

Escape Room Python

PROTOCOLO GENESIS 2026 - TERMINAL DE ACESSO

Status do Sistema: CRÍTICO

Objetivo: Restaurar os 5 níveis de segurança.

INSTRUÇÕES:

1. Execute a célula abaixo para registrar sua dupla.
2. Use as células de cada nível para resolver os desafios do Google Forms.
3. A resposta que o seu código "printar" é a senha para o próximo nível.

```
# EXECUTE ESTA CÉLULA PRIMEIRO PARA FAZER LOGIN
print(">>> SISTEMA DE IDENTIFICAÇÃO INICIADO...")

# A função input() pausa o sistema e aguarda a digitação do aluno
AGENTE_1 = input("Digite o nome do Agente 1 (Piloto): ").strip().title()
AGENTE_2 = input("Digite o nome do Agente 2 (Co-piloto): ").strip().title()

print("-" * 50)
print(f"✅ IDENTIDADE CONFIRMADA.")
print(f"👤 Agente 1: {AGENTE_1}")
print(f"👤 Agente 2: {AGENTE_2}")
print("-" * 50)
print(f"🔒 Acesso concedido ao Terminal Genesis. Carregando desafios...")
```

NÍVEL 1: A Criptografia Matemática

A IA alterou o algoritmo de geração de senhas. Agora, a chave depende de cálculos matemáticos precisos nos índices das strings.

Missão:

O sistema exige um Token de Acesso formato por 3 partes unidas:

1. A letra do `nome_alvo` que está na posição raiz quadrada de 36.
2. A idade atual do alvo (cálculo simples).
3. A letra do `sobrenome_alvo` que está na posição 2 elevado a 3.

Atenção aos tipos de dados! Índices precisam ser números inteiros.

```
# MÓDULOS NECESSÁRIOS
import math # Dica: use math.sqrt() ou a potência ** 0.5

# DADOS DO ALVO (Não altere os dados)
nome_alvo = "Bartholomew"
sobrenome_alvo = "Frederick"
ano_nascimento = 1980
ano_atual = ???

# --- ESCREVA SEU CÓDIGO ABAIXO ---
# Gere e printe o Token.
# Lembrete: Se o cálculo der 6.0, o Python não aceita como índice. É preciso transformar em inteiro.
```

NÍVEL 2: O Firewall Numérico

O firewall está bloqueando conexões. A senha é uma sequência de resultados.

Missão: Para a lista de portas abaixo, seu código deve percorrer (loop) e verificar:

1. Se o número for PAR e MAIOR que 2000 → Printar "LIBERADO"

2. Se o número for ÍMPAR e DIVISÍVEL por 5 → Printar "QUARENTENA"

3. Qualquer outro caso → Printar "BLOQUEADO"

A senha no formulário será a sequência das 5 palavras, separadas por espaço.

```
# LISTA DE PORTAS (Não altere)
portas_acesso = [1002, 2024, 3055, 15, 888]

# --- ESCREVA SEU CÓDIGO ABAIXO ---
# Dica: Use um laço 'for' e condicionais 'if/elif/else'
```

NÍVEL 3: O Algoritmo de Fibonacci

A senha de acesso ao banco de dados está escondida em uma sequência matemática clássica.

A IA usa a Sequência de Fibonacci para criptografar os dados.

Missão:

1. Gere os primeiros 15 números da sequência de Fibonacci. (Lembrete: A sequência começa com 1, 1 e o próximo número é sempre a soma dos dois anteriores: 1, 1, 2, 3, 5...)

2. A senha é a **SOMA** de todos os números **PARES** contidos nesses 15 primeiros termos.

Senha: Um número inteiro único.

```
# CONFIGURAÇÃO DA SEQUÊNCIA
quantidade_termos = 15

# Dica de início: Você vai precisar de variáveis para o termo 'atual' e o 'anterior'
# e uma variável para acumular a soma dos pares.

# --- ESCREVA SEU CÓDIGO ABAIXO ---
```

NÍVEL 4: O Banco de Dados Sujo

Conseguimos recuperar uma lista de arquivos, mas ela está cheia de vírus.

Missão: Limpe a lista. Remova qualquer string que contenha a palavra "virus" (pode ser maiuscula ou minuscula).

Senha: Quantos arquivos sobraram no final? (A resposta é um número).

```
# DADOS CORROMPIDOS (Não altere)
sistemas_arquivos = [
    "sys_config.exe", "virus_trojan.py", "dados_clientes.csv",
    "VIRUS_WORM.bat", "backup_2025.tar", "virus_spy.dll",
    "foto_férias.png", "relatorio_final.docx", "Virus_x.vbs"
]

# --- ESCREVA SEU CÓDIGO ABAIXO ---
# Dica: Crie uma nova lista apenas com os arquivos limpos ou conte diretamente.
```

NÍVEL 5: A Matriz de Coordenadas (O Chefe Final)

Chegamos ao núcleo do sistema. A estrutura de dados aqui é bidimensional (uma Matriz).

Precisamos mapear os setores críticos e organizar isso em um Dicionário para o reinício do sistema.

Missão:

1. Abaixo existe uma matriz 3×3 chamada `setores_rede`.

2. Sua tarefa é calcular a soma da DIAGONAL PRINCIPAL (os números que cruzam do canto superior esquerdo ao inferior direito: índices `[0][0]`, `[1][1]`, `[2][2]`).

3. Com essa soma, crie um Dicionário exatamente assim:

```
'config = {"status": "ONLINE", "nivel_energia": valor_da_soma}'
```

4. Printe esse dicionário na tela.

Senha do Forms:* O valor numérico da chave 'nivel_energia' (a soma que você calculou).

```
# MATRIZ DO NÚCLEO (Lista de Listas)
setores_rede = [
    [1050, 200, 350], # Linha 0
    [400, 1500, 600], # Linha 1
    [700, 800, 2026] # Linha 2
]

# --- ESCREVA SEU CÓDIGO ABAIXO ---
# Passo 1: Acesse os índices específicos para somar a diagonal (1050 + 1500 + 2026).
# Passo 2: Crie o dicionário e printe.
```

🏆 NÍVEL 6: O Protocolo Final

AGENTE 1: {AGENTE_1}
AGENTE 2: {AGENTE_2}

O núcleo foi restaurado, mas para reiniciar o servidor sem apagar os dados, é necessária a *Chave Mestre Binária.

Missão Final:

1. A chave mestre é o número 5762 convertido para BINÁRIO.
2. Você deve calcular/converter esse valor (pode abrir uma nova célula de código para fazer a conta se quiser).
3. Insira o código binário (apenas os números 0 e 1) quando solicitado abaixo.

Dica: A função `bin()` do Python pode ajudar, mas lembre-se de limpar o prefixo '0b' se houver.

```
import time
import base64

# --- CONFIGURAÇÃO DE SEGURANÇA (OFUSCADA) ---
# A senha está codificada abaixo. Se você tentar ler, não vai entender.
# Isso simula um banco de dados de senhas reais.
SENHA_CRIPTOGRAFADA = b'MTAxMTAxMDAwMDAxMA=='

def verificar_acesso():
    print(f"🔒 {AGENTE_1.upper()} e {AGENTE_2.upper()}, insiram a CHAVE MESTRE.")

    # Input da senha
    tentativa = input("Digite o código binário de 5762: ").strip()

    # Tratamento: Remove '0b' caso o aluno copie direto do output do Python
    if tentativa.startswith("0b"):
        tentativa = tentativa[2:]

    # Decodificando a senha real para comparar (o aluno não vê isso acontecendo)
    senha_real = base64.b64decode(SENHA_CRIPTOGRAFADA).decode('utf-8')

    if tentativa == senha_real:
        animacao_vitoria()
    else:
        print("\n🚫 ERRO FATAL: Chave binária incorreta.")
        print("O sistema entrará em auto-destruição em breve se não corrigir...")

def animacao_vitoria():
```

```

print("\nAUTENTICANDO...")
time.sleep(1)
print("✓ HASH CONFIRMADO.")
time.sleep(0.5)

print("\nREINICIANDO SISTEMA...")
barrinha = "[          ]"
for i in range(1, 21):
    barrinha = barrinha.replace(" ", "█", 1)
    # O caractere \r faz o print sobreescriver a mesma linha (efeito de loading)
    print(f"\rCARREGANDO: {barrinha} {i*5}%", end="")
    time.sleep(0.15)

cod_final = 5**3;
print("\n\n")
print("██████████")
print("||      SISTEMA GENESIS : ONLINE      ||")
print("||██████████")
print(f"|| HERÓIS DO DIA: {AGENTE_1.ljust(26)} ||") # ljust alinha o texto
print(f"||           {AGENTE_2.ljust(26)} ||")
print(f"||      CÓDIGO FINAL DA MISSÃO É {cod_final}      ||")
print("||██████████")
print("||\o/  MISSÃO CUMPRIDA!  \o/      ||")

# Executa a verificação
try:
    verificar_acesso()
except NameError:
    print("⚠ ERRO: Você não rodou a primeira célula para definir os nomes dos Agentes!")

```

PROTOCOLO GENESIS 2026 - Acesso Restrito

Insira as credenciais descriptografadas no terminal Python para avançar.

Nome da Dupla: _____

Nível 1 - Acesso ao Mainframe

Rode o código do Nível 1 no Colab. Qual foi o Username gerado para o alvo?

Insira o Username gerado: 146k

Nível 2 - Firewall Ativo

Analise a saída do seu terminal para as portas [1002, 2024, 3055, 15, 888].

Qual a sequência de status gerada? (Digite separado por espaços): ^BLOQUEADO LIBERADO QUARENTENA QUARENTENA BLOQUEADO\$

Nível 3 - Sequência Lógica

O sistema pede a SOMA dos números PARES nos primeiros 15 termos de Fibonacci.

Insira o valor da soma: 798

Nível 4 - Limpeza de Dados

Após remover os arquivos infectados com "virus", quantos arquivos seguros sobraram?

Quantidade de arquivos limpos: 5

Nível 5 - A Matriz Mestra

Qual o valor da energia (soma da diagonal principal) calculado no dicionário de configuração?

Valor da energia: 4576

Nível 6 - PROTOCOLO GENESIS RESTAURADO

Parabéns, Agentes! O sistema está online.

Insira o código final da missão para confirmar a reinicialização para registrar seu tempo: 125