

// laboratório codex 02 · princípios de gestão

python + ia

>_ Laboratório Codex 02

Ferramentas de gestão em Python, com IA

SWOT, plano de ação 5W2H, organograma, liderança, indicadores e stakeholders: as quatro funções do gestor (planejar, organizar, dirigir e controlar) em programas que você cria com IA.

// apostila do aluno II · trilha de exercícios

01 Análise SWOT



03 Organograma



05 Indicadores e semáforo



07 Os 7S da McKinsey



02 Plano de ação 5W2H



04 Liderança (Likert)



06 Stakeholders



08 Relatório do gestor



Prof. Caio Haffner

Mestre em Desenvolvimento Econômico · Unicamp

1 Como usar esta apostila

LEIA TUDO, NA ORDEM

Leia esta apostila **desde a primeira palavra** e faça tudo o que ela manda, na ordem. Não pule direto para os exercícios: cada página prepara a seguinte. Se ainda não fez a **Apostila I** (inclusive a página "PARE. Leia antes de tudo"), comece por ela.

Esta é a **continuação da Apostila I** (Laboratório Codex 01). Lá você instalou as ferramentas e aprendeu a conversar com o Codex. Aqui você vai **praticar**: cada exercício é uma **ferramenta do dia a dia do gestor**, na ordem do processo de gestão: planejar, organizar, dirigir e controlar. Quase todos usam o Café Bom Dia, uma empresa fictícia.

1. Revisar

Checklist e lembretes da Apostila I

2. Planejar

SWOT e plano 5W2H

3. Organizar e dirigir

Organograma e liderança

4. Controlar

Indicadores e stakeholders

5. Integrar

O relatório do gestor

Os sinais desta apostila

Quando aparecer	O que significa
Cartão roxo · PROMPT	Texto para copiar e colar no Codex. Leia antes de enviar.
Quadrado tracejado · CHECKPOINT	O que você deve estar vendo. Se não estiver, pare e peça ajuda.
Caixa azul · DICA	Atalho, pergunta para pensar ou jeito mais fácil de fazer.
Caixa laranja · ATENÇÃO	Ponto em que muita gente erra (e a IA também).

Ao final, você deverá conseguir

Habilidade	Como saber que consegui	Ok
Fazer uma SWOT	Organizo os quatro quadrantes e indico a postura da empresa.	☐
Montar um plano de ação	Cadastro ações no 5W2H, com prazos, responsáveis e custos.	☐
Ler uma estrutura	Desenho o organograma e calculo a margem de controle.	☐
Reconhecer estilos de liderança	Classifico a gestão nos 4 sistemas de Likert.	☐
Controlar resultados	Calculo o atingimento das metas e uso o semáforo.	☐
Pensar em sistema	Classifico os stakeholders por poder e interesse.	☐

1

REGRA DE OURO

Você é responsável pelo código que entrega, mesmo quando a IA o escreveu. **Não entregue código que você não consegue explicar.** E aqui vale mais uma: **a ferramenta não decide pelo gestor**; ela organiza a informação para você decidir.

2

Antes de começar

A instalação está na **Apostila I** (Laboratório Codex 01, seções 3 a 7). Se alguma coisa abaixo não estiver pronta, volte lá antes de seguir.

O que precisa estar pronto	Onde está na Apostila I	Ok
VS Code instalado e abrindo	Seção 4	☐
Python funcionando no terminal (<code>python --version</code> ou <code>py --version</code>)	Seção 5	☐
Extensão do Codex instalada e com login feito	Seção 6	☐
Você sabe criar uma pasta de projeto e abrir no VS Code	Seção 7	☐

Crie a pasta desta apostila

- 1 Na pasta Documentos, crie uma pasta chamada `Laboratorio_Codex_PG`.
- 2 No VS Code, use **Arquivo** → **Abrir Pasta** e escolha essa pasta.

- 3 Abra o painel do Codex e confira que ele está conectado à sua conta.



CHECKPOINT

O Explorer mostra a pasta **Laboratorio_Codex_PG** e o Codex responde a um "olá".



NÃO TEM A APOSTILA I?

Peça ao professor o PDF do **Laboratório Codex 01 · Apostila do aluno**. As seções 3 a 7 levam cerca de 30 minutos.

3

Lembretes da Apostila I



O CICLO DE TRABALHO COM A IA

Pedir um **plano** → ler → aprovar → **executar** → **testar** com contas feitas por você → **explicar** o código com suas palavras.



MÉTODO DE DEPURACÃO

Diagnóstico → explicação → solução → alteração → teste. **Peça o diagnóstico antes da correção.**



A VÍRGULA BRASILEIRA

No Python, o decimal é com **ponto** (**13.75**). Em todos os programas desta apostila, peça ao Codex para **aceitar vírgula ou ponto**. E digite os milhares **sem ponto** (**1000** , e não **1.000**): um programa que aceita vírgula ou ponto pode ler 1.000 como 1.



SEGURANÇA

Só aprove o que você entende. Trabalhe só na pasta do projeto. Nunca coloque senhas ou chaves no código.

PROMPT UNIVERSAL DE DIAGNÓSTICO

serve para qualquer erro

Estou com este problema no meu projeto.

Erro/mensagem: [COLE A MENSAGEM]

Arquivo envolvido: [NOME DO ARQUIVO]

Antes de alterar qualquer coisa:

1. explique a causa provável;
2. diga como confirmar a causa;
3. proponha a menor correção;
4. aguarde minha autorização para alterar o arquivo.

4

Exercício 1 · Planejar: a análise SWOT

Antes de planejar, o gestor olha para dentro (**forças e fraquezas**) e para fora (**oportunidades e ameaças**). Dando uma nota de importância a cada item, a SWOT indica a postura da empresa.



saldo interno = Σ forças - Σ fraquezas · **saldo externo = Σ oportunidades - Σ ameaças**

Interno + e externo + → desenvolvimento (ofensiva) · interno + e externo - → manutenção (confronto)

interno - e externo + → crescimento (reforço) · interno - e externo - → sobrevivência (defesa)

PROMPT 1

Crie um programa em Python chamado swot.py.

O usuário informa os itens de cada quadrante da SWOT (forças, fraquezas, oportunidades

e ameaças), cada um com uma nota de importância de 1 a 5. O programa deve:

1. mostrar a matriz SWOT organizada em 4 quadrantes;
 2. calcular o saldo interno e o saldo externo;
 3. indicar a postura estratégica (desenvolvimento, manutenção, crescimento ou sobrevivência);
 4. apontar o item de maior nota em cada quadrante.
- Antes de criar o arquivo, apresente um plano curto.



OS DADOS DO CAFÉ BOM DIA (EMPRESA FICTÍCIA)

Forças: café de qualidade (5), atendimento simpático (4). **Fraquezas:** caixa desorganizado (4), sem presença digital (3). **Oportunidades:** escritórios novos no bairro (5), delivery por aplicativo (4). **Ameaças:** franquia de café abrindo na esquina (4), alta do preço do café (3).

Cenário	Saldo interno	Saldo externo	Postura
Dados acima			
As duas ameaças com nota 5			



PARA PENSAR

A SWOT mostra uma foto. O que muda no planejamento quando a foto muda (segundo cenário)?

5

Exercício 2 · Planejar: o plano de ação 5W2H

O planejamento só sai do papel com um **plano de ação**: o que será feito, por quê, onde, quando, por quem, como e quanto custa.



What · Why · Where · When · Who · How · How much

O que · Por quê · Onde · Quando · Quem · Como · Quanto custa

PROMPT 2

Crie um programa em Python chamado `plano_5w2h.py`.

O usuário cadastra ações com os 7 campos do 5W2H (a data no formato dd/mm/aaaa e o custo em reais). O programa deve:

1. salvar as ações em `acoes.csv` e carregá-las de novo ao abrir;
2. mostrar o custo total do plano e o custo por responsável (Who);
3. listar as ações por prazo, do mais próximo ao mais distante;
4. marcar como ATRASADA a ação com prazo anterior à data de hoje.

Antes de escrever o código, apresente um plano curto.



O PLANO DO CAFÉ BOM DIA

1) Criar o perfil no Instagram · Ana · 15/11/2026 · R\$ 0. 2) Organizar o fluxo de caixa em planilha · Bruno · 30/11/2026 · R\$ 0. 3) Treinar a equipe em atendimento · Ana · 10/12/2026 · R\$ 600. 4) Cadastrar o café num aplicativo de delivery · Carla · 05/12/2026 · R\$ 250.
Complete o Why, o Where e o How de cada ação.

Resultado	Você espera	O programa mostrou
Custo total do plano		
Custo das ações da Ana		
Ação com o prazo mais próximo		
Ações atrasadas hoje		

6

Exercício 3 · Organizar: organograma e margem de controle

Depois de planejar, o gestor distribui o trabalho. O **organograma** mostra quem responde a quem, e a **margem de controle** mostra quantas pessoas cada gestor lidera.



margem de controle = número de subordinados diretos de um gestor

Margem muito pequena (1) pode indicar um nível desnecessário; muito grande (mais de 7), sobrecarga

PROMPT 3

Crie um programa em Python chamado `organograma.py`.

O usuário informa as pessoas no formato nome; cargo; chefe (o dono não tem chefe).

O programa deve:

1. desenhar o organograma no terminal, com recuos por nível;
2. contar os níveis hierárquicos;
3. mostrar a margem de controle de cada gestor e a média;
4. emitir um alerta para gestores com 1 subordinado ou com mais de 7.

Antes de escrever o código, apresente um plano curto.



A EQUIPE DO CAFÉ BOM DIA

Marcos (dono) · Ana (gerente; chefe: Marcos) · Bruno (financeiro; chefe: Marcos) · Carla (barista-chefe; chefe: Ana) · Diego e Elisa (atendentes; chefe: Ana) · Fábio, Gabi e Hugo (baristas; chefe: Carla) · Iara (auxiliar administrativa; chefe: Bruno).

Resultado	Você espera	O programa mostrou
Número de níveis		
Maior margem de controle (quem?)		
Margem média dos gestores		
Alertas		



PARA PENSAR

O alerta do Bruno é um problema de verdade? Que mudança na estrutura você sugeriria?

7

Exercício 4 · Dirigir: o perfil de liderança (Likert)

Rensis Likert classificou a gestão em **4 sistemas**, do autoritário coercitivo ao participativo, olhando para decisões, comunicação, relações e recompensas.



média das 8 respostas (1 a 4) · abaixo de 1,75 → Sistema 1
· até 2,49 → Sistema 2 · até 3,24 → Sistema 3 · 3,25 ou mais
→ Sistema 4

Sistema 1: autoritário coercitivo · 2: autoritário benevolente · 3: consultivo · 4: participativo



AS 8 AFIRMAÇÕES (RESPOSTA DE 1 = NUNCA A 4 = SEMPRE)

1) A equipe participa das decisões. 2) As metas são discutidas com quem vai cumpri-las. 3) A informação circula de baixo para cima. 4) O gestor explica o porquê das decisões. 5) Há confiança entre gestor e equipe. 6) As pessoas trabalham em grupo, não isoladas. 7) Reconhecimento é mais usado que punição. 8) As recompensas dependem de metas combinadas.

PROMPT 4

Crie um programa em Python chamado `lideranca.py`.

O programa faz as 8 afirmações acima, uma de cada vez, e aceita respostas de 1 a 4 (repete a pergunta se a resposta for inválida). Depois deve:

1. calcular a média geral e a média de cada variável de Likert (decisões: 1 e 2 · comunicação: 3 e 4 · relações: 5 e 6 · recompensas: 7 e 8);
2. classificar o sistema de gestão (1 a 4) pela média geral;
3. apontar a variável mais fraca e sugerir uma atitude para melhorá-la.

Antes de escrever o código, apresente um plano curto.

Respostas	Média	Sistema
3 · 3 · 4 · 3 · 2 · 3 · 3 · 4		
1 · 2 · 1 · 1 · 2 · 1 · 1 · 2		
4 em todas		

8

Exercício 5 · Controlar: indicadores e semáforo

O controle compara o **realizado** com a **meta**. Cuidado: em alguns indicadores, quanto menor, melhor (reclamações, desperdício).



maior é melhor: $\text{atingimento} = \text{realizado} \div \text{meta}$ · menor é melhor: $\text{atingimento} = \text{meta} \div \text{realizado}$

Semáforo: verde a partir de 100% · amarelo de 90% a 99,9% · vermelho abaixo de 90%

PROMPT 5

Crie um programa em Python chamado indicadores.py.

O usuário informa cada indicador com nome, meta, realizado e se "maior é melhor" ou "menor é melhor". O programa deve calcular o atingimento (%) de cada um, mostrar o semáforo (verde, amarelo ou vermelho), contar quantos estão em cada cor e listar primeiro os vermelhos, para o gestor agir. Antes de escrever o código, apresente um plano curto.

Indicador (Café Bom Dia, outubro)	Meta	Realizado	Atingimento	Semáforo
Vendas (R\$) · maior é melhor	60.000	54.000		
Clientes por dia · maior é melhor	200	210		
Reclamações no mês · menor é melhor	10	14		
Desperdício (kg) · menor é melhor	30	27		
Ticket médio (R\$) · maior é melhor	25,00	23,50		



CONFIRA A IA

Se o programa disser que 14 reclamações (meta 10) atingiram 140%, ele tratou o indicador como "maior é melhor". Corrija o prompt.

9

Exercício 6 · Pensamento sistêmico: o mapa dos stakeholders

A empresa é um sistema ligado a clientes, funcionários, fornecedores, governo e comunidade. A matriz **poder x interesse** mostra como tratar cada parte interessada.



poder e interesse de 1 a 5 (alto = 3 ou mais)

Alto poder e alto interesse → gerenciar de perto · alto poder e baixo interesse → manter satisfeito

baixo poder e alto interesse → manter informado · baixo poder e baixo interesse → monitorar

PROMPT 6

Crie um programa em Python chamado stakeholders.py.

O usuário informa cada stakeholder com o nome, o poder (1 a 5) e o interesse (1 a 5). O programa deve classificar cada um no quadrante da matriz poder x interesse, mostrar a lista agrupada por quadrante e sugerir uma ação de comunicação para cada grupo.

Antes de escrever o código, apresente um plano curto.

Stakeholder do Café Bom Dia	Poder	Interesse	Quadrante
Clientes	4	5	
Funcionários	3	5	
Fornecedor de café	4	2	
Prefeitura	5	1	
Vizinhos do bairro	2	4	
Concorrente	2	2	

10 Desafio extra · Os 7S da McKinsey

Para a McKinsey, a organização funciona bem quando sete elementos estão **alinhados**: estratégia, estrutura, sistemas, habilidades, pessoas, estilo e valores compartilhados.

PROMPT 7

desafio

Crie um programa em Python chamado `sete_s.py`.

O usuário dá uma nota de 1 a 5 para cada um dos 7S. O programa deve calcular a média, apontar o S mais forte e o mais fraco, calcular a amplitude (maior nota - menor nota) e avisar que a organização está desalinhada se a amplitude for 2 ou mais.

Teste com: estratégia 4, estrutura 3, sistemas 2, habilidades 4, pessoas 3, estilo 4, valores compartilhados 5. Explique o modelo dos 7S antes do código.

Resultado

Resposta

Média

S mais forte e S mais fraco

Amplitude e diagnóstico

11 Use o Codex como professor

Escolha um dos seus programas e use os três prompts abaixo. Anote no diário de prompts o que aprendeu.

EXPLICAR

Explique este código linha por linha para um estudante de Princípios de Gestão que conhece o conteúdo, mas nunca programou em Python. Não reescreva o código.

REVISAR O CONTEÚDO

Revise meu programa como um professor de Princípios de Gestão. Não faça alterações. Confira se os conceitos e as contas estão corretos (as regras de cada ferramenta (quadrantes, limites do semáforo, sentido dos indicadores) e os dados do Café Bom Dia) e liste os riscos de entrada inválida.

TESTAR

Crie uma tabela de casos de teste para este programa, com o resultado esperado de cada caso: um caso normal, um caso-limite, valores com centavos ou decimais e uma entrada inválida.

**CONFIRA A TABELA DE TESTES**

Os "resultados esperados" que a IA escreve também podem estar errados. Confira pelo menos dois na calculadora.

Atividade final · O relatório do gestor

Junte as quatro funções do gestor num programa só: a partir dos **indicadores** e do **plano de ação**, ele mostra o semáforo, aponta onde agir e confere se cada problema já tem uma ação no plano.

Requisitos

- ☐ Ler indicadores.csv (nome, meta, realizado, sentido) e acoes.csv (o 5W2H)
- ☐ Calcular atingimento e semáforo de cada indicador
- ☐ Para cada indicador vermelho ou amarelo, procurar no plano uma ação relacionada
- ☐ Sugerir uma nova ação 5W2H quando não houver nenhuma
- ☐ Mostrar o custo total do plano e as ações atrasadas
- ☐ Escrever um parágrafo de conclusão para o dono, com os números



O CICLO COMPLETO

É o processo de gestão inteiro: o **controle** mostra o desvio, o **planejamento** cria a ação, a **organização** define o responsável e a **direção** acompanha a equipe.

PROMPT A

planejamento

Quero criar em Python um "relatório do gestor" para uma pequena empresa. Ele lê indicadores.csv e acoes.csv, calcula o semáforo dos indicadores, verifica se cada indicador vermelho ou amarelo tem uma ação no plano 5W2H e sugere ações novas. Antes de escrever código, proponha as funções (reaproveitando os exercícios 2 e 5) e os testes com os dados do Café Bom Dia. Não escreva o código ainda.

PROMPT B

implementação

Agora implemente em relatorio_gestor.py e termine com a conclusão escrita pelo programa, com os números.

Com os dados do Café Bom Dia

Resultado

Indicadores verdes / amarelos / vermelhos

Indicador vermelho sem ação no plano

Ação sugerida pelo programa

Custo total do plano

13 Entrega e reflexão

Diário de prompts

#	O que pedi (resumo)	O que a IA fez	Precisei corrigir?
1			
2			
3			

O que entregar

Item	Entregue
Pasta do projeto com os arquivos .py dos exercícios e da atividade final	☐
As tabelas de teste desta apostila preenchidas	☐
Um erro de conteúdo que a IA cometeu (ou que você encontrou) e como foi corrigido	☐
O diário de prompts (acima) e a reflexão final (abaixo)	☐
Registro da auditoria humana (opcional, vale ponto extra)	☐

Reflexão individual

1. Qual ferramenta de gestão ficou mais clara depois que você a programou?

2. Em que momento a IA errou uma regra (por exemplo, o sentido de um indicador), e como você percebeu?

3. Que postura a SWOT indicou para o Café Bom Dia? Você concorda?

4. Que informação faltou ao relatório do gestor para decidir melhor?

14 Como você será avaliado

Critério	0	1	2	3
Conteúdo da disciplina	Conceitos errados	Acertou alguns	Acertou quase todos	Todos corretos e conferidos
Uso do Codex	Não conseguiu	Só copiou	Usou com alguma compreensão	Usou de forma crítica
Programação	Não funciona	Funciona em parte	Funciona	Funciona e está bem organizado
Testes	Não testou	Um teste	Vários testes	Testes e casos-limite
Depuração	Não identificou	Precisou da solução pronta	Identificou com ajuda	Diagnosticou e corrigiu
Explicação	Não explica	Explica pouco	Explica a lógica	Explica a lógica e o significado para o gestor

Total máximo: 18 pontos, convertidos para a escala da disciplina.

As contas desta apostila

SWOT

saldo interno e
externo

5W2H

7 perguntas por ação

Margem de
controle

subordinados diretos

Likert

média de 1 a 4

Atingimento

realizado/meta ou
meta/realizado

Stakeholders

poder × interesse

Comandos

TERMINAL

```
python arquivo.py    # executa um  
programa  
Ctrl + C             # interrompe um  
programa travado
```

Glossário

SWOT

Forças, fraquezas, oportunidades e ameaças.

5W2H

Plano de ação com 7 perguntas.

Organograma

Desenho da estrutura hierárquica.

Margem de controle

Número de subordinados diretos de um gestor.

Sistemas de Likert

Quatro estilos de gestão, do coercitivo ao participativo.

Indicador

Número que mede um resultado.

Stakeholder

Parte interessada na organização.

7S

Modelo da McKinsey com sete elementos que precisam estar alinhados.

1

LEMBRE-SE

A IA é uma ferramenta. Conhecimento, julgamento, validação e responsabilidade continuam sendo de quem programa, e de quem toma as decisões na organização.

Todo material produzido com IA precisa de auditoria humana. Sempre. A IA escreve rápido e com segurança, mas pode errar uma conta, inventar uma regra ou usar um dado desatualizado. E ela escreve o erro com a mesma confiança do acerto.

Esta apostila foi escrita com ajuda de IA e revisada pelo professor, conta por conta. Mesmo assim, algum erro pode ter escapado. Agora o auditor é você: encontre, comprove e relate.



REGRA DO LABORATÓRIO: IA SEMPRE PASSA POR AUDITORIA HUMANA

A IA não sabe quando errou: o texto errado sai tão convincente quanto o certo.

Quem assina o resultado é uma pessoa: você no trabalho, o analista no relatório, o professor na apostila.

Auditar não é desconfiar da ferramenta. É a etapa que transforma um rascunho da IA num material confiável.

Vale para tudo: código, contas, textos, gráficos e esta apostila.

O que procurar

Tipo de erro	Exemplo	Como conferir
Conta ou número	Um total que não fecha, uma porcentagem errada	Refaça na calculadora
Regra da disciplina	Fórmula trocada, conceito mal explicado	Compare com o livro ou o material da aula
Referência	"Veja a seção 9", quando o assunto está na 11	Vá até a seção indicada
Prompt	Um prompt que faz o Codex errar sempre do mesmo jeito	Rode o prompt duas vezes
Dado desatualizado	Selic, IPCA, regra de imposto	Confira no site oficial (Banco Central, IBGE, Receita Federal)
Texto	Frase confusa, erro de português	Leia o trecho em voz alta

Como auditar com o Codex

PROMPT DE AUDITORIA

use com cuidado

Vou colar um trecho da minha apostila de [DISCIPLINA].

Atue como revisor. Não reescreva o texto.

1. refaça cada conta e mostre o cálculo;
2. aponte afirmações que podem estar erradas ou desatualizadas;
3. para cada problema, diga como eu posso confirmar sozinho.

Trecho: [COLE AQUI]



A IA TAMBÉM ERRA NA AUDITORIA

O Codex pode apontar erros que não existem. Só relate o que você confirmou com uma conta, uma fonte oficial ou um teste.

Registro da auditoria

#	Seção e página	O que está escrito	O que deveria ser	Como confirmei
1				
2				
3				

Como enviar

- 1 Preencha a tabela acima.
- 2 Envie cada erro pelo formulário: **laboratorio-codex.pages.dev/auditoria**
- 3 Aguarde a confirmação do professor.



O QUE VOCÊ GANHA

Cada erro confirmado vale **+1 ponto extra** na avaliação (até 3 pontos), e o seu nome entra na **lista de revisores** da próxima versão da apostila. Se duas pessoas relatarem o mesmo erro, vale o primeiro envio.