# PLANO SUGERIDO

### **🚀 Etapa 1: Fundamentos**

* **Python**: aprender lógica e bibliotecas básicas.
  + Começar com: pandas, numpy, matplotlib, seaborn
* **Excel** (se ainda não domina): filtros, tabelas dinâmicas, funções.
* **Power BI** ou **Tableau**: visualização de dados interativa.

### **🧪 Etapa 2: Prática com dados**

* Fazer projetos simples com bases reais (ex: dados do IBGE, Kaggle, etc.)
* Criar dashboards com Power BI ou gráficos com Python

### **🧠 Etapa 3: Estatística básica**

* Média, mediana, desvio padrão, correlação
* Distribuições e análise exploratória de dados

### **💼 Etapa 4: Portfólio e LinkedIn**

* Colocar projetos no GitHub
* Mostrar resultados visuais (imagens, links) no LinkedIn

### **Plano semanal sugerido (10–12h/semana) – 1º mês**

#### **Semana 1 – Fundamentos de Python**

* Curso: [Python Básico – Curso em Vídeo](https://www.youtube.com/playlist?list=PLHz_AreHm4dlKP6QQCekuIPky1CiwmdI6)
* Conteúdo:
  + Variáveis, tipos de dados, operadores, listas, dicionários
  + Condições (if, else), laços (for, while)
* Prática: criar scripts simples (ex: calculadora, média de notas)

#### **Semana 2 – Bibliotecas para dados: Pandas e NumPy**

* Curso: [Curso gratuito de Pandas – Hashtag Treinamentos](https://www.youtube.com/playlist?list=PLyqOvdQmGdTQ3mU3Bz2jTl5b0iQ40nT2N)
* Conteúdo:
  + Leitura de arquivos (.csv, .xlsx)
  + Filtragem, agrupamentos, limpeza
* Projeto: analisar um conjunto de dados simples (ex: vendas ou notas)

#### **Semana 3 – Visualização de dados com Matplotlib e Seaborn**

* Curso: [Visualização de Dados com Python – Data Bootcamp](https://www.youtube.com/watch?v=3Xc3CA655Y4)
* Conteúdo:
  + Criar gráficos: barras, linhas, pizza, dispersão
  + Customizar títulos, cores, eixos
* Projeto: criar um relatório gráfico com dados abertos

#### **Semana 4 – Power BI básico**

* Curso: [Power BI Completo – Hashtag Treinamentos](https://www.youtube.com/watch?v=5g0LNrAhs0Y)
* Conteúdo:
  + Importar dados, criar gráficos e dashboards
  + Filtros, segmentações e visualizações interativas
* Projeto: construir um dashboard com dados reais

### **🧩 Desafios e Projetos Reais**

* **Kaggle** ([kaggle.com](https://www.kaggle.com/)): plataforma com datasets e competições
  + Começar com datasets simples (vendas, clima, esportes)
* **Projetos sugeridos**:
  + Análise de dados do ENEM
  + Evolução do COVID no Brasil
  + Desempenho de vendas de uma loja fictícia
  + Análise de gastos públicos (Portal da Transparência)

### **🛠️ Ferramentas recomendadas**

* Google Colab: pra rodar código Python sem instalar nada
* [GitHub](https://github.com/): pra criar seu portfólio
* [Data Hackers](https://www.datahackers.com.br/): comunidade brasileira top de dados

### **Plano de estudos completo – 3 a 5 meses**

#### **1º Mês – Fundamentos de Python e Pandas**

**Semana 1: Introdução ao Python**

* **Segunda-feira a Quarta-feira**:
  + Vídeos do curso Python Básico (3–4h)
  + Exercícios: listas, dicionários, laços, condições
* **Quinta-feira a Sexta-feira**:
  + Criar pequenos programas (calculadora, análise simples de dados)
  + Revisão e prática adicional (2h)
* **Sábado e Domingo**:
  + Projetos práticos (ex: calculadora com várias funções, organizar lista de contatos)

**Semana 2: Pandas e NumPy**

* **Segunda-feira a Quarta-feira**:
  + Vídeos sobre Pandas e NumPy (3–4h)
  + Leitura de arquivos .csv, .xlsx
  + Limpeza de dados, agrupamentos e ordenação
* **Quinta-feira a Sexta-feira**:
  + Prática com datasets reais (como CSV de vendas, de esportes)
  + Exercícios de transformação e manipulação de dados
* **Sábado e Domingo**:
  + Projeto prático: Análise de um dataset simples (vendas, dados públicos)

**Semana 3: Visualização de Dados com Matplotlib e Seaborn**

* **Segunda-feira a Quarta-feira**:
  + Estudo de gráficos básicos (barras, pizza, dispersão) (3–4h)
  + Customização de gráficos (cores, eixos, títulos)
* **Quinta-feira a Sexta-feira**:
  + Prática com gráficos para visualizar resultados de datasets
* **Sábado e Domingo**:
  + Projeto prático: criar gráficos para mostrar insights de um dataset (ex: evolução do número de vendas)

**Semana 4: Power BI Básico**

* **Segunda-feira a Quarta-feira**:
  + Introdução ao Power BI (3–4h)
  + Importar dados, criar gráficos e dashboards
* **Quinta-feira a Sexta-feira**:
  + Exercícios práticos com Power BI (criação de painéis, filtros)
* **Sábado e Domingo**:
  + Projeto prático: construir um dashboard interativo com dados reais

#### **2º Mês – Análise Exploratória, Estatísticas e Introdução a Machine Learning**

**Semana 5: Análise Exploratória de Dados (EDA)**

* **Segunda-feira a Quarta-feira**:
  + Estudo de conceitos de EDA (correlação, valores ausentes, boxplot, histograma)
  + Prática com pandas para explorar dados (3–4h)
* **Quinta-feira a Sexta-feira**:
  + Identificação de outliers, distribuição de dados
* **Sábado e Domingo**:
  + Análise de um dataset real (ex: performance de vendas)

**Semana 6: Estatísticas Básicas**

* **Segunda-feira a Quarta-feira**:
  + Média, mediana, desvio padrão, distribuição normal (3–4h)
  + Visualização e interpretação desses dados
* **Quinta-feira a Sexta-feira**:
  + Correlação e causalidade
  + Prática com pandas, numpy e gráficos
* **Sábado e Domingo**:
  + Projeto: análise estatística de um conjunto de dados (ex: dados de saúde ou educação)

**Semana 7: Introdução a Machine Learning (ML)**

* **Segunda-feira a Quarta-feira**:
  + Conceitos básicos de Machine Learning (supervisionado vs não supervisionado)
  + Estudo sobre regressão linear e classificação
* **Quinta-feira a Sexta-feira**:
  + Implementação de um modelo simples de regressão com scikit-learn
* **Sábado e Domingo**:
  + Projeto: construção de um modelo simples para prever preços de casas, por exemplo

**Semana 8: Trabalhando com Modelos e Avaliação**

* **Segunda-feira a Quarta-feira**:
  + Métricas de avaliação de modelos (acurácia, precisão, recall, F1-score)
  + Validação cruzada e overfitting
* **Quinta-feira a Sexta-feira**:
  + Prática com modelos de classificação (ex: previsão de compra)
* **Sábado e Domingo**:
  + Projeto prático: melhorar um modelo já feito (ex: prever resultados de um campeonato de futebol)

#### **3º Mês – Prática com Projetos Reais e Ferramentas de Mercado**

**Semana 9: Projetos Reais e Portfólio**

* **Segunda-feira a Quarta-feira**:
  + Escolher um projeto real de seu interesse (ex: análise de dados públicos ou dados do mercado)
  + Construir um portfólio no GitHub
* **Quinta-feira a Sexta-feira**:
  + Refinamento do portfólio: incluir gráficos, relatórios e código bem comentado
* **Sábado e Domingo**:
  + Finalizar o projeto e preparar apresentação visual para LinkedIn

**Semana 10: Trabalhando com APIs e Automação**

* **Segunda-feira a Quarta-feira**:
  + Aprender a consumir dados de APIs (ex: dados financeiros ou de clima)
* **Quinta-feira a Sexta-feira**:
  + Projeto: extrair dados de uma API e fazer uma análise
* **Sábado e Domingo**:
  + Automatizar um processo de coleta e análise de dados

**Semana 11: Prática com Power BI Avançado**

* **Segunda-feira a Quarta-feira**:
  + Filtros avançados, integração de dados de várias fontes
  + Criação de relatórios dinâmicos
* **Quinta-feira a Sexta-feira**:
  + Refinamento de dashboards
* **Sábado e Domingo**:
  + Projeto: dashboard interativo e dinâmico com Power BI

**Semana 12: Revisão e Preparação para o Mercado de Trabalho**

* **Segunda-feira a Quarta-feira**:
  + Revisão dos conceitos mais importantes e projetos anteriores
* **Quinta-feira a Sexta-feira**:
  + Aperfeiçoamento de seu LinkedIn e perfil de GitHub
  + Aplique para vagas de estágio ou júnior
* **Sábado e Domingo**:
  + Preparação para entrevistas (respostas comuns, testes técnicos)

# DESAFIOS

Existem várias plataformas online que oferecem **desafios de Python**, desde o nível básico até o avançado. Essas plataformas são ótimas para praticar e melhorar suas habilidades de programação de forma gradual, e muitas delas têm **exercícios** e **problemas** organizados por **nível de dificuldade**. Aqui estão algumas das melhores opções:

### **1. HackerRank**

* **Descrição**: Uma das plataformas mais populares para aprender e praticar programação. Possui uma ampla gama de desafios em Python, organizados por **nível de dificuldade** (Fácil, Médio, Difícil).
* **Níveis**: Começa com desafios simples e vai até problemas mais complexos, abordando conceitos como estruturas de dados, algoritmos, e muito mais.
* **Link**: HackerRank - Python

### **2. Exercism**

* **Descrição**: Oferece exercícios de Python em diversos níveis de dificuldade. A plataforma também oferece **mentoria** gratuita de programadores mais experientes para revisar e discutir suas soluções.
* **Níveis**: Possui uma série de exercícios que começam no nível básico e vão se tornando progressivamente mais difíceis.
* **Link**: Exercism - Python

### **3. LeetCode**

* **Descrição**: Uma plataforma popular para praticar **algoritmos e estruturas de dados**. LeetCode é amplamente usada por pessoas que se preparam para entrevistas de emprego, mas também oferece desafios em Python em vários níveis.
* **Níveis**: Os desafios são classificados por dificuldade (Fácil, Médio, Difícil), o que permite um progresso gradual.
* **Link**: LeetCode - Python

### **4. Codewars**

* **Descrição**: Codewars oferece desafios de programação, chamados de **katas**, que são classificados por **nível de dificuldade**. Os desafios de Python começam do nível mais básico e progridem até questões mais complexas.
* **Níveis**: Começa no nível **8kyu** (mais fácil) e vai até **1kyu** (mais difícil).
* **Link**: [Codewars - Python](https://www.codewars.com/)

### **5. Project Euler**

* **Descrição**: Uma plataforma que foca em problemas matemáticos e de lógica que você pode resolver usando Python (ou outras linguagens). Ideal para quem quer trabalhar em questões desafiadoras que envolvem matemática e raciocínio lógico.
* **Níveis**: Os problemas variam de fáceis a muito difíceis, e você pode começar com os mais simples e avançar conforme seu progresso.
* **Link**: [Project Euler](https://projecteuler.net/)

### **6. Codeforces**

* **Descrição**: Codeforces é uma plataforma que oferece competições e desafios em tempo real, focada em **algoritmos e programação competitiva**. Você pode resolver problemas em Python e melhorar suas habilidades de codificação.
* **Níveis**: Embora seja voltado para programadores mais experientes, a plataforma tem uma boa variedade de problemas para iniciantes também.
* **Link**: [Codeforces](https://codeforces.com/)

### **7. Kaggle**

* **Descrição**: Principalmente focado em **data science e aprendizado de máquina**, Kaggle oferece notebooks e desafios que exigem o uso de Python para resolver problemas relacionados a **análise de dados**, **machine learning** e **inteligência artificial**.
* **Níveis**: A plataforma oferece **competências práticas** e **problemas de codificação** com vários níveis de complexidade.
* **Link**: [Kaggle](https://www.kaggle.com/)

### **8. Real Python**

* **Descrição**: Real Python oferece tutoriais, artigos e desafios que cobrem conceitos fundamentais de Python e avançam para tópicos mais complexos. Além disso, oferece cursos pagos com desafios integrados.
* **Níveis**: Os tutoriais e desafios cobrem todos os níveis, desde iniciantes até programadores avançados.
* **Link**: [Real Python](https://realpython.com/)

### **9. PyBites**

* **Descrição**: PyBites oferece uma plataforma para aprender Python com **exercícios práticos**. A ideia é melhorar suas habilidades por meio de **desafios pequenos** e práticos, cobrindo desde o básico até tópicos mais avançados.
* **Níveis**: Possui desafios para iniciantes e para programadores mais avançados.
* **Link**: [PyBites](https://codechalleng.es/)

### **10. Advent of Code**

* **Descrição**: Advent of Code é um desafio anual, mas os problemas ficam disponíveis o ano todo. São problemas de programação que começam com **dificuldade baixa** e aumentam conforme o calendário de dezembro avança.
* **Níveis**: Começa fácil, mas fica mais difícil à medida que você avança.
* **Link**: [Advent of Code](https://adventofcode.com/)

### **Dicas para aproveitar os desafios:**

1. **Comece devagar**: Se você é iniciante, comece com desafios fáceis e vá aumentando a dificuldade aos poucos.
2. **Pratique regularmente**: Tente resolver desafios diariamente ou semanalmente para fortalecer suas habilidades.
3. **Pesquise**: Se você ficar preso em um problema, não tenha medo de pesquisar soluções ou discutir com outros programadores. O objetivo é aprender, não apenas vencer o desafio.
4. **Reveja soluções**: Depois de resolver um desafio, procure outras soluções e compare para aprender novas abordagens.

Essas plataformas são ótimas para melhorar suas habilidades em Python, resolver problemas reais e se preparar para entrevistas de emprego em tecnologia. Se precisar de mais detalhes sobre alguma dessas opções ou sugestões de como começar, me avise!

# DATA SCIENCE EM 5 MESES

## **🎯 Objetivo final:**

Dominar os fundamentos de Data Science, ter um portfólio com pelo menos **2 projetos próprios** e estar pronto para aplicar para **vagas júnior** ou **freelas**.

## **📅 Mês 1: Fundamentos de Python e Matemática**

### **Habilidades:**

* Lógica de programação
* Sintaxe do Python
* Funções, loops, listas, dicionários
* Álgebra linear e estatística básica

### **Recursos sugeridos:**

* Curso: [Python para Data Science - DataCamp (grátis)](https://www.datacamp.com/)
* Exercícios: [Exercism.io](https://exercism.io/), [HackerRank](https://www.hackerrank.com/)
* Matemática: [Khan Academy - Álgebra e Estatística](https://pt.khanacademy.org/)

## **📅 Mês 2: Bibliotecas de Análise de Dados**

### **Habilidades:**

* Manipulação de dados com Pandas
* Visualização com Matplotlib e Seaborn
* Leitura de arquivos .csv, limpeza de dados

### **Recursos:**

* Curso: Pandas - Kaggle Micro-Courses
* Exercícios com datasets públicos (Kaggle)
* Mini-projetos: análise de dados do COVID, vendas, etc.

## **📅 Mês 3: Estatística aplicada e Exploração de Dados**

### **Habilidades:**

* Probabilidade, distribuições, testes de hipótese
* Análise exploratória de dados (EDA)
* Correlação, outliers, normalização

### **Recursos:**

* Curso: [StatQuest (YouTube)](https://www.youtube.com/user/joshstarmer)
* Projeto: Análise exploratória de um dataset do Kaggle

## **📅 Mês 4: Introdução ao Machine Learning**

### **Habilidades:**

* Modelos supervisionados: regressão linear, árvores, KNN
* Métricas: acurácia, precisão, recall, f1-score
* Scikit-learn

### **Recursos:**

* Curso: Machine Learning - Kaggle
* Mini-projetos com datasets do Kaggle (Titanic, Iris)

## **📅 Mês 5: Portfólio, SQL e preparação para o mercado**

### **Habilidades:**

* SQL para análise (SELECT, JOIN, GROUP BY)
* Construção de dashboards com Power BI / Tableau
* Soft skills e currículo para área de dados

### **Projeto final:**

* Escolha um problema real (ex: análise de dados de aluguel, esportes, saúde)
* Crie um notebook no GitHub bem documentado
* Crie um dashboard com os resultados no Power BI ou Tableau

## **📚 Extras semanais:**

* 1 desafio Python ou Kaggle por semana
* 1 artigo técnico ou vídeo sobre carreira em dados
* Revisar com anotações (use Notion, Obsidian ou caderno)