

// laboratório codex 05 · finanças 4.0

python + ia + app

>_ Laboratório Codex 05

A carteira inteligente, com IA

Banco de dados, previsão, carteira de menor risco, painel de risco × retorno, alertas e relatório executivo: o simulador que ajuda a decidir.

// apostila do aluno V · o projeto em etapas

01 Banco de dados

02 Previsão do patrimônio

03 Carteira de menor risco

04 Risco × retorno

05 Alertas

06 Relatório executivo

07 Índice de Sharpe

08 Apresentação



Prof. Caio Haffner

Mestre em Desenvolvimento Econômico · Unicamp

1 Como usar esta apostila

Esta é a **Apostila V de Mercado Financeiro**, a etapa avançada da trilha: ela continua a **Apostila IV** (o Simulador de Carteira no ar (Apostila IV)). Agora você vai dar **inteligência** ao Simulador de Carteira: um banco de dados com o histórico, a previsão do patrimônio, a carteira de menor risco, um painel de risco × retorno, alertas automáticos e um relatório executivo para o investidor.

1. Guardar

Banco de dados

2. Prever

Patrimônio da carteira

3. Otimizar

Carteira de menor risco

4. Vigiar

Alertas

5. Decidir

Relatório executivo



O PROJETO: A CARTEIRA INTELIGENTE

Nas Apostilas III e IV, o simulador mostra quanto a carteira rendeu. Agora ele passa a ajudar a decidir: prevê o patrimônio, calcula a mistura de menor risco, avisa quando algo sai do padrão e escreve o relatório do mês.

Ao final, você deverá conseguir

Habilidade	Como saber que consegui	Ok
Usar um banco de dados	O histórico fica num banco SQLite e o app consulta com SQL.	☐
Prever	Comparo média móvel e tendência e meço o erro.	☐
Otimizar a carteira	Calculo a mistura de ativos de menor risco (Markowitz).	☐
Mostrar num painel	Os resultados aparecem num painel interativo.	☐
Criar alertas	O app avisa quando um número sai do padrão.	☐
Relatar	O app gera o relatório executivo do mês.	☐

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.** Aqui vale mais uma: **previsão, simulação e otimização são apoio à decisão.** Quem decide continua sendo uma pessoa, que entende os números.

2

Antes de começar



LEIA TUDO, NA ORDEM

Leia esta apostila **desde a primeira palavra** e faça tudo o que ela manda, na ordem. Cada etapa usa o que a anterior construiu.

A Apostila V começa onde a IV termina: **o Simulador de Carteira no ar, no GitHub, com testes e dados reais**. Se ainda não está assim, termine a IV antes.

Instale as bibliotecas novas (uma vez só)

TERMINAL

```
python -m pip install numpy scipy plotly fpdf2
```

Biblioteca	Para que serve
sqlite3	Banco de dados num arquivo (já vem com o Python).
numpy e scipy	Contas, simulação e otimização (a melhor decisão dentro dos limites).
plotly	Gráficos interativos dentro do app: zoom, filtro e valor ao passar o mouse.
fpdf2	Gerar o relatório executivo em PDF.

Ao longo das etapas, a pasta do projeto vai ganhar:

TERMINAL

```
Laboratorio_Codex_MF3/  
├─ dados/historico.db ← novo: banco de dados  
├─ calculos.py  
├─ previsao.py        ← novo  
├─ otimizacao.py      ← novo  
├─ painel.py          ← novo  
├─ alertas.py         ← novo  
├─ app.py  
└─ tests/
```



LEMBRETES DAS APOSTILAS ANTERIORES

Peça sempre um **plano** antes do código · teste com contas feitas por você · peça o **diagnóstico antes da correção** · nada vai para a internet sem passar nos testes · nunca coloque senhas no código.

3

Etapa 1 • Guardar: o banco de dados

Planilhas soltas não aguentam anos de histórico. Um **banco de dados** guarda tudo e responde perguntas em segundos.

PROMPT 1

Crie dados/historico.db (SQLite, que já vem com o Python) com as tabelas ativos (nome, classe, taxa), movimentos (data, ativo, valor) e cotacoes (data, ativo, preço), importando os dados das Apostilas III e IV. Crie banco.py com funções que usem SQL para:

1. o saldo por ativo no fim de cada mês;
2. a rentabilidade da carteira mês a mês;
3. a participação (%) de cada classe: renda fixa, ações e fundos.

Antes de criar, explique em 5 linhas o que são tabela, linha, coluna e consulta SQL.



CHECKPOINT

O saldo do último mês é igual ao que o simulador da Apostila IV mostra.



LEIA O SQL

Peça ao Codex: "explique esta consulta SQL linha por linha". SELECT escolhe as colunas, WHERE filtra as linhas, GROUP BY agrupa e ORDER BY ordena.

4 Etapa 2 · Prever: o que vem por aí?

Para decidir o próximo mês, é preciso uma **previsão**. Duas formas simples e honestas: média móvel e tendência.



média móvel de 3 períodos = média dos 3 últimos ·
tendência: valor = $a + b \times \text{período}$ (regressão)

Erro absoluto médio (EAM) = média de $|\text{real} - \text{previsto}|$ nos períodos em que dá para testar



PATRIMÔNIO DA CARTEIRA (R\$ MIL, 12 MESES)

100 · 101 · 103 · 102 · 105 · 107 · 106 · 109 · 111 · 110 · 113 · 115

PROMPT 2

Crie `previsao.py` com:

1. `media_movel(serie, n=3)` e a previsão do período 13;
2. `tendencia(serie)`: a e b da reta por mínimos quadrados e a previsão dos períodos 13 e 14;
3. o erro absoluto médio da média móvel nos períodos 4 a 12.

No app, mostre um gráfico com o histórico, as duas previsões e uma frase dizendo qual erra menos.

Resultado	Esperado	Calculado
Média móvel (período 13)	112,67	
Tendência: a e b	$a \approx 98,24 \cdot b \approx 1,32$	
Tendência (períodos 13 e 14)	115,42 e 116,75	
Erro absoluto médio da média móvel	2,59	



PARA PENSAR

O patrimônio cresce com os aportes e com o rendimento. Se a previsão for usada para prometer um valor ao cliente, que cuidado é preciso ter?

5

Etapa 3 · Otimizar: a carteira de menor risco

Misturar ativos que não andam juntos reduz o risco. É a ideia de **Markowitz**, que deu um Nobel de Economia.



DOIS ATIVOS

Ações: retorno esperado 12% ao ano, risco (σ) 20% · Renda fixa: retorno 8%, risco 10% · correlação entre eles: 0,2.

PROMPT 3

Crie `otimizacao.py` com:

1. `risco_carteira(peso_acoes)`: o risco (σ) da carteira com os dois ativos;
2. `menor_risco()`: o peso das ações que dá o menor risco, com `scipy.optimize.minimize_scalar`, conferido pela fórmula fechada de Markowitz;
3. a fronteira: retorno e risco para pesos de 0% a 100% de ações, de 5 em 5 pontos. Explique antes, em 5 linhas, por que a mistura pode ter risco menor que o do ativo mais seguro.

Resultado	Esperado	Calculado
Peso das ações na carteira de menor risco	$\approx 14,3\%$	
Risco (σ) dessa carteira	$\approx 9,56\%$	
Retorno esperado dessa carteira	$\approx 8,57\%$	



PARA PENSAR

A carteira de menor risco tem risco **menor que o da renda fixa sozinha** (10%). Por que isso acontece? E por que nem todo investidor escolhe essa carteira?

6

Etapa 4 · Mostrar: o painel de risco x retorno

Um gráfico mostra o que a tabela esconde: onde a carteira do cliente está e onde poderia estar.

PROMPT 4

Acrescente ao app a aba "Risco x retorno" com plotly: a fronteira da Etapa 3 como linha, a carteira de menor risco destacada e a carteira atual do cliente como um ponto vermelho. Ao passar o mouse, mostre o peso das ações, o risco e o retorno. Um controle deslizante muda o peso das ações.



CHECKPOINT

O gráfico abre no app, o ponto da carteira atual muda quando o peso muda e o ponto de menor risco fica em $\approx 14\%$ de ações.

Etapa 5 · Vigiar: alertas automáticos

Ninguém olha todos os números todos os dias. Os **alertas** mostram só o que pede atenção.

PROMPT 5

Crie alertas.py com regras que o usuário pode ajustar na barra lateral:

1. um ativo com mais de 40% da carteira (concentração);
2. perda maior que 5% num mês;
3. rendimento do mês abaixo da poupança;
4. patrimônio previsto abaixo da meta do cliente.

No topo do app, mostre os alertas em vermelho (crítico) e amarelo (atenção), do mais grave para o menos grave.

Alerta	Quantos apareceram	Faz sentido?
Concentração acima de 40%		
Perda acima de 5% no mês		
Rendimento abaixo da poupança		
Previsão abaixo da meta		



ALERTA DEMAIS É ALERTA NENHUM

Se aparecerem 30 alertas, ninguém olha mais. Ajuste as regras para mostrar só o que exige decisão.

Etapa 6 · Decidir: o relatório executivo

Quem decide não quer tabelas: quer saber o que aconteceu, o que vem aí e o que fazer.

PROMPT 6

Instale fpdf2 (python -m pip install fpdf2) e acrescente o botão "Relatório executivo (PDF)",
de uma página: o patrimônio e a rentabilidade do mês (contra o CDI e a poupança), a
previsão dos próximos 2 meses,
a carteira de menor risco comparada com a atual, os 3 alertas mais graves e 3
recomendações escritas pelo app,
cada uma com o número que a justifica. Atualize os testes e publique a nova versão
(como na Apostila IV).



CHECKPOINT

O PDF tem uma página, os números batem com o app e a nova versão está no ar pelo link da Apostila IV.

9

Desafio extra · Três ativos e o índice de Sharpe

Com três ativos, a fronteira vira uma nuvem de carteiras. A melhor relação entre retorno e risco é medida pelo **índice de Sharpe**.

PROMPT 7

desafio

Acrescente um terceiro ativo (fundo imobiliário: retorno 10%, risco 15%, correlação 0,3 com as ações e 0,1 com a renda fixa). Gere 5.000 carteiras com pesos aleatórios (numpy), desenhe a nuvem no plotly e destaque a de menor risco e a de maior Sharpe = $(\text{retorno} - \text{Selic}) \div \text{risco}$, usando a Selic real da Apostila IV. Explique o resultado em 3 linhas antes de mostrar o código.



CHECKPOINT

Os pesos de cada carteira somam 100%, nenhum é negativo e a de maior Sharpe tem retorno maior que a de menor risco.

10

Use o Codex como revisor do projeto

REVISAR A ORGANIZAÇÃO

Revise o meu projeto como um programador experiente revisaria o de um iniciante. Não faça alterações. Aponte: funções muito longas, código repetido, nomes pouco claros e cálculos misturados com a tela. Separe o que é obrigatório do que é opcional.

REVISAR O CONTEÚDO

Revise os cálculos do meu projeto como um professor de Mercado Financeiro. Não faça alterações. Confira o SQL, a previsão (e o erro medido), a fórmula do risco da carteira e as regras dos alertas e liste o que pode estar errado.



GRÁFICO BONITO NÃO É GRÁFICO CERTO

Confira pelo menos um número de cada gráfico com a calculadora ou com a Apostila II. A IA erra eixos, unidades e somas com frequência.

Entrega: apresente o seu projeto

Um projeto só existe quando alguém o vê funcionando. Prepare uma **apresentação de 3 minutos** e um registro para o seu portfólio.

Roteiro de 3 minutos

Tempo	O que mostrar
30 s	O problema que o app resolve e para quem.
1 min 30 s	O app funcionando: mexa nos controles e mostre a carteira de menor risco no gráfico de risco x retorno.
30 s	Um erro que a IA cometeu e como você descobriu.
30 s	O que você faria a seguir, se tivesse mais tempo.

O que entregar

- ☐ A pasta do projeto, com todos os arquivos .py e os dados
- ☐ O relatório em Excel gerado pelo app
- ☐ Três prints do app (telas principais)
- ☐ Um arquivo LEIAME.md explicando o que o app faz e como rodar (peça um rascunho ao Codex e revise)
- ☐ O diário de prompts: pelo menos 5 prompts importantes e o que aconteceu



PARA O SEU PORTFÓLIO

Um post no LinkedIn com um vídeo curto do app funcionando ("Construí em Python, com ajuda de IA, um app de investimentos com otimização e previsão") mostra a recrutadores algo que poucos estudantes têm: **uma ferramenta real feita por você.**

Reflexão individual

1. Qual etapa foi a mais difícil, e como você a resolveu?

2. Em que momento a IA errou, e como você percebeu?

3. O que você aprendeu de Mercado Financeiro usando previsão e otimização que não tinha aprendido nas apostilas anteriores?

12 Como você será avaliado

Critério	0	1	2	3
O app funciona	Não abre	Abre com erros	Funciona	Funciona e trata entradas inválidas
Conteúdo	Cálculos errados	Alguns certos	Quase todos certos	Todos certos e conferidos
Gráficos	Nenhum	Sem títulos ou eixos	Corretos	Corretos e fáceis de ler
Relatório	Não gera	Gera incompleto	Gera completo	Completo e organizado em abas
Uso do Codex	Não usou	Só copiou	Usou com compreensão	Usou de forma crítica
Apresentação	Não apresentou	Confusa	Clara	Clara, no tempo e com o erro da IA

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

13 Consulta rápida

Comandos

TERMINAL

```
python -m pip install numpy scipy
plotly fpdf2
python -m pytest                # roda
os testes
python -m streamlit run app.py  # abre
o app
git add . ; git commit -m "..."; git
push # publica
```

Streamlit

<code>st.title · st.header</code>	Títulos
<code>st.number_input · st.slider</code>	Entradas de número
<code>st.selectbox · st.file_uploader</code>	Escolha e upload
<code>st.metric · st.dataframe</code>	Indicador e tabela
<code>st.pyplot(fig)</code>	Mostrar um gráfico
<code>st.tabs · st.sidebar</code>	Abas e barra lateral
<code>st.download_button</code>	Botão para baixar o Excel

Glossário

Biblioteca

Código pronto que você instala para ganhar recursos (gráficos, telas).

pip

O instalador de bibliotecas do Python.

DataFrame

Tabela do pandas, com linhas e colunas.

App web

Programa que roda no navegador.

SQLite

Banco de dados num arquivo só, sem servidor.

SQL

Linguagem para consultar tabelas (SELECT, WHERE, GROUP BY).

Média móvel

Média dos últimos períodos, usada para prever o próximo.

Erro absoluto médio

Quanto a previsão erra, em média, para mais ou para menos.

Markowitz

Método que combina ativos para reduzir o risco da carteira.

Desvio padrão (σ)

Medida do risco: quanto o retorno costuma variar.

1

LEMBRE-SE

A IA é uma ferramenta. Conhecimento, julgamento, validação e responsabilidade continuam sendo de quem programa. Nada nesta apostila é recomendação profissional: é aprendizado.