

Complementos de Bases de Dados 2022/2023

Licenciatura em Eng^a. Informática

1^a Fase Relatório Técnico

Turma: 6

Horário de Laboratório: 6^afeira - 16:30

Docente: Sara Batista

Grupo

202002203, Fernando Ramalho

202001870, Guilherme Bernardino

1. Introdução

Este projeto tem como objetivo satisfazer a avaliação da cadeira Complementos Base de Dados, do curso Engenharia de Software, Escolha Superior de Tecnologias no Instituto Politécnico de Setúbal. Para esse fim, foi pedido aos alunos que desenvolvessem e administrassem uma nova base de dados para a empresa denominada *Wide World Importers* (WWI), uma empresa importadora e distribuidora de produtos, que opera no mercado de vendas e retalho.

Com suporte de ficheiros Excel, texto e antiga base de dados da WWI, os alunos foram desafiados a desenvolver uma nova base de dados, atualizada, normalizada e escalável.

Ao longo deste documento vão ser relatadas as implementações nesta nova base de dados.

2. Especificação de Requisitos

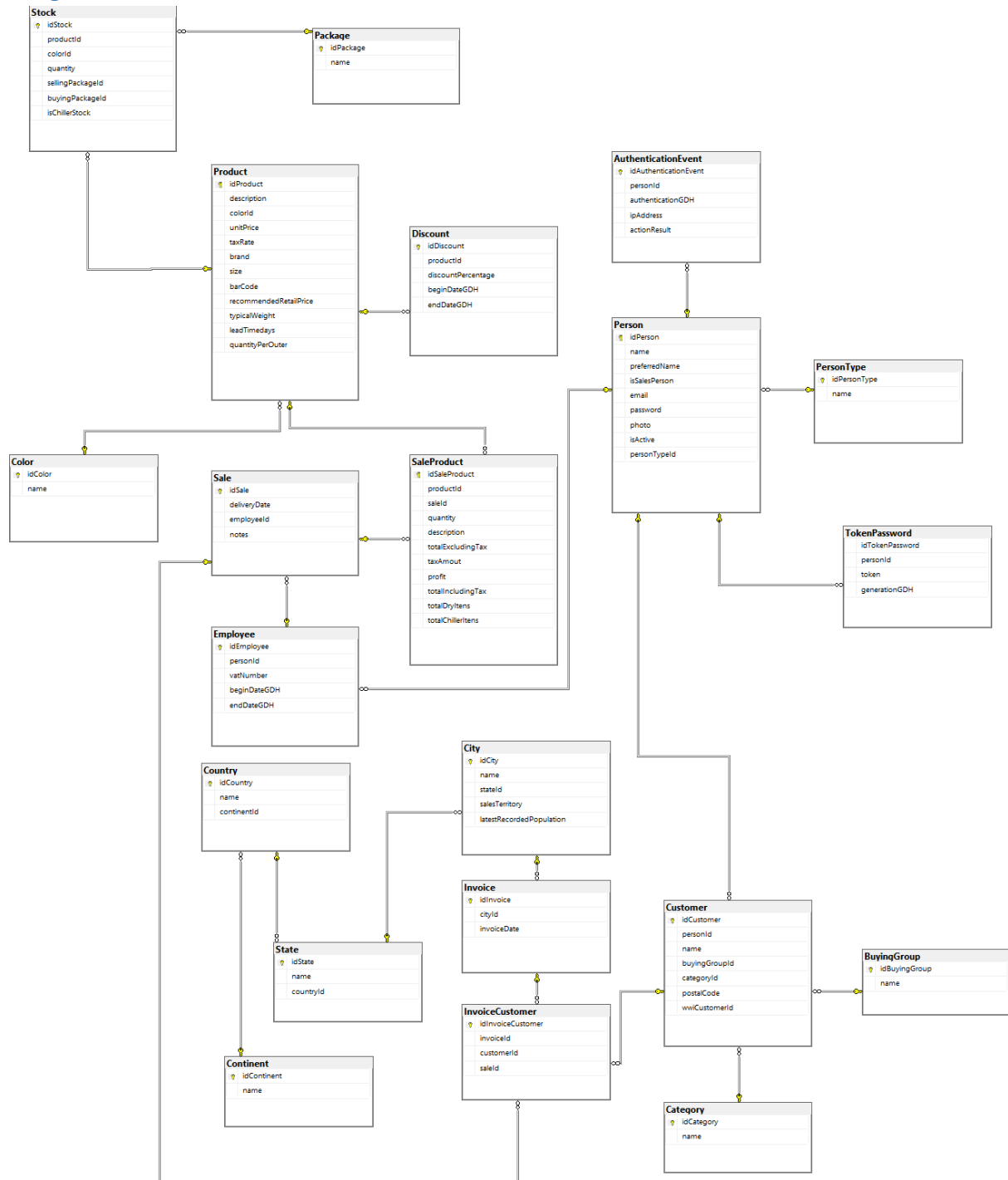
ID	Descrição	Implementado (S/N)
R01	O sistema deve permitir a autenticação com recurso à conta de email e password	S
R02	O sistema deve permitir adicionar um utilizador	S
R03	O sistema deve permitir atualizar um utilizador	S
R04	O sistema deve permitir remover um utilizador	S
R05	O sistema deve permitir recuperar um utilizador através de um <i>token</i> que permite alterar a password	S
R06	O sistema deve permitir definir uma promoção sobre um ou mais produtos	S
R07	O sistema deve permitir alterar as datas de início e fim de uma promoção	S
R08	O sistema deve permitir criar uma venda	S
R09	O sistema deve permitir adicionar um produto a uma venda	S
R10	O sistema deve permitir alterar a quantidade de um produto numa venda	S

1ª Fase Relatório Técnico – Complementos de Bases de Dados

R11	O sistema deve permitir remover um produto de uma venda.	S
R12	O sistema deve permitir verificar se uma venda contém 0 produtos associados para se poder remover	S
R13	O sistema deve permitir calcular o preço total de uma venda	S
R14	O sistema deve permitir verificar se a data de entrega está de acordo com o tempo previsto de entrega de um produto (“Lead Time Days”)	S
R15	O sistema não deve permitir uma venda conter produtos com e sem “Chiller Stock”	S
RM01	O sistema deve verificar na inserção da tabela de utilizadores se já existe um utilizador com um email duplicado	S

3. Modelo Relacional (*Modelo de dados*)

3.1 Diagrama do Modelo Relacional



4. Definição do Layout

4.1 Identificação do espaço ocupado por tabela

Nome Tabela	Dimensão do Registo (Bites)	Nº de Registos
AuthenticationEvent	8	0
BuyingGroup	36 000	2
Category	15 000	5
City	60	116294
Color	800	9
Continent	1 000	7
Country	60	249
Customer	100	402
Discount	8 000	1
Employee	30	212
Invoice	20	70510
InvoiceCustomer	20	228265
Package	44 564 000	6
Person	80	615
PersonType	4 000	2

1ª Fase Relatório Técnico – Complementos de Bases de Dados

Product	100	671
Sale	100	228266
SaleProduct	100	228266
State	100	57
Stock	8	0
TokenPassword	8	0

4.2 Especificação dos Filegroups

Nome Filegroup	Tabelas associadas	Parâmetros
ProductFileGroup	Stock, Package, Product, Category e Color	20 MB / 400 MB / 5 %
PersonFileGroup	Person, AuthenticationEvent, PersonType, TokenPassowrd, Employee e Customer	30 MB / 600 MB / 5 %
InvoiceFileGroup	City, States, Invoice, InvoiceCustomer, Country e Continent	25 MB / 500 MB / 5 %
SaleFileGroup	Discount, Sale e SaleProduct	15 MB / 300 MB / 5 %

4.3 Schemas

Nome	Descrição
dbo	User de origem, owner. Também designado como user servidor.
FernandoRamalho	User de acesso, owner. User utilizado pelo aluno Fernando Ramalho para aceder ao servidor.

GuilhermeBernardino

User de acesso, owner. Use utilizado pelo aluno Guilherme Bernardino para aceder ao servidor.

5. Verificação da migração de dados

5.1 Consultas sobre a base de dados original

5.1.1. Nº de Customers

```
SELECT count([CUSTOMER KEY]) as [Number Of Customers On Old WWI]
FROM [WWI_DS].[dbo].[Customer]
```

	Number Of Customers On Old WWI
1	402

5.1.2. Nº de Customers por categoria

```
SELECT [Category] , count([CUSTOMER KEY]) as [Number Of Customers On Old WWI]
FROM [WWI_DS].[dbo].[Customer]
GROUP BY [Category]
```

	Category	Number Of Customers On Old WWI
1	24H Shop	73
2	Gas Station Shop	54
3	Gift Shop	62
4	Kiosk	62
5	Novelty Shop	151

5.1.3. Total de Vendas por Employee

```
SELECT Emp.[Employee], count(sa.[Salesperson Key]) FROM [WWI_DS].[dbo].[Sale] as sa
join [WWI_DS].[dbo].[Employee] as Emp on Emp.[Employee Key] = sa.[Salesperson Key]
group by Emp.[Employee]
```

1ª Fase Relatório Técnico – Complementos de Bases de Dados

	Employee	(No column name)
1	Amy Trefl	22444
2	Anthony Grosse	22521
3	Archer Lamble	23331
4	Hudson Hollinworth	22902
5	Hudson Onslow	22681
6	Jack Potter	22784
7	Kayla Woodcock	23079
8	Lily Code	22642

5.1.4. Total monetário de vendas por “Stock Item”

```
SELECT si.[Stock Item], sum(sale.[Quantity] * sale.[Unit Price]) as [MonetaryValue]
FROM [WWI_DS].[dbo].[Sale] as sale
JOIN [WWI_DS].[dbo].[Stock Item] as si on si.[Stock Item Key] = sale.[Stock Item Key]
group by si.[Stock Item]
```

	Stock Item	MonetaryValue
1	Fury gorilla with big eyes slippers (Black) XL	185024.00
2	Plush shark slippers (Gray) L	179424.00
3	Ogre battery-powered slippers (Green) L	189856.00
4	Red and white urgent heavy despatch tape 48mmx...	533032.80
5	Black and orange handle with care despatch tape 4...	565308.00
6	Clear packaging tape 48mmx100m	400750.00
7	DBA joke mug - mind if I join you? (White)	79768.00
8	USB food flash drive - donut	184859.20

5.1.5. Total monetário de vendas por ano por “Stock Item”

```
SELECT si.[Stock Item], YEAR(sale.[Delivery Date Key]) AS YearOfSale, sum(sale.[Quantity] * sale.[Unit Price]) as [MonetaryValue]
FROM [WWI_DS].[dbo].[Sale] as sale
JOIN [WWI_DS].[dbo].[Stock Item] as si on si.[Stock Item Key] = sale.[Stock Item Key]
group by si.[Stock Item], YEAR(sale.[Delivery Date Key])
order by YEAR(sale.[Delivery Date Key]) desc
```


1ª Fase Relatório Técnico – Complementos de Bases de Dados

	Stock Item	YearOfSale	MonetaryValue
1	Medium sized bubblewrap roll 20m	2016	150400.00
2	Furry animal socks (Pink) XL	2016	41640.00
3	Plush shark slippers (Gray) XL	2016	23424.00
4	Alien officer hoodie (Black) 4XL	2016	20825.00
5	Plush shark slippers (Gray) L	2016	20832.00
6	Clear packaging tape 48mmx75m	2016	67256.80
7	"The Gu" red shirt XML tag t-shir...	2016	145152.00
8	Developer joke mug - old C dev...	2016	8502.00

5.1.6. Total monetário de vendas por ano por "City"

```
SELECT c.[City], YEAR(sale.[Delivery Date Key]) AS YearOfSale, sum(sale.[Quantity] * sale.[Unit Price]) as  
[MonetaryValue]
```

```
FROM [WWI_DS].[dbo].[Sale] as sale
```

```
JOIN [WWI_DS].[dbo].[City] as c on c.[City Key] = sale.[City Key]
```

```
group by c.[City], YEAR(sale.[Delivery Date Key])
```

```
order by YEAR(sale.[Delivery Date Key]) desc
```

	City	YearOfSale	MonetaryValue
1	Twodot	2016	46996.10
2	Valdese	2016	18005.90
3	Cadogan	2016	36721.45
4	Raton	2016	30257.20
5	Bock	2016	8198.65
6	Airport ...	2016	38580.45
7	Guin	2016	35886.70
8	Newberg	2016	47676.80

5.2 Consultas sobre a nova base de dados

5.2.1. Nº de Customers

```
SELECT count(idCustomer) as [Number Of Customers On New WWI]
```

```
FROM [WWI].[dbo].[Customer]
```

1ª Fase Relatório Técnico – Complementos de Bases de Dados

	Number Of Customers On New WWI
1	402

5.2.2. Nº de Customers por categoria

```
SELECT ca.name , count(c.idCustomer) as NumberOfCustomersOnNewWWI
FROM [WWI].[dbo].[Customer] as c
JOIN [WWI].[dbo].[Category] as ca on ca.idCategory = c.categoryId
group by ca.name
```

	name	NumberOfCustomersOnNewWWI
1	24H Shop	73
2	Gas Station Shop	54
3	Gift Shop	62
4	Kiosk	62
5	Novelty Shop	151

5.2.3. Total de Vendas por Employee

```
SELECT Person.name , count(Sale.employeeId) as [Total of Sales]
FROM [WWI].[dbo].[Sale] as Sale
join [WWI].[dbo].[Employee] as Employee on Employee.idEmployee = Sale.employeeId
join [WWI].[dbo].[Person] as Person on Person.idPerson = Employee.personId
group by Person.name
```

	name	Total of Sales
1	Amy Trefl	22444
2	Anthony Grosse	22521
3	Archer Lamble	23331
4	Hudson Hollinworth	22902
5	Hudson Onslow	22681
6	Isabella Rupp	1
7	Jack Potter	22784
8	Kayla Woodcock	23079

5.2.4. Total monetário de vendas por “Stock Item”

1ª Fase Relatório Técnico – Complementos de Bases de Dados

```
SELECT p.description , sum(sp.quantity * p.unitPrice) as [MonetaryValue]
```

```
FROM [WWI].[dbo].[SaleProduct] as sp
```

```
JOIN [WWI].[dbo].[Product] as p on p.idProduct = sp.productId
```

```
group by p.description
```

- OU

```
SELECT p.description , sum(totalExcludingTax) as [MonetaryValue]
```

```
FROM [WWI].[dbo].[SaleProduct] as sp
```

```
JOIN [WWI].[dbo].[Product] as p on p.idProduct = sp.productId
```

```
group by p.description
```

	description	MonetaryValue
1	"The Gu" red shirt XML tag t-shirt (Black) 3XL	1246536,00
2	"The Gu" red shirt XML tag t-shirt (Black) 3XS	1213704,00
3	"The Gu" red shirt XML tag t-shirt (Black) 4XL	418392,00
4	"The Gu" red shirt XML tag t-shirt (Black) 5XL	1184328,00
5	"The Gu" red shirt XML tag t-shirt (Black) 6XL	1261440,00
6	"The Gu" red shirt XML tag t-shirt (Black) 7XL	1307448,00
7	"The Gu" red shirt XML tag t-shirt (Black) L	1195776,00
8	"The Gu" red shirt XML tag t-shirt (Black) M	1245456,00

5.2.5. Total monetário de vendas por ano por "Stock Item"

```
SELECT p.description , YEAR(s.deliveryDate) as YearOfSale, sum(sp.quantity * p.unitPrice) as [MonetaryValue]
```

```
FROM [WWI].[dbo].[SaleProduct] as sp
```

```
JOIN [WWI].[dbo].[Product] as p on p.idProduct = sp.productId
```

```
JOIN [WWI].[dbo].[Sale] as s on s.idSale = sp.saleId
```

```
group by p.description, YEAR(s.deliveryDate)
```

```
order by YEAR(s.deliveryDate) desc
```

- OU

```
SELECT p.description , YEAR(s.deliveryDate) as YearOfSale, sum(totalExcludingTax) as [MonetaryValue]
```

```
FROM [WWI].[dbo].[SaleProduct] as sp
```

```
JOIN [WWI].[dbo].[Product] as p on p.idProduct = sp.productId
```

```
JOIN [WWI].[dbo].[Sale] as s on s.idSale = sp.saleId
```

1ª Fase Relatório Técnico – Complementos de Bases de Dados

group by p.description, YEAR(s.deliveryDate)

order by YEAR(s.deliveryDate) desc

	description	YearOfSale	MonetaryValue
1	Packing knife with metal insert blade (Yellow) 18mm	2022	200,00
2	RC big wheel monster truck with remote control (B...	2016	31905,00
3	Plush shark slippers (Gray) M	2016	17856,00
4	Developer joke mug - there are 10 types of people...	2016	9373,00
5	Red and white urgent despatch tape 48mmx75m	2016	68109,60
6	RC toy sedan car with remote control (Red) 1/50 ...	2016	16950,00
7	"The Gu" red shirt XML tag t-shirt (White) 6XL	2016	167400,00
8	Developer joke mug - (hip, hip, array) (Black)	2016	9776,00

5.2.6. Total monetário de vendas por ano por "City"

```
SELECT c.name , YEAR(s.deliveryDate) as YearOfSale, sum(sp.quantity * p.unitPrice) as [MonetaryValue]
```

```
FROM [WWI].[dbo].[SaleProduct] as sp
```

```
JOIN [WWI].[dbo].[Product] as p on p.idProduct = sp.productId
```

```
JOIN [WWI].[dbo].[Sale] as s on s.idSale = sp.saleId
```

```
JOIN [WWI].[dbo].[InvoiceCustomer] as ic on ic.saleId = s.idSale
```

```
JOIN [WWI].[dbo].[Invoice] as i on i.idInvoice = ic.invoiceId
```

```
JOIN [WWI].[dbo].[City] as c on c.idCity = i.cityId
```

```
group by c.name, YEAR(s.deliveryDate)
```

```
order by YEAR(s.deliveryDate) desc
```

- OU

```
SELECT c.name , YEAR(s.deliveryDate) as YearOfSale, sum(totalExcludingTax) as [MonetaryValue]
```

```
FROM [WWI].[dbo].[SaleProduct] as sp
```

```
JOIN [WWI].[dbo].[Product] as p on p.idProduct = sp.productId
```

```
JOIN [WWI].[dbo].[Sale] as s on s.idSale = sp.saleId
```

```
JOIN [WWI].[dbo].[InvoiceCustomer] as ic on ic.saleId = s.idSale
```

```
JOIN [WWI].[dbo].[Invoice] as i on i.idInvoice = ic.invoiceId
```

```
JOIN [WWI].[dbo].[City] as c on c.idCity = i.cityId
```

```
group by c.name, YEAR(s.deliveryDate)
```

1ª Fase Relatório Técnico – Complementos de Bases de Dados

order by YEAR(s.deliveryDate) desc

	name	YearOfSale	MonetaryValue
1	North Eastham	2016	27551,80
2	Bratt	2016	40995,75
3	Bokeelia	2016	23158,20
4	Bakers Mill	2016	49482,00
5	Drakesboro	2016	23206,60
6	Shell	2016	16258,60
7	Schoharie	2016	22058,65
8	Willow Valley	2016	13047,00

6. Programação

6.1 Views

Nome	Descrição
dbo.ViewRecordOfWWI	Vizualizar a inserção mais recente de dados na tabela de Records, ou seja, Select na tabela RecordOfWWI
dbo.ViewNumberOfRecords	Vizualizar a inserção mais recente de dados na tabela de Records, ou seja, Select na tabela NumberOfRecords

6.2 Functions

Nome	Atributos	Requisito	Descrição
dbo.returnIfSaleEmpty	@idSale int	R12	Função que retorna um valor verdadeiro ou falso se a venda conter produtos ou não
dbo.totalSalePriceFunction	@saleId int	R13	Função que retorna o valor total de uma venda (sum de todos os totais por produto na venda)
dbo.verifyLeadTimeDaysFunction	@idSale int	R14	Função que retorna uma mensagem caso a diferença entre data de entrega e a data corrente de uma venda não esteja de acordo com o leadTimeDays de um produto

1ª Fase Relatório Técnico – Complementos de Bases de Dados

dbo.verifySaleContainsChillerStockFunction	@idSale int	R15	Função que verifica se uma venda contém produtos com chiller stock e sem chiller stock ao mesmo tempo
--	-------------	-----	---

6.3 Stored procedures

Nome	Atributos	Requisito	Descrição
dbo.Person_insertInto	@name nvarchar(100) @preferredName nvarchar(50) @isSalesPerson bit @email nvarchar(50) @password nvarchar(50) @photo varbinary(max) @isActive bit @personTypeId int	R02	Stored Procedure que cria um utilizador (Person)
dbo.Person_update	@idPerson int @name nvarchar(100) @preferredName nvarchar(50) @isSalesPerson bit @email nvarchar(50) @password nvarchar(50) @photo varbinary(max) @isActive bit @personTypeId int	R03	Stored Procedure que atualiza um utilizador (Person)
dbo.Person_removeFrom	@idPerson int	R04	Stored Procedure que remove um utilizador (Person)

1ª Fase Relatório Técnico – Complementos de Bases de Dados

dbo.newDiscount	@productId int @discountPercentage int @beginDateGDH datetime @endDateGDH datetime	R06	Stored Procedure que cria uma promoção (Discount)
dbo.updateDiscountDates	@idDiscount int, @beginDateGDH datetime, @endDateGDH datetim	R07	Stored Procedure que atualiza as datas de início e fim de uma promoção (Discount)
dbo.createSale	@idSale int, @deliveryDate datetime @employeeId int @notes nvarchar(500)	R08	Stored Procedure que cria uma venda (Sale)
dbo.addProductToSale	@productId int @saleId int @quantity int @description nvarchar(250) @totalExcludingTax money @taxAmount money @profit money @totalIncludingTax money @totalDryItens int @totalChillerItens int	R09	Stored Procedure que cria uma venda relacionada a um produto (SaleProduct)
dbo.updateProductQuantity OnSale	@idSaleProduct int @quantity int @totalExcludingTax money	R10	Stored Procedure que atualiza a quantidade de um produto da venda (SaleProduct)

1ª Fase Relatório Técnico – Complementos de Bases de Dados

	@taxAmount money @profit money @totalIncludingTax money @totalDryItens int @totalChillerItens int		
dbo.deleteFromSaleProduct Table	@idSaleProduct int @idSale int	R11	Stored Procedure que remove uma relacionada a um produto (SaleProduct)
dbo.totalSalePriceProcedure	@saleId int	R13	Stored Procedure que retorna o valor total de uma venda (sum de todos os totais por produto na venda)
dbo.authenticateUser	@email nvarchar(50) @password nvarchar(50)	R01	Stored Procedure para verificar a existência de um utilizador com através de email e password
dbo.tokenGeneration	@email nvarchar(50)	R05	Stored Procedure que gera um token na tabela TokenPassword para recuperar a password agregada a um email da conta de utilizador

6.4 Triggers

Nome	Tipo	Tabela	Requisito	Descrição
dbo.dbo_Insert_User_Email_TriggerDuplicate	AFTER INSERT	[dbo].[Person]	RM01	Trigger para verificar a existência de emails duplicados em utilizadores (Person)

7. Catálogo/Metadados

7.1 Geradores

1ª Fase Relatório Técnico – Complementos de Bases de Dados

Nome	Atributos	Descrição
dbo.sp_insertProcedure	@table_name NVARCHAR(255)	Gera um sp de Insert Into @table_name, e obtém e armazena informação em variáveis da tabela escolhida: parâmetros, colunas e tipos de dados associados
dbo.sp_updateProcedure	@table_name NVARCHAR(255)	Gera um sp de Update @table_name, e obtém e armazena informação em variáveis da tabela escolhida: parâmetros, coluna e tipo de dados no ID, colunas e tipos de dados associados
dbo.sp_deleteProcedure	@table_name NVARCHAR(255)	Gera um sp de Delete From @table_name, e obtém e armazena informação em variáveis da tabela escolhida: parâmetro ID, coluna do ID e tipo de dado no ID

7.2 Monitorização

Nome	Atributos	Descrição
dbo.sp_RecordOfDatabase	Nenhum	Cria uma tabelas dedicadas (se não existir) que guarda os registos e os registos anteriores de cada tabela da base de dados (nome da tabela, colunas, etc). Se já existirem, atualiza os dados das tabelas criadas.
dbo. sp_NumberOfRecords	Nenhum	Cria uma tabela (se não existir) que guarda o número de registos de cada tabela da base de dados. Se existir, guarda os dados anteriores na coluna RowCount_Previous.

8. Descrição da Demonstração

8.1 Script de demonstração

-- Procedures

-- Person_insertInto

EXEC [WWI].[dbo].Person_insertInto 'Luís', 'Luís', 0, 'lui.luis@hotmail.com', '123', null, 0, 1

SELECT * FROM [WWI].[dbo].[Person]

1ª Fase Relatório Técnico – Complementos de Bases de Dados

WHERE name = 'Luís'

	idPerson	name	preferredName	isSalesPerson	email	password	photo	isActive	personTypeId
1	615	Luis	Luis	0	lui.luis@hotmail.com	123	NULL	0	1

-- Person_update

EXEC [WWI].[dbo].Person_update 615, 'Luís', 'Lui', 0, 'lui.luis@hotmail.com', '123', null, 0, 2

SELECT * FROM [WWI].[dbo].[Person]

WHERE name = 'Luís'

	idPerson	name	preferredName	isSalesPerson	email	password	photo	isActive	personTypeId
1	615	Luis	Lui	0	lui.luis@hotmail.com	123	NULL	0	2

-- Person_removeFrom

EXEC [WWI].[dbo].Person_removeFrom 615

SELECT * FROM [WWI].[dbo].[Person]

WHERE name = 'Luís'

	idPerson	name	preferredName	isSalesPerson	email	password	photo	isActive	personTypeId
--	----------	------	---------------	---------------	-------	----------	-------	----------	--------------

-- newDiscount

EXEC [WWI].[dbo].newDiscount 1, 10, '2022-11-20 00:00:00', '2022-11-30 00:00:00'

SELECT * FROM [WWI].[dbo].[Discount]

WHERE productId = 1

	idDiscount	productId	discountPercentage	beginDateGDH	endDateGDH
1	1	1	10	2022-11-20 00:00:00.000	2022-11-30 00:00:00.000

-- updateDiscountDates

EXEC [WWI].[dbo].updateDiscountDates 1, '2022-11-25 00:00:00', '2022-12-05 00:00:00'

SELECT * FROM [WWI].[dbo].[Discount]

WHERE productId = 1

1ª Fase Relatório Técnico – Complementos de Bases de Dados

	idDiscount	productId	discountPercentage	beginDateGDH	endDateGDH
1	1	1	10	2022-11-25 00:00:00.000	2022-12-05 00:00:00.000

-- createSale

EXEC [WWI].[dbo].createSale 600000, '2022-11-25 00:00:00', 2, 'Notas'

SELECT * FROM [WWI].[dbo].[Sale]

where idSale = 600000

	idSale	deliveryDate	employeeId	notes
1	600000	2022-11-25 00:00:00.000	2	Notas

-- addProductToSale

EXEC [WWI].[dbo].addProductToSale 10, 600000, 5, 5, 200, 10, 0, 220, 0, 0

SELECT * FROM [WWI].[dbo].[SaleProduct]

WHERE saleId = 600000

	idSaleProduct	productId	saleId	quantity	description	totalExcludingTax	taxAmount	profit	totalIncludingTax	totalDryItens	totalChillerItens
1	228266	10	600000	5	5	200,00	10,00	0,00	220,00	0	0

-- updateProductQuantityOnSale

EXEC [WWI].[dbo].updateProductQuantityOnSale 228266, 12, 0,0, 0, 0, 0, 0

SELECT * FROM [WWI].[dbo].[SaleProduct]

WHERE saleId = 600000

	idSaleProduct	productId	saleId	quantity	description	totalExcludingTax	taxAmount	profit	totalIncludingTax	totalDryItens	totalChillerItens
1	228266	10	600000	12	5	0,00	0,00	0,00	0,00	0	0

-- deleteFromSaleProductTable

EXEC [WWI].[dbo].deleteFromSaleProductTable 228266, 600000

SELECT * FROM [WWI].[dbo].[SaleProduct]

WHERE saleId = 600000

idSaleProduct	productId	saleId	quantity	description	totalExcludingTax	taxAmount	profit	totalIncludingTax	totalDryItens	totalChillerItens
---------------	-----------	--------	----------	-------------	-------------------	-----------	--------	-------------------	---------------	-------------------

-- totalSalePriceProcedure

1ª Fase Relatório Técnico – Complementos de Bases de Dados

EXEC [WWI].[dbo].totalSalePriceProcedure 600000

	Sale ID	Total Sale Price Taxed	Total Price NotTax
1	600000	220,00	200,00

-- authenticateUser

EXEC [WWI].[dbo].authenticateUser 'lui.luis@hotmail.com', '123'

Aunthentication valid!

(1 row affected)

Completion time: 2022-11-21T19:31:42.0180708+00:00

SELECT * FROM [WWI].[dbo].Person

WHERE email = 'lui.luis@hotmail.com'

SELECT * FROM [WWI].[dbo].AuthenticationEvent

	idPerson	name	preferredName	isSalesPerson	email	password	photo	isActive	personTypeId
1	1233	Luis	Luis	0	lui.luis@hotmail.com	123	NULL	0	2

	idAuthenticationEvent	personId	authenticationGDH	ipAddress	actionResult
1	1	1233	2022-11-21 19:30:05.187	0.0.0.0	1
2	2	1233	2022-11-21 19:31:41.147	0.0.0.0	1

-- tokenGeneration

EXEC [WWI].[dbo].tokenGeneration 'lui.luis@hotmail.com'

SELECT * FROM [WWI].[dbo].TokenPassword

	idTokenPassword	personId	token	generationGDH
1	1	1233	7B129EEB-57ED-423E-9208-493E9A293470	2022-11-21 19:32:33.143

-- Functions

-- returnIfSaleEmpty

select [WWI].[dbo].returnIfSaleEmpty (600000)

	(No column name)
1	0

----> not empty

1ª Fase Relatório Técnico – Complementos de Bases de Dados

-- totalSalePriceFunction

select [WWI].[dbo].totalSalePriceFunction (600000) [Total Price Including Tax]

	Total Price Including Tax
1	220

-- verifyLeadTimeDaysFunction

select [WWI].[dbo].verifyLeadTimeDaysFunction (600000)

	(No column name)
1	Lead Time Days doesnt correspond with delivery date!

-- verifySaleContainsChillerStockFunction

select [WWI].[dbo].verifySaleContainsChillerStockFunction (600000)

	(No column name)
1	0

SELECT * FROM [WWI].[dbo].[SaleProduct]

join [WWI].[dbo].[Sale] as sp on sp.idSale = saleId

WHERE saleId = 600000

	idSaleProduct	productId	saleId	quantity	description	totalExcludingTax	taxAmount	profit	totalIncludingTax	totalDryItens	totalChillerItens	idSale	deliveryDate	employeeId	notes
1	228267	10	600000	5	5	200,00	10,00	0,00	220,00	0	0	600000	2022-11-25 00:00:00.000	2	Notas

--Triggers

-- dbo_Insert_User_Email_TriggerDuplicate

INSERT INTO [WWI].[dbo].[Person] VALUES ('Guilherme', 'Gui', 0, 'guib@hotmail.com', '123', null, 0, 1)

SELECT * FROM [WWI].[dbo].[Person]

WHERE name = 'Guilherme'

	idPerson	name	preferredName	isSalesPerson	email	password	photo	isActive	personTypeId
1	616	Guilherme	Gui	0	guib@hotmail.com	123	NULL	0	1

--> 1º Insert

Msg 50000, Level 16, State 1, Procedure dbo_Insert_User_Email_TriggerDuplicate, Line 8 [Batch Start Line 81]
Email already exists!

Msg 3609, Level 16, State 1, Line 82

The transaction ended in the trigger. The batch has been aborted.

-->2º Insert

-- Metadata/Catalog Procedures

Ano Letivo 2022/23

Pág.: 21 de 26

1ª Fase Relatório Técnico – Complementos de Bases de Dados

--Insert Generator

EXEC [WWI].[dbo].sp_insertProcedure 'Color'

```
CREATE PROCEDURE Color_Insert (@idColor int,@name nvarchar(100))
AS BEGIN
    INSERT INTO Color (idColor,name)
    VALUES (@idColor,@name)
END
```

Completion time: 2022-11-20T23:16:20.4664990+00:00

EXEC Color_Insert '10', 'NavyBlue'

SELECT * FROM [WWI].[dbo].[Color]

	idColor	name
1	0	N/A
2	1	Black
3	2	Blue
4	3	Gray
5	4	Light Brown
6	5	Red
7	6	Steel Gray
8	7	White
9	8	Yellow
10	10	NavyBlue

--Update Generator

EXEC [WWI].[dbo].sp_updateProcedure 'Color'

```
CREATE PROCEDURE Color_Update (@idColor int,@name nvarchar(100))
AS BEGIN
    UPDATE Color
    SET idColor=@idColor,name=@name
    WHERE (idColor = @idColor)
END
```

Completion time: 2022-11-20T23:17:34.2665737+00:00

EXEC Color_Update '10', 'LightBlue'

SELECT * FROM [WWI].[dbo].[Color]

1ª Fase Relatório Técnico – Complementos de Bases de Dados

	idColor	name
1	0	N/A
2	1	Black
3	2	Blue
4	3	Gray
5	4	Light Brown
6	5	Red
7	6	Steel Gray
8	7	White
9	8	Yellow
10	10	LightBlue

-- Remove Generator

EXEC [WWI].[dbo].sp_deleteProcedure 'Color'

```
CREATE PROCEDURE Color_Delete (@idColor int)
AS BEGIN DELETE FROM Color
WHERE (idColor = @idColor)
END
```

Completion time: 2022-11-20T23:18:05.7229621+00:00

EXEC Color_Delete '10'

SELECT * FROM [WWI].[dbo].[Color]

	idColor	name
1	0	N/A
2	1	Black
3	2	Blue
4	3	Gray
5	4	Light Brown
6	5	Red
7	6	Steel Gray
8	7	White
9	8	Yellow

-- Record Database

EXEC [WWI].[dbo].sp_RecordOfDatabase

1ª Fase Relatório Técnico – Complementos de Bases de Dados

	TABLE_CATALOG	TABLE_SCHEMA	COLUMN_NAME	COLLATION_NAME	IS_NULLABLE	DATA_TYPE	CHARACTER_MAXIMUM_LENGTH	CONSTRAINT_CATALOG	CONSTRAINT_SCHEMA	CONSTRAINT_NAME
1	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_AuthenticationEvent_Person
2	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_AuthenticationEvent_Person
3	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_AuthenticationEvent_Person
4	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_AuthenticationEvent_Person
5	WWI	dbo	stateId	Latin1_General_CI_AS	NO	varchar	2	WWI	dbo	FK_City_State
6	WWI	dbo	continentId	NULL	NO	int	NULL	WWI	dbo	FK_Country_Continent
7	WWI	dbo	buyingGroupId	NULL	NO	int	NULL	WWI	dbo	FK_Customer_BuyingGroup
8	WWI	dbo	categoryId	NULL	NO	int	NULL	WWI	dbo	FK_Customer_Category
9	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_Customer_Person

	TABLE_CATALOG	TABLE_SCHEMA	COLUMN_NAME	COLLATION_NAME	IS_NULLABLE	DATA_TYPE	CHARACTER_MAXIMUM_LENGTH	CONSTRAINT_CATALOG	CONSTRAINT_SCHEMA	CONSTRAINT_NAME
1	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_AuthenticationEvent_Person
2	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_AuthenticationEvent_Person
3	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_AuthenticationEvent_Person
4	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_AuthenticationEvent_Person
5	WWI	dbo	stateId	Latin1_General_CI_AS	NO	varchar	2	WWI	dbo	FK_City_State
6	WWI	dbo	continentId	NULL	NO	int	NULL	WWI	dbo	FK_Country_Continent
7	WWI	dbo	buyingGroupId	NULL	NO	int	NULL	WWI	dbo	FK_Customer_BuyingGroup
8	WWI	dbo	categoryId	NULL	NO	int	NULL	WWI	dbo	FK_Customer_Category
9	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_Customer_Person

SELECT * FROM ViewRecordOfWWI

	TABLE_CATALOG	TABLE_SCHEMA	COLUMN_NAME	COLLATION_NAME	IS_NULLABLE	DATA_TYPE	CHARACTER_MAXIMUM_LENGTH	CONSTRAINT_CATALOG	CONSTRAINT_SCHEMA	CONSTRAINT_NAME
1	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_AuthenticationEvent_Person
2	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_AuthenticationEvent_Person
3	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_AuthenticationEvent_Person
4	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_AuthenticationEvent_Person
5	WWI	dbo	stateId	Latin1_General_CI_AS	NO	varchar	2	WWI	dbo	FK_City_State
6	WWI	dbo	continentId	NULL	NO	int	NULL	WWI	dbo	FK_Country_Continent
7	WWI	dbo	buyingGroupId	NULL	NO	int	NULL	WWI	dbo	FK_Customer_BuyingGroup
8	WWI	dbo	categoryId	NULL	NO	int	NULL	WWI	dbo	FK_Customer_Category
9	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_Customer_Person
10	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_Customer_Person
11	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_Customer_Person
12	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_Customer_Person
13	WWI	dbo	productId	NULL	NO	int	NULL	WWI	dbo	FK_Discount_Product
14	WWI	dbo	productId	NULL	NO	int	NULL	WWI	dbo	FK_Discount_Product
15	WWI	dbo	productId	NULL	NO	int	NULL	WWI	dbo	FK_Discount_Product
16	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_Employee_Person
17	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_Employee_Person
18	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_Employee_Person
19	WWI	dbo	personId	NULL	NO	int	NULL	WWI	dbo	FK_Employee_Person

-- Record number of records per table

EXEC [WWI].[dbo].sp_NumberOfRecords

1ª Fase Relatório Técnico – Complementos de Bases de Dados

	TableName	Row_Count	previousRowCount
1	AuthenticationEvent	0	0
2	BuyingGroup	2	2
3	Category	5	5
4	City	116294	116294
5	Color	9	9
6	Continent	7	7
7	Country	249	249
8	Customer	402	402
9	Discount	1	0

	TableName	NewCount
1	AuthenticationEvent	0
2	BuyingGroup	2
3	Category	5
4	City	116294
5	Color	9
6	Continent	7
7	Country	249
8	Customer	402
9	Discount	1

SELECT * FROM ViewNumberOfRecords

	TableName	Row_Count	previousRowCount
1	AuthenticationEvent	0	0
2	BuyingGroup	2	2
3	Category	5	5
4	City	116294	116294
5	Color	9	9
6	Continent	7	7
7	Country	249	249
8	Customer	402	402
9	Discount	1	0
10	Employee	212	212
11	Invoice	70510	70510
12	InvoiceCustomer	228265	228265
13	NumberOfRecords	25	25
14	Package	6	6
15	Person	615	614
16	PersonType	2	2
17	Product	671	671
18	RecordOfWWI	75	75
19	RecordOfWWI_Previous	75	75

9. Conclusões

Com este relatório pode-se concluir o desenvolvimento e a administração de uma Base de Dados, destinada a uma empresa que trabalha sobre o tema de importações e exportações.

Pode se concluir também, ao longo do projeto, como uma base de dados deve ser organizada em termos de Layouts e ficheiros, assim como devem ser desenvolvidas as suas stored procedures, triggers etc.

No decorrer desta primeira fase o grupo encontrou diversas dificuldades tais como:

- Qual seria a melhor abordagem para o desenvolvimento do modelo de dados;
- Que Stored Procedures seriam significativas para o cliente;
- Quais as Views que seriam pertinentes para o cliente;
- Que tamanho deveriam limitar Files e Filesgroups.

Estas dificuldades listadas foram algumas de muitas, aos quais o grupo através de pesquisa, e assistência de colegas/professores, conseguiram ultrapassar.