**Documentação do Código - Automação de Solicitações de Pagamento e Integração com Notion e Banco Inter**

**Situação Problema**

A empresa realiza pagamentos recorrentes para fornecedores, mas o processo manual de lançamento e aprovação de pagamentos frequentemente causa atrasos e erros, impactando a eficiência financeira e a conciliação bancária. O fluxo manual é suscetível a falhas de comunicação, e o status de pagamentos nem sempre é atualizado corretamente no sistema.

**Objetivo**

Este código tem como objetivo automatizar o processamento de pagamentos para fornecedores utilizando o Notion e o Banco Inter, realizando as seguintes tarefas:

* **Obter dados de pagamentos pendentes e dados bancários dos fornecedores no Notion.**
* **Criar e enviar payloads de pagamento para a API do Banco Inter, utilizando Pix ou dados bancários.**
* **Atualizar o status dos itens de solicitação de pagamento no Notion.**

**Como Funciona**

1. **Acesso ao Notion**: O código acessa o Notion para obter itens de solicitação de pagamento com status "Pagamento em processamento".
2. **Verificação de Dados**: Verifica se os dados bancários ou a chave Pix do fornecedor estão presentes.
3. **Criação de Payload**: Cria um payload de pagamento baseado nas informações disponíveis (Chave Pix ou Dados Bancários).
4. **Envio de Pagamento**: Envia o payload de pagamento à API do Banco Inter.
5. **Atualização de Status**: Atualiza o status do item de solicitação de pagamento no Notion para "Pagamento em aprovação" após o lançamento do pagamento.

**Requisitos Obrigatórios para Funcionamento do Código**

**Configurações Necessárias no Notion**

* **API do Banco Inter**: A Client ID, Cliente Secret, Certificado Digital (.crt) e (.key) fornecidos pelo internet Banking na ativação da aplicação API.
* **API do Notion**: A chave da API e os IDs das bases de dados são necessários.
* **Base de Dados de Solicitação de Pagamento**: O código acessa a base de dados para identificar pagamentos a serem realizados.
* **Base de Dados Bancária**: O código acessa a base de dados bancária para obter os dados bancários ou chave Pix dos fornecedores.

**Ferramentas Instaladas**

* **Python 3.x**: A versão recomendada é 3.6 ou superior.
* **Bibliotecas Python**: requests, json, time, msvcrt.

**Arquivos de Configuração**

* **notion\_credenciais.json**: Contém as credenciais da API do Notion e os IDs das bases de dados.
* **inter\_credenciais.json**: Contém as credenciais da API do Banco Inter, incluindo client\_id, client\_secret e caminhos para os certificados necessários.

**Estrutura das Bases de Dados no Notion**

* **Base de Dados de Solicitação de Pagamento**: Deve ter as propriedades Status (select), Chave Pix e Conta (textos, dependendo do caso) e outras informações relacionadas ao pagamento.
* **Base de Dados Bancária**: Contém as informações dos fornecedores, como número de conta, chave Pix, agência e ISPB.

**Setup dos Arquivos JSON e Configuração de Caminhos**

**1. Arquivos JSON Necessários**

**O código utiliza dois arquivos JSON para armazenar credenciais e configurações essenciais. Esses arquivos devem ser criados e configurados pelo usuário antes de executar o programa.**

**a) notion\_credenciais.json**

**Este arquivo contém as credenciais da API do Notion e os IDs das bases de dados. O usuário deve criar este arquivo no formato JSON e definir os seguintes campos:**

**{**

**"Pagamentos\_de\_Projetos\_Key": "sua\_chave\_da\_api\_notion\_aqui",**

**"Pagamentos\_de\_Projetos\_ID": "id\_da\_base\_de\_dados\_de\_pagamentos",**

**"Dados\_Bancarios\_Fornecedores\_ID": "id\_da\_base\_de\_dados\_bancarios"**

**}**

* **Pagamentos\_de\_Projetos\_Key: Chave da API do Notion.**
* **Pagamentos\_de\_Projetos\_ID: ID da base de dados de solicitação de pagamentos.**
* **Dados\_Bancarios\_Fornecedores\_ID: ID da base de dados bancários de fornecedores.**

**b) inter\_credenciais.json**

**Este arquivo contém as credenciais da API do Banco Inter e os caminhos para os certificados. O usuário deve criar este arquivo no formato JSON e definir os seguintes campos:**

**{**

**"CONTA\_CORRENTE": "sua\_conta\_corrente\_aqui",**

**"CLIENT\_ID": "seu\_client\_id\_aqui",**

**"CLIENT\_SECRET": "seu\_client\_secret\_aqui"**

**}**

* **CONTA\_CORRENTE: Número da conta corrente no Banco Inter.**
* **CLIENT\_ID: Client ID fornecido pelo Banco Inter.**
* **CLIENT\_SECRET: Client Secret fornecido pelo Banco Inter.**

**2. Configuração dos Caminhos no Código**

**O código precisa saber onde os arquivos JSON e os certificados estão localizados. Para isso, o usuário deve definir os caminhos corretos no código. Abaixo estão os trechos do código que precisam ser ajustados:**

**a) Caminho para notion\_credenciais.json**

**No código, o caminho para o arquivo notion\_credenciais.json é definido na variável notion\_credenciais\_json. O usuário deve alterar o caminho para refletir a localização do arquivo no seu sistema.**

**notion\_credenciais\_json = "C:\\Users \\Scripts\\notion\_credenciais.json"**

**b) Caminho para inter\_credenciais.json**

**O caminho para o arquivo inter\_credenciais.json é definido na variável inter\_credenciais\_json. O usuário deve ajustar o caminho conforme necessário.**

**inter\_credenciais\_json = "C:\\Users\\Scripts\\Inter\_API-Chave\_e\_Certificado\\inter\_credenciais.json"**

**c) Caminho para os Certificados do Banco Inter**

**O código utiliza dois arquivos fornecidos pelo Banco Inter: o certificado (.CRT) e a chave privada (.KEY). O usuário deve definir os caminhos corretos para esses arquivos nas variáveis cert\_path e key\_path.**

**cert\_path = "C:\\Users\\Scripts\\Inter\_API-Chave\_e\_Certificado\\Certificado.crt"**

**key\_path = "C:\\Users\\Scripts\\Inter\_API-Chave\_e\_Certificado\\Chave.key"**

**3. Passos para Configuração**

1. **Crie os Arquivos JSON:**
   * **Crie os arquivos notion\_credenciais.json e inter\_credenciais.json com as credenciais necessárias.**
   * **Certifique-se de que os arquivos estejam no formato JSON válido.**
2. **Defina os Caminhos no Código:**
   * **Localize as variáveis notion\_credenciais\_json, inter\_credenciais\_json, cert\_path e key\_path no código.**
   * **Altere os caminhos para refletir a localização dos arquivos no seu sistema.**
3. **Obtenha os Certificados do Banco Inter:**
   * **Certifique-se de que os arquivos .CRT e .KEY fornecidos pelo Banco Inter estejam no local especificado nos caminhos.**
4. **Execute o Código:**
   * **Após configurar os arquivos e caminhos, o código estará pronto para ser executado.**

**Exemplo de Estrutura de Pastas**

**Aqui está um exemplo de como a estrutura de pastas e arquivos pode ser organizada:**

**C:\Users\Scripts\**

**│**

**├── notion\_credenciais.json**

**├── Inter\_API-Chave\_e\_Certificado\**

**│ ├── inter\_credenciais.json**

**│ ├── Certificado.crt**

**│ └── Chave.key**

**└── script\_automacao\_pagamentos.py**

**Observações Importantes**

* **Segurança: Certifique-se de que os arquivos JSON e certificados estejam armazenados em um local seguro, pois eles contêm informações sensíveis.**
* **Caminhos Absolutos vs. Relativos: O código utiliza caminhos absolutos. Se preferir usar caminhos relativos, ajuste as variáveis conforme necessário.**
* **Permissões de Arquivos: Verifique se o código tem permissão para acessar os arquivos JSON e certificados.**

**Passo a Passo do Funcionamento**

**Descrição Geral**

Este script automatiza o processo de:

1. **Obtenção de Dados do Notion**: Coleta dados de pagamentos a serem feitos e dados bancários dos fornecedores.
2. **Criação de Payloads de Pagamento**: Criação do payload com base nas informações de pagamento disponíveis (Chave Pix ou Dados Bancários).
3. **Envio de Pagamentos para o Banco Inter**: Realiza o lançamento do pagamento utilizando a API do Banco Inter.
4. **Atualização de Status no Notion**: Atualiza o status do item para "Pagamento em aprovação".

**Principais Funcionalidades**

1. **Leitura de Dados do Notion**:
   * Recupera itens de solicitação de pagamento com status "Pagamento em processamento".
   * Obtém dados bancários ou chave Pix do fornecedor a partir da base de dados bancária.
2. **Criação de Payloads**:
   * Cria payloads com base nas informações obtidas, formatando o pagamento com chave Pix ou dados bancários, conforme necessário.
3. **Envio para o Banco Inter**:
   * Envia a solicitação de pagamento via API do Banco Inter.
4. **Atualização do Status no Notion**:
   * Após o pagamento ser enviado, o status é atualizado para "Pagamento em aprovação".

**Dependências**

**Bibliotecas Python**

* **requests**: Para fazer requisições HTTP.
* **json**: Para manipulação de dados no formato JSON.
* **time**: Para manipulação de tempo.
* **msvcrt**: Para captura de entradas no Windows.

**Configuração**

**Arquivos Necessários**

* **notion\_credenciais.json**: Contém chaves de autenticação da API do Notion e IDs da base de dados.
* **inter\_credenciais.json**: Contém as credenciais da API do Banco Inter, incluindo client\_id, client\_secret e certificados.

**Caminhos e Comandos Importantes**

* **token\_inter.json**: Caminho para o arquivo de credenciais do Banco Inter.
* **notion\_credenciais.json**: Caminho para o arquivo de credenciais do Notion.

**Estrutura da Base de Dados do Notion**

* A base de dados de solicitação de pagamento deve ter as propriedades Status e campos de dados bancários, como Chave Pix, Conta, Agência etc.

**Funções do Código**

1. **get\_notion\_data(database\_id)**:
   * Recupera todos os itens de uma base de dados do Notion.
   * Utiliza a API do Notion para acessar a base de dados e recuperar os itens necessários.
2. **extract\_property\_value(page\_id, property\_name)**:
   * Extrai o valor de uma propriedade específica de uma página do Notion.
3. **get\_relation\_names(relation\_ids)**:
   * Obtém os nomes relacionados a IDs de relação no Notion.
4. **get\_property\_value(properties, key, default='')**:
   * Obtém o valor de uma propriedade específica de um objeto de propriedades.
5. **get\_text\_value(properties, key, default='')**:
   * Obtém o valor de texto de uma propriedade específica.
6. **atualizar\_status\_notion(page\_id, novo\_status)**:
   * Atualiza o status de uma página no Notion.
7. **inserir\_id\_transacao(page\_id, codigoSolicitacao)**:
   * Insere o ID da transação no Notion.
8. **obter\_token()**:
   * Obtém o token de acesso do Banco Inter.
9. **carregar\_token()**:
   * Carrega o token do arquivo ou obtém um novo se necessário.
10. **Fluxo Principal**:
    * Coordena a execução do fluxo do script, incluindo a leitura de dados, criação de payloads e envio para o Banco Inter.

**Fluxo do Script**

1. **Leitura de Dados**:
   * Acessa as bases de dados do Notion para obter itens com status "Pagamento em processamento" e dados bancários dos fornecedores.
2. **Criação do Payload**:
   * Cria um payload com as informações do pagamento, dependendo de se há chave Pix ou dados bancários.
3. **Envio do Pagamento**:
   * Envia a solicitação de pagamento para a API do Banco Inter.
4. **Atualização de Status**:
   * Atualiza o status do item no Notion para "Pagamento em aprovação" após o pagamento ser lançado.

**Erros e Exceções**

* **Erro ao acessar o Notion**: Erros de requisição são tratados com requests.exceptions.RequestException.
* **Erro ao criar payload de pagamento**: Erros de formatação de payload são tratados e logados.
* **Falha ao enviar pagamento ao Banco Inter**: Erros de comunicação com a API são tratados e logados.
* **Status do item no Notion**: Caso o status não seja atualizado corretamente, o erro é registrado e o item é ignorado.

**Observações**

* **Autenticação com o Banco Inter**: A autenticação usa client\_id, client\_secret e certificados digitais para garantir a segurança da transação.
* **Formato de Dados**: O código garante que os dados de pagamento sejam formatados corretamente antes de serem enviados para o Banco Inter.