

---

# Laboratório 15: APIs e Prompts Básicos na Prática

---

## Objetivo do Laboratório

Neste laboratório, vamos nos conectar pela primeira vez a um modelo avançado na nuvem. O objetivo é configurar um ambiente Python que consome a API oficial do Google Generative AI (Gemini) ou OpenAI, enviar uma pergunta técnica e recuperar a resposta com segurança (sem vaziar credenciais).

## Atividade Prática

### 1. Configuração do Ambiente e Chaves

- 1 Obtenha uma API Key válida num provedor moderno (ex: Google AI Studio com o Gemini, que é gratuita para uso moderado, ou OpenAI).
- 2 No diretório do laboratório, crie o arquivo `.env` contendo:

```
API_KEY=sua-chave-aqui-sem-aspas
```

- 3 Verifique que o `.env` está listado no seu arquivo `.gitignore`.
- 4 Instale a biblioteca do provedor e a de leitura do ambiente:

```
pip install google-generativeai python-dotenv
```

### 2. O Código: Conectando à API

Crie o arquivo `consumo_api.py`:

```
import os
import google.generativeai as genai
from dotenv import load_dotenv
```

---

```
# 1. Carregando variáveis sensíveis do arquivo .env
load_dotenv()
API_KEY = os.getenv("API_KEY")

# 2. Configurando o SDK do Google
genai.configure(api_key=API_KEY)

# 3. Instanciando o Modelo
modelo = genai.GenerativeModel('gemini-1.5-flash')

# 4. A Requisição
prompt = "Me explique em um único parágrafo o que é REST API e como ela se comunica."
print(f"Enviando o prompt: '{prompt}'...")

# 5. Resposta
resposta = modelo.generate_content(prompt)
print("\n--- Resposta da IA ---")
print(resposta.text)
```

### 3. Entrega via Git

- 1 Execute o arquivo usando `python consumo_api.py` e verifique se a resposta textual condiz com uma inteligência de ponta.
- 2 Certifique-se de que o arquivo `.env` **NÃO** apareça ao rodar `git status`.
- 3 Adicione as mudanças no rastreamento: `git add consumo_api.py`.
- 4 Realize o commit no repositório local: `git commit -m "Cria script básico de consumo de LLM na nuvem usando dotenv"`.
- 5 Envie para o servidor: `git push origin main`.

### Critério de Avaliação

O código `consumo_api.py` deve estar no repositório Git sem nenhuma API Key *hardcoded* (`.env` devidamente ignorado). A sua submissão demonstrará a capacidade de abstração do *cloud computing* para inferência em IA na sua máquina local.