada tradicional', 'muçarela-g',
'tradicional-salgada'),
(14, 7, 'pizza salgada especial', 'presunto-p', 'especial-salgada'),
(15, 8, 'pizza doce tradicional', 'banana-g', 'tradicional-doce'),
(16, 8, 'pizza salgada especial', 'mineira-g', 'especial-salgada'),
(17, 9, 'pizza doce tradicional', 'prestigio-m', 'tradicional-doce'),
(18, 9, 'pizza salgada especial', 'lombo-canadense-g',
'especial-salgada'),
(19, 10, 'pizza salgada tradicional', 'escandinava-m',
'tradicional-salgada'),
(20, 10, 'pizza salgada especial', 'carne-seca-g',
'especial-salgada');
```

INSERT INTO TRADICIONAL VALUES (1),

(3),

(5),

(7),

(9),



(11),  
(13),  
(15),  
(17),  
(19);

INSERT INTO ESPECIAL VALUES (2),

(4),  
(6),  
(8),  
(10),  
(12),  
(14),  
(16),  
(18),  
(20);

INSERT INTO ACOMPANHAMENTO VALUES (1, 1, 'molho-barbecue', 'molho  
barbecue para pizza', 'molho', 3.90),

(2, 1, 'bread-stick', 'pãozinho crocante para aperitivo',  
'salgadinho', 1.50),

(3, 1, 'coxinha', 'coxinha de frango', 'salgadinho', 1.00),

(4, 2, 'doce-leite-ninho', 'docinho de leite ninho', 'sobremesa',  
2.50),

(5, 2, 'brigadeiro', 'brigadeiro de sobremesa', 'sobremesa', 2.00),

(6, 4, 'refri lata', 'refrigerante lata de 350ml', 'bebida', 3.50),

(7, 6, 'croquete', 'salgadinho frito com recheio de queijo',  
'salgadinho', 3.00),

(8, 8, 'espetinho', 'espetinho de churrasco 5 pedaços',  
'aperitivo', 5.90),

(9, 8, 'doce de leite', 'doce de leite em barrinha', 'sobremesa',  
4.00),

(10, 4, 'salada de folhas', 'salada com couve, alface e outras  
folhas', 'salada', 6.90);

INSERT INTO CONSUMIDOR\_FAMINTO VALUES (11, 'Rua Bergamo, 1832, Jardim  
Europa'),

(12, 'Rua Manoel Machado Filho, 236, Chácaras Tubalina e  
Quartel'),

(13, 'Rua Siomara Carla de Jesus, 564, Shopping Park'),

(14, 'Rua Bergamo, 208, Jardim Europa'),

(15, 'Praça dos Araújo, 243, Mansões Aeroporto'),

(16, 'Rua do Estivador, 303, Jardim das Palmeiras'),

(17, 'Rua Francino Vieira de Barros, 1670, Tocantins'),

(18, 'Rua Bergamo, 1976, Jardim Europa'),

(19, 'Praça dos Araújo, 2349, Mansões Aeroporto'),

(20, 'Rua Dirceu Alves Pereira, 1674, Chácaras Tubalina e  
Quartel');

INSERT INTO PEDIDO VALUES (1, 11, 1, '18:30', 38.90, '03/09/2019', '17:00'),

(2, 14, 1, '20:00', 44.00, '04/09/2019', '16:07'),

(3, 19, 2, '22:30', 56.90, '23/10/2019', '20:40'),

(4, 11, 1, '08:30', 22.60, '07/11/2019', '6:03'),

(5, 13, 2, '18:00', 38.60, '08/11/2019', '16:00'),

(6, 14, 1, '21:30', 27.50, '08/11/2019', '19:30'),

(7, 11, 1, '19:45', 54.90, '09/11/2019', '16:30'),

(8, 19, 3, '18:30', 67.80, '15/11/2019', '17:00'),

(9, 13, 1, '12:00', 29.90, '25/11/2019', '6:55'),

(10, 20, 1, NULL, 34.90, '25/12/2019', '18:00'),

(11, 12, 1, '20:00', 34.90, '25/12/2019', '19:00'),

```

(12, 15, 1, '16:30', 34.90, '27/12/2019', '19:00'),
(13, 16, 1, NULL, 34.90, '28/12/2019', '06:30'),
(14, 12, 1, NULL, 34.90, '28/12/2019', '09:00'),
(15, 17, 1, NULL, 34.90, '28/12/2019', '10:00'),
(16, 19, 1, NULL, 34.90, '28/12/2019', '10:15'),
(17, 18, 1, NULL, 34.90, '28/12/2019', '10:30'),
(18, 11, 1, NULL, 34.90, '28/12/2019', '11:30'),
(19, 17, 1, NULL, 34.90, '29/12/2019', '19:00'),
(20, 14, 1, NULL, 34.90, '29/12/2019', '20:45');

```

```

INSERT INTO ACOMPA_PEDIDO VALUES (1,1, 2, 2),

```

```

(2,1, 2, 2),
(3,1, 3, 2),
(4,1, 3, 2),
(5,1, 5, 2),
(6,1, 7, 2),
(7,1, 7, 2),
(8,1, 7, 2),
(9,1, 9, 2),
(10,1, 10, 2);

```

```

INSERT INTO PREPARACAO_PEDIDO VALUES (1,1, 1, 'marguerita-g', 'fina',
'normal', 'pouco', 37.90 ),

```

```

(2,2, 1, 'calabresa-m', 'média', 'recheada com catupiry',
'extra',23.50),
(3,3, 2, 'napolitana-g', 'grossa', 'recheada com catupiry',
'extra',41.99),
(4,4, 2, 'marguerita-p','média', 'recheada com catupiry',
'pouco',23.00),
(5,5, 3, 'brasileira-g', 'fina', 'normal', 'extra',31.90),

```

(6,6, 3, 'marguerita-g', 'grossa', 'normal', 'pouco',30.90),  
(7,7, 4, 'espanhola-g','média', 'recheada com catupiry',  
'normal',47.90),  
(8,8, 4, 'bacon-m', 'fina', 'normal', 'pouco',30.90),  
(9,9, 5, 'frango&catupiry-g', 'grossa', 'recheada com catupiry',  
'normal',33.90),  
(10,10, 5, 'alemã-g', 'fina', 'normal', 'pouco',34.90);

INSERT INTO INGREDIENTE\_EXTRA VALUES (1, '10 fatias de calabresa', 6.50, 1),  
(2, 'cobertura de cheddar na borda', 4.90, 2),  
(3, 'frango desfiado', 5.75, 3),  
(4, 'frango desfiado', 5.75, 3),  
(5, 'cobertura de cheddar na borda', 4.90, 2),  
(6, '10 fatias de calabresa', 6.50, 1),  
(7, '6 ovos de codorna', 5.50, 4),  
(8, '5 camarões', 12.50, 5),  
(9, '6 ovos de codorna', 5.50, 4),  
(10, 'pedacinhos de abacaxi', 8.60, 6);

INSERT INTO PEDIDO\_COMBOS VALUES (1),  
(3),  
(5),  
(7),  
(9),  
(11),  
(13),  
(15),  
(17),

(19);

```
INSERT INTO COMBOS VALUES (1, 'marguerita-g + refri 2L',37.90),  
    (2, 'calabresa-m + 1 pizza doce especial m',37.50),  
    (3, 'napolitana-g + acompanhamento',42.99),  
    (4, 'marguerita-p + refri 2L',27.90),  
    (5, 'brasileira-g + refri 3L',33.90),  
    (6, 'marguerita-g + refri 600ml',35.90),  
    (7, 'espanhola-g + 2 refri 2L',47.90),  
    (8, 'bacon-m + 3 docinhos',31.50),  
    (9, 'frango&catupiry-g + suco de laranja natural 900ml',40.90),  
    (10, 'alemã-g + 5 breadsticks',40.00);
```

```
INSERT INTO TEM_COMBO VALUES (1, 1, 1),  
    (2, 3, 2),  
    (3, 5, 1),  
    (4, 7, 1),  
    (5, 9, 3),  
    (6, 11, 2),  
    (7, 13, 1),  
    (8, 15, 4),  
    (9, 17, 1),  
    (10, 19, 1);
```

```
INSERT INTO PIZZA_COMBO VALUES (1, 1, 'marguerita-g'),  
    (2, 1, 'calabresa-m'),  
    (3, 2, 'napolitana-g'),  
    (4, 2, 'marguerita-p'),  
    (5, 3, 'brasileira-g'),
```

(6, 3, 'marguerita-g'),  
(7, 4, 'espanhola-g'),  
(8, 4, 'bacon-m'),  
(9, 5, 'frango&catupiry-g'),  
(10, 5, 'alemã-g');

INSERT INTO ANIMADOR VALUES (21, 'O Russo', 'Imigrante russo, cuspo fogo com vodka', 15.60),

(22, 'Dom Fernand', 'Monarquista, herdeiro do trono do Brasil', 12.90),

(23, 'O Cobrador', 'Eu me finjo de agiota', 15.00),

(24, 'Ernesto Cabreiro', 'Nasci e cresci no circo, hoje te trago alegria', 18.60),

(25, 'Everest', 'Escalei o monte everest e hoje compartilho a história com você', 22.00),

(26, 'John Luck', 'Sou o homem mais sortudo do mundo', 10.30),

(27, 'Felipão', 'Me fantasio de técnico da seleção', 11.60),

(28, 'Sextador', 'Te trago a alegria da sexta-feira independente do dia', 19.90),

(29, 'Valquíria', 'Me fantasio de valquíria nórdica', 25.40),

(30, 'Madelyne Star', 'Sou cantora e dançarina formada pela', 45.60);

INSERT INTO RETIRADA VALUES (1, '18:30'),

(3, '22:30'),

(5, '19:45'),

(7, '18:30'),

(9, '18:30'),

(11, '18:30'),

(13, '18:30'),

(15, '18:30'),  
(17, '18:30'),  
(19, '18:30');

INSERT INTO ENTRETENIMENTO VALUES (2, 21, 2,'palhaço'),

(4, 27, 1,'magico'),  
(6, 22, 3,'cantor'),  
(8, 22, 4,'canto'),  
(10, 24, 7,'magico'),  
(12, 21, 8,'serenata'),  
(14, 27, 10,'serenata'),  
(16, 23, 2,'magico'),  
(18, 27, 1,'cantor'),  
(20, 26, 2,'palhaço');

INSERT INTO TRABALHA VALUES (21, 1, 'segunda e terça'),

(27, 2, 'segunda e quarta'),  
(22, 4, 'segunda e quinta'),  
(22, 7, 'terça e quarta'),  
(22, 8, 'sexta e sábado'),  
(24, 10, 'segunda à sexta'),  
(23, 3, 'segunda à sábado'),  
(26, 1, 'segunda à sexta'),  
(21, 2, 'quarta à sexta'),  
(27, 3, 'quarta e quinta');