

Sistemas Multimédia

**Fábrica 1   
(Gestão dos clientes e suas encomendas)**

*Daniel José Guedes Carneiro*

Índice

[Índice III](#_Toc29808927)

[Índice de figuras VI](#_Toc29808928)

[1. Introdução VIII](#_Toc29808929)

[1.1 Objetivo VIII](#_Toc29808930)

[1.2 Organização do Relatório VIII](#_Toc29808931)

[2. Desenho e Conceção da Base de Dados IX](#_Toc29808932)

[2.1 Descrição das entidades e atributos: XI](#_Toc29808933)

[3. Normalização de Tabelas XII](#_Toc29808934)

[3.1 Forma não normalizada XII](#_Toc29808935)

[3.2 Primeira forma normalizada XII](#_Toc29808936)

[3.3 Segunda forma normalizada XIII](#_Toc29808937)

[3.4 Terceira forma normalizada XIII](#_Toc29808938)

[4. Diagrama de Relação entre tabelas XIV](#_Toc29808939)

[5. Criação da Base de Dados (SQL) XV](#_Toc29808940)

[5.1 Código SQL XV](#_Toc29808941)

[5.1.2 – Tabela de Códigos Postais XV](#_Toc29808942)

[5.1.3 – Tabela de Clientes XV](#_Toc29808943)

[5.1.4 – Tabela de Produtos XVI](#_Toc29808944)

[5.1.5 – Tabela de Encomendas XVI](#_Toc29808945)

[5.1.4 – Tabela de Fornecedores XVII](#_Toc29808946)

[5.1.4 – Tabela de Relação entre Produtos e Fornecedores XVII](#_Toc29808947)

[5.1.4 – Tabela de relação entre Encomendas, Produtos e Clientes XVII](#_Toc29808948)

[5.2 Consultas SQL XVIII](#_Toc29808949)

[5.2.1 Listar o nome de todos os clientes XVIII](#_Toc29808950)

[5.2.2 Listar o nome dos produtos com quantidade superior a 150.000 XVIII](#_Toc29808951)

[5.2.3 Listar, sem repetições, a localidade dos códigos postais superiores a 5000. XVIII](#_Toc29808952)

[5.2.4 Listar o número total de Produtos e associar a tabela 'total\_prod'. XVIII](#_Toc29808953)

[5.2.5 Listar todos os produtos com quantidade superior a 120000 e o nome termina em 'o'. XIX](#_Toc29808954)

[5.2.6 Listar todos os clientes cuja morada contém a palavra 'das'. XIX](#_Toc29808955)

[5.2.7 Número de encomendas feitas em 2020. XIX](#_Toc29808956)

[5.2.8 Nomes de clientes residentes no Porto. XIX](#_Toc29808957)

[5.2.9 Qual o preço da Quantidade de produto superior à média. XX](#_Toc29808958)

[5.2.10 Nomes dos produtos encomendados pelo cliente nº 3. XX](#_Toc29808959)

[5.2.11 Lista de fornecedores e produtos por eles fornecidos. XX](#_Toc29808960)

[5.2.12 Lista do número de encomendas feitas for cada cliente. XXI](#_Toc29808961)

[5.2.13 Nomes de clientes que fizeram mais do que uma encomenda. XXI](#_Toc29808962)

[5.2.14 Lista de todas as encomendas efetuadas em 2020 ordenadas pelo número de cliente e incluindo as datas e o nome dos produtos encomendados. XXI](#_Toc29808963)

[6. Conclusão XXII](#_Toc29808964)

[7. Referências e Bibliografia XXIII](#_Toc29808965)

Índice de figuras

[Figura 1 - Diagrama de Fluxo de dados da Fábrica 1 – Gestão de encomendas de clientes X](#_Toc29808920)

[Figura 2 - Diagrama de Relação entre tabelas - Gerado automáticamente XIV](#_Toc29808921)

# Introdução

No âmbito da avaliação da unidade curricular de Bases Dados 1 foram dadas algumas sugestões para a elaboração deste trabalho, o exemplo sugerido que foi escolhido foi “Fábrica 1 (gestão dos clientes e das suas encomendas) ”.

Para a elaboração deste trabalho recorri ao *software* *Draw.io* para a elaboração do diagrama de fluxo de dados, para a criação e desenho das relações da base de dados foi utilizei o *Microsoft SQL Management Studio*.

## Objetivo

Considera-se o objetivo deste trabalho a demonstração e a criação de uma base de dados começando por criar um diagrama de fluxo de dados, as normalizações das tabelas, e também um diagrama de relação entre as tabelas. Após os anteriores procedimentos concluídos, a elaboração de uma base de dados em SQL através da criação de um script para ser demonstrado neste trabalho.

## Organização do Relatório

Para a realização deste relatório foi dividido em três fases: desenho e conceção onde é apresentada uma imagem gráfica das entidades e os atributos necessários, a normalização das tabelas onde se passa do desenho para uma fase final da base de dados e finalmente a implementação, criação e consultas da base de dados.

# Desenho e Conceção da Base de Dados

O foque da conceção desta base de dados está na gestão dos clientes e das suas encomendas. Para ajudar a entender o funcionamento desta gestão começamos por fazer algumas afirmações:

1. Um cliente pode fazer várias encomendas;
2. Uma encomenda pertence apenas a um cliente;
3. Um fornecedor pode fornecer vários produtos;
4. As encomendas podem ter vários produtos.

Para elaborar uma base de dados e fazer relações entre as tabelas é primeiro necessário criar um diagrama com todas a relações para que o criador da base de dados tenha toda essa ideia em mente.

Neste sistema em concreto e tendo em conta as afirmações efetuadas anteriormente, podemos concluir que lidamos com quatro entidades principais: clientes, produtos, encomendas e fornecedores. Então já temos um ponto de partida para criação de um diagrama e podemos começar por elaborar o desenho do diagrama de fluxo de dados.

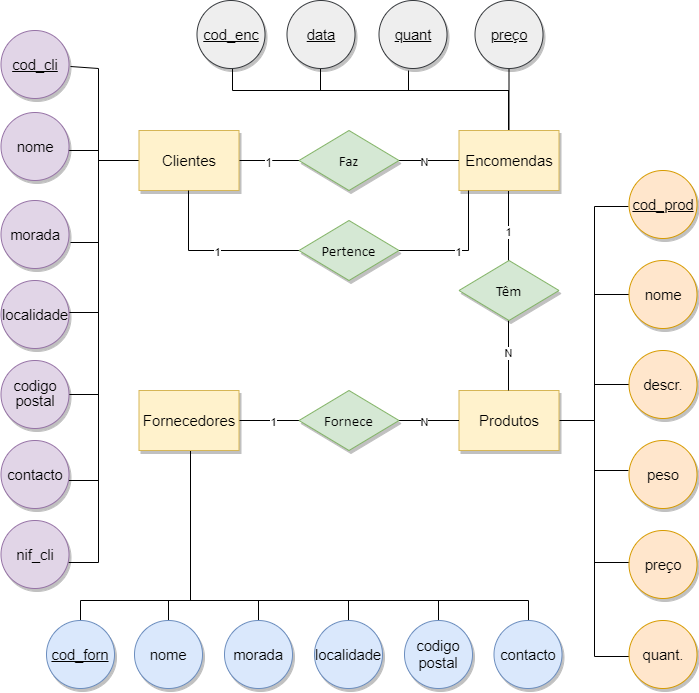


Figura 1 - Diagrama de Fluxo de dados da Fábrica 1 – Gestão de encomendas de clientes

## 2.1 Descrição das entidades e atributos:

* **Clientes** – Entidade que representa todos os clientes
  + **cod\_cli** – código do cliente (chave primária)
  + **nome\_cli** – nome do cliente
  + **morada\_cli** – morada do cliente
  + **localidade** – localidade do cliente
  + **cod\_postal** – código postal do cliente
  + **contacto\_cli** – contacto do cliente
  + **nif\_cli** – numero de identificação fiscal do cliente
* **Encomendas** – Entidade que regula todas as encomendas
  + **cod\_enc** – código da encomenda (chave primária)
  + **data\_enc** – data da encomenda
  + **preco\_enc** – preço da encomenda
  + **quantidade\_prod** – quantidade do produto
* **Produtos** – Entidade que representa todos os produtos
  + **cod\_prod** – código do produto (chave primária)
  + **nome\_prod** – nome do produto
  + **descricao\_prod** – descrição do produto
  + **peso\_prod** – peso do produto
  + **preco\_prod** – preço do produto
  + **quantidade\_prod** – quantidade do produto
* **Fornecedores** – Entidade que representa todos os fornecedores
  + **cod\_forn** – código do fornecedor (chave primária)
  + **nome\_forn** – nome do fornecedor
  + **morada\_forn** – morada do fornecedor
  + **localidade** – localidade do fornecedor
  + **cod\_postal** – código postal do fornecedor
  + **contacto\_forn** – contacto do fornecedor

# Normalização de Tabelas

Para a criação de base de dados uma parte fundamental é a criação conceptual das tabelas, sendo é importante fazer a normalização das tabelas visto que esta vai ajudar a partir de um conceito, criar uma estrutura real para o auxílio na criação da base de dados.

## 3.1 Forma não normalizada

Com o auxílio da descrição das entidades e dos atributos foi possível então criar uma lista de todos os atributos necessários para esta base de dados.

cod\_cli, cod\_prod, cod\_enc, cod\_forn, nome\_cli, nome\_prod, data\_enc, nome\_forn, morada\_cli, descricao\_prod, cod\_prod, morada\_forn, localização, peso\_prod, cod\_cli, localização, cod\_postal, preco\_prod, quantidade\_prod, cod\_postal, contacto\_cli, quantidade\_prod, preco\_enc, contacto\_forn, nif\_cli, cod\_forn.

## 3.2 Primeira forma normalizada

Depois de analisar todos esses atributos necessários para esta base de dados foram agrupados em quatro tabelas e associadas as chaves primárias:

|  |  |  |  |
| --- | --- | --- | --- |
| **Clientes** | **Produtos** | **Encomendas** | **Fornecedores** |
| # cod\_cli | # cod\_prod | # cod\_enc | # cod\_forn |
| nome\_cli | nome\_prod | data\_enc | nome\_forn |
| morada\_cli | descricao\_prod | cod\_prod | morada\_forn |
| localidade | peso\_prod | cod\_cli | localidade |
| cod\_postal | preco\_prod | quantidade\_prod | cod\_postal |
| contacto\_cli | quantidade\_prod | preco\_enc | contacto\_forn |
| nif\_cli | cod\_forn |  |  |

## 3.3 Segunda forma normalizada

Após uma análise a todos os atributos, foram verificados se os campos não-chave dependem da totalidade das chaves. Sendo que foram criadas 2 tabelas:

* Prod-Forn – estabelece uma relação entre os Produtos e o Fornecedores
* End-Prod-Cli – estabelece uma relação entre as Encomendas, Produtos e Clientes.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Clientes** | **Produtos** | **Encomendas** | **Fornecedores** | **ProdForn** | **EncProdCli** |
| # cod\_cli | # cod\_prod | # cod\_enc | # cod\_forn | cod\_forn | cod\_cli |
| nome\_cli | nome\_prod | data\_enc | nome\_forn | cod\_prod | cod\_prod |
| morada\_cli | descricao\_prod | quantidade\_prod | morada\_forn |  | cod\_enc |
| localizacao | peso\_prod | preco\_enc | localizacao |  |  |
| cod\_postal | preco\_prod |  | cod\_postal |  |  |
| contacto\_cli | quantidade\_prod |  | contacto\_forn |  |  |
| nif\_cli |  |  |  |  |  |

## 3.4 Terceira forma normalizada

Depois de verificar a segunda forma normalizada foram analisados os campos não chave e verificou-se se existia dependência entre eles. Foi criada a tabela Codigos\_Postais e todas as referências de localização foram removidas para evitar duplicações. Assim nesta terceira forma podemos ter a vista da estrutura da base de dados.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Clientes** | **Produtos** | **Encomendas** | **Fornecedores** | **ProdForn** | **EncProdCli** | **CodPostais** |
| # cod\_cli | # cod\_prod | # cod\_enc | # cod\_forn | cod\_forn | cod\_cli | # cod\_postal |
| nome\_cli | nome\_prod | data\_enc | nome\_forn | cod\_prod | cod\_prod | localidade |
| morada\_cli | descricao\_prod | quantidade\_prod | morada\_forn |  | cod\_enc |  |
| cod\_postal | peso\_prod | preco\_enc | cod\_postal |  |  |  |
| contacto\_cli | preco\_prod |  | contacto\_forn |  |  |  |
| nif\_cli | quantidade\_prod |  |  |  |  |  |

# Diagrama de Relação entre tabelas

A ferramenta *Microsoft SQL Management Studio* permite criar um desenho de todas as tabelas para verificar se as ligações entre as mesmas fazem sentido. Na seguinte imagem podemos identificar que as relações que anteriormente foram requeridas na normalização fazem mais sentido quando interligadas entre si.

Para que seja possível criar este diagrama é primeiro necessário criar a base de dados e inserir todas as suas relações, no próximo ponto iniciamos a criação.

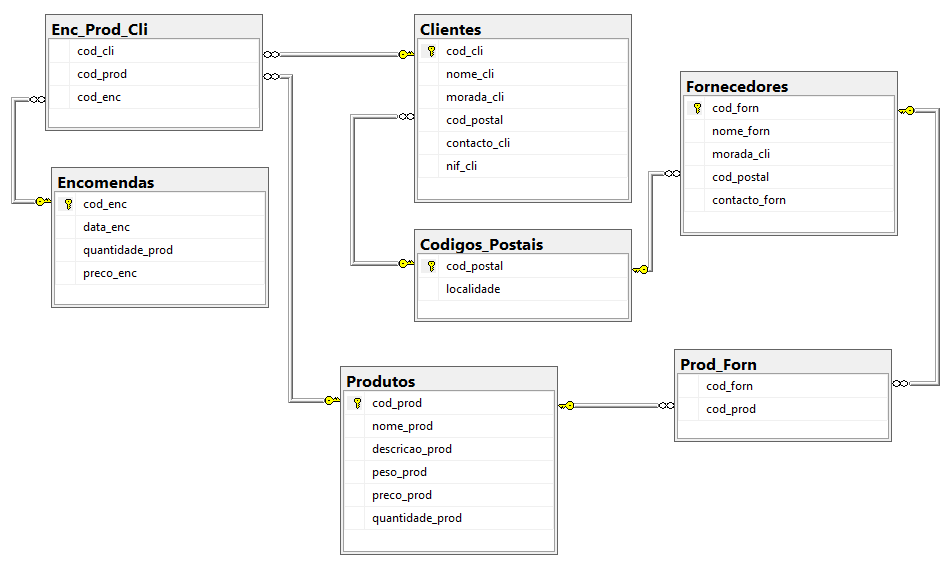


Figura 2 - Diagrama de Relação entre tabelas - Gerado automáticamente

# Criação da Base de Dados (SQL)

Para a criação da base de dados em SQL foi utilizado o *Microsoft SQL Management Studio*, este permitiu através de um código a criação da mesma.

## 5.1 Código SQL

Para fazer a criação da base de dados comecei por executar o seguinte código:

-- CRIAR BASE DE DADOS

CREATE DATABASE fabrica

Depois a criação das tabelas e a inserção de dados nas mesmas:

### 5.1.2 – Tabela de Códigos Postais

-- CRIAR TABELA CODIGOS\_POSTAIS

create table Codigos\_Postais (

cod\_postal nvarchar(8) primary key not null,

localidade nvarchar(30) not null

);

-- INSERIR CODIGOS POSTAIS

insert into Codigos\_Postais values

('4460-280','Barbeitos'),

('2799-554','Linda-a-Velha'),

('7370-170', 'Campo Maior'),

('4100-001','Lisboa'),

('4400-096','V.N.Gaia'),

('4000-034','Porto');

### 5.1.3 – Tabela de Clientes

-- CRIAR TABELA CLIENTES

create table Clientes(

cod\_cli int primary key not null,

nome\_cli nvarchar(35) not null,

morada\_cli nvarchar(35) null,

cod\_postal nvarchar(8) foreign key references Codigos\_Postais(cod\_postal) null,

contacto\_cli int null,

nif\_cli int null

);

-- CRIAR CLIENTES

insert into Clientes values

(1,'Diogo Ramiro','Rua de Alverca','4100-001', 225695555, 235265585),

(2,'Jaime Delgado','Rua das violetas','4400-096', 935262352, 265851455),

(3,'Artur Lima','Rua das fontainhas','4000-034', 965232255, 265851520),

(4,'Gerson Magno','Rua das fontainhas','4000-034', 965232255, 265851520);

### 5.1.4 – Tabela de Produtos

-- CRIAR TABELA PRODUTOS

create table Produtos(

cod\_prod int primary key not null,

nome\_prod nvarchar(35) not null,

descricao\_prod nvarchar(50) null,

peso\_prod decimal (16, 2) null,

preco\_prod money default 0,

quantidade\_prod int null

);

-- CRIAR PRODUTOS

insert into Produtos values

(1, 'Delta', 'Capsulas de Cafe', 0.50, 0.20, 300000),

(2, 'Buondi', 'Capsulas de Cafe', 0.55, 0.25, 200000),

(3, 'Nespresso', 'Capsulas de Cafe', 0.55, 0.25, 100000),

(4, 'Dulce Gusto', 'Capsulas de Cafe', 0.55, 0.30, 500000);

### 5.1.5 – Tabela de Encomendas

-- CRIAR TABELA ENCOMENDAS

create table Encomendas(

cod\_enc int primary key not null,

data\_enc datetime null,

quantidade\_prod int null,

preco\_enc money default 0

);

-- CRIAR ENCOMENDAS

insert into Encomendas values

(1, '2019-01-01 10:30:00', 100, 20),

(2, '2020-01-02 11:30:00', 200, 50),

(3, '2020-01-03 12:30:00', 300, 75),

(4, '2020-01-04 13:30:00', 1000, 250);

### 5.1.4 – Tabela de Fornecedores

-- CRIAR TABELA FORNECEDORES

create table Fornecedores(

cod\_forn int primary key not null,

nome\_forn nvarchar(35) not null,

morada\_cli nvarchar(35) null,

cod\_postal nvarchar(8) foreign key references Codigos\_Postais(cod\_postal) null,

contacto\_forn int null,

);

-- CRIAR FORNECEDORES

insert into Fornecedores values

(1, 'Delta Cafe', 'Campo Maior Rua B, nº 4', '7370-170', 268009200),

(2, 'Nestlé Portugal SA', 'R. Alexandre Herculano 8', '2799-554', 808200644);

### 5.1.4 – Tabela de Relação entre Produtos e Fornecedores

-- CRIAR TABELA DE RELACAO PRODUTOS + FORNECEDORES

create table Prod\_Forn(

cod\_forn int foreign key references Fornecedores(cod\_forn) not null,

cod\_prod int foreign key references Produtos(cod\_prod) not null,

);

-- CRIAR RELACAO ENTRE FORNECEDORES + PRODUTOS

insert into Prod\_Forn values

(1, 1),

(2, 2),

(2, 3),

(2, 4);

### 5.1.4 – Tabela de relação entre Encomendas, Produtos e Clientes

-- CRIAR TABELA FORNECEDORES

create table Enc\_Prod\_Cli(

cod\_cli int foreign key references Clientes(cod\_cli) not null,

cod\_prod int foreign key references Produtos(cod\_prod) not null,

cod\_enc int foreign key references Encomendas(cod\_enc) not null

);

-- CRIAR RELACAO ENTRE ENCOMENDAS + PRODUTOS + CLIENTES

insert into Enc\_Prod\_Cli values

(1, 1, 1),

(2, 2, 2),

(3, 3, 3),

(3, 4, 4);

## 5.2 Consultas SQL

Para verificar se as relações entre as tabelas estão a funcionar podemos recorrer às consultas nas tabelas. Em seguida vou demonstrar o código para fazer uma determinada consulta e o seu resultado representado numa tabela.

### 5.2.1 Listar o nome de todos os clientes

select nome\_cli from Clientes

|  |  |
| --- | --- |
|  | **nome\_cli** |
| **1** | Diogo Ramiro |
| **2** | Jaime Delgado |
| **3** | Artur Lima |
| **4** | Gerson Magno |

### 5.2.2 Listar o nome dos produtos com quantidade superior a 150.000

select nome\_prod, quantidade\_prod from Produtos

where quantidade\_prod > 150000

|  |  |  |
| --- | --- | --- |
|  | **nome\_prod** | **quantidade\_prod** |
| **1** | Delta | 300000 |
| **2** | Buondi | 200000 |
| **3** | Dulce Gusto | 500000 |

### 5.2.3 Listar, sem repetições, a localidade dos códigos postais superiores a 5000.

select distinct(localidade) from Codigos\_Postais

where cod\_postal > '5000'

|  |  |
| --- | --- |
|  | **localidade** |
| **1** | Campo Maior |

### 5.2.4 Listar o número total de Produtos e associar a tabela 'total\_prod'.

select count(\*) as total\_prod from Produtos

|  |  |
| --- | --- |
|  | **total\_prod** |
| **1** | 4 |

### 5.2.5 Listar todos os produtos com quantidade superior a 120000 e o nome termina em 'o'.

select \* from Produtos

where quantidade\_prod > 120000 and lower(nome\_prod) like('%o')

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | cod\_prod | nome\_prod | descricao\_prod | peso\_prod | preco\_prod | quantidade\_prod |
| **1** | 4 | Dulce Gusto | Capsulas de Cafe | 0.55 | 0.3 | 500000 |

### 5.2.6 Listar todos os clientes cuja morada contém a palavra 'das'.

select \* from Clientes where lower(morada\_cli) like('%das%')

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | cod\_cli | nome\_cli | morada\_cli | cod\_postal | contacto\_cli | nif\_cli |
| **1** | 2 | Jaime Delgado | Rua das violetas | 4400-096 | 935262352 | 265851455 |
| **2** | 3 | Artur Lima | Rua das fontainhas | 4000-034 | 965232255 | 265851520 |
| **3** | 4 | Gerson Magno | Rua das fontainhas | 4000-034 | 965232255 | 265851520 |

### 5.2.7 Número de encomendas feitas em 2020.

select count(\*) as numero\_enc from Encomendas

where YEAR(data\_enc) = 2020

|  |  |
| --- | --- |
|  | numero\_enc |
| **1** | 3 |

### 5.2.8 Nomes de clientes residentes no Porto.

select nome\_cli, localidade from Clientes, Codigos\_Postais

where localidade = 'Porto'

AND Clientes.cod\_postal = Codigos\_Postais.cod\_postal

|  |  |  |
| --- | --- | --- |
|  | nome\_cli | localidade |
| **1** | Artur Lima | Porto |
| **2** | Gerson Magno | Porto |

### 5.2.9 Qual o preço da Quantidade de produto superior à média.

select preco\_enc from Encomendas

where quantidade\_prod >

(select AVG(quantidade\_prod) from Encomendas)

|  |  |
| --- | --- |
|  | preco\_enc |
| **1** | 250 |

### 5.2.10 Nomes dos produtos encomendados pelo cliente nº 3.

select nome\_prod from Produtos, Enc\_Prod\_Cli

where cod\_cli = 3 and Enc\_Prod\_Cli.cod\_prod = Produtos.cod\_prod

|  |  |
| --- | --- |
|  | nome\_prod |
| **1** | Nespresso |
| **2** | Dulce Gusto |

### 5.2.11 Lista de fornecedores e produtos por eles fornecidos.

select nome\_forn, nome\_prod

from Produtos, Fornecedores, Prod\_Forn

where Fornecedores.cod\_forn = Prod\_Forn.cod\_forn

and Produtos.cod\_prod = Prod\_Forn.cod\_prod

|  |  |  |
| --- | --- | --- |
|  | nome\_forn | nome\_prod |
| **1** | Delta Cafe | Delta |
| **2** | Nestlé Portugal SA | Buondi |
| **3** | Nestlé Portugal SA | Nespresso |
| **4** | Nestlé Portugal SA | Dulce Gusto |

### 5.2.12 Lista do número de encomendas feitas for cada cliente.

select nome\_cli, count(\*) as numero\_enc

from Encomendas, Enc\_Prod\_Cli, Clientes

where Enc\_Prod\_Cli.cod\_cli = Clientes.cod\_cli

and Encomendas.cod\_enc = Enc\_Prod\_Cli.cod\_enc

group by nome\_cli

|  |  |  |
| --- | --- | --- |
|  | nome\_cli | numero\_enc |
| **1** | Artur Lima | 2 |
| **2** | Diogo Ramiro | 1 |
| **3** | Jaime Delgado | 1 |

### 5.2.13 Nomes de clientes que fizeram mais do que uma encomenda.

select nome\_cli from Clientes, Enc\_Prod\_Cli

where Clientes.cod\_cli = Enc\_Prod\_Cli.cod\_cli

group by Clientes.cod\_cli, nome\_cli

having count(cod\_enc) > 1

|  |  |
| --- | --- |
|  | nome\_cli |
| **1** | Artur Lima |

### 5.2.14 Lista de todas as encomendas efetuadas em 2020 ordenadas pelo número de cliente e incluindo as datas e o nome dos produtos encomendados.

select nome\_cli from Clientes, Enc\_Prod\_Cli

where Clientes.cod\_cli = Enc\_Prod\_Cli.cod\_cli

group by Clientes.cod\_cli, nome\_cli

having count(cod\_enc) > 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | cod\_cli | nome\_cli | data\_enc | nome\_prod |
| **1** | 2 | Jaime Delgado | 2020-01-02 11:30:00 | Buondi |
| **2** | 3 | Artur Lima | 2020-01-03 12:30:00 | Nespresso |
| **3** | 3 | Artur Lima | 2020-01-04 13:30:00 | Dulce Gusto |

# Conclusão

Este trabalho demonstrou-se bastante entusiasmante de se concretizar, uma vez que à medida que o estava a fazer, aprendi a criação básica de uma base de dados, desde a conceção em papel até à fase final. A conceção das normalizações das tabelas também ficaram bastante mais claras quando essas passaram de tabelas para a base de dados e foram representadas num diagrama.

Importante também salientar que o uso de diversas consultas SQL tornou-se mais evidente de como funcionam as relações entre as tabelas e da própria sintaxe SQL.

# Referências e Bibliografia

Academy, C. (04 de 01 de 2019). *List of SQL Commands*. Obtido de Codecademy: https://www.codecademy.com/articles/sql-commands

Microsoft. (24 de 09 de 2018). *Create a New Database Diagram (Visual Database Tools)*. Obtido em 05 de 01 de 2020, de Microsoft - SQL Docs: https://docs.microsoft.com/en-us/sql/ssms/visual-db-tools/create-a-new-database-diagram-visual-database-tools?view=sql-server-ver15