**Passo 1: Criando um Gerador de Senhas Seguras**

O objetivo aqui é gerar senhas aleatórias que sejam seguras e imprevisíveis. Para isso, usaremos conceitos fundamentais como **entropia, aleatoriedade e força de senha**.

**🔍 Conceitos Importantes**

**1 - O que faz uma senha ser segura?**

Uma senha segura precisa ser **longa, complexa e imprevisível**. Isso significa:  
✅ **Comprimento:** Quanto mais longa, melhor (mínimo recomendado de 12 caracteres).  
✅ **Variedade de caracteres:** Letras maiúsculas, minúsculas, números e símbolos.  
✅ **Ausência de padrões fáceis de prever:** Nada de "123456" ou "password".  
✅ **Unicidade:** Não usar senhas repetidas entre sites.

**2️ - Aleatoriedade e a Biblioteca random vs. secrets**

Para gerar senhas seguras, precisamos de números realmente imprevisíveis. O Python tem duas formas principais de gerar valores aleatórios:

🔸 **random (não recomendado para segurança):** Usa um algoritmo pseudoaleatório, que pode ser previsível se alguém descobrir a "semente" (seed).  
🔸 **secrets (mais seguro):** Projetado para criptografia e segurança, gera valores realmente imprevisíveis.

➡️ **Conclusão:** Usaremos a biblioteca secrets para evitar previsibilidade.

**3 - Entropia e Segurança das Senhas**

A **entropia** mede o nível de imprevisibilidade de uma senha. Quanto maior a entropia, mais difícil será quebrá-la.

A fórmula básica para calcular a entropia de uma senha é:

Entropia = log2(tamanho da senha)

- Uma senha de 8 caracteres apenas com letras minúsculas (26 opções) tem:

Log2(26^8) = 37.6 bits de entropia

- Já uma senha com 12 caracteres com maiúsculas, minúsculas, números e símbolos (94 opções) tem:

Log2(94^12) = 78.8 bits de entropia

🔹 **Quanto maior a entropia, mais difícil de ser descoberta por ataques de força bruta.**

**4️ - Ataques Contra Senhas**

Entender como senhas são quebradas nos ajuda a criar defesas melhores:

🔹 **Ataque de Força Bruta:** Testa todas as combinações possíveis. Quanto maior a senha, mais difícil.  
🔹 **Ataque de Dicionário:** Usa listas de senhas vazadas para tentar encontrar combinações comuns.  
🔹 **Ataque de Engenharia Social:** Pessoas usam informações previsíveis (data de nascimento, nomes, etc.).

➡️ **Conclusão:** Senhas devem ser **longas, aleatórias e únicas** para evitar esses ataques.