

**Base de Dados**

**2019/2020**

**Cadeia de Supermercados**

Turma 3 – Grupo 7

201806582 – Diogo Guimarães do Rosário

201806451 – Gonçalo de Batalhão Alves

201806227 – Pedro Jorge Fonseca Seixas

Índice

[**Introdução** 2](#_Toc40880814)

[**Especificação da Base de Dados** 3](#_Toc40880815)

[**Localização** 3](#_Toc40880816)

[**Pessoa** 3](#_Toc40880817)

[**Supermercado** 3](#_Toc40880818)

[**Funcionário** 3](#_Toc40880819)

[**Cliente** 3](#_Toc40880820)

[**Horário** 3](#_Toc40880821)

[**Caixa** 3](#_Toc40880822)

[**Secção** 4](#_Toc40880823)

[**Compra** 4](#_Toc40880824)

[**Produto** 4](#_Toc40880825)

[**Esquema Inicial** 5](#_Toc40880826)

[**Esquema Relacional** 7](#_Toc40880827)

[**Dependências Funcionais e Análise de Forma Normal** 9](#_Toc40880828)

[**Restrições** 11](#_Toc40880829)

[**Queries** 14](#_Toc40880830)

[**Triggers** 15](#_Toc40880831)

**Introdução**

Para o desenvolvimento de um projeto no âmbito da unidade curricular “Base de Dados”, foi-nos permitido escolher um tema, de modo a criar e desenvolver uma base de dados.

Deste modo, escolhemos desenvolver uma base dados para uma cadeia de supermercados, pois além de nos parecer um tópico interessante, também seria útil na vida real.

Na nossa base de dados é possível aceder a informação sobre: um supermercado, um produto, uma caixa, uma compra, entre outros. Na página seguinte, é possível ver mais em detalhe como está organizada a nossa base de dados.

**Especificação da Base de Dados**

## **Localização**

Uma Localização tem um [**Supermercado**](#_Supermercado) e pode ter várias [**Pessoas**](#_Pessoa). A classe da Localização é constituída por: uma morada, uma localidade e um código-postal.

## **Pessoa**

Uma Pessoa é constituída por: um nome, uma data de nascimento, um número de telefone, um NIF e um género. A classe Pessoa é uma generalização de um [**Funcionário**](#_Funcionário) e de um [**Cliente**](#_Cliente).

## **Supermercado**

Um Supermercado tem um [**Horário**](#_Horário) de funcionamento, um diretor (um [**Funcionário**](#_Funcionário) específico), várias [**Caixas**](#_Caixa) e várias [**Secções**](#_Secção). A classe Supermercado é constituída por um nome.

## **Funcionário**

Um Funcionário tem um [**Horário**](#_Horário) associado a si, é responsável por uma [**Caixa**](#_Caixa) (podendo estar a trabalhar numa caixa manual ou não) e está associado a diferentes [**Secções**](#_Secção). A classe Funcionário, para além dos atributos que herda da classe [**Pessoa**](#_Pessoa), também é constituída por um salário.

## **Cliente**

Um Cliente pode fazer várias [**Compras**](#_Compra). A classe Cliente, para além dos atributos que herda da classe[**Pessoa**](#_Pessoa), também é constituída por um número de cliente.

## **Horário**

Cada [**Supermercado**](#_Supermercado), assim como cada [**Funcionário**](#_Funcionário), têm um Horário associados a si. A classe Horário é constituída por uma hora inicial, uma hora final.

## **Caixa**

Uma Caixa é constituída por um número e por uma variável “Aberta” que, tal como o nome indica, representa o estado de funcionamento desta. Além disso, cada Caixa está associada a uma [**Compra**](#_Compra). A classe Caixa é uma generalização de uma caixa **Automática** e de uma caixa **Manual**.

## **Secção**

Uma Secção é uma parte íntegra de um [**Supermercado**](#_Supermercado), onde trabalham diversos [**Funcionários**](#_Funcionário) e onde se encontram diversos [**Produtos**](#_Produto). A classe Secção é constituída por um nome.

## **Compra**

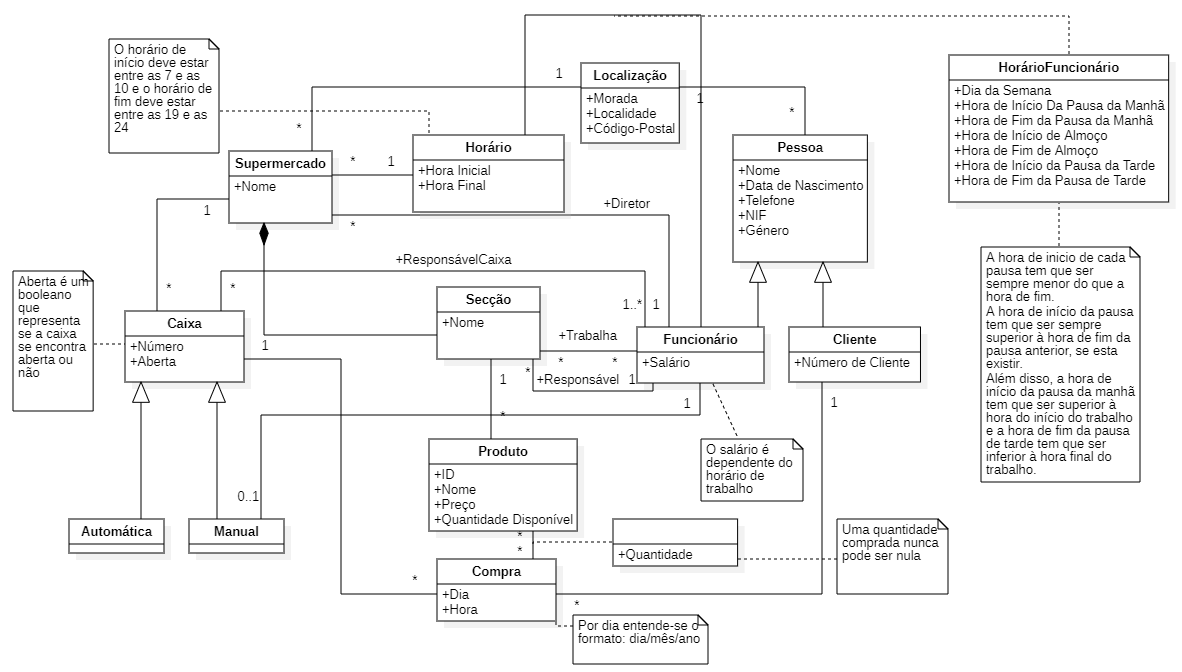
Uma Compra está associada a uma [**Caixa**](#_Caixa), a um [**Cliente**](#_Cliente) e a vários [**Produtos**](#_Produto), sendo possível saber a quantidade que foi comprada de cada um destes últimos. A classe Compra é constituída por um dia e uma hora.

## **Produto**

Um Produto está associado a uma [**Secção**](#_Secção) e a uma [**Compra**](#_Compra). A classe Produto é constituída por um ID, um preço e também pela quantidade disponível.

**Uma imagem com texto, mapa

Descrição gerada automaticamenteEsquema Inicial**

**Esquema Revisto**

**Esquema Relacional**

**Localização** (IDLocalização, Código-postal, Morada, Localidade)

IDLocalização (Chave Primária) → Código-postal, Morada, Localidade

Morada → Código-postal

Código-postal → Localidade

**Supermercado** (IDSupermercado, Nome, IDLocalização → Localização, IDHorário → Horário)

IDSupermercado (Chave Primária) → Nome, IDLocalização, IDHorário

IDLocalização e IDHorário são chaves estrangeiras

**Pessoa** (IDPessoa, NIF, Nome, Data de Nascimento, Telefone, Género, IDLocalização → Localização)

IDPessoa (Chave Primária) → NIF, Nome, Data de Nascimento, Telefone, Género, IDLocalização

NIF → IDPessoa

Nome, Data de Nascimento → IDPessoa

Telefone → IDPessoa

IDLocalização é uma chave estrangeira

**Funcionário** (IDPessoa → Pessoa, Salário, IDSupermercado → Supermercado, IDHorário → Horário)

IDPessoa (Chave Primária) → Salário, IDSupermercado, IDHorário

IDPessoa, IDSupermercado e IDHorário são chaves estrangeira

**Cliente** (IDPessoa → Pessoa, Número de Cliente)

IDPessoa (Chave Primária) → Número de Cliente

IDPessoa é uma chave estrangeira

**Horário** (IDHorário, Hora Inicial, Hora Final)

IDHorário (Chave Primária) → Hora Inicial, Hora Final

**HorárioFuncionário** (IDPessoa→ Funcionário, IDHorário → Horário, Dia da Semana, Hora de Início Da Pausa da Manhã, Hora de Fim da Pausa da Manhã, Hora de Início de Almoço, Hora de Fim de Almoço, Hora de Início da Pausa da Tarde, Hora de Fim da Pausa de Tarde)

IDPessoa, IDHorário (Chaves Primárias) → Dia da Semana, Hora de Início Da Pausa da Manhã, Hora de Fim da Pausa da Manhã, Hora de Início de Almoço, Hora de Fim de Almoço, Hora de Início da Pausa da Tarde, Hora de Fim da Pausa de Tarde

IDPessoa e IDHorário são chaves estrangeiras

**Caixa** (IDCaixa, Número, Aberta, IDSupermercado → Supermercado)

IDCaixa (Chave Primária) → Número, Aberta, IDSupermercado

IDSupermercado é uma chave estrangeira

**Automática** (IDCaixa → Caixa);

IDCaixa (Chave Primária) é uma chave estrangeira

**Manual** (IDCaixa → Caixa, IDPessoa→ Funcionário);

IDCaixa e IDPessoa (Chave Primárias) são chaves estrangeiras

**Secção** (IDSecção, Nome, IDSupermercado → Supermercado);

IDSecção (Chave Primária) → Nome, IDSupermercado

IDSupermercado é uma chave estrangeira

**Compra** (IDCompra, Dia, Hora, IDCliente → Cliente, IDCaixa -> Caixa)

IDCompra (Chave Primária) → Dia, Hora, IDCliente, IDCaixa

IDPessoa e IDCaixa são chaves estrangeiras

**Produto** (IDProduto, Nome, Preço, Quantidade Disponível, IDSecção → Secção);

IDProduto (Chave Primária) → Preço, Quantidade Disponível, IDSecção

IDSecção é uma chave estrangeira

**Quantidade** (IDProduto → Produto, IDCompra → Compra, Quantidade)

IDProduto, IDCompra (Chaves Primárias) → Quantidade

IDProduto e IDCompra são chaves estrangeiras

**ResponsávelCaixa** (IDCaixa → Caixa, IDPessoa → Funcionário)

IDCaixa e IDPessoa (Chaves Primárias) são chaves estrangeiras

**Trabalha** (IDPessoa→ Funcionário, IDSecção → Secção)

IDPessoa e IDSecção (Chaves Primárias) são chaves estrangeiras

**Responsável** (IDSecção → Caixa, IDPessoa → Funcionário)

IDSecção e IDPessoa (Chaves Primárias) são chaves estrangeiras

**Dependências Funcionais e Análise de Forma Normal**

Em cada relação descrita no ponto anterior, o lado esquerdo das dependências funcionais é a chave para a relação. Deste modo, o fecho dos atributos do lado esquerdo são todos os atributos da relação, como se pode ver de seguida:

**Localização**

{IDLocalização}⁺ = {IDLocalização, Código-postal, Morada, Localidade}

**Supermercado**

{IDSupermercado}⁺ = {IDSupermercado, Nome, IDLocalização, IDHorário}

**Pessoa**

{IDPessoa}⁺ = {IDPessoa, NIF, Nome, Data de Nascimento, Telefone, Género, IDLocalização}

**Funcionário**

{IDPessoa}⁺ = {IDPessoa, Salário, IDSupermercado, IDHorário, IDHorárioFuncionário}

**Cliente**

{IDPessoa }⁺ = {IDPessoa, Número de Cliente}

**Horário**

{IDHorário}⁺ = {IDHorário, Hora Inicial, Hora Final}

**HorárioFuncionário**

{IDPessoa, IDHorário}⁺ = {IDPessoa, IDHorário, Dia da Semana, Hora de Início Da Pausa da Manhã, Hora de Fim da Pausa da Manhã, Hora de Início de Almoço, Hora de Fim de Almoço, Hora de Início da Pausa da Tarde, Hora de Fim da Pausa de Tarde}

**Caixa**

{IDCaixa}⁺ = {IDCaixa, Número, Aberta, IDSupermercado}

**Automática**

{IDCaixa}⁺ = {IDCaixa}

**Manual**

{IDCaixa, IDPessoa}⁺ = {IDCaixa, IDPessoa}

**Secção**

{IDSecção}⁺ = {IDSecção, Nome, IDPessoa, IDSupermercado}

**Compra**

{IDCompra}⁺ = {IDCompra, Dia, Hora, IDPessoa, IDCaixa}

**Produto**

{IDProduto}⁺ = {IDProduto, Nome, Preço, Quantidade Disponível, IDSecção}

**Quantidade**

{IDProduto, IDCompra}⁺ = {IDProduto, IDCompra, Quantidade}

**ResponsávelCaixa**

{IDCaixa, IDPessoa}⁺ = {IDCaixa, IDPessoa}

**Trabalha**

{IDPessoa, IDSecção}⁺ = {IDPessoa, IDSecção}

**Responsável**

{IDSecção, IDPessoa}⁺ = {IDSecção, IDPessoa}

Deste modo, como, em cada relação, o lado esquerdo da dependência funcional é uma chave para essa relação, o modelo relacional já se encontra na **Forma Normal de Boyce-Codd** (não existem violações). E como esta forma é um subconjunto da **Terceira Forma Normal**, então também se encontra nesta forma.

# **Restrições**

**Localização**

IDLocalização é a chave primária (PRIMARY KEY);

O código-postal, a morada e a localidade têm que, obrigatoriamente, ter um valor (NOT NULL);

**Supermercado**

IDSupermercado é a chave primária (PRIMARY KEY);

O Nome tem que, obrigatoriamente, um valor (NOT NULL);

IDHorário e IDLocalização são chaves estrangeiras (integridade referencial, FOREIGN KEY) ;

**Pessoa**

IDPessoa é a chave primária (PRIMARY KEY);

O NIF é um número único (UNIQUE) que se encontra entre *100000000* e *4000000* (CHECK (*NIF > 100000000*) AND (*NIF < 400000000*)). Esta variável tem sempre um valor (NOT NULL);

O Nome tem que, obrigatoriamente, ter um valor (NOT NULL);

A Data de Nascimento é uma data, tendo sempre um valor (NOT NULL);

O conjunto (Nome, Data de Nascimento) é único (UNIQUE);

O Telefone tem que, obrigatoriamente, ter um valor e este tem de ser único (UNIQUE);

O Género é um carácter cujo valor por definição é *‘?’* e que tem de ter sempre valor (NOT NULL);

IDLocalização é uma chave estrangeira (integridade referencial, FOREIGN KEY);

**Funcionário**

IDPessoa é a chave primária (PRIMARY KEY)e também é uma chave estrangeira (integridade referencial, FOREIGN KEY);

Salário é um número que tem, obrigatoriamente, um valor (NOT NULL) e este tem de ser maior que 700 (CHECK *Salário>=700*);

IDSupermercado e IDHorário são chaves estrangeiras (integridade referencial, FOREIGN KEY);

**Cliente**

IDPessoa é a chave primária (PRIMARY KEY)e também é uma chave estrangeira (integridade referencial, FOREIGN KEY);

Número de Cliente tem que, obrigatoriamente, ter um valor (NOT NULL);

**Horário**

IDHorário é a chave primária (PRIMARY KEY);

Hora Inicial e Hora Final são datas e têm de ter sempre um valor (NOT NULL). Além disso, a Hora Inicial tem de ser menor que a Hora Final (CHECK (*Hora Inicial < Hora Final*));

**HorárioFuncionário**

IDHorário e IDPessoa formam a chave primária composta (PRIMARY KEY (IDHorário, IDPessoa)) e são ambas chaves estrangeiras (integridade referencial, FOREIGN KEY);

Dia da Semana tem de ter um valor (NOT NULL) e tem que ter um dos seguintes valores (*“Segunda”, “Terca”,”Quarta”,”Quinta”, “Sexta”, “Sabado”, “Domingo”* ) (CHECK ((*Dia da Semana == "Segunda"*)OR (*Dia da Semana == "Terca"*) OR (*Dia da Semana == "Quarta"*) OR (*Dia da Semana == "Quinta"*) OR (*Dia da Semana == "Sexta"*) OR (*Dia da Semana == "Sabado"*) OR (*Dia da Semana == "Domingo"*)));

Hora de Início Da Pausa da Manhã, Hora de Fim da Pausa da Manhã, Hora de Início de Almoço, Hora de Fim de Almoço, Hora de Início da Pausa da Tarde e Hora de Fim da Pausa de Tarde são datas, tendo sempre valores (NOT NULL). Além disso, é verificado se a hora de inicio de cada pausa é menor do que a hora de fim, e se a hora de início da pausa é superior à hora de fim da pausa anterior, se esta existir (CHECK ((*Hora Inicial da Pausa da Manhã < Hora Final da Pausa da Manhã*) AND (*Hora Final da Pausa da Manhã < Hora Inicial da Pausa de Almoço*) AND (*Hora Inicial da Pausa de Almoço < Hora Final da Pausa de Almoço*) AND (*Hora Final da Pausa de Almoço < Hora Inicial da Pausa de Tarde*) AND (*Hora Inicial da Pausa da Tarde < Hora Final da Pausa de Tarde*)));

**Caixa**

IDCaixa é a chave primária (PRIMARY KEY);

Número tem de ter, obrigatoriamente, um valor (NOT NULL);

Aberta é um booleano que tem por definição o valor *false* (DEFAULT) e tem de ter sempre um valor (NOT NULL);

IDSupermercado é uma chave estrangeira (integridade referencial, FOREIGN KEY);

**Automática**

IDCaixa é a chave primária (PRIMARY KEY) e também é uma chave estrangeira (integridade referencial, FOREIGN KEY);

**Manual**

IDCaixa é a chave primária (PRIMARY KEY) e também é uma chave estrangeira (integridade referencial, FOREIGN KEY);

**Secção**

IDSecção é a chave primária (PRIMARY KEY);

Nome tem de ter, obrigatoriamente, um valor (NOT NULL);

IDSupermercado é uma chave estrangeira (integridade referencial, FOREIGN KEY);

**Compra**

IDCompra é a chave primária (PRIMARY KEY)

Dia e Hora são datas e têm de ter, obrigatoriamente, um valor (NOT NULL);

IDPessoa e IDCaixa são chaves estrangeiras (integridade referencial, FOREIGN KEY)

**Produto**

IDProduto é a chave primária (PRIMARY KEY);

Nome, Preço e Quantidade Disponível têm de ter, obrigatoriamente, um valor (NOT NULL). Além disso, a Quantidade Disponível tem de ser superior ou igual a *0* (CHECK (*Quantidade Disponível >= 0*));

IDSecção é uma chave estrangeira (integridade referencial, FOREIGN KEY);

**Quantidade**

IDProduto e IDCompra formam a chave primária composta (PRIMARY KEY (IDProduto, IDCompra)) e são chaves estrangeiras (integridade referencial, FOREIGN KEY);

Quantidade tem de ter, obrigatoriamente, um valor (NOT NULL) e este tem de ser superior a *0* (CHECK (*Quantidade > 0*));

**ResponsávelCaixa**

IDCaixa, IDPessoa formam a chave primária composta (PRIMARY KEY (IDCaixa, IDPessoa)). Além disso, IDPessoa e IDCaixa são chaves primárias (integridade referencial, FOREIGN KEY);

**Trabalha**

IDPessoa e IDSecção formam a chave primária composta (PRIMARY KEY (IDPessoa, IDSecção)). Além disso, IDPessoa e IDSecção são chaves primárias (integridade referencial, FOREIGN KEY);

**Responsável**

IDPessoa e IDSecção formam a chave primária composta (PRIMARY KEY (IDPessoa, IDSecção)). Além disso, IDPessoa e IDSecção são chaves primárias (integridade referencial, FOREIGN KEY);

***Queries***

Abaixo estão apresentadas as 10 “*queries*” implementadas, com uma breve explicação:

1. Obtenção de todas as compras efetuadas por um certo cliente. É apresentado ao utilizador: o nome do cliente, o dia, a hora e o custo a que efetuou a compra. Esta “*query*” pode ser adaptada para qualquer cliente trocando o NIF para aquele que se deseja.
2. Obtenção dos horários dos funcionários. São apresentados, ao utilizador, todos os funcionários e os seus respetivos horários: hora de entrada, hora de saída e dia da semana.
3. Obtenção do número de funcionários de cada supermercado. São apresentados o nome do supermercado e quantos funcionários este emprega.
4. Obtenção dos 5 produtos mais vendidos. São apresentados ao utilizador os 5 produtos mais vendidos, ordenados de forma descendente.
5. Obtenção do número de caixas e secções a cargo de cada funcionário. São apresentados todos os funcionários com a quantidade de caixas e secções onde estes trabalham. Por exemplo: um funcionário X, que trabalha numa caixa manual e numa secção, será apresentado como: “X 1 1”.
6. Obtenção do lucro total dos supermercados. São apresentados todos os supermercados e o lucro que obtiveram.

|  |  |
| --- | --- |
| Matosinhos | 25 |
| Senhora da Hora | 25 |
| Leça da Palmeira | 50 |

1. Obtenção da frequência de clientes, em relação à sua localidade. São apresentadas as localidades dos clientes e a percentagem de clientes que são dessa localidade. Por exemplo: Se no total existirem 4 clientes: 1 de Matosinhos, 1 da Senhora da Hora e 2 de Leça da Palmeira, a tabela apresentada teria um resultado semelhante a:
2. Obtenção das horas úteis dos trabalhadores. São apresentados todos os trabalhadores e o número de horas úteis que cada um trabalha, considerando as pausas que estes efetuam.
3. Obtenção de todos os clientes. São apresentados todos os clientes e as suas informações
4. Obtenção da compra mais cara de cada cliente. São apresentados todos os clientes e a sua compra mais cara.

***Triggers***

Abaixo estão apresentados os 3 “*triggers*”, com uma breve explicação:

1. Este t*rigger* verifica se, quando se faz uma compra e se adiciona um tuplo quantidade, a quantidade comprada não excede a quantidade disponível de um produto em stock. Se esta última situação não se verificar, lança uma mensagem de erro, alertando o utilizador.
2. Este *trigger* atualiza a quantidade total do produto e da quantidade disponível desse mesmo produto quando se introduz um tuplo quantidade já existente, aumentando-se a essa compra a quantidade do produto comprada e diminuindo a quantidade disponível.
3. Este *trigger* atualiza o salário de um funcionário, quando o seu horário é alterado