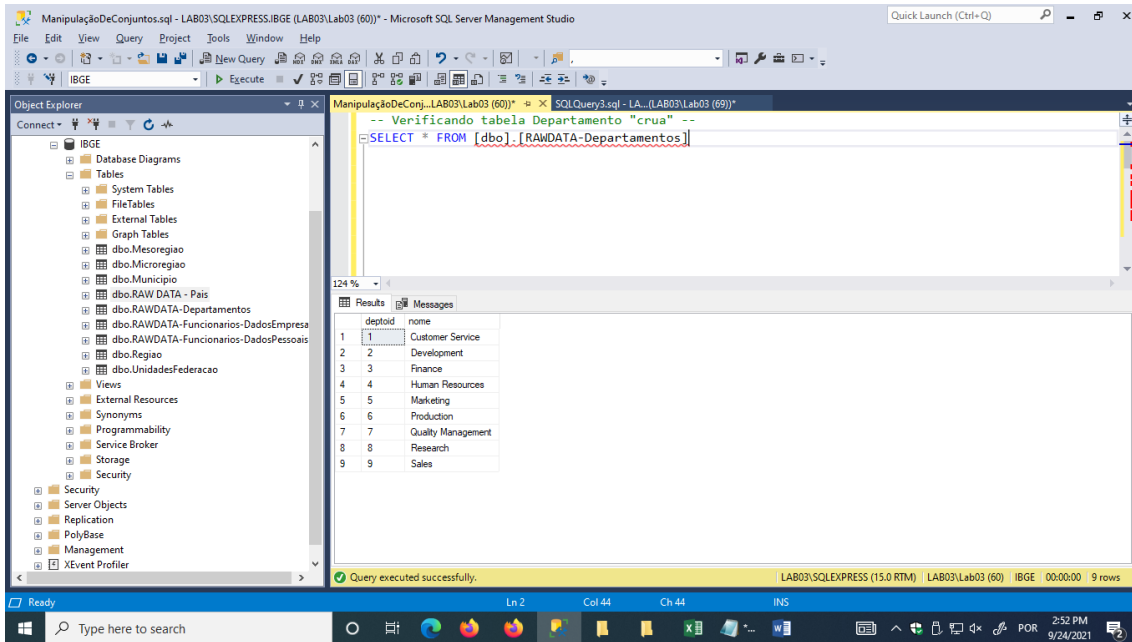


Passo 1 - Colhendo os dados “Crus”



The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left displays the database structure for 'IBGE'. The query editor in the center contains the following SQL code:

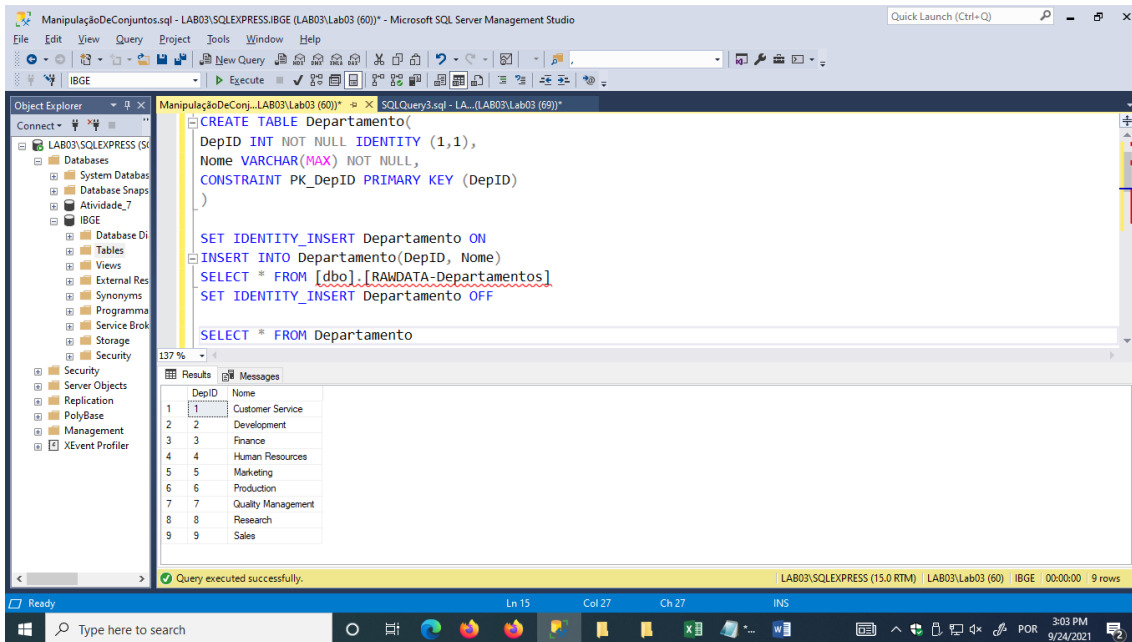
```
-- Verificando tabela Departamento "crua" --  
SELECT * FROM [dbo].[RAWDATA-Departamentos]
```

The Results pane at the bottom shows the output of the query, which is a table with 9 rows and 2 columns: 'deptid' and 'nome'.

deptid	nome
1	Customer Service
2	Development
3	Finance
4	Human Resources
5	Marketing
6	Production
7	Quality Management
8	Research
9	Sales

The status bar at the bottom indicates that the query was executed successfully, returning 9 rows.

Passo 2 - Criando a nova tabela Departamento, inserindo e pesquisando os dados.



The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor in the center contains the following SQL code:

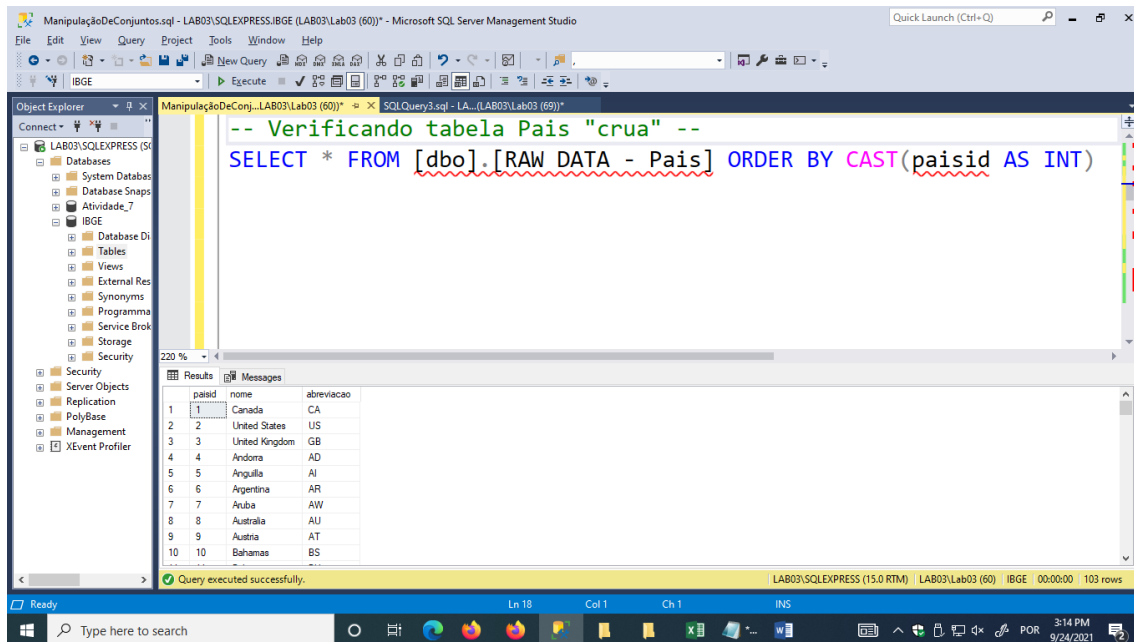
```
CREATE TABLE Departamento(  
    DepID INT NOT NULL IDENTITY (1,1),  
    Nome VARCHAR(MAX) NOT NULL,  
    CONSTRAINT PK_DepID PRIMARY KEY (DepID)  
)  
  
SET IDENTITY_INSERT Departamento ON  
INSERT INTO Departamento(DepID, Nome)  
SELECT * FROM [dbo].[RAWDATA-Departamentos]  
SET IDENTITY_INSERT Departamento OFF  
  
SELECT * FROM Departamento
```

The Results pane at the bottom shows the output of the query, which is a table with 9 rows and 2 columns: 'DepID' and 'Nome'.

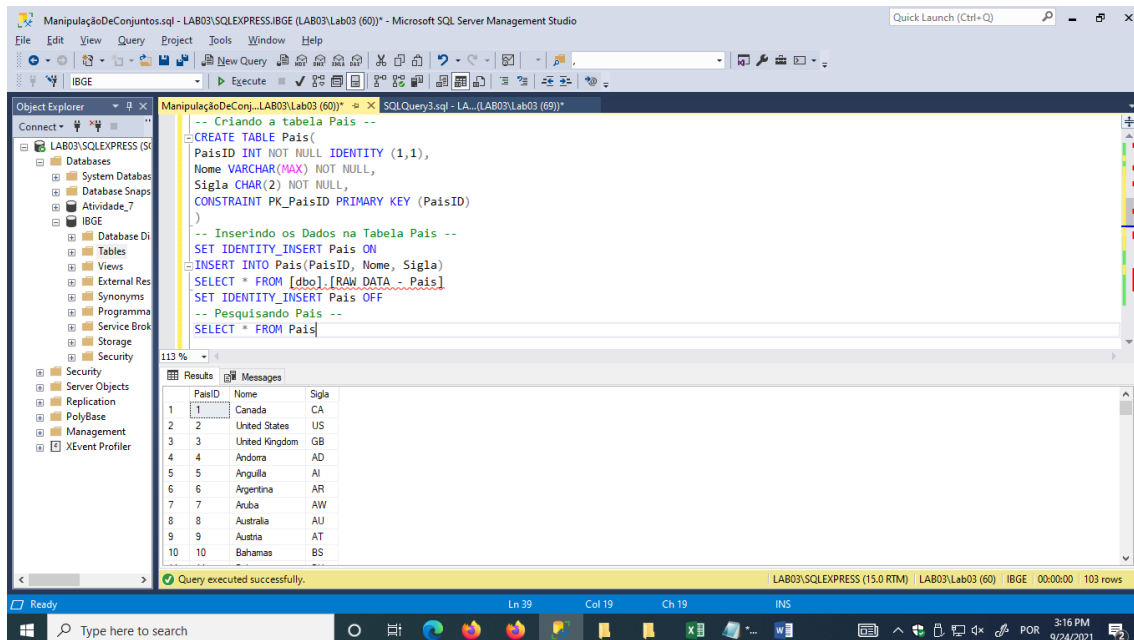
DepID	Nome
1	Customer Service
2	Development
3	Finance
4	Human Resources
5	Marketing
6	Production
7	Quality Management
8	Research
9	Sales

The status bar at the bottom indicates that the query was executed successfully.

Passo 1.1 - Colhendo os dados “Crus”



Passo 2.1 - Criando a nova tabela Pais, inserindo e pesquisando os dados.



Passo 1.2 - Colhendo os dados "Crus"

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
-- Verificando tabela Funcionarios-DadosEmpresa "crua" --  
SELECT * FROM [dbo].[RAWDATA-Funcionarios-DadosEmpresa]
```

The query has been executed successfully, returning 120,670 rows. The results grid displays the following columns:

	chave	nome	sobrenome	dataadmissao	ctps	ctpsnum	ctpsserie
1	20100150001	Georgi	Facello	1986-06-26 00:00:00.0000000	49001540741/380	49001540741	380
2	20100150002	Bezel	Simmel	1985-11-21 00:00:00.0000000	38050201548/584	38050201548	584
3	20100150003	Pato	Bamford	1986-08-28 00:00:00.0000000	45152038868/403	45152038868	403
4	20100150004	Christian	Koblick	1986-12-01 00:00:00.0000000	93993062385/891	93993062385	891
5	20100150005	Kyoichi	Maliak	1989-09-12 00:00:00.0000000	64885131268/229	64885131268	229
6	20100150006	Anneke	Preusig	1989-06-02 00:00:00.0000000	32097781440/565	32097781440	565
7	20100150007	Tzvetan	Zielinski	1989-02-10 00:00:00.0000000	45521773009/130	45521773009	130
8	20100150008	Saniya	Kaloufi	1994-09-15 00:00:00.0000000	61346916570/146	61346916570	146
9	20100150009	Sumant	Peac	1985-02-18 00:00:00.0000000	97392551134/059	97392551134	059
10	20100150010	Duangkaew	Piveteau	1989-08-24 00:00:00.0000000	67883327357/459	67883327357	459

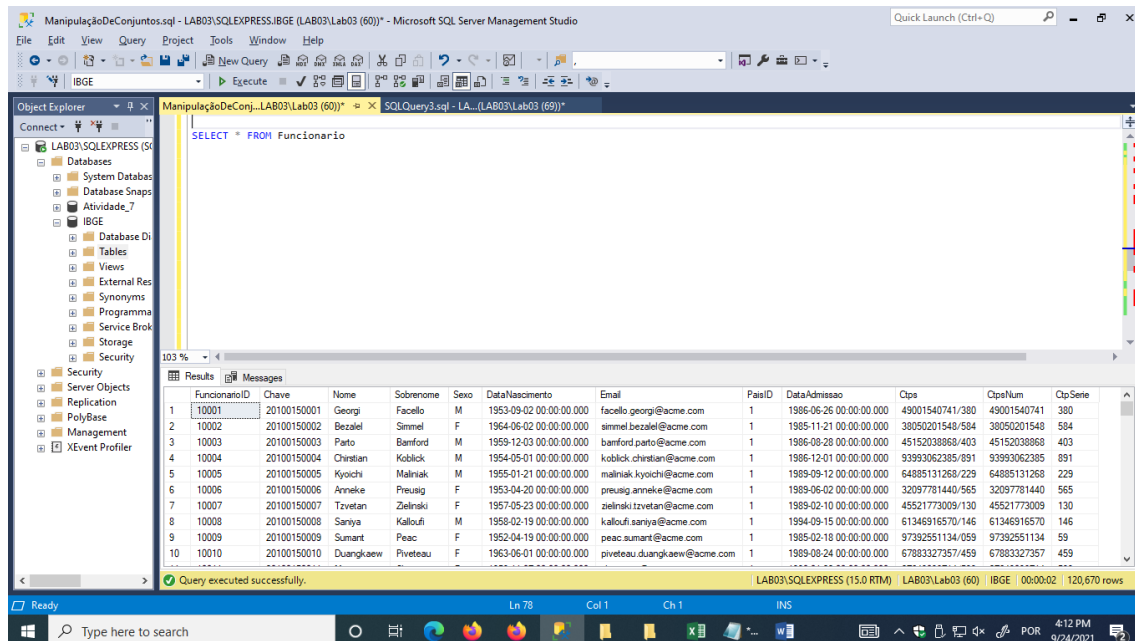
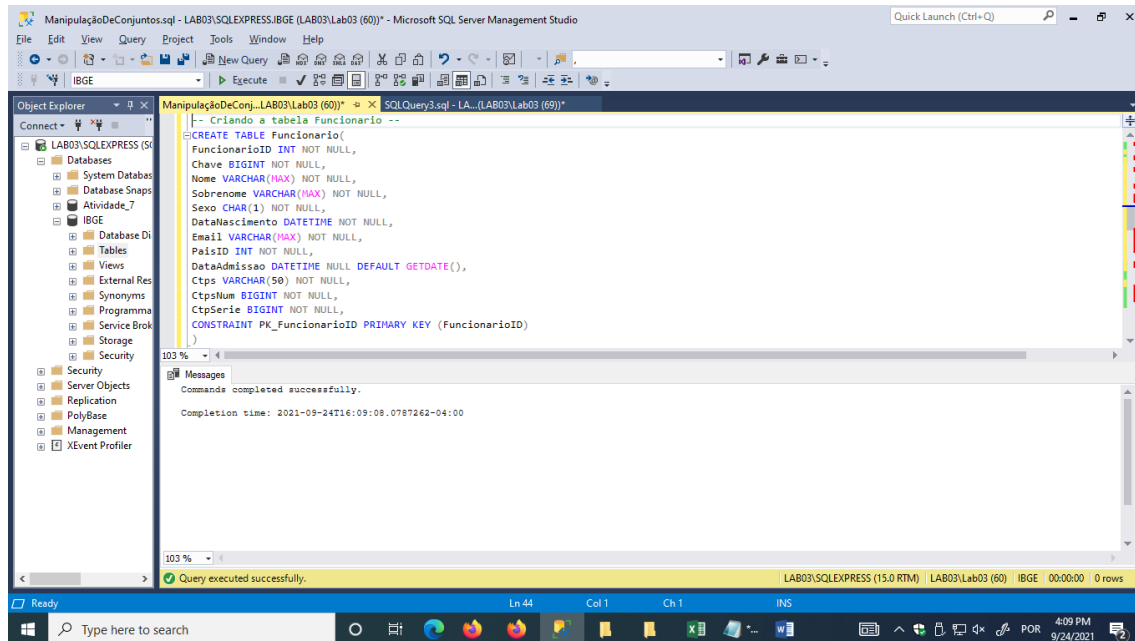
The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
-- Verificando tabela Funcionarios-DadosPessoais "crua" --  
SELECT * FROM [dbo].[RAWDATA-Funcionarios-DadosPessoais]
```

The query has been executed successfully, returning 120,670 rows. The results grid displays the following columns:

	funcid	chave	nome	sobrenome	sexo	datanascimento	email	paisid
1	10001	20100150001	Georgi	Facello	M	1986-06-26 00:00:00.0000000	facello.georgi@acme.com	1
2	10002	20100150002	Bezel	Simmel	F	1985-11-21 00:00:00.0000000	simmel.bezel@acme.com	1
3	10003	20100150003	Pato	Bamford	M	1986-08-28 00:00:00.0000000	bamford.pato@acme.com	1
4	10004	20100150004	Christian	Koblick	M	1986-12-01 00:00:00.0000000	koblick.christian@acme.com	1
5	10005	20100150005	Kyoichi	Maliak	M	1989-09-12 00:00:00.0000000	maliak.kyoichi@acme.com	1
6	10006	20100150006	Anneke	Preusig	F	1989-06-02 00:00:00.0000000	preusig.anneke@acme.com	1
7	10007	20100150007	Tzvetan	Zielinski	F	1989-02-10 00:00:00.0000000	zielinski.tzvetan@acme.com	1
8	10008	20100150008	Saniya	Kaloufi	M	1994-09-15 00:00:00.0000000	kaloufi.saniya@acme.com	1
9	10009	20100150009	Sumant	Peac	F	1985-02-18 00:00:00.0000000	peac.sumant@acme.com	1
10	10010	20100150010	Duangkaew	Piveteau	F	1989-08-24 00:00:00.0000000	piveteau.duangkaew@acme.com	1

Passo 2.2 - Criando a nova tabela Funcionário, inserindo e pesquisando os dados.



Passo 3 - Criando a VIEW e exibindo os dados dos funcionários.

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
CREATE VIEW ExibeFuncionario
AS
SELECT Funcionario.*, Pais.Nome AS NomePais FROM Funcionario
INNER JOIN Pais
ON Funcionario.Paisid = Pais.PaisID
GO

SELECT * FROM ExibeFuncionario
```

The Results pane displays a table with 13 columns: FuncionarioID, Chave, Nome, Sobrenome, Sexo, DataNascimento, Email, PaisID, DataAdmissao, DataInss, Ctps, CtpsNum, CtpSerie, and Nome. The table contains 18 rows of data, including employee details and their associated company information.

Passo 3.1 - Criando a VIEW e exibindo os dados de empresa dos funcionários.

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
-- Criando a VIEW --
CREATE VIEW ExibeEmpresa
AS
SELECT FuncionarioID, Chave, DataAdmissao, Ctps, CtpsNum, CtpSerie FROM Funcionario
-- Exibindo os dados da VIEW --
SELECT * FROM ExibeEmpresa
```

The Results pane displays a table with 6 columns: FuncionarioID, Chave, DataAdmissao, Ctps, CtpsNum, and CtpSerie. The table contains 13 rows of data, showing employee IDs, keys, and company-related information.

Passo 4 - Criando a PROCEDURE para buscar e exibir os dados dos funcionários a partir do FuncionarioID.

The screenshot shows the SQL Server Management Studio interface. The query editor contains the following T-SQL code:

```
-- Criação da Procedure para buscar Funcionário --  
CREATE PROCEDURE BuscaFuncionario  
@FuncionarioID INT  
AS  
SELECT * FROM Funcionario  
WHERE (@FuncionarioID = Funcionario.FuncionarioID)  
-- Execução da Procedure para buscar Funcionário --  
EXEC BuscaFuncionario 10001
```

The Results pane shows the output of the query, displaying a single row of employee data:

FuncionarioID	Chave	Nome	Sobrenome	Sexo	DataNascimento	Email	PaisID	DataAdmissao	DataInsert	Ctps	CtpsNum	CtpSerie
1	10001	Georgi	Facello	M	1953-09-02 00:00:00.000	facello.georgi@scme.com	1	1986-06-26 00:00:00.000	2021-09-24 16:31:11.040	49001540741/380	49001540741	380

The status bar at the bottom indicates "Query executed successfully." and "1 rows".

Passo 4.1 - Criando a PROCEDURE para buscar e exibir os dados dos funcionários a partir da Chave.

The screenshot shows the SQL Server Management Studio interface. The query editor contains the following T-SQL code:

```
-- Criação da Procedure para buscar Funcionário pela chave --  
CREATE PROCEDURE BuscaFuncionarioChave  
@FuncionarioChave BIGINT  
AS  
SELECT * FROM Funcionario  
WHERE (Funcionario.Chave = @FuncionarioChave)  
-- Execução da Procedure para buscar Funcionário --  
EXEC BuscaFuncionarioChave 20100150001
```

The Results pane shows the output of the query, displaying a single row of employee data:

FuncionarioID	Chave	Nome	Sobrenome	Sexo	DataNascimento	Email	PaisID	DataAdmissao	DataInsert
1	10001	Georgi	Facello	M	1953-09-02 00:00:00.000	facello.georgi@scme.com	1	1986-06-26 00:00:00.000	2021-09-24 16:31:11.040

The status bar at the bottom indicates "Query executed successfully." and "1 rows".