




# CODE ONE



LET'S CODE

## Desafio Code One - Python

Fala, pessoal! Sejam muito bem-vindos ao Desafio de Python do Code One! Se você chegou aqui, então já deve ter acompanhado as aulas do nosso curso, certo? Bom, agora, é hora de colocar seus novos conhecimentos em prática! Vamos nessa?! 🚀 😊

Antes de tudo, é válido destacar que por se tratar de um desafio, é essencial que você troque uma ideia com outras pessoas, caso sinta dificuldades. Para isso, conte com o [Let's Code Pass](#). Ah, e claro, você também pode ficar de olho nas dúvidas de outras pessoas para ajudá-las por lá. Tudo isso vai te fazer avançar cada vez mais rápido rumo ao seu objetivo! 🎯

### Conhecendo o seu desafio

Você já deve ter jogado o Jogo da Forca, certo? O que você acha de desenvolver o seu próprio Jogo da Forca em Python? Esse será o seu desafio!


Em resumo, ao iniciar o jogo, o jogador receberá uma palavra "codificada", de forma que ele saiba apenas o número de letras que a compõem. Com base nisso, o jogador deve escolher uma letra que acredita fazer parte daquela palavra. O jogo acaba quando o jogador erra a letra pela sexta vez ou quando acerta todas as letras da palavra.

### Como desenvolver o seu Jogo da Forca?

Para te ajudar com isso, vamos te passar algumas diretrizes para que você tenha uma noção clara de como o jogo deve funcionar e de quais etapas você deve seguir para atingir esse resultado. Vamos lá!

1. No início do código, você pode solicitar o nome do jogador. Assim, você pode imprimir uma mensagem de boas-vindas. Outra opção, caso queira incluir, é imprimir uma mensagem explicando como o jogo funciona.
2. Você precisa definir uma palavra para o jogador descobrir, certo? Para isso, você pode ter uma lista com várias palavras e, no início do programa, sortear uma delas.
3. Escolhida uma palavra para o jogador descobrir, você já pode mostrar para ele quantas letras a palavra tem. Vai ser importante ter uma outra variável que consiste na palavra que o jogador está tentando acertar. Você pode utilizar uma lista, que inicia com vários `█`, por exemplo; sendo o número de `█` igual ao número de caracteres da palavra que ele precisa descobrir.
4. A partir daí, o jogo começa. Ou seja, você irá pedir que o usuário informe uma letra repetidas vezes, até que ele erre 6 vezes ou acerte todas as letras da palavra (já consegue imaginar como vai ser essa estrutura de repetição?).
5. Lembre-se que, ao término dessa repetição, você deve mostrar que o usuário perdeu, caso ele tenha errado 6 vezes, ou que ele acertou a palavra, caso ele a tenha completado. Além disso, é importante que você informe qual era a palavra a ser descoberta.

Pronto! Essas são algumas diretrizes para que você possa se guiar no desenvolvimento do seu desafio.

 **Importante:** apesar das diretrizes elencadas acima, você pode utilizar sua criatividade para deixar esse jogo com a sua cara. Fique à vontade para personalizar o funcionamento dele, de acordo com o que você acredita que vai te ajudar a extrair o máximo de conhecimento dessa experiência.

## Para inspirar

Para te inspirar de maneira você possa ter uma referência ao longo do desenvolvimento desta atividade,, assista a este vídeo: [https://youtu.be/Oc\\_HCXAYdxq](https://youtu.be/Oc_HCXAYdxq)

## Dicas

Você deve ter notado no vídeo acima que existem alguns recursos do Python que estão sendo utilizados, mas que você pode não ter visto durante o curso. Não se preocupe; no primeiro momento, você não precisa se preocupar em utilizá-los ou até mesmo em fazer da mesma forma que está no vídeo. Lembre-se que é apenas uma referência.

Porém, tendo certeza que você vai querer ir além, já preparamos algumas dicas que podem te ajudar:

## Qual comando eu poderia utilizar no Python para limpar o console?

Para isso, você deve utilizar um comando do sistema, o que é possível de ser feito por meio do módulo `os` do Python. Nesse módulo, existe uma função chamada `system` que te permite utilizar comandos do sistemas (comandos que você utilizaria no cmd, prompt de comando ou Terminal). Sendo assim, veja o exemplo abaixo:

```
import os
```

```
# o comando abaixo vai limpar o seu console, caso você esteja utilizando o Windows os.system('cls')
```

```
# o comando abaixo vai limpar o seu console, caso você esteja utilizando o Linux/Mac os.system('clear')
```

## Como eu posso fazer para "congelar" a execução do meu programa por um tempo específico?

Você pode fazer isso utilizando a função `sleep` do módulo `time`. Esse módulo já vem instalado com o Python, portanto, basta você fazer a importação desse módulo ou apenas da função `sleep` (como no exemplo abaixo).

```
from time import sleep
```

```
sleep(3)
```

```
print('Essa mensagem só aparece após 3 segundos..')
```

## LET'S CODE

Observando o exemplo acima, você pode notar que a função `sleep` deve receber um parâmetro, que é o tempo em segundos que o programa irá "dormir". Ou seja, "segurar" o seu fluxo de execução. Portanto, o `print` que vem logo abaixo da função `sleep` será executado apenas 3 segundos após a chamada dessa função `sleep`.