**Documentação de Atividades - Python e Microsoft Suite**

**Índice**

1. Extração de Notas dos Alunos da PUC
2. Top 250 Filmes do IMDB
3. Envio de E-mail e Mensagem no Teams com o CommonKit
4. Aplicação de Batida de Ponto no PowerApps
5. Trigger de Power Automate a partir do Teams
6. Criação de API com FastAPI para Consultar Logs de Erro no MongoDB

**Extração de Notas dos Alunos da PUC**

**Descrição**: Extrair as notas do portal WebAluno da PUC e salvá-las em um arquivo Excel ou CSV.

**Requisitos**: Usuário deve ser um aluno ativo da PUC.

**Passo a Passo**:

1. Acessar o portal WebAluno da PUC: https://432f5d.mannesoftprime.com.br/webaluno/.
2. Navegar até a seção de notas.
3. Realizar a extração das notas:

- Utilizar a funcionalidade de exportação do portal, se disponível.

- Caso contrário, copiar os dados manualmente.

1. Salvar os dados coletados em uma planilha Excel ou em formato CSV para armazenamento e análise futura.

**Top 250 Filmes do IMDB**

**Descrição**: Extrair os 250 filmes mais populares do IMDB, salvar o nome do filme e sua nota em um banco de dados.

**Requisitos**: Acesso ao site IMDB.

**Passo a Passo**:

1. Acessar a página de Top 250 Filmes no IMDB: https://www.imdb.com/chart/top/?ref\_=nv\_mv\_250.
2. Extrair as informações de nome do filme e nota.
3. Salvar as informações no banco de dados chamado Atividades:

- Estrutura sugerida para o banco de dados: Tabela Filmes com colunas nome (string) e nota (float).

1. Verificar a consistência dos dados e garantir que todos os filmes e notas foram salvos corretamente.

**Envio de E-mail e Mensagem no Teams com o CommonKit**

**Descrição**: Utilizar o CommonKit para automatizar o envio de uma mensagem para si mesmo via e-mail e para o canal de teste no Teams.

**Requisitos**: Acesso à biblioteca CommonKit e ao canal de teste no Microsoft Teams.

**Passo a Passo**:

1. Usar o CommonKit para configurar o envio de um e-mail para o próprio usuário.

- Conteúdo do e-mail:

"Olá, sou o robô. Se eu puder te ajudar, basta me chamar."

1. Enviar uma mensagem no canal de teste no Teams com o mesmo conteúdo.
2. Verificar se ambas as mensagens (e-mail e Teams) foram enviadas e recebidas corretamente.

**Aplicação de Batida de Ponto no PowerApps**

**Descrição**: Desenvolver uma aplicação no PowerApps para registrar as batidas de ponto (entrada e saída) e armazenar os dados no SharePoint.

**Requisitos**: Acesso ao PowerApps e ao SharePoint para criação da lista.

**Passo a Passo**:

1. Criar uma aplicação no PowerApps para controle de batidas de ponto.
2. Configurar a aplicação para:

- Registrar data e hora de entrada e saída.

- Salvar os dados na lista do SharePoint:

- Estrutura sugerida para a lista: Colunas data (data), hora\_entrada (data/hora), hora\_saida (data/hora).

1. Configurar uma mensagem automática no Teams:

- Após o registro de saída, enviar uma mensagem para o próprio usuário no Teams informando que a batida foi registrada com sucesso.

**Trigger de Power Automate a partir do Teams**

**Descrição**: Utilizar o Power Automate para responder automaticamente a uma mensagem enviada no canal de testes do Teams.

**Requisitos**: Acesso ao Power Automate e ao canal de teste no Teams.

**Passo a Passo**:

1. Criar um fluxo no Power Automate que seja disparado por uma mensagem específica enviada ao canal de testes do Teams.

- Definir a mensagem do exercício anterior como o gatilho do fluxo.

1. Configurar o fluxo para responder automaticamente à mensagem original no Teams com uma resposta personalizada.
2. Testar o fluxo enviando uma mensagem de teste e verificando se a resposta automática é disparada corretamente.

**Criação de API com FastAPI para Consultar Logs de Erro no MongoDB**

**Descrição**: Criar uma API usando o FastAPI com um endpoint que consulta o MongoDB para retornar informações de todas as collections filtradas por logs de erro da última semana. A conexão com o banco de dados deve utilizar a implementação do CommonKit.

**Requisitos**:

* Conhecimento básico em FastAPI e MongoDB.
* Projeto configurado com a conexão ao banco via CommonKit.

**Passo a Passo**:

1. **Configuração do Projeto**:

- Configurar uma aplicação FastAPI.

- Garantir que o projeto tenha acesso ao CommonKit para conectar ao MongoDB.

1. **Implementação do Endpoint**:

- Criar um endpoint /logs/erro que consulte todas as collections no MongoDB.

- O filtro de data deve ser ajustado para retornar apenas logs de erro da última semana.

- O parâmetro de filtro (data e tipo de log) deve ser passado como parâmetro do endpoint.

1. **Filtragem de Logs**:

- Utilizar o filtro de logs de erro e filtrar as entradas pela última semana.

- Se necessário, permitir flexibilidade para customizar o intervalo de datas.

1. **Bibliotecas de Qualidade de Código**:

- Configurar o projeto para ser validado pelas seguintes bibliotecas de qualidade:

- **Black** para formatação de código.

- **Mypy** para verificação de tipagem estática.

- **Pylint** para análise estática de código.

- **Flake8** para checagem de conformidade com PEP8.

- **Bandit** para análise de segurança.

- **Pydocstyle** para garantir a padronização da documentação.

1. **Testes de Qualidade**:

- Rodar as bibliotecas de qualidade de código após a implementação do endpoint.

- Corrigir quaisquer alertas ou erros apontados por cada uma das ferramentas configuradas.

**Exemplo de Request**:

* **Endpoint**: /logs/erro
* **Parâmetros**:

- start\_date: data inicial do filtro (por padrão, uma semana atrás).

- end\_date: data final do filtro (por padrão, data atual).

- log\_type: tipo de log para filtrar (por padrão, erro).

**Exemplo de Uso**:

Realizar uma requisição GET para o endpoint /logs/erro com os parâmetros de data e log\_type para obter logs de erro da última semana.