

<https://github.com/LeaGuima/jogo.git>

Arquivo 1: jogo_da_forca.py

```
import random
```

```
def escolher_palavra():
```

```
    """Escolhe uma palavra aleatória da lista."""
```

```
    palavras = ['python', 'programacao', 'computador', 'desafio', 'jogo']
```

```
    return random.choice(palavras)
```

```
def processar_palpite(palpite, palavra_secreta, letras_descobertas, letras_erradas, tentativas_restantes):
```

```
    """Processa o palpite do jogador, atualizando o estado do jogo."""
```

```
    if len(palpite) == 1: # Palpite é uma letra
```

```
        if palpite in letras_descobertas or palpite in letras_erradas:
```

```
            return "Você já tentou essa letra. Tente outra.", letras_descobertas, letras_erradas, tentativas_restantes
```

```
        if palpite in palavra_secreta:
```

```
            for i, l in enumerate(palavra_secreta):
```

```
                if l == palpite:
```

```
                    letras_descobertas[i] = palpite
```

```
            return "Boa! Você acertou uma letra.", letras_descobertas, letras_erradas, tentativas_restantes
```

```
        else:
```

```
            letras_erradas.append(palpite)
```

```
            return "Ops! Essa letra não está na palavra.", letras_descobertas, letras_erradas, tentativas_restantes - 1
```

```
    elif len(palpite) == len(palavra_secreta): # Palpite é uma palavra
```

```
        if palpite == palavra_secreta:
```

```
            return "Você adivinhou a palavra corretamente!", letras_descobertas, letras_erradas, tentativas_restantes
```

```
        else:
```

```
        return "Não é a palavra correta.", letras_descobertas, letras_erradas,
tentativas_restantes - 1
```

```
    return "Entrada inválida. Tente novamente.", letras_descobertas, letras_erradas,
tentativas_restantes
```

Arquivo 2: main.py

```
from jogo_da_forca import escolher_palavra, processar_palpite
```

```
def jogo_da_forca():
```

```
    palavra_secreta = escolher_palavra()
```

```
    letras_descobertas = ['_' for _ in palavra_secreta]
```

```
    letras_erradas = []
```

```
    tentativas_restantes = 6
```

```
    tentativas_totais = 0
```

```
    print("Bem-vindo ao jogo da forca!")
```

```
    print("Adivinhe a palavra abaixo:")
```

```
    print(' '.join(letras_descobertas))
```

```
    while tentativas_restantes > 0:
```

```
        print(f"\nTentativas restantes: {tentativas_restantes}")
```

```
        print(f"Letras erradas: {' '.join(letras_erradas)}")
```

```
        if tentativas_totais >= 4:
```

```
            palpite = input("Digite uma letra ou tente adivinhar a palavra inteira: ").lower()
```

```
        else:
```

```
            palpite = input("Digite uma letra: ").lower()
```

```
        mensagem, letras_descobertas, letras_erradas, tentativas_restantes =
processar_palpite(
```

```

        palpite, palavra_secreta, letras_descobertas, letras_erradas,
tentativas_restantes
    )

    print(mensagem)
    tentativas_totais += 1

    # Mostra o progresso
    print(' '.join(letras_descobertas))

    # Verifica se o jogador ganhou
    if '_' not in letras_descobertas:
        print("\nParabéns! Você descobriu a palavra!")
        break
    else:
        print("\nVocê perdeu! A palavra era:", palavra_secreta)

# Executa o jogo
if __name__ == "__main__":
    jogo_da_forca()

```

Executar o jogo com :

```
python main.py
```

Projeto: Jogo da Forca

O **Jogo da Forca** é um projeto desenvolvido em Python com o objetivo de recriar um dos jogos mais populares e simples, mas com uma abordagem que prioriza a organização e boas práticas de programação. O jogo oferece uma experiência interativa para o jogador, que precisa adivinhar uma palavra oculta, uma letra por vez, dentro de um número limitado de tentativas.

Funcionalidades do Projeto

1. Escolha Aleatória de Palavras

O jogo inicia escolhendo automaticamente uma palavra de uma lista pré-definida. Essa seleção aleatória garante que cada partida seja diferente, proporcionando uma experiência variada para o jogador.

2. Interação com o Jogador

O jogador interage diretamente com o jogo, inserindo palpites no formato de letras. O programa informa se a letra faz parte da palavra oculta ou não. Além disso, após quatro tentativas (certas ou erradas), o jogador pode tentar adivinhar a palavra inteira, oferecendo uma dinâmica adicional e mais opções para resolver o desafio.

3. Controle de Tentativas

O jogador tem um limite inicial de seis tentativas. Cada erro resulta na redução desse número, e o jogo termina quando as tentativas se esgotam. Esse controle adiciona um elemento de tensão e estratégia, incentivando o jogador a pensar bem antes de fazer um palpite.

4. Feedback em Tempo Real

Durante o jogo, o programa mantém o jogador informado sobre o progresso. Ele exibe:

- As letras corretas já descobertas.
- As letras erradas que foram tentadas.
- A palavra oculta, com os espaços ainda não revelados destacados.

Além disso, mensagens claras indicam se o palpite foi válido, correto ou incorreto.

5. Finalização do Jogo

O jogo termina de duas formas possíveis:

- Vitória: o jogador descobre todas as letras da palavra ou adivinha a palavra inteira corretamente.
- Derrota: o jogador esgota todas as tentativas sem descobrir a palavra.

Estrutura e Etapas de Desenvolvimento

1. Planejamento e Design

A primeira etapa foi definir a lógica principal do jogo e identificar as

funcionalidades desejadas, como escolha aleatória de palavras, controle de tentativas e feedback para o jogador.

2. Divisão do Código

Para garantir um código organizado e reutilizável, o projeto foi dividido em dois arquivos:

- **jogo_da_forca.py**: Contém as funções responsáveis pela lógica do jogo, como a escolha da palavra e o processamento dos palpites do jogador.
- **main.py**: Gerencia o loop principal do jogo, lida com a interação do jogador e utiliza as funções do arquivo anterior.

3. Desenvolvimento do Código

Cada parte do código foi escrita para cumprir uma responsabilidade específica. Por exemplo:

- A função **escolher_palavra** é responsável por selecionar a palavra aleatória.
- A função **processar_palpite** lida com as regras de validação do palpite e atualiza o estado do jogo.

4. Testes e Ajustes

O jogo foi testado extensivamente para garantir que:

- Letras e palavras fossem validadas corretamente.
- O jogador recebesse feedback claro em todas as etapas.
- A funcionalidade de adivinhar a palavra inteira após quatro tentativas funcionasse sem erros.

5. Melhoria da Experiência do Jogador

Foram adicionadas mensagens informativas e ajustes na interface do jogo para tornar a experiência mais amigável e intuitiva.

Conclusão

Este projeto não só recria o clássico Jogo da Forca, mas também demonstra a importância de uma abordagem estruturada no desenvolvimento de software. Com a separação do código em módulos e foco na organização, o projeto é fácil de manter e expandir, permitindo adicionar futuras melhorias, como carregar palavras de um arquivo externo ou criar um modo de dificuldade.

O desenvolvimento do **Jogo da Força** foi um projeto que, embora não tenha sido extremamente difícil, apresentou uma boa dose de complexidade. Houve desafios ao longo do caminho, especialmente na validação das entradas dos jogadores, no controle do estado do jogo e na organização do código em módulos separados para garantir clareza e manutenção.

Enfrentar essas dificuldades foi uma parte importante do aprendizado. Para resolvê-las, recorri a pesquisas em fontes confiáveis, como documentação oficial e fóruns de programação, além de trocar ideias e discutir soluções com amigos que também têm experiência na área. Essas conversas foram muito enriquecedoras, pois me ajudaram a enxergar o problema por diferentes perspectivas.

No final, o resultado foi um projeto bem estruturado e funcional, que me proporcionou uma experiência prática de aplicar lógica de programação e boas práticas de organização de código. Apesar das dificuldades, cada desafio superado foi uma oportunidade de aprendizado, e a satisfação de ver o jogo funcionando compensou todo o esforço.