Excelente escolha! Dominar **Python com pandas** é uma base sólida para depois avançar para PySpark e outras ferramentas de dados. Abaixo está um **leque completo de exercícios**, separados por níveis e focados em treinar tudo que você precisa para se tornar fluente com pandas.

**🟢 Nível 1 – Fundamentos (Python + pandas básico)**

**1. Leitura de dados**

* 📄 Leia um arquivo CSV (ex: alunos.csv) e exiba as 5 primeiras linhas.
* ✅ Verifique o tipo de dados de cada coluna (.dtypes).
* 🔍 Mostre o resumo estatístico dos dados (.describe()).

**2. Seleção de dados**

* 🔢 Selecione uma coluna específica.
* 🧩 Selecione várias colunas.
* 🔎 Filtre linhas com base em uma condição (ex: alunos com nota > 7).

**3. Criação e modificação de colunas**

* ➕ Crie uma nova coluna com base em cálculos (ex: nota\_final = (nota1 + nota2)/2).
* 🔄 Altere valores com .apply() ou .map().

**4. Ordenação e reset de índice**

* 📊 Ordene os dados pela nota final, do maior para o menor.
* 🔢 Resete o índice do DataFrame.

**🟡 Nível 2 – Intermediário (limpeza, agrupamento, estatísticas)**

**5. Tratamento de dados nulos**

* ❓ Conte quantos valores nulos existem por coluna.
* 🧼 Preencha nulos com média ou valor fixo.
* 🗑️ Remova linhas ou colunas com muitos valores nulos.

**6. Agrupamento de dados**

* 📚 Agrupe alunos por curso e tire a média da nota final.
* 📆 Agrupe vendas por mês e calcule o total.

**7. Conversões de tipo**

* ⏱️ Converta uma coluna de data de string para datetime.
* 🧠 Transforme valores categóricos em numéricos com .astype() ou LabelEncoder.

**8. Indexação e slicing avançado**

* 🎯 Use .loc[] e .iloc[] para selecionar intervalos de linhas e colunas.
* 📌 Filtre com múltiplas condições (&, |, ~).

**🔵 Nível 3 – Avançado (análise e transformação complexa)**

**9. Merge e Join de DataFrames**

* 🔗 Una dois DataFrames (ex: alunos + cursos) com merge().
* 🔄 Experimente tipos diferentes de join: inner, left, right, outer.

**10. Pivot e crosstab**

* 🔄 Crie uma tabela dinâmica com .pivot\_table().
* 🧮 Conte quantos alunos há por curso com pd.crosstab().

**11. Análise de séries temporais**

* 📆 Defina uma coluna como índice de data.
* 📉 Calcule média móvel (rolling().mean()) e resample para outra frequência.

**12. Exportação**

* 💾 Exporte os dados para .csv, .json ou .xlsx.

**🧪 Projeto prático final (nível consolidado)**

Crie um mini projeto com os seguintes passos:

1. **Tema:** "Análise de Desempenho de Estudantes"
2. Baixe ou crie um CSV com colunas como: nome, curso, nota1, nota2, frequência, status, data de matrícula.
3. Faça:
   * Cálculo da média
   * Classificação de aprovado/reprovado
   * Agrupamento por curso
   * Gráficos com matplotlib ou seaborn (opcional)
   * Exportação do relatório final para CSV

**📁 Arquivos de treino**

Se você quiser, posso gerar datasets sintéticos (ex: alunos.csv, vendas.csv, filmes.csv) para que você pratique direto com eles. Deseja que eu envie alguns exemplos prontos para você começar agora?