

# **Apresentando análises e resultados**

# Comunicar Dados de Forma Eficaz

Criar o dashboard é apenas metade do trabalho. O verdadeiro sucesso vem quando os gestores **entendem** os dados, **confiam** nos seus achados e **tomam ações** baseadas neles.

Para isso, seu dashboard precisa ser mais do que uma coleção de gráficos. Ele precisa:

1. **Ser Intuitivo:** Seguir princípios de visualização eficaz.
2. **Contar uma História:** Usar o storytelling para guiar a interpretação.
3. **Ter uma Estrutura Lógica:** Organizar a informação do mais geral para o mais específico.

Vamos ver como aplicar isso no seu app Streamlit.

---

# Storytelling com Dados: Guiando a Decisão

Seu dashboard não deve ser um quebra-cabeça. Ele deve contar uma história clara que leve a uma conclusão.

**A Situação (O Contexto):** Comece com a visão geral. Qual é o principal objetivo de negócio?

**O Problema ou a Oportunidade (O Conflito):** Aponte para a principal descoberta. Onde algo inesperado aconteceu?

**A Análise Profunda (O Clímax):** Permita que o gestor explore o "porquê".

**A Resolução (A Chamada para Ação):** Sugira os próximos passos com base nos dados.

.....

# Apresentando o Streamlit

**Streamlit é um framework open-source que transforma scripts de dados em aplicações web compartilháveis em minutos.**

Sua filosofia é simples:

- **Fácil de aprender:** Se você sabe escrever um script em Python, já sabe usar o Streamlit.
  - **Rápido de desenvolver:** Crie um app com poucas linhas de código. O fluxo é "script -> app".
  - **Totalmente em Python:** Não é necessário saber HTML, CSS ou JavaScript.
  - **Resultados instantâneos:** Cada vez que você salva o script, o app atualiza automaticamente.
-

# Anatomia de um App Streamlit

Um app Streamlit é, essencialmente, um script Python que é executado de cima para baixo.

Os comandos do Streamlit são fáceis de reconhecer: `st.*`

- `st.title("Meu Dashboard")`: Adiciona um título.
- `st.write("Qualquer texto, tabela ou gráfico")`: O comando "mágico" que exibe quase tudo.
- `st.sidebar`: Organiza widgets em uma barra lateral.
- `st.slider("Escolha um valor")`: Cria um controle deslizante interativo.
- `st.button("Clique aqui")`: Cria um botão.
- `st.plotly_chart(fig)`: Exibe um gráfico interativo do Plotly.

O Streamlit cuida de toda a complexidade de criar a interface web, permitindo que você se concentre na lógica e nos dados.

---

# Exemplo Prático no Colab

# Como compartilhar seu dashboard?

Um dashboard só é útil se puder ser compartilhado. O Streamlit torna isso fácil.

## 1. **Streamlit Community Cloud (Grátis):**

- Hospede seu código em um repositório público no GitHub.
- Conecte sua conta GitHub ao Streamlit Community Cloud.
- Faça o "deploy" com alguns cliques.
- Seu dashboard estará online com um link compartilhável (ex: `seu-nome-seu-app.streamlit.app`).

## 2. **Outras Opções:**

- Hospedagem em servidores próprios (AWS, Google Cloud, Azure).
  - Uso de contêineres (Docker) para facilitar a implantação.
-