### **Documento 5: Relatório de Lições Aprendidas (Análise de Erros)**

**Projeto:** Análise de Impacto e Estratégia de Mitigação de Riscos Cambiais e Tarifários para Importações dos EUA **Data:** 27 de Julho de 2025 **Assunto:** Análise dos Erros de Programação Encontrados e Soluções Aplicadas

**1. Introdução:** Este documento resume os principais erros de programação encontrados durante a fase de análise de dados deste projeto. O objetivo é documentar as causas, as soluções aplicadas e as lições aprendidas, criando uma base de conhecimento para acelerar a resolução de problemas em projetos futuros.

**2. Erros Analisados:**

* **Erro 1: NameError: name '...' is not defined**
  + **Significado:** O Python tentou usar uma variável que não havia sido criada (definida) na memória da sessão atual.
  + **Causa no Projeto:** Este erro ocorreu repetidamente ao reiniciar o trabalho no Google Colab. Executamos uma célula que tentava modificar uma tabela (ex: df\_importacao) antes de executar a célula que a criava (lendo o arquivo com pd.read\_excel).
  + **Solução e Lição Aprendida:** A solução foi sempre garantir a **ordem de execução sequencial** das células, começando pela célula de "Setup" que carrega todos os dados para a memória. A lição é que o ambiente de um notebook não tem memória entre sessões, e o fluxo de trabalho deve ser executado de cima para baixo.
* **Erro 2: UnicodeDecodeError: 'utf-8' codec can't decode...**
  + **Significado:** O Python tentou ler um arquivo usando o "idioma" (encoding) padrão UTF-8, mas o arquivo foi salvo em um idioma diferente.
  + **Causa no Projeto:** O arquivo CSV baixado do Banco Central do Brasil estava codificado em latin-1.
  + **Solução e Lição Aprendida:** Adicionamos o parâmetro encoding='latin-1' ao comando pd.read\_csv(). A lição é que, ao trabalhar com fontes de dados diversas (especialmente brasileiras), é fundamental estar preparado para testar diferentes encodings.
* **Erro 3: ValueError: time data '...' doesn't match format...**
  + **Significado:** A função pd.to\_datetime encontrou um texto em uma coluna que não correspondia ao formato de data esperado.
  + **Causa no Projeto:** O arquivo do BCB continha linhas de rodapé com texto (a palavra "Fonte"), que foram lidas como se fossem dados.
  + **Solução e Lição Aprendida:** Tornamos o processo de conversão mais robusto usando errors='coerce', que transforma textos inválidos em valores nulos, e em seguida usamos .dropna() para remover essas linhas. A lição é que arquivos de dados "reais" raramente são perfeitamente limpos e nosso código precisa ser robusto para lidar com rodapés e outros "lixos".
* **Erro 4: KeyError: '...'**
  + **Significado:** O código tentou acessar uma coluna em um DataFrame usando um nome que não existia na lista de colunas.
  + **Causa no Projeto:** Ocorreu quando tentamos usar o nome 'taxa\_dolar\_bcb' antes que a coluna 'Valor' (ou o nome longo da série) fosse efetivamente renomeada.
  + **Solução e Lição Aprendida:** Usamos df.columns para diagnosticar os nomes reais das colunas e garantimos que a célula de renomeação (.rename()) fosse executada antes das células que usavam o novo nome. A lição é a importância da precisão nos nomes e, novamente, da ordem de execução.
* **Erro 5: AttributeError: '...' object has no attribute '...'**
  + **Significado:** O código tentou usar um método ou atributo (um "comando") que não pertence àquele objeto.
  + **Causa no Projeto:** Ocorreu por dois erros de digitação: primeiro, df.heaed() em vez de df.head(); segundo, ao usar o parâmetro xytext (que pertence à função plt.annotate) na função plt.text.
  + **Solução e Lição Aprendida:** Corrigimos o erro de digitação e ajustamos o código para usar os parâmetros corretos da função plt.text. A lição é que a programação exige precisão e é crucial consultar a documentação das funções para usar seus parâmetros corretamente.
* **Erro 6: SyntaxError: incomplete input**
  + **Significado:** O Python chegou ao final de uma célula sem que um comando ou bloco de código fosse finalizado corretamente.
  + **Causa no Projeto:** Um erro de copiar-colar onde o final de um bloco de código else e o resto do script não foram colados na célula.
  + **Solução e Lição Aprendida:** Revisamos o código, identificamos a parte faltante e colamos o script completo. A lição é a importância de revisar o código após colar para garantir sua integridade.