



25-10-2024

Análise e Limpeza de Dados

E-Commerce

Integração de Sistemas de Informação

Licenciatura: Engenharia de Sistemas Informáticos

Ano Letivo: 2024 / 2025

Trabalho realizado por:

- Pedro Henrique da Costa Vieira, 25626.
- 

Índice

Índice de Figuras	2
Introdução	3
Tema escolhido	4
Problema a resolver	5
Objetivos de problema	6
Estratégia a utilizar	7
Ferramenta utilizada	8
Transformações	9
Primeira transformação - DisplayAllDataByCategory	10
Segunda transformação - AllowHTMLtoEmail	12
Terceira transformação - FilterWithRegularExpression	13
Jobs	15
Primeiro Job - SendBDTableToEmail	16
Node-Red	18
Estrutura do Node-Red	19
Primeiro Nó da estrutura: Node-red	21
Segundo Nó da estrutura: Node-red	21
Terceiro Nó da estrutura: Node-red	21
Conteúdo dos Nós da estrutura	22
Vídeo	0
Conclusão	0
Webgrafia	1

Índice de Figuras

Figura 1- DisplayAllDataByCategory	11
Figura 2 - AllowHTMLtoEmail.....	12
Figura 3 - FilterWithRegularExpression	14
Figura 4 - SendBDTableToEmail.....	17
Figura 5 - Estrutura do Node-Red	19
Figura 6 - Primeiro Nó da estrutura.....	21
Figura 7 - Segundo Nó da estrutura	21
Figura 8 - Terceiro Nó da estrutura	21
Figura 9 - Conexão com a BD.....	22
Figura 10 - Criação da Conexão	23
Figura 11 - Função da query da BD.....	24

Introdução

A crescente adoção de plataformas de **e-commerce** nos últimos anos resultou na geração de grandes volumes de dados provenientes de diversas fontes, como transações de compras, inventário de produtos, comportamento de clientes e interações com a plataforma.

Neste contexto, o **processo de análise e limpeza de dados** torna-se crucial. A **limpeza de dados** é o primeiro passo para garantir que as informações utilizadas para análise sejam precisas, completas e utilizáveis. Isso envolve a identificação e correção de erros, dados mal formatados e valores fora dos intervalos esperados, assegurando que a base de dados mantenha um nível adequado de integridade.

O presente trabalho tem como objetivo aplicar técnicas de análise e limpeza de dados em uma base de dados de um e-commerce, focando principalmente em garantir que os valores relacionados aos produtos (como preço, **stock** e categorias) estejam devidamente formatados e consistentes. A utilização de ferramentas como expressões regulares (Regex) para validação de campos e a aplicação de regras de negócio para verificar a consistência dos dados são elementos centrais da abordagem adotada neste projeto.

Ao garantir a integridade e a qualidade dos dados, este processo permite que a empresa otimize a operação do e-commerce, melhore a experiência dos clientes e tome decisões estratégicas fundamentadas em dados confiáveis.

Tema escolhido

Em e-commerce, os dados são produzidos em larga escala e de maneira contínua, sendo provenientes de diversas fontes, como transações, interações com clientes e tráfego do site. A capacidade de processar e analisar esses dados corretamente pode diferenciar uma empresa de sucesso de uma que enfrenta dificuldades.

A transformação de dados brutos em informações úteis começa pela limpeza, que garante a integridade dos dados e, consequentemente, a confiabilidade das análises futuras.

No meu caso, em que trabalhei diretamente com a limpeza de dados de uma base de produtos de um e-commerce, o foco é garantir que as informações de produtos estejam corretas, como preços e descrições. Isso é essencial para garantir que a experiência do utilizador na plataforma seja fluida, sem erros que possam prejudicar a navegação, compra ou a confiança na loja.

Problema a resolver

Consequências da Falta de Análise e Limpeza:

Se os dados não forem adequadamente validados e limpos, a loja online enfrentará problemas como:

- **Experiência de compra ruim:** clientes podem não encontrar produtos devido à categorização inadequada ou ver informações incorretas sobre preço e disponibilidade.
- **Perdas financeiras:** erros de preços ou **stock** podem levar à perda de vendas ou problemas logísticos.
- **Decisões de negócio prejudicadas:** com dados incorretos, as empresas podem tomar decisões erradas sobre marketing, promoções, gestão de inventário e campanhas de vendas.

Assim, o objetivo do projeto é criar processos que assegurem a validação automática dos dados, principalmente para o campo de preços e **stock**, garantindo que o banco de dados contenha apenas informações corretas e relevantes. Isso aumentará a eficiência das operações, proporcionará uma experiência de compra mais fluida e evitará problemas operacionais.

Objetivos de problema

O projeto de Análise e Limpeza de Dados de E-Commerce tem como principais objetivos:

1. Assegurar a integridade e qualidade dos dados:

Garantir que os dados inseridos no sistema estejam corretos, válidos e em um formato padronizado. Isso inclui a verificação de campos essenciais como preço, **stock** e categoria dos produtos, evitando informações erradas que possam impactar o negócio.

2. Implementar verificações de formato e conteúdo de dados, como:

- Verificação de formato correto em campos numéricos, como preços e pesos (e.g., valores decimais formatados corretamente).
- Verificação de categorização correta para garantir que os produtos sejam classificados adequadamente.

3. Prevenir a entrada de dados inválidos no banco:

Assegurar que as regras de validação estejam implementadas em processos automáticos antes de qualquer inserção ou atualização de registos.

Ao atingir esses objetivos, a loja online poderá funcionar de maneira mais eficiente, proporcionando uma experiência de usuário melhor e uma gestão mais precisa de seus produtos e operações.

Estratégia a utilizar

Para garantir a eficácia da Análise e Limpeza de Dados de E-Commerce, a estratégia a ser adotada envolve uma abordagem estruturada e automatizada para identificar, validar e corrigir inconsistências nos dados. Os principais pontos da estratégia são:

1. Coleta de dados e pré-processamento:

- **Coleta de dados brutos** da base de dados do e-commerce, como informações de produtos (nome, descrição, categoria, preço, etc.). Isso inclui a ingestão de dados através de processos automatizados que obtêm os registos diretamente da fonte.

2. Aplicação de regras de validação:

- **Expressões regulares (Regex)** para validar campos específicos (ex.: garantir que o valor do preço segue o formato decimal adequado).

3. Limpeza e padronização dos dados:

4. Implementação de mecanismos de feedback e logs:

5. Testes e validação final:

- **Simulação de casos de teste** com diferentes tipos de entradas de dados para assegurar que o sistema de validação funcione corretamente sob diferentes cenários (e.g., preços com muitos dígitos, categorias mal definidas).

Essa abordagem automatizada não apenas ajuda a garantir a qualidade dos dados, mas também reduz erros manuais e aumenta a eficiência operacional, permitindo que os gestores do e-commerce confiem nos dados para tomadas de decisões estratégicas.

Ferramenta utilizada

O **Pentaho Data Integration (PDI)**, também conhecido como **Pentaho Kettle**, é uma ferramenta de **ETL (Extract, Transform, Load)** amplamente utilizada para integrar, transformar e manipular dados de diferentes fontes. O PDI faz parte da suite **Pentaho**, uma plataforma de **Business Intelligence (BI)** que oferece soluções para análise de dados, relatórios e **dashboards**.

Características Principais:

1. Ambiente Gráfico Intuitivo;
2. Capacidades de Integração;
3. Transformações e Limpeza de Dados;
4. Automação e Orquestração de Processos;
5. Escalabilidade e Desempenho;
6. Comunidade e Suporte.

Aplicação no Projeto:

No contexto do projeto de Análise e Limpeza de Dados de **E-Commerce**, o **Pentaho** foi fundamental para:

1. **Validar os dados de produtos**, como preço e peso, utilizando expressões regulares.
2. **Limpar dados inconsistentes**, como descrições mal formatadas ou categorias incorretas.
3. **Gerar relatórios automatizados** que permitiram garantir a integridade e qualidade dos dados.
4. **Exportar os dados limpos** para bases de dados e formatos como **JSON**, facilitando a integração com outros sistemas.

Com essas capacidades, o **Pentaho Data Integration** provou ser uma ferramenta robusta e eficiente para manter a qualidade de dados e otimizar a gestão de informações em plataformas de e-commerce.

Transformações

As transformações no **Pentaho Kettle** são **workflows** que processam dados de diferentes fontes, realizando uma série de operações, como limpeza, transformação, agregação e carga de dados para outros sistemas ou bancos de dados. Cada transformação é composta por uma sequência de passos que executam tarefas específicas, desde a extração dos dados até o carregamento final.

Aqui estão alguns pontos principais sobre as transformações no Pentaho:

1. Passos de Extração de Dados (**Input**);
2. Passos de Transformação (**Processing**);
3. Passos de Carga de Dados (**Output**);
4. Uso de Expressões Regulares (**Regex**);
5. Orquestração de Transformações.

Benefícios das Transformações no Pentaho:

- **Automatização:** Permite a automação de processos repetitivos de limpeza e análise de dados.
- **Integração Fácil:** Pode-se integrar com diversas fontes de dados e exportar para vários destinos.
- **Interface Visual:** O ambiente gráfico do **Pentaho** facilita a criação de fluxos complexos sem a necessidade de programação manual.

Primeira transformação - *DisplayAllDataByCategory*

A transformação ***DisplayAllDataByCategory***, mostrada no diagrama, é composta por várias etapas que têm o objetivo de processar e organizar dados de produtos extraídos via uma **API REST** e, em seguida, armazená-los em diferentes formatos de saída (como Excel, XML e arquivos de texto).

Objetivo Geral:

- O objetivo principal dessa transformação é integrar e processar dados de produtos de um **e-commerce**, categorizando-os e salvando-os em diferentes formatos para facilitar sua análise e utilização.

Explicação Geral do Fluxo:

1. Entrada de Dados (URL Products):

Objetivo: O fluxo começa com a leitura de dados de uma API ou uma URL que contém informações sobre produtos.

2. REST Client (Cliente REST):

Objetivo: Esse componente consulta a URL fornecida para obter dados de produtos.

3. Products JSON (Processamento de JSON):

Objetivo: Decodificar e processar o conteúdo JSON retornado pelo cliente REST.

4. Insert / Update (Inserir ou Atualizar):

Objetivo: Esta etapa insere novos dados ou atualiza registos já existentes em uma base de dados.

5. Select Fields (Selecionar Campos):

Objetivo: Selecionar apenas os campos necessários para as operações seguintes.

6. Sort By Category (Ordenar por Categoria):

Objetivo: Ordenar os dados dos produtos com base na categoria.

7. Saídas de Dados:

- **Microsoft Excel Output (Saída para Excel):**

Objetivo: Exportar os dados organizados para um arquivo Excel.

- **Text File Output (Saída para Arquivo de Texto):**

Objetivo: Salvar os dados processados em um arquivo de texto.

- **XML Output (Saída para XML):**

Objetivo: Exportar os dados processados para um arquivo XML. Esse formato pode ser utilizado para integração com outros sistemas que necessitem desse tipo de estrutura.

Considerações:

1. Objetivo Principal:

A transformação tem como objetivo principal coletar dados de produtos de uma **API REST**, processar esses dados e organizá-los por categoria. Ela também atualiza um banco de dados existente e oferece diversas saídas para exportação, como Excel, arquivo de texto e XML.

2. Atualização de Dados (Insert/Update):

Uma funcionalidade importante desta transformação é o passo de **Insert/Update**, que mantém a base de dados sincronizada com os dados mais recentes recebidos da **API REST**. Isso garante que as informações sobre os produtos estejam sempre atualizadas.

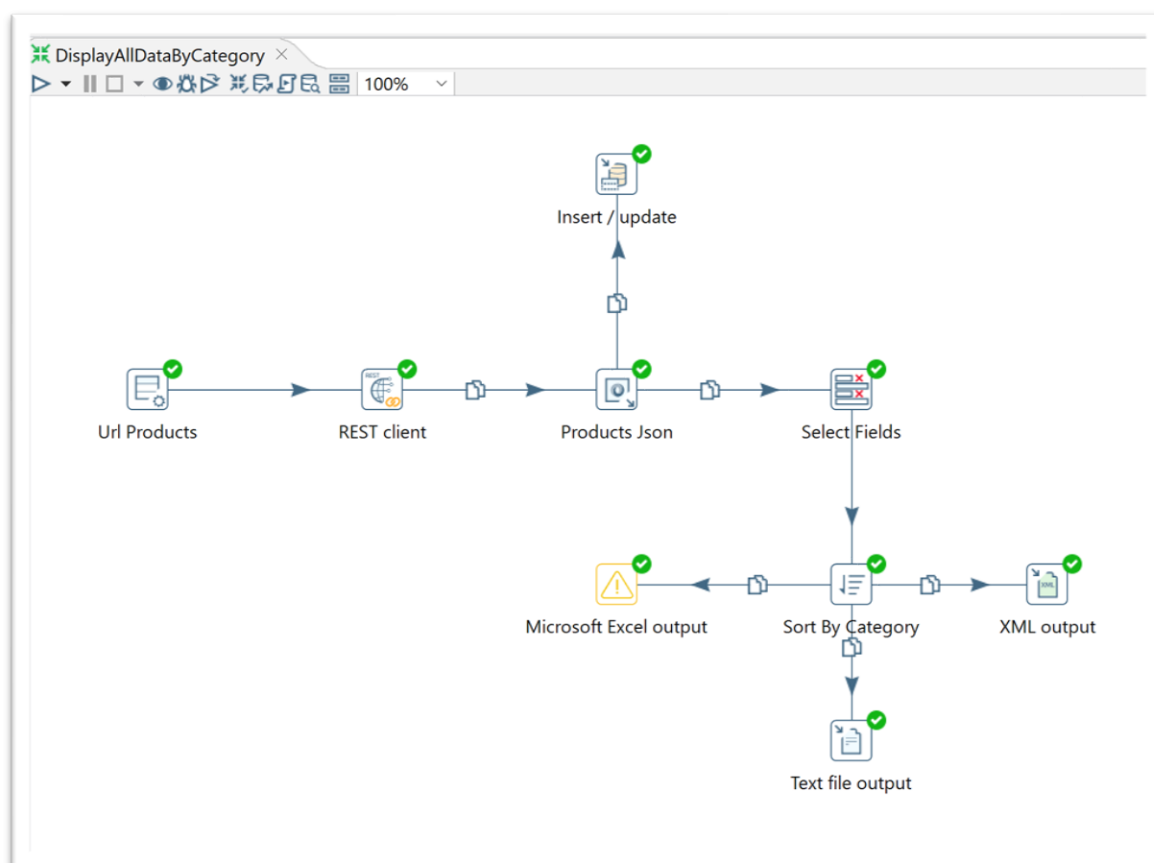


Figura 1- DisplayAllDataByCategory

Segunda transformação - *AllowHTMLtoEmail*

Esta transformação está a realizar um fluxo de leitura de dados de um arquivo, organizando os dados através de um agrupamento, e finalizando com um ajuste nos dados strings para serem utilizados ou enviados em um passo subsequente. Se o objetivo for gerar um relatório para permitir que o HTML seja enviado por email (como o nome sugere), o último componente pode estar formatando strings para esse propósito.

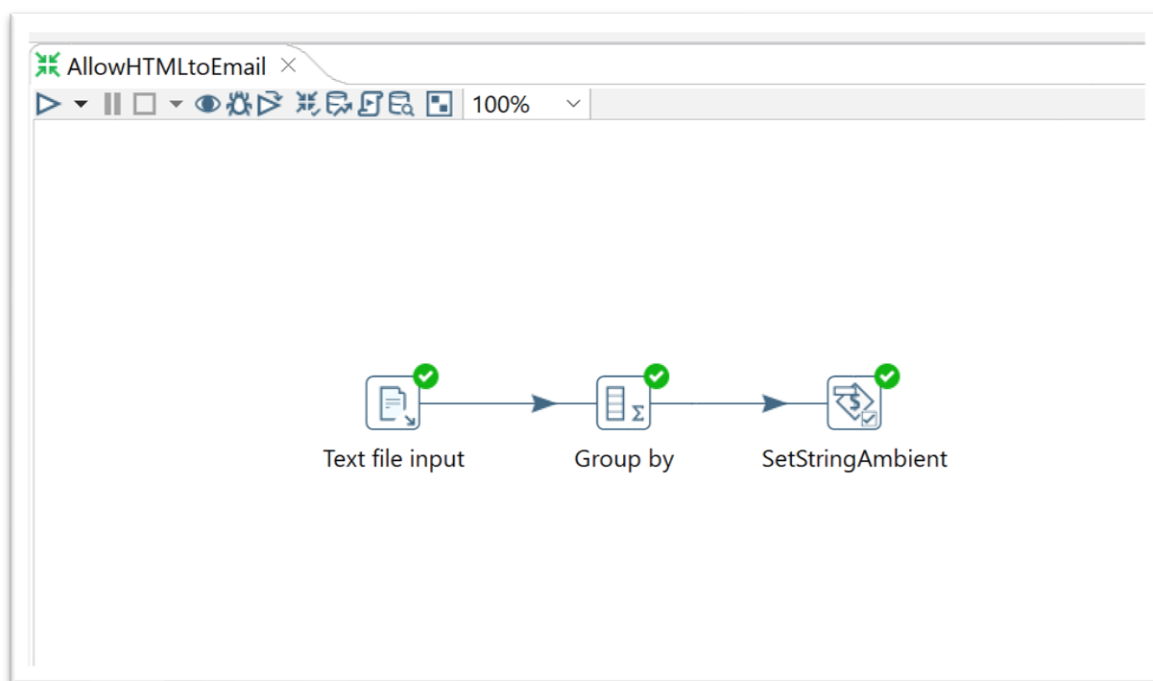


Figura 2 - AllowHTMLtoEmail

Terceira transformação - *FilterWithRegularExpression*

Explicação Geral do Fluxo:

1. **Entrada de Dados (API REST):** A transformação começa com a obtenção de dados de produtos de uma API REST, usando um cliente REST para consultar um endpoint que retorna um arquivo JSON.
2. **Processamento de Dados:** Os dados JSON são processados, extraindo campos específicos dos produtos, como título, preço, peso e marca.
3. **Verificações e Filtragem com Expressões Regulares:** Expressões regulares são aplicadas a vários campos para verificar se os dados estão corretos e no formato esperado. Produtos que não atendem aos critérios definidos são filtrados.
4. **Classificação:** Os dados filtrados são classificados por marca.
5. **Exportação:** Os dados finais são exportados para diferentes formatos, como XML, arquivo de texto e Excel, e também podem ser armazenados em um banco de dados.

Considerações:

- **Objetivo Principal:** A transformação visa processar dados de produtos de uma API REST, validá-los e filtrá-los usando expressões regulares, e exportar os resultados para múltiplos formatos.
- **Ponto de Atenção:** Há um alerta na saída para Excel, o que sugere que pode haver um problema de configuração nesta parte. Isso deve ser investigado para garantir que os dados sejam exportados corretamente.
- **Expressões Regulares:** As expressões regulares desempenham um papel central na validação dos dados, assegurando que apenas os produtos que atendem a certos critérios sejam incluídos nos resultados finais.
- **Flexibilidade:** A transformação inclui múltiplas opções de saída (XML, texto, Excel), permitindo flexibilidade na maneira como os dados são utilizados ou analisados posteriormente.

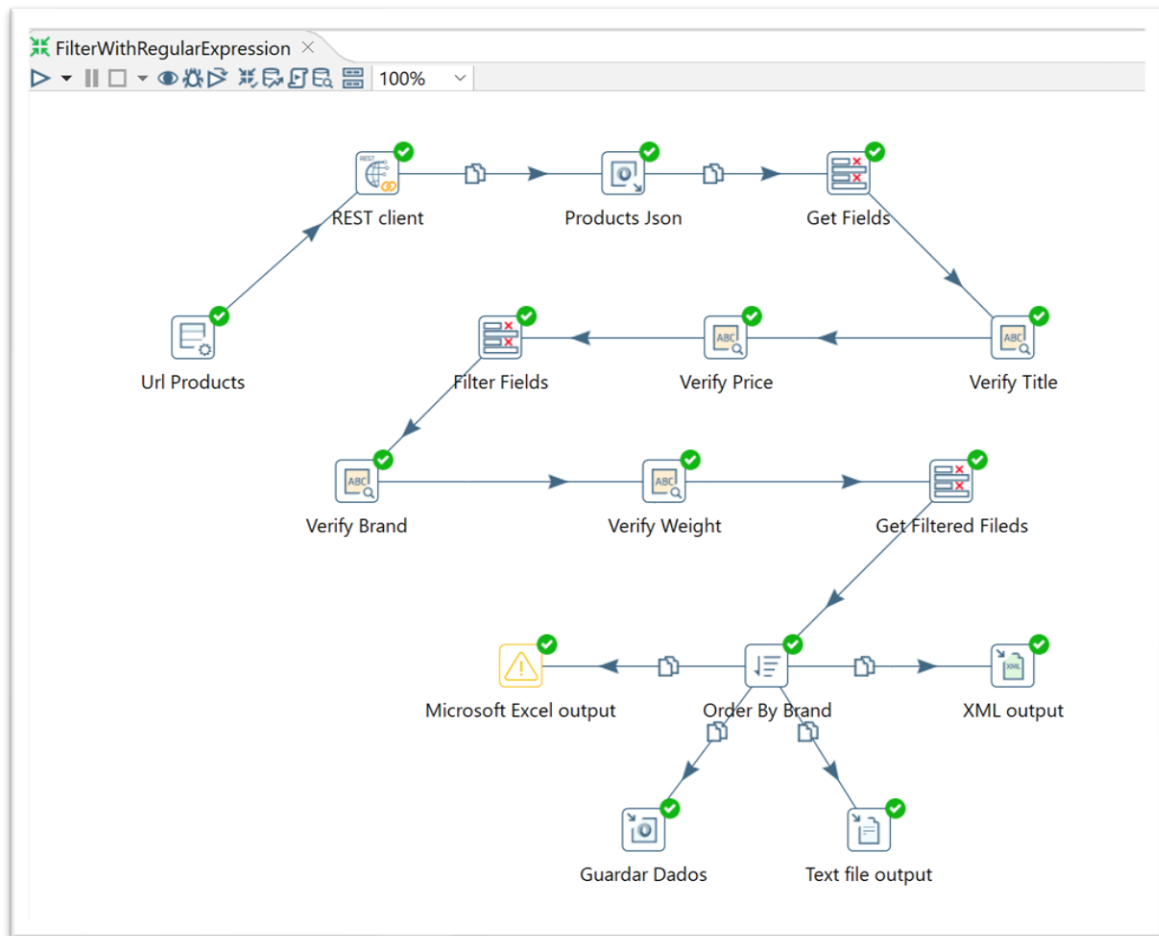


Figura 3 - FilterWithRegularExpression

Jobs

Assim como as transformações no **Pentaho Kettle** são **workflows** que realizam o processamento de dados, os Jobs no Pentaho Kettle têm um papel complementar, atuando como uma camada de orquestração e controle de fluxo. Enquanto as transformações se concentram em operar diretamente sobre os dados, os jobs gerem a execução das transformações e outros processos externos, permitindo a construção de pipelines complexos de ETL (Extração, Transformação e Carga).

Benefícios dos Jobs no Pentaho:

- **Orquestração Completa:** Os jobs permitem a orquestração de fluxos ETL complexos, controlando a execução de múltiplas transformações, sistemas e processos externos.
- **Automatização de Processos:** Jobs automatizam processos repetitivos, como mover arquivos, executar scripts e coordenar a execução de transformações com base em condições pré-definidas.
- **Integração com Sistemas Externos:** A capacidade de interagir com sistemas externos (através de comandos de sistema, verificações de arquivos, etc.) torna os jobs extremamente flexíveis e úteis em cenários de automação.
- **Condicionalidade e Controle:** Jobs fornecem um controle sofisticado sobre o fluxo de execução, permitindo tomar decisões dinâmicas com base no resultado de tarefas anteriores.
- **Notificações Automatizadas:** Jobs podem automatizar a comunicação e a geração de relatórios, enviando e-mails ou gerando logs em caso de falhas ou sucessos.

Primeiro Job - SendBDTableToEmail

Contexto:

Este job automatiza o processo de:

- Gerar um arquivo com dados de uma base de dados (provavelmente em formato XML ou HTML).
- Transformar esse arquivo para um formato adequado para envio por e-mail.
- Verificar e gerenciar a existência de arquivos anteriores.
- Enviar o resultado final como um anexo de e-mail.

Explicação Geral:

Este job automatiza um fluxo de ETL para gerir e enviar uma tabela ou conjunto de dados categorizados via e-mail. Ele:

- Garante que os arquivos antigos (XML ou HTML) sejam excluídos para evitar conflitos.
- Executa uma transformação para buscar e organizar os dados por categoria.
- Verifica se os arquivos necessários existem após a execução da transformação.
- Transforma os dados XML gerados em um arquivo HTML mais legível através de uma transformação XSLT.
- Prepara o arquivo HTML para envio por e-mail.
- Envia o arquivo ou seu conteúdo formatado por e-mail para um destinatário.

Considerações:

- **Eficiência:** O job é altamente automatizado, removendo arquivos antigos, gerando novos dados, verificando a existência dos arquivos e enviando-os automaticamente via e-mail.
- **Transformação XSLT:** A etapa de transformação XSLT garante que os dados sejam visualmente estruturados, permitindo que o destinatário receba um e-mail com conteúdo formatado adequadamente.
- **Robustez:** A verificação de arquivos existentes (etapas "File Exists") adiciona um nível de controle e evita que o job falhe por ausência ou conflitos de arquivos.

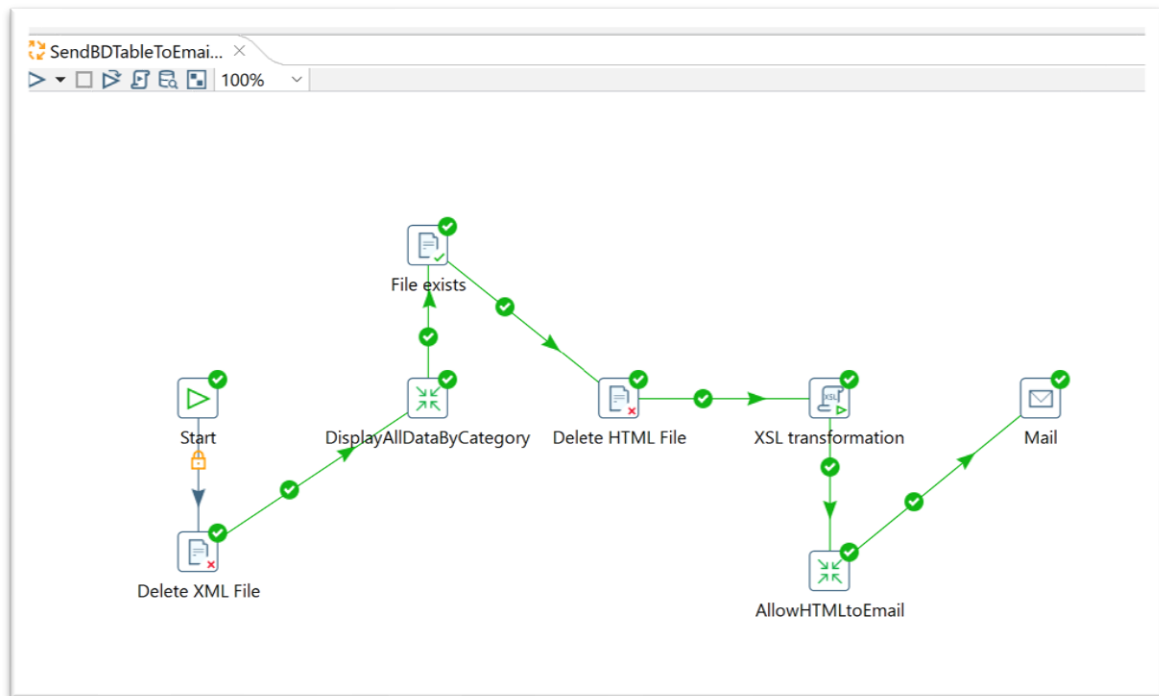


Figura 4 - SendBDTableToEmail

Finalidade:

Esse processo é útil para empresas ou projetos que precisam enviar relatórios automatizados por e-mail, atualizando informações de tabelas ou bases de dados diretamente para destinatários interessados. O uso de XSLT garante que os dados estejam em um formato adequado e visualmente apresentável no e-mail.

Node-Red

O **Node-RED** é uma ferramenta de desenvolvimento baseada em fluxo, que permite a conexão e orquestração de dispositivos, **APIs** e serviços online de forma visual. Desenvolvido originalmente pela IBM, o Node-RED é amplamente utilizado em projetos de automação, **IoT (Internet of Things)**, integração de sistemas e **workflows** de dados. Através de sua interface gráfica baseada em nós (nodes), usuários podem construir fluxos de trabalho de maneira intuitiva e sem a necessidade de programação complexa.

Características Principais:

1. Ambiente Visual de Fluxo;
2. Integração com **APIs** e Dispositivos;
3. Flexibilidade para Scripts e Código Customizado;
4. Processamento em Tempo Real;
5. Facilidade na Automação;
6. Open **Source** e Comunidade Ativa.

Aplicação no Projeto:

1. Orquestrar **Workflows** Automatizados;
2. Integração com Serviços Externos e **APIs**;
3. Monitoramento e Reação a Eventos em Tempo Real;
4. Automação de Processos de Notificação;
5. Execução de Lógica Customizada;

Benefícios para o Projeto:

Interface Simples e Flexível: A facilidade de uso do Node-RED, com sua interface de desenvolvimento baseada em nós, permitiu criar rapidamente fluxos de trabalho complexos, economizando tempo de desenvolvimento e simplificando a manutenção dos sistemas.

Automatização e Monitoramento Eficiente: Foi possível automatizar tarefas repetitivas e monitorar eventos em tempo real, o que melhorou a eficiência operacional e permitiu respostas rápidas a eventos críticos no processo de negócio.

Escalabilidade e Integração: Através da vasta biblioteca de nós e sua capacidade de integração com **APIs** e sistemas de terceiros, o Node-RED se mostrou altamente escalável e adaptável às necessidades crescentes do projeto.

Código Aberto e Comunidade Ativa: Como uma plataforma open source, o Node-RED oferece a vantagem de ter uma comunidade ativa, o que facilitou a resolução de problemas e a adição de funcionalidades personalizadas ao projeto.

Estrutura do Node-Red

Estrutura Geral:

Este fluxo parece ser replicado em três instâncias paralelas, possivelmente para realizar diferentes consultas SQL em bases de dados semelhantes ou em tabelas diferentes da mesma base de dados. Cada fluxo segue o mesmo padrão:

1. Um gatilho relacionado a Produtos inicia o processo.
2. Uma função formata ou ajusta a consulta SQL.
3. O nó **MSSQL-PLUS** executa a consulta no banco de dados SQL Server.
4. Os resultados são exibidos ou processados pelo nó Information.

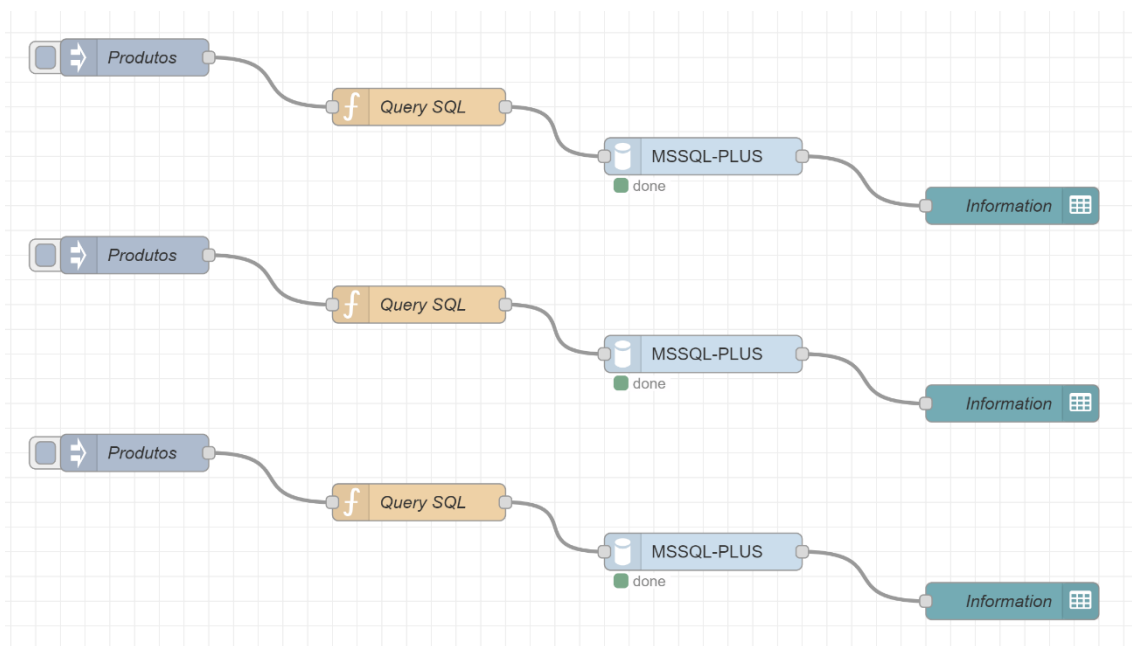


Figura 5 - Estrutura do Node-Red

Considerações:

- **Consulta de Dados:** O fluxo está claramente configurado para consultar informações de um banco de dados Microsoft SQL Server.
- **Escalabilidade e Paralelismo:** O fato de haver três fluxos paralelos sugere que diferentes consultas estão sendo executadas simultaneamente ou com variações mínimas, permitindo lidar com múltiplos tipos de dados ou tabelas relacionadas a "Produtos".
- **Modularidade:** O uso de um nó de função para gerar as consultas SQL antes de enviá-las ao banco de dados oferece flexibilidade, permitindo que o fluxo se adapte facilmente a diferentes cenários de consulta ou manipulação de dados.

Componentes do Fluxo:

1. Produtos (Nó de Entrada):

- **Objetivo:** Este nó representa a origem dos dados ou um gatilho para iniciar o processo. Ele pode estar recebendo uma solicitação ou atuando como um ponto de partida quando algo relacionado aos produtos é acionado.

2. Query SQL (Função):

- **Objetivo:** O nó **Query SQL** parece ser uma função personalizada que constrói ou ajusta uma consulta SQL a ser enviada para a base de dados.

3. MSSQL-PLUS (Conexão com SQL Server):

- **Objetivo:** Este é o nó responsável por conectar-se a uma base de dados Microsoft SQL Server e executar a consulta SQL gerada no nó anterior.

4. Information (Exibição ou Processamento dos Resultados):

- **Objetivo:** Este nó provavelmente exibe os dados obtidos da consulta SQL ou os processa de alguma forma. Ele pode estar a formatar os resultados para serem apresentados em uma interface, exportados para outro sistema ou armazenados em um arquivo.

Primeiro Nó da estrutura: Node-red

id	title	description	category	price	stock	brand	weight
41	Tissue Paper Box	Convenient tissue ...	groceries	2.49	97		3
47	Table Lamp	The Table Lamp is ...	home-decoration	49.99	84		2
75	Tray	The Tray is a functi...	kitchen-accessories	16.99	48		8
112	TV Studio Camera ...	The TV Studio Ca...	mobile-accessories	499.99	45	ProVision	7
151	Tennis Ball	The Tennis Ball is a...	sports-accessories	6.99	99		3
152	Tennis Racket	The Tennis Racket ...	sports-accessories	49.99	86		2
166	Tartan Dress	The Tartan Dress f...	tops	39.99	100		4
184	Tropical Earring	The Tropical Earrin...	womens-jewellery	19.99	76		9

Figura 6 - Primeiro Nó da estrutura

Segundo Nó da estrutura: Node-red

id	title	description	category	price	stock	brand	weight
2	Eyeshadow Palett...	The Eyeshadow P...	beauty	19.99	44	Glamour Beauty	3
3	Powder Canister	The Powder Canis...	beauty	14.99	59	Velvet Touch	8
4	Red Lipstick	The Red Lipstick is...	beauty	12.99	68	Chic Cosmetics	5
5	Red Nail Polish	The Red Nail Polis...	beauty	8.99	71	Nail Couture	9
7	Chanel Coco Noir ...	Coco Noir by Cha...	fragrances	129.99	41	Chanel	4
8	Dior J'adore	J'adore by Dior is ...	fragrances	89.99	91	Dior	10
10	Gucci Bloom Eau de	Gucci Bloom by G...	fragrances	79.99	93	Gucci	10
11	Annibale Colomb...	The Annibale Col...	furniture	1899.99	47	Annibale Colombo	3
14	Knoll Saarinen Exe...	The Knoll Saarine...	furniture	499.99	47	Knoll	3
15	Wooden Bathroo...	The Wooden Bath...	furniture	799.99	95	Bath Trends	6
17	Beef Steak	High-quality beef ...	groceries	12.99	96		9
19	Chicken Meat	Fresh and tender ...	groceries	9.99	69		7
22	Dog Food	Specially formulat...	groceries	10.99	40		2
24	Fish Steak	Quality fish steak, ...	groceries	14.99	99		8
25	Green Bell Pepper	Fresh and vibrant ...	groceries	1.29	89		7
28	Ice Cream	Creamy and delici...	groceries	5.49	76		5
29	Juice	Refreshing fruit ju...	groceries	3.99	99		2
32	Milk	Fresh and nutritio...	groceries	3.49	57		1
33	Mulberry	Sweet and juicy m...	groceries	4.99	79		4
36	Protein Powder	Nutrient-packed ...	groceries	19.99	65		2

Figura 7 - Segundo Nó da estrutura

Terceiro Nó da estrutura: Node-red

id	title	description	category	price	stock	brand	weight
83	Blue & Black Chec...	The Blue & Black ...	mens-shirts	29.99	44	Fashion Trends	6
84	Gigabyte Aorus M...	The Gigabyte Aor...	mens-shirts	24.99	64	Gigabyte	2
85	Man Plaid Shirt	The Man Plaid Shi...	mens-shirts	34.99	65	Classic Wear	1
86	Man Short Sleeve ...	The Man Short Sle...	mens-shirts	19.99	20	Casual Comfort	2
87	Men Check Shirt	The Men Check Sh...	mens-shirts	27.99	69	Urban Chic	1
172	Blue Women's Ha...	The Blue Women's...	womens-bags	49.99	76	Fashionista	2
173	Heshe Women's L...	The Heshe Wome...	womens-bags	129.99	9	Heshe	5
174	Prada Women Bag	The Prada Women...	womens-bags	599.99	43	Prada	4
175	White Faux Leathe...	The White Faux Le...	womens-bags	39.99	89	Urban Chic	1
176	Women Handbag ...	The Women Hand...	womens-bags	59.99	24	Elegance Collection	1

Figura 8 - Terceiro Nó da estrutura

Conteúdo dos Nós da estrutura

Edit MSSQL node

Delete

Cancel

Done

⚙️ Properties

⚙️ 📄 🖨️

🌐 Connection

ISIECommerce

✎

+

🏷️ Name

Name

☰ Query mode

▼ Query

💬 Query

▼ msg. payload

📋 Parameters

▼ None

🏷️ Parse Mustache

☒

➡️ Output property

msg. payload

➡️ Output type

Original output

▼

⚠️ Error Handling

Throw error

▼

Figura 9 - Conexão com a BD

A imagem mostra a configuração de um nó MSSQL no Node-RED, utilizado para executar consultas SQL em uma base de dados Microsoft SQL Server com a conexão chamada **ISIECommerce**. A consulta SQL é recebida dinamicamente através do campo **msg.payload**, permitindo que o SQL seja definido em tempo de execução. A opção **Parse Mustache** está ativada, o que possibilita o uso de variáveis dinâmicas na consulta. Os resultados da consulta são retornados no **msg.payload** e o tratamento de erros está configurado para interromper o fluxo caso haja falhas na execução da consulta.

Edit MSSQL node > Edit MSSQL-CN node

Delete Cancel Update

⚙ Properties

🔑 Name ISIECommerce

🖨 Server localhost

🔌 Port 1433

👤 Username ISIECommerce

🔒 Password

👤 Domain

🗄 Database ISIECommerce

📋 TDS Version 7_4 (SQL Server 2012 ~ 2022) ▾

🔒 Use Encryption? ☒
 SQL Databases hosted on Azure will need this checked.

🔒 Trust Certificate? ☒
 If unchecked, SQL Server will try to validate the server SSL certificate and will terminate the connection if validation fails

🕒 Assume UTC? ☒
 Pass time values in UTC or local time.

📄 Enabled 3 On all flows ▾

Figura 10 - Criação da Conexão

A imagem mostra a configuração de uma conexão MSSQL no Node-RED para o banco de dados **ISIECommerce**. A conexão usa o servidor local (**localhost**), porta padrão do SQL Server (1433), com o usuário **ISIECommerce** e senha configurada. A base de dados acedida é o **ISIECommerce**, e a versão do protocolo TDS está definida como **7.4 (SQL Server 2012 ~ 2022)**. A opção de criptografia está ativada, garantindo que a conexão seja segura, e o campo. Assume UTC está habilitado, para garantir que os valores de tempo sejam tratados no fuso UTC.

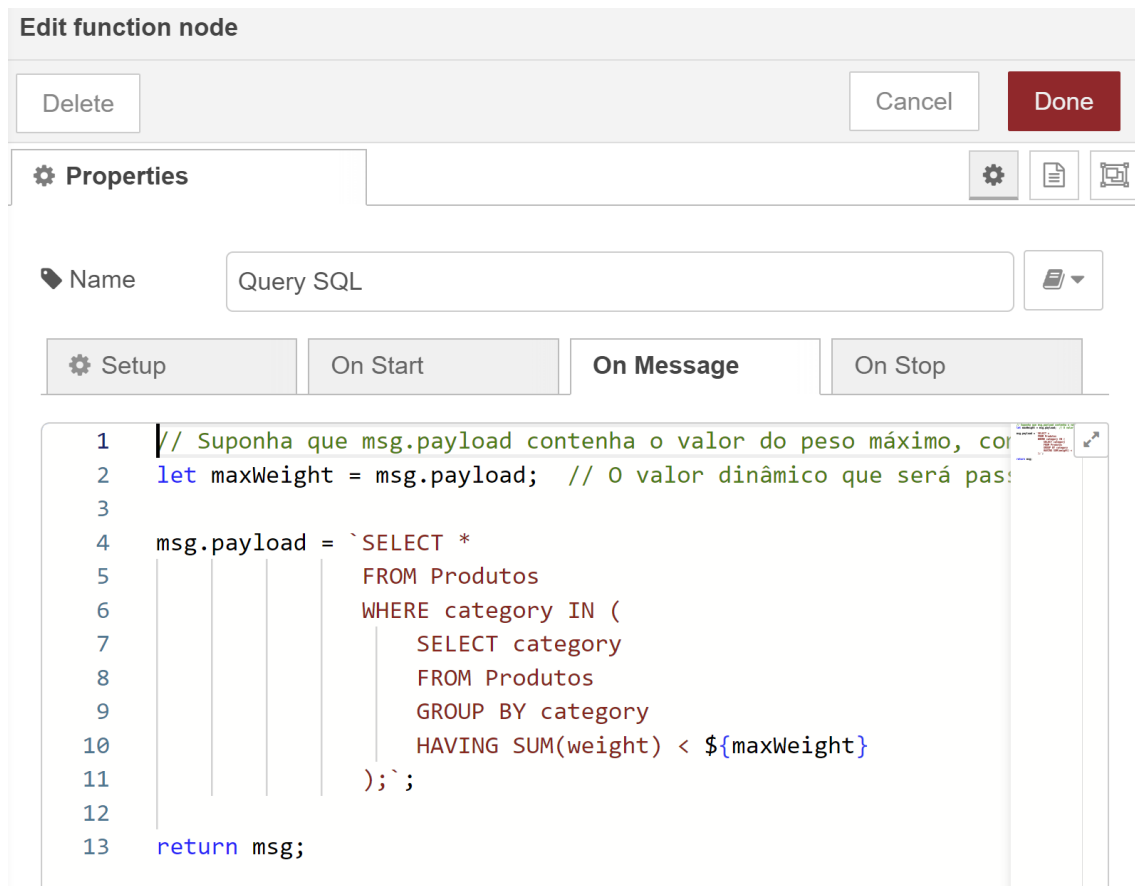


Figura 11 - Função da query da BD

A imagem mostra a configuração de um nó de função no **Node-RED** chamado **Query SQL**. Nesse nó, uma consulta SQL dinâmica é construída com base em um valor recebido no **msg.payload**, que contém o valor de um peso máximo (**maxWeight**). A consulta seleciona todos os produtos da tabela **Produtos** que pertencem a categorias onde a soma dos pesos de todos os produtos é menor que esse valor de **maxWeight**. O valor dinâmico é inserido na consulta utilizando **template strings** em JavaScript, e a consulta final é retornada no **msg.payload** para ser executada em um banco de dados SQL.

Vídeo

Scan para
visualizar o
vídeo



Conclusão

Neste relatório, foi demonstrado como a análise e limpeza de dados desempenham um papel fundamental na gestão eficaz de uma base de dados de um **e-commerce**. A aplicação de técnicas de validação de dados e a utilização de ferramentas como expressões regulares (**Regex**) permitiram garantir que as informações essenciais, como preço, **stock** e categorias, estejam formatadas corretamente e dentro dos parâmetros esperados. Isso assegura que o banco de dados contenha apenas informações relevantes e precisas, evitando problemas e melhorando a experiência do cliente na plataforma.

Em termos práticos, este projeto evidenciou a importância de se garantir a integridade dos dados como parte de uma estratégia maior de tomada de decisão baseada em dados confiáveis. Ao assegurar que os dados estejam corretos, a empresa poderá tomar decisões mais assertivas sobre marketing, gestão de inventário e estratégias de vendas, otimizando os resultados do negócio.

Webgrafia

Para auxílio na implementação deste relatório foi utilizado a AI:

- **Chat Open AI:** <https://chat.openai.com/>;

Para criação de nós em Node-Red, foi utilizada a plataforma:

- **Node-RED:** <https://nodered.org/>

Para criação da base de dados, foi utilizada a plataforma:

- **SQL Server Management Studio (SSMS):** <https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16>

Para criação do QR Code para o vídeo, foi utilizado o site:

- **QR Code Generator:** <https://goqr.me/#t=url>