



# **Relatório Trabalho Prático 1**

## **Integração de Sistemas de Informação**

**Aluno:**

**Thiago Yabuki de Araujo – a24207**

**Docente:**

**Luís Ferreira**

**Licenciatura em Engenharia de Sistemas Informáticos**

**3º ano**

**Barcelos | Outubro, 2024**



FIGURA 1 - DESCRIÇÃO PARA A IA DO MOCKAROO .....	6
FIGURA 2 - GERAÇÃO DOS DADOS UTILIZANDO A IA DO MOCKAROO .....	6
FIGURA 3 - TRANSFORMAÇÃO DE FICHEIRO CSV PARA XML COM INSERTS PARA A BASE DE DADOS.....	8
FIGURA 4 - SCRIPT SQL .....	9
FIGURA 5 - CAPTURA DE TELA DO FICHEIRO XML ANTES DA ALTERAÇÃO .....	9
FIGURA 6 - ALTERAÇÃO DOS DADOS PARA "INTEGER" .....	10
FIGURA 7 - CAPTURA DE TELA DO FICHEIRO XML APÓS A ALTERAÇÃO .....	10
FIGURA 8 - DETALHES DA SEPARAÇÃO DO CAMPO FULL_NAME PARA FIRST_NAME E LAST_NAME .....	10
FIGURA 9 - DETALHES DA FORMATAÇÃO DAS LETRAS MAIÚSCULAS .....	11
FIGURA 10 - DETALHES DA IMPORTAÇÃO DO FICHEIRO CSV.....	11
FIGURA 11 - DETALHES DA CRIAÇÃO DO FICHEIRO XML .....	12
FIGURA 12 - TRANSFORMAÇÃO DE UM FICHEIRO XML PARA DOIS FICHEIROS XLS, UM COM FILTRAGEM E OUTRO SEM ALTERAÇÕES .....	13
FIGURA 13 - DETALHES DA LEITURA DO FICHEIRO XML .....	13
FIGURA 14 - DETALHES DO SELECT VALUES .....	14
FIGURA 15 - DETALHES DA IF FIELD VALUE IS NULL .....	15
FIGURA 16 – REMOÇÃO DO HÍFEN NO CAMPO POSTAL_CODE .....	15
FIGURA 17 - REMOÇÃO DOS CARACTERES CEDEX E PRÓXIMOS DÍGITOS NO CAMPO POSTAL_CODE .....	15
FIGURA 18 - REMOÇÃO DE ESPAÇOS NO CAMPO POSTAL_CODE .....	16
FIGURA 19 - DETALHES DA FILTER ROWS .....	16
FIGURA 20 - DETALHES DO FICHEIRO XLS COM A QUANTIDADE MAIOR QUE 25 .....	17
FIGURA 21 - DETALHES DA CRIAÇÃO DO FICHEIRO XLS COM A QUANTIDADE MENOR QUE 25 .....	18
FIGURA 22 - ENTRADA DO FICHEIRO EXCEL, REMOÇÃO DE CAMPOS E FILTRAGEM DE ESTADO DO PEDIDO. ....	19
FIGURA 23 - DETALHE DA IMPORTAÇÃO DO FICHEIRO XLS .....	20
FIGURA 24 - REMOÇÃO DOS CAMPOS.....	20
FIGURA 25 - FILTRAGEM DO CAMPO DE STATUS DO PEDIDO “DELIVERED” .....	21
FIGURA 26 - CRIAÇÃO DO FICHEIRO EXCEL COM OS ENVIOS COMPLETOS.....	21
FIGURA 27 - FILTRAGEM PARA O CAMPO DE STATUS DO PEDIDO "SHIPPED" .....	22
FIGURA 28 - CRIAÇÃO DO FICHEIRO EXCEL COM OS ENVIOS DESPACHADOS .....	23
FIGURA 29 - CRIAÇÃO DO FICHEIRO EXCEL COM OS ENVIOS PENDENTES .....	24
FIGURA 30 – VISÃO GERAL DO JOB .....	26
FIGURA 31 - CAPTURA DE TELA DO EMAIL RECEBIDO .....	26
FIGURA 32 - QR CODE .....	27

## Índice

1. Introdução .....	5
2. Geração de dados aleatórios.....	6
3. Manipulações no Kettle .....	7
4. Transformações.....	8
4.1. Transformação 1 - CSV to XML .....	8
4.2. Transformação 2 - XML to XLS .....	13
4.3. Transformação 3 – XLS Edit .....	19
5. Job .....	25
6. Vídeo de funcionamento do programa .....	27
7. Conclusão .....	28

## **1. Introdução**

O primeiro trabalho da Unidade Curricular de Integração de Sistemas de Informação tem como objetivo consolidar o conhecimento teórico e prático sobre as ferramentas de transformação de dados, que foram abordadas durante as aulas. Este trabalho propõe a utilização de ferramentas de ETL (Extract, Transformation and Load) para a manipulação e integração de dados.

Para o desenvolvimento deste trabalho, foi utilizado a plataforma Pentaho Data Integration, também conhecida como Kettle ou Spoon. O Pentaho oferece um conjunto diversificado de recursos para manipulação de dados, como a integração com diversos tipos de bancos de dados, criação e manipulação de tabelas, manipulação de ficheiros, além de funcionalidades para enviar e-mails e executar pings em sistemas remotos.

## 2. Geração de dados aleatórios

Para criação de dados aleatórios foi utilizado o [Mockaroo](#). Este website permite a geração de dados aleatórios levando em consideração o tipo de dados de cada campo. Também permite gerar os campos através de inteligência artificial, que foi a utilização abordada neste trabalho, basta informar o problema, especificar a quantidade de campos que contém e ele retorna os campos preenchidos com os dados.

Figura 1 - Descrição para a IA do Mockaroo

Field Name	Type	Options
beneficiary_id	Row Number	blank: 0 %
first_name	First Name	blank: 0 %
last_name	Last Name	blank: 0 %
age	Number	min: 18 max: 85 decimals: 0 blank: 0 %
email	Email Address	blank: 0 %
race	Custom List	Caucasian, African American, Hispanic, Asian, Middle Eastern, Native American, Pacific Islander, Mixed random blank: 0 %
address	Street Address	blank: 0 %
city	City	blank: 0 %
country	Country	restrict countries... blank: 0 %
postal_code	Postal Code	blank: 0 %
order_id	Row Number	blank: 0 %
order_date	Datetime	01/01/2020 to 12/31/2021 format: m/d/yyyy blank: 0 %
order_total	Number	min: 0 max: 10000 decimals: 2 blank: 0 %
product_name	Custom List	Shoes, Shirts, Pants, Dresses, Accessories random blank: 0 %
quantity	Number	min: 1 max: 10 decimals: 0 blank: 0 %
order_status	Custom List	Pending, Completed random blank: 0 %

Figura 2 - Geração dos dados utilizando a IA do Mockaroo

### **3. Manipulações no Kettle**

No Kettle temos dois tipos de ficheiros, o .ktr que pertence a uma Transformação e o .kjb que pertence a um Job. Como citado em aula, o Job é como se fosse um ficheiro “main.c” e a Transformação a uma função de um programa. O Job é responsável por chamar as Transformações que precisemos de usar. Foram realizadas três transformações nesse trabalho prático, falaremos sobre cada uma delas.

## 4. Transformações

### 4.1. Transformação 1 - CSV to XML

Esta transformação inicial faz uma leitura do ficheiro CSV com os dados gerados por IA, realiza um script em SQL para criar as tabelas, caso não exista na base de dados, a seguir divide o campo **full\_name** do beneficiário em dois campos: **first\_name** e **last\_name**, formatando ambos em letras maiúsculas. Em seguida, o ficheiro XML é convertido em um ficheiro CSV. Por fim, é realizado um **INSERT** caso as tabelas estejam vazias, ou um **UPDATE** se já houver dados na base de dados.

Durante o processo, encontrei dificuldades com campos do tipo “Number” devido ao XML adicionar zeros extras para preencher o tamanho especificado pelo atributo “Length”. Isso afetou as etapas seguintes. Para resolver essa questão, mudamos o tipo dos dados para “Integer”, eliminando a necessidade de zeros adicionais e facilitando o processamento.

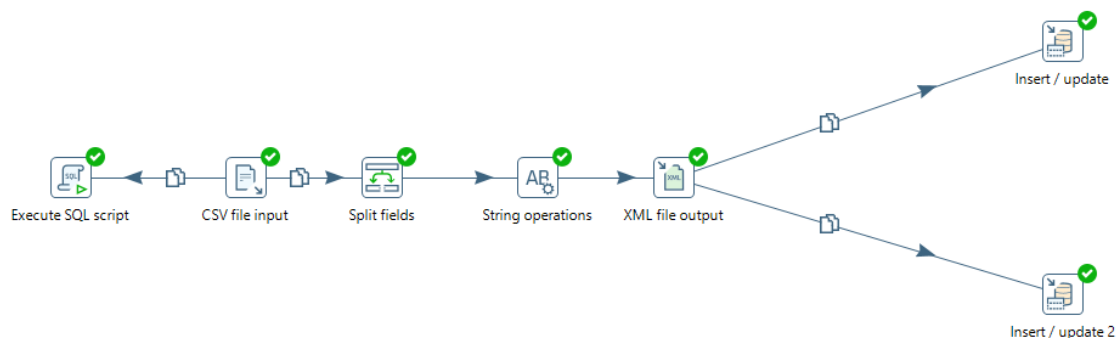


Figura 3 - Transformação de ficheiro CSV para XML com inserts para a base de dados



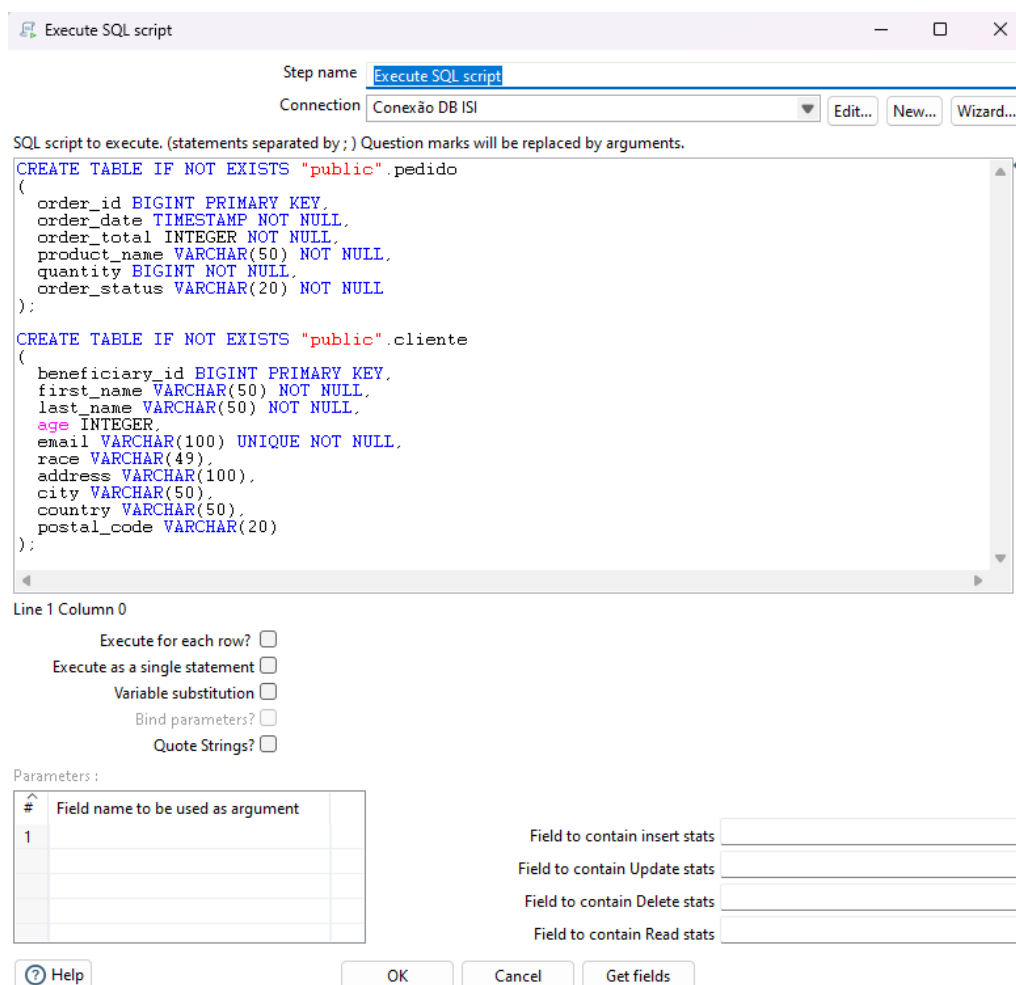


Figura 4 - Script SQL

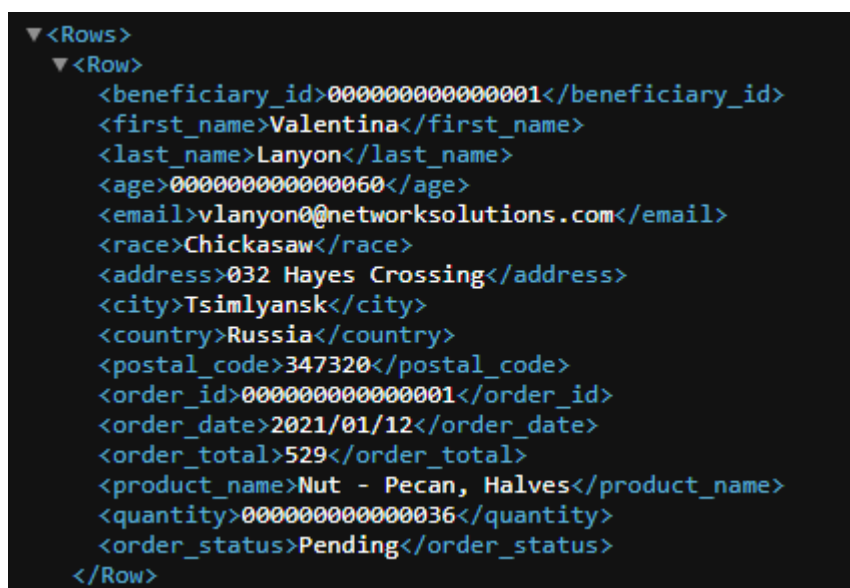


Figura 5 - Captura de tela do ficheiro XML antes da alteração

#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	beneficiary_id	Integer	#	15	0	USD	.	,	none
2	first_name	String		11		USD	.	,	none
3	last_name	String		11		USD	.	,	none
4	age	Integer	#	15	0	USD	.	,	none
5	email	String		31		USD	.	,	none
6	race	String		49		USD	.	,	none
7	address	String		26		USD	.	,	none
8	city	String		18		USD	.	,	none
9	country	String		14		USD	.	,	none
10	postal_code	String		13		USD	.	,	none
11	order_id	Integer	#	15	0	USD	.	,	none
12	order_date	Date	MM/dd/yyyy			USD	.	,	none
13	order_total	Integer	#	15	0	USD	0		none
14	product_name	String		32		USD	.	,	none
15	quantity	Integer	#	15	0	USD	0		none
16	order_status	String		9		USD	.	,	none

*Figura 6 - Alteração dos dados para "Integer"*

```

▼<Rows>
  ▼<Row>
    <beneficiary_id>1</beneficiary_id>
    <first_name>Valentina</first_name>
    <last_name>Lanyon</last_name>
    <age>60</age>
    <email>vlanyon@networksolutions.com</email>
    <race>Chickasaw</race>
    <address>032 Hayes Crossing</address>
    <city>Tsimlyansk</city>
    <country>Russia</country>
    <postal_code>347320</postal_code>
    <order_id>1</order_id>
    <order_date>2021/01/12</order_date>
    <order_total>529</order_total>
    <product_name>Nut - Pecan, Halves</product_name>
    <quantity>36</quantity>
    <order_status>Pending</order_status>
  </Row>

```

*Figura 7 - Captura de tela do ficheiro XML após a alteração*

[illegible]

Figura 8 - Detalhes da separação do campo full name para first name e last name

## Relatório Trabalho Prático 1 – Integração de Sistemas de Informação

String operations

Step name String operations

The fields to process:

#	In stream field	Out stream field	Trim type	Lower/Upper	Padding	Pad char	Pad Length	InitCap	Escape	Digits	Remove Special character
1	first_name			upper							
2	last_name			upper							

? Help

OK

Get fields

Cancel

*Figura 9 - Detalhes da formatação das letras maiúsculas*

**CSV file input**

Step name: **CSV file input**

Filename:  Browse...

Delimiter:  Insert TAB

Enclosure:

NIO buffer size:

Lazy conversion? ☒

Header row present? ☒

Add filename to result ☐

The row number field name (optional)

Running in parallel? ☐

New line possible in fields? ☐

Format:

File encoding:

#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	beneficiary_id	Integer	#	15	0	USD	.	,	none
2	first_name	String		11		USD	.	,	none
3	last_name	String		11		USD	.	,	none
4	age	Integer	#	15	0	USD	.	,	none
5	email	String		31		USD	.	,	none
6	race	String		49		USD	.	,	none
7	address	String		26		USD	.	,	none
8	city	String		18		USD	.	,	none
9	country	String		14		USD	.	,	none
10	postal_code	String		13		USD	.	,	none
11	order_id	Integer	#	15	0	USD	.	,	none
12	order_date	Date	MM/dd/yyyy			USD	.	,	none
13	order_total	Integer	#	15	0	USD	0		none
14	product_name	String		32		USD	.	,	none
15	quantity	Integer	#	15	0	USD	0		none
16	order_status	String		9		USD	.	,	none

Help OK Get Fields Preview Cancel

*Figura 10 - Detalhes da importação do ficheiro CSV*

The screenshot shows a dialog box titled "XML output" with a standard Windows-style title bar (minimize, maximize, close buttons). Below the title bar, there is a "Step name" field containing the text "XML file output". The main area of the dialog is divided into three tabs: "File", "Content", and "Fields", with "File" currently selected. Under the "File" tab, the following options are visible:

- Filename:** A text field containing the expression `${Internal.Entry.Current.Directory}/Generated Data/XML_Generated`, followed by a "Browse..." button.
- Do not create file at start:** A checkbox that is currently unchecked.
- Pass output to servlet:** A checkbox that is currently unchecked.
- Extension:** A text field containing the value "xml", followed by a small diamond icon.
- Include stepnr in filename?:** A checkbox that is currently unchecked.
- Include date in filename?:** A checkbox that is currently unchecked.
- Include time in filename?:** A checkbox that is currently unchecked.
- Specify Date time format:** A checkbox that is currently unchecked.
- Date time format:** A text field that is currently empty.
- Show filename(s):** A button.
- Add filenames to result:** A checkbox that is currently unchecked.

At the bottom of the dialog, there are three buttons: "Help" (with a question mark icon), "OK", and "Cancel".

*Figura 11 - Detalhes da criação do ficheiro XML*

## 4.2. Transformação 2 - XML to XLS

Nesta segunda transformação, estamos a importar o ficheiro XML que foi criado na transformação anterior, o select values está “formatando” o formato da data do pedido obtido no ficheiro XML. Logo após temos a eliminação de valores nulos no campo de código postal, ou seja, todos os valores nulos vão ser substituídos por “N/A”, após, temos três sequências de normalização utilizando expressões regulares, uma para remover espaços, outra para remover os hífens e outra para remover caracteres do campo postal\_code. A seguir ocorre uma filtragem, caso a quantidade de pedidos feitas seja maior que 25, vão ser criados dois ficheiros Excel, um contendo os dados da quantidade de pedidos maior que 25 e outro armazenando os dados que são menor que 25.

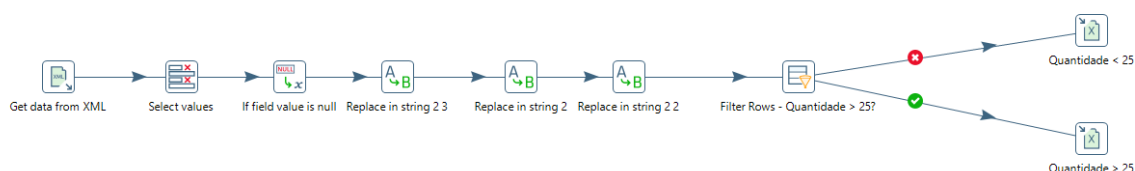


Figura 12 - Transformação de um ficheiro XML para dois ficheiros XLS, um com filtragem e outro sem alterações

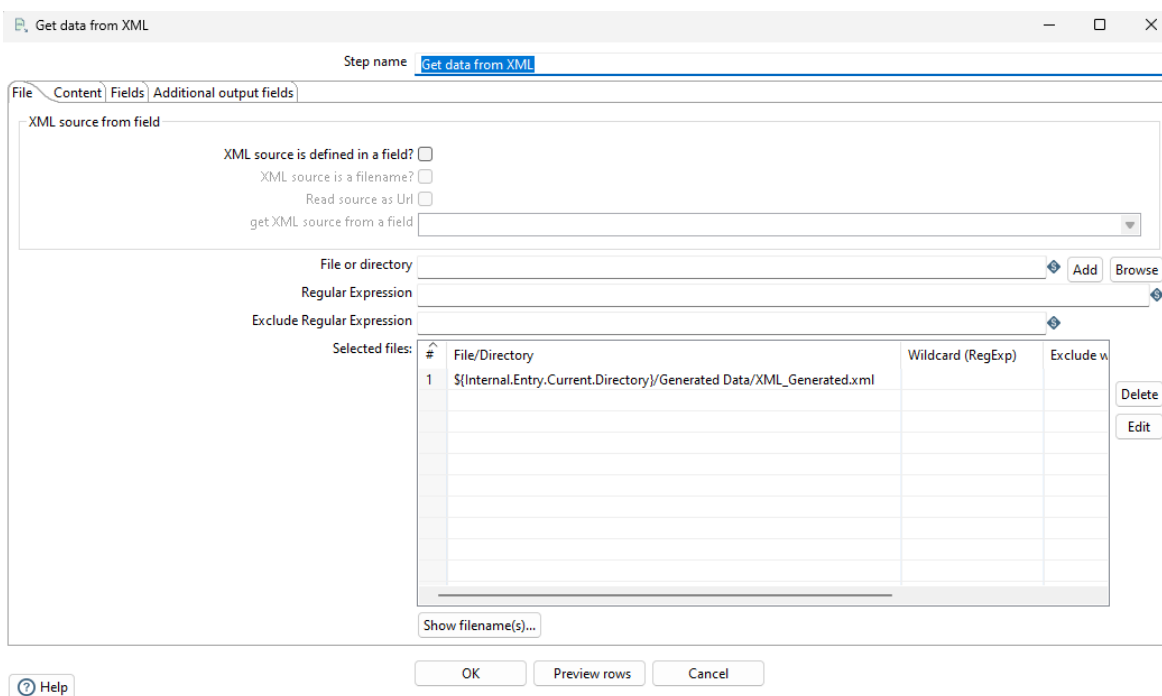


Figura 13 - Detalhes da leitura do ficheiro XML

Select values

Step name Select values

Select & Alter Remove Meta-data

Fields to alter the meta-data for :

#	Fieldname	Rename to	Type	Length	Precision	Binary to
1	beneficiary_id		Integer		0	N
2	first_name		None			N
3	last_name		None			N
4	age		None		0	N
5	email		None			N
6	race		None			N
7	address		None			N
8	city		None			N
9	country		None			N
10	postal_code		None			N
11	order_id		None		0	N
12	order_date		Date			N
13	order_total		None		0	N
14	product_name		None			N
15	quantity		None		0	N
16	order_status		None			N

Get fields to change

Help OK Cancel

Figura 14 - Detalhes do Select Values

If field value is null

Step name: If field value is null

Replace Null for all fields

Replace by value: N/A

Set empty string? ☐

Mask (Date):

Select fields ☐

Select value type ☐

Value types

#	Type	Replace by value	Conversion mask (Date)	Set empty string?

Fields

#	Field	Replace by value	Conversion mask (Date)	Set empty string?
1				

Help OK Get Fields Cancel

Figura 15 - Detalhes da If field value is null

Replace in string

Step name: Replace in string 2

Fields string

#	In stream field	Out stream field	use RegEx	Search	Replace with	Set empty string?	Replace with field	Whole Word	Case sensitive	Is Unicode
1	postal_code		Y	-		N		N	N	N

Help OK Get fields Cancel

Figura 16 – Remoção do hífen no campo postal\_code

Replace in string

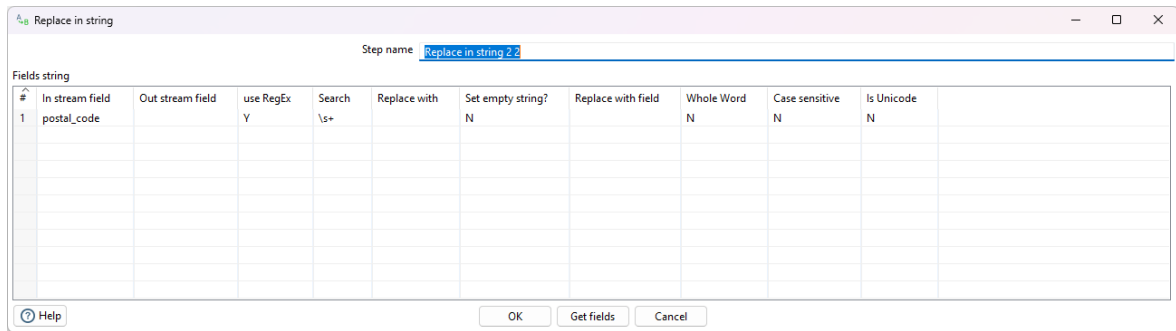
Step name: Replace in string 2

Fields string

#	In stream field	Out stream field	use RegEx	Search	Replace with	Set empty string?	Replace with field	Whole Word	Case sensitive	Is Unicode
1	postal_code		Y	CEDEX\d*		N		N	N	N

Help OK Get fields Cancel

Figura 17 - Remoção dos caracteres CEDEX e próximos dígitos no campo postal\_code

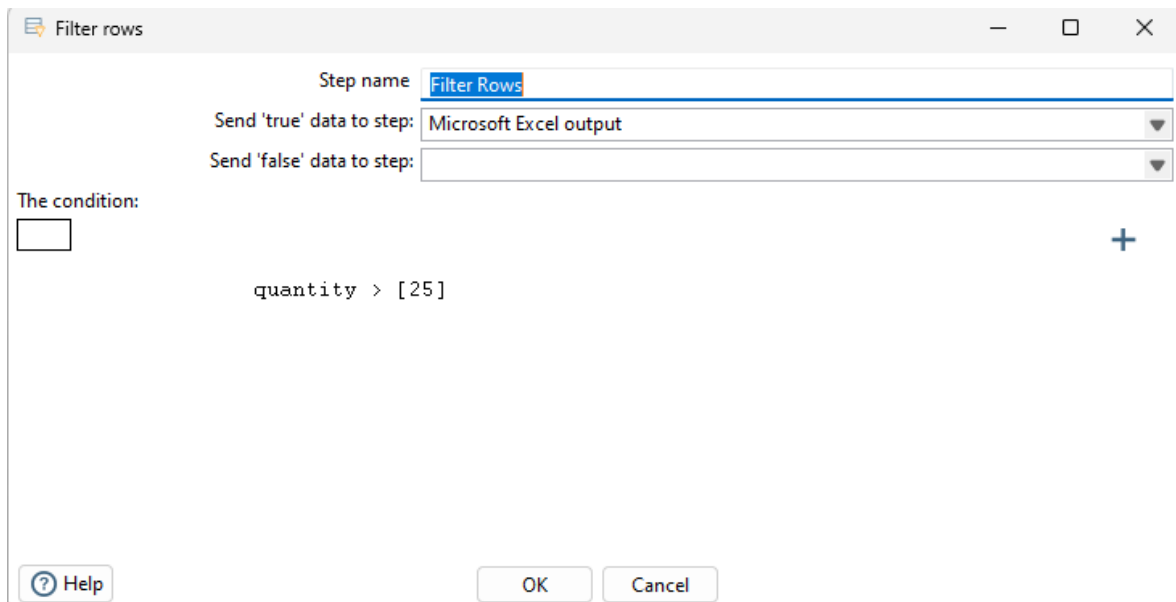


Step name: Replace in string 2

#	In stream field	Out stream field	use RegEx	Search	Replace with	Set empty string?	Replace with field	Whole Word	Case sensitive	Is Unicode
1	postal_code		Y	\s+		N		N	N	N

Buttons: Help, OK, Get fields, Cancel

Figura 18 - Remoção de espaços no campo postal\_code



Step name: Filter Rows

Send 'true' data to step: Microsoft Excel output

Send 'false' data to step:

The condition:

quantity > [25]

Buttons: Help, OK, Cancel

Figura 19 - Detalhes da Filter Rows



Microsoft Excel writer

Step name **Quantidade > 25**

File & Sheet Content

File

Filename

Extension

Create parent folder ☐

Stream XLSX data ☐

Split every ... data rows

Include stepnr in filename? ☐

Include date in filename? ☐

Include time in filename? ☐

Specify Date time format ☐

Date time format

If output file exists

Wait for first row before creating file ☐

Add filenames to result ☒

Sheet

Sheet name (max. 31 characters)

Make this the active sheet ☒

If sheet exists in output file

Protect sheet? (XLS format only) ☐

Protected by user

Password

Template

Use template when creating new files ☐

Template file

Use template when creating new sheets ☐

Template sheet

Hide Template Sheet ☐

Help OK Cancel

Figura 20 - Detalhes do ficheiro XLS com a quantidade maior que 25

Microsoft Excel writer

Step name: Quantidade < 25

File & Sheet | Content

File

Filename: \${Internal.Entry.Current.Directory}/Generated Data/XLS\_Generated\_MinorQuantity Browse...

Extension: xls [Excel 97 and above]

Create parent folder ☐

Stream XLSX data ☐

Split every ... data rows: 0

Include stepnr in filename? ☐

Include date in filename? ☐

Include time in filename? ☐

Specify Date time format ☐

Date time format:

Show filename(s)...

If output file exists: replace with new output file

Wait for first row before creating file ☐

Add filenames to result ☒

Sheet

Sheet name (max. 31 characters): Sheet1

Make this the active sheet ☒

If sheet exists in output file: replace with new sheet

Protect sheet? (XLS format only) ☐

Protected by user:

Password:

Template

Use template when creating new files ☐

Template file: template.xls Browse...

Use template when creating new sheets ☐

Template sheet:

Hide Template Sheet ☐

Help OK Cancel

Figura 21 - Detalhes da criação do ficheiro XLS com a quantidade menor que 25

### 4.3. Transformação 3 – XLS Edit

Esta é a última transformação, onde campos irrelevantes para o usuário final do arquivo XLS são removidos, sendo eles os campos: **age**, **email**, **race**, **address**, **city**, **country**, e **postal\_code**. Em seguida, o processo realiza uma filtragem com base no status do pedido. Se o status for **Delivered**, é criado um ficheiro Excel com os envios completos. Caso contrário, o fluxo segue para uma nova filtragem. Se o status for **Shipped**, são gerados dois ficheiros Excel: um para pedidos que cumprem a condição e outro para os que não cumprem.

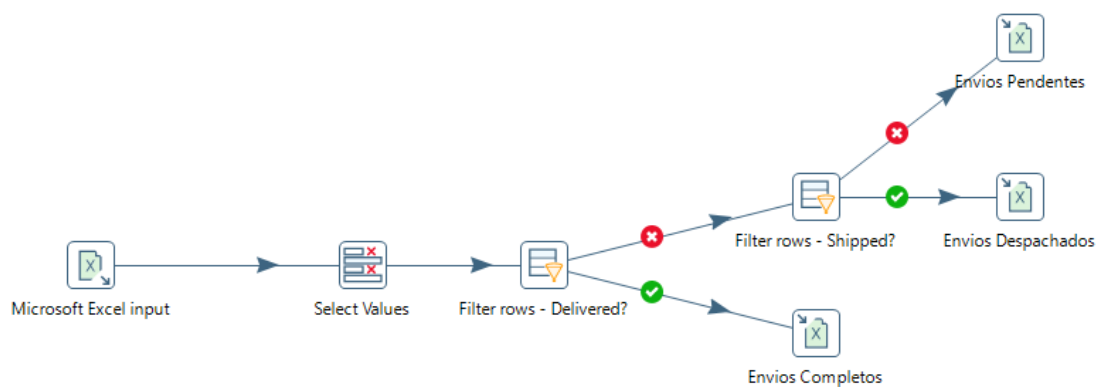


Figura 22 - Entrada do ficheiro Excel, remoção de campos e filtragem de estado do pedido.

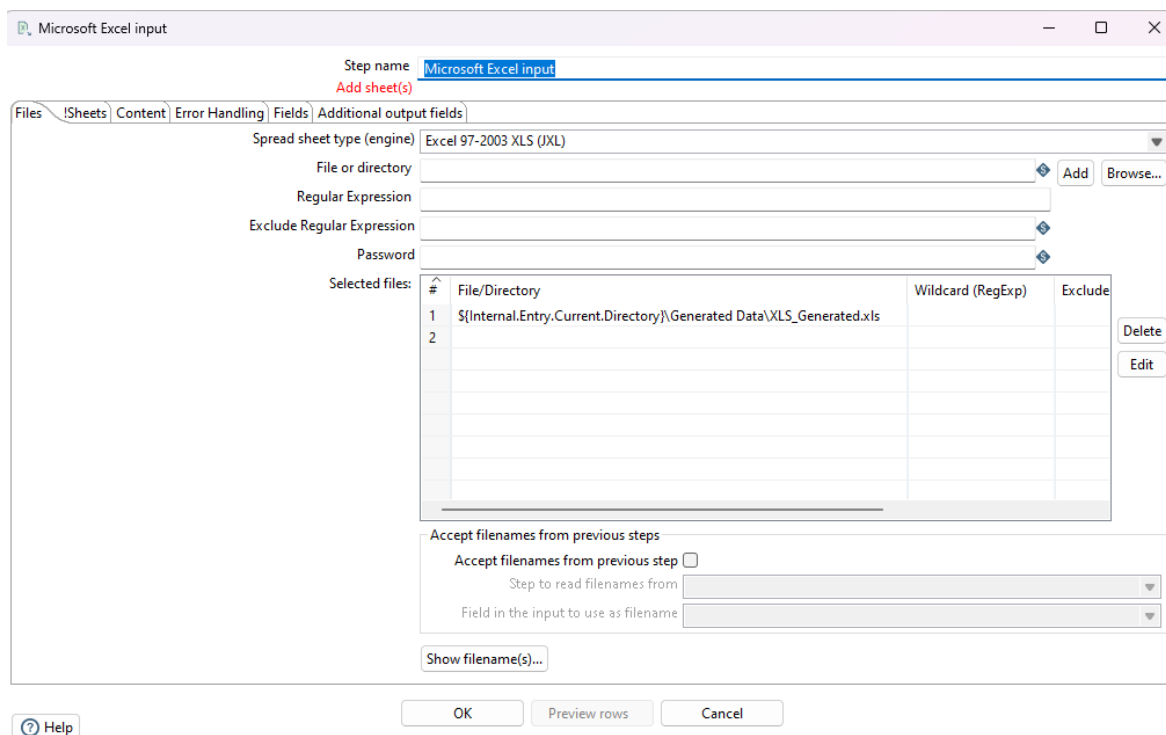


Figura 23 - Detalhe da importação do ficheiro XLS

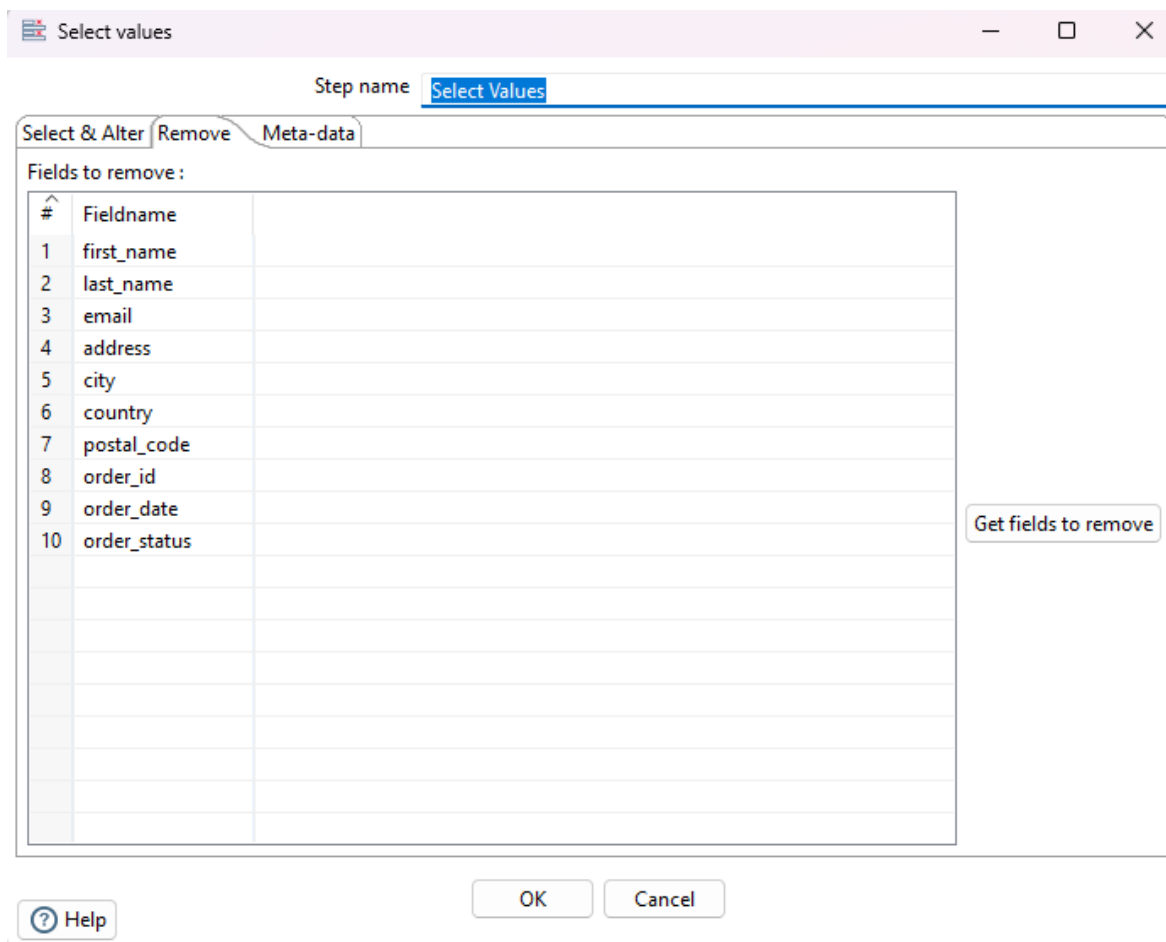


Figura 24 - Remoção dos campos

Filter rows

Step name: Filter rows - Delivered?

Send 'true' data to step: Envios Completos

Send 'false' data to step: Filter rows - Shipped?

The condition:

☐ order\_status =  Delivered (String)

Figura 25 - Filtragem do campo de status do pedido "Delivered"

Microsoft Excel writer

Step name: Envios Completos

File & Sheet Content

File

Filename: S:\Internal.Entry.Current.Directory\Generated Data\XLS\_Generated\_Delivered

Extension: xls [Excel 97 and above]

Create parent folder ☐

Stream XLSX data ☐

Split every ... data rows: 0

Include stepnr in filename? ☐

Include date in filename? ☐

Include time in filename? ☐

Specify Date time format ☐

Date time format:

Show filename(s)...

If output file exists: replace with new output file

Wait for first row before creating file ☐

Add filenames to result ☒

Sheet

Sheet name (max. 31 characters): Sheet1

Make this the active sheet ☒

If sheet exists in output file: replace with new sheet

Protect sheet? (XLS format only) ☐

Protected by user:

Password:

Template

Use template when creating new files ☒

Template file: template.xls

Use template when creating new sheets ☐

Template sheet:

Hide Template Sheet ☐

Figura 26 - Criação do ficheiro Excel com os envios completos

Filter rows

Step name:

Send 'true' data to step:

Send 'false' data to step:

The condition:

=  (String)

Figura 27 - Filtragem para o campo de status do pedido "Shipped"

Microsoft Excel writer

Step name: Envios Despachados

File & Sheet | Content

File

Filename:  Browse...

Extension: xls [Excel 97 and above]

Create parent folder ☐

Stream XLSX data ☐

Split every ... data rows:

Include stepnr in filename? ☐

Include date in filename? ☐

Include time in filename? ☐

Specify Date time format ☐

Date time format:

Show filename(s)...

If output file exists: replace with new output file

Wait for first row before creating file ☐

Add filenames to result ☒

Sheet

Sheet name (max. 31 characters):

Make this the active sheet ☒

If sheet exists in output file: replace with new sheet

Protect sheet? (XLS format only) ☐

Protected by user:

Password:

Template

Use template when creating new files ☐

Template file:  Browse...

Use template when creating new sheets ☐

Template sheet:

Hide Template Sheet ☐

Help OK Cancel

Figura 28 - Criação do ficheiro Excel com os envios despachados

Microsoft Excel writer

Step name: Envios Pendentes

File & Sheet | Content

File

Filename:  Browse...

Extension: xls [Excel 97 and above]

Create parent folder ☐

Stream XLSX data ☐

Split every ... data rows:

Include stepnr in filename? ☐

Include date in filename? ☐

Include time in filename? ☐

Specify Date time format ☐

Date time format:

Show filename(s)...

If output file exists: replace with new output file

Wait for first row before creating file ☐

Add filenames to result ☒

Sheet

Sheet name (max. 31 characters):

Make this the active sheet ☒

If sheet exists in output file: replace with new sheet

Protect sheet? (XLS format only) ☐

Protected by user:

Password:

Template

Use template when creating new files ☐

Template file:  Browse...

Use template when creating new sheets ☐

Template sheet:

Hide Template Sheet ☐

Help OK Cancel

Figura 29 - Criação do ficheiro Excel com os envios pendentes



## 5. Job

Na criação do Job, iremos chamar todas as transformações que referenciamos anteriormente e também foram utilizadas algumas ferramentas que o Kettle nos fornece.

Para executar o Job, é necessário abrir com o Kettle o ficheiro START\_JOB.kjb e clicar em Run.

O fluxo do Job é estruturado da seguinte forma:

1. Start
2. É criada uma pasta chamada “Generated Data” para armazenar os dados gerados
  - a. Caso não seja possível realizar a criação da pasta, o programa mostrará um erro e o Job será abortado.
3. É checada a conexão com a base de dados
  - a. Caso não seja possível conectar com a base de dados, o programa mostrará um erro e o Job será abortado.
4. Verificação da existência do ficheiro CSV.
  - a. Caso não seja encontrada a existência do ficheiro CSV, o programa mostrará uma mensagem de erro e o Job será abortado.
5. Faz uma chamada para a Transformação 1 – CSV to XML.
6. Verificação da existência do ficheiro XML
  - a. Caso não seja encontrada a existência do ficheiro XML, o programa mostrará uma mensagem de erro e o Job será abortado.
7. Faz uma chamada para a Transformação 2 – XML to XLS
8. Verificação da existência do ficheiro XLS
  - a. Caso não seja encontrada a existência do ficheiro XLS, o programa mostrará uma mensagem de erro e o Job será abortado.
9. Faz uma chamada para a Transformação 3 – XLS Edit
10. Verificação da existência do ficheiro XLS editado.
  - a. Caso não seja encontrada a existência do ficheiro XLS editado, o programa mostrará uma mensagem de erro e o Job será abortado.
11. Realiza um ping ao Google para verificar a ligação à internet.
  - a. Caso não haja ligação à internet, o programa mostrará uma mensagem de erro e o último ficheiro criado é movido para a raiz do projeto para ser armazenado localmente e a pasta que foi criada no início do programa será excluída.

12. Envia um email para [a24207@alunos.ipca.pt](mailto:a24207@alunos.ipca.pt) através do endereço de email [a24207@sapo.pt](mailto:a24207@sapo.pt) com três ficheiros criados anteriormente, o XLS\_Generated\_Backup (XLS original), o XLS\_Generated (XLS com filtro da quantidade de produtos pedidos) e o XLS\_Generated\_Edit (XLS com a limpeza de campos). Nesse passo foi criado e utilizado um email da [Sapo](mailto:Sapo).
  - a. Caso ocorra algum erro ao enviar o email, o programa mostrará uma mensagem de erro e o último ficheiro criado é movido para a raiz do projeto para ser armazenado localmente e a pasta que foi criada no início do programa será excluída.
13. A pasta chamada “Generated Data” criada no início do programa é excluída
14. O Job é finalizado com sucesso.

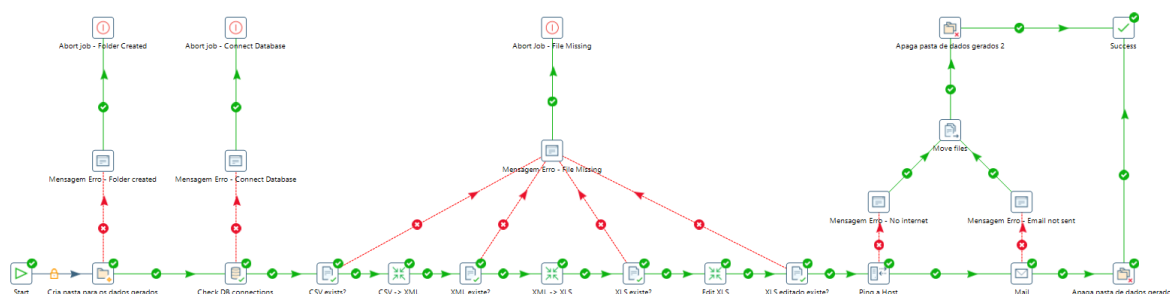


Figura 30 – Visão geral do Job

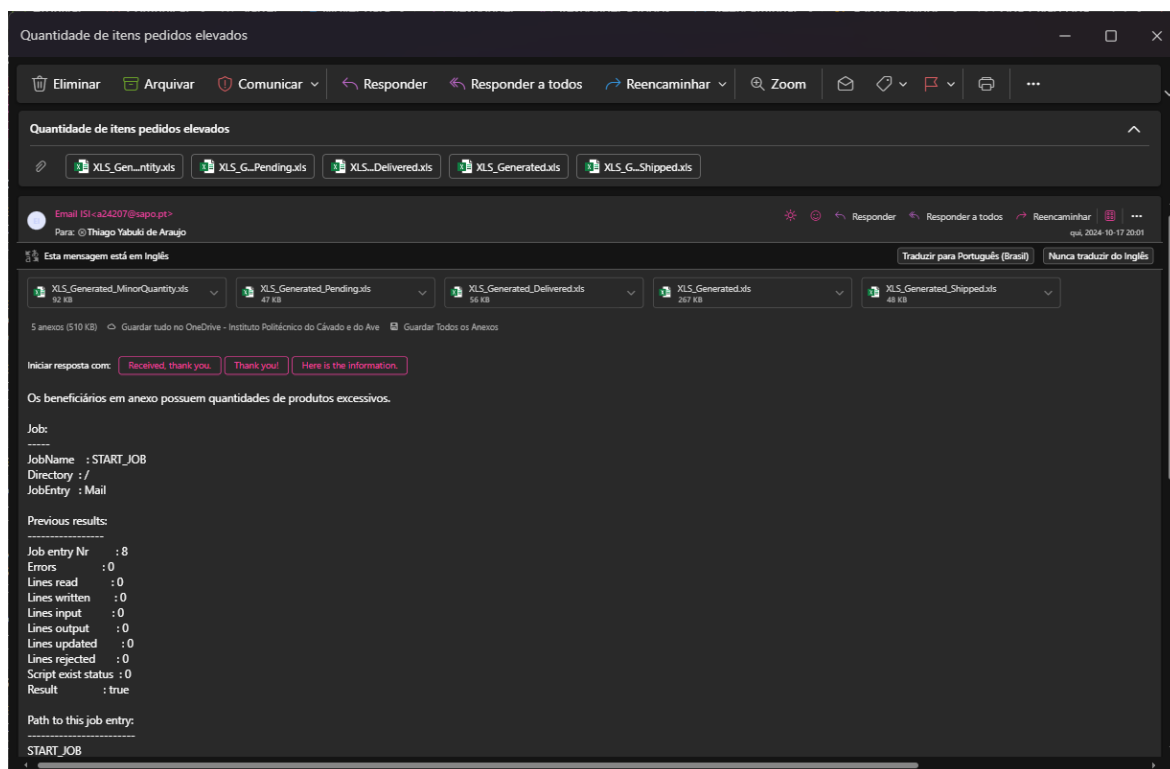


Figura 31 - Captura de tela do Email recebido

## 6. Vídeo de funcionamento do programa



*Figura 32 - QR Code*

## 7. Conclusão

Este trabalho prático permitiu consolidar o conhecimento sobre a utilização de ferramentas de ETL adquirido nas aulas da disciplina de Integração de Sistemas de Informação, em especial a plataforma **Pentaho Data Integration**, uma plataforma com ampla variedade de funcionalidades para integração e manipulação de dados. Através das atividades desenvolvidas, foi possível gerar dados fictícios, realizar transformações de ficheiros de formatos variados e automatizar processos que facilitam a integração de sistemas de informação. A utilização de técnicas como a conversão de ficheiros CSV para XML e XML para XLS foi eficaz para entender a manipulação e filtragem de dados em diferentes formatos. Além disso, o Job criado demonstra a importância de uma estrutura sequencial para garantir que cada etapa do processo seja completada com sucesso, fornecendo mecanismos de verificação e manipulação dos dados.