

Atividades do Módulo 4 - QA NA PRÁTICA

Esse [repositório](#) é dedicado às atividades realizadas durante o Módulo 4 - M4 - LÓGICA DE PROGRAMAÇÃO COM PYTHON do curso de Quality Assurance oferecido pelo [Instituto JogaJunto](#).

Descrição da 21ª Atividade:

🔍 Faça uma pesquisa sobre ESTRUTURAS DE REPETIÇÃO E ITERAÇÃO, identificando: O que são estruturas de repetição e iteração? | Quando são usadas? | Quais os principais tipos de estruturas de repetição?

Estruturas de Repetição e Iteração

As estruturas de repetição e iteração são fundamentais na programação e são usadas para executar um conjunto de instruções repetidamente, com base em uma condição específica. Essas estruturas são utilizadas quando se deseja automatizar tarefas que precisam ser realizadas várias vezes ou quando se precisa percorrer elementos em uma coleção de dados, como uma lista ou um conjunto.

As estruturas de repetição podem ser usadas em uma variedade de situações, incluindo:

- **Processamento de Dados:** Para processar cada elemento de uma lista, arquivo ou conjunto de dados.
- **Iteração de Loops:** Para criar loops que executam um conjunto de instruções até que uma condição seja atendida.
- **Validação de Entradas:** Para garantir que o usuário insira dados corretos ou para verificar entradas em um formulário, repetindo até que sejam válidas.
- **Implementação de Algoritmos:** Para implementar algoritmos que envolvem repetição, como ordenação, busca e cálculos iterativos.

Principais Tipos de Estruturas de Repetição

For Loop

Utilizado quando você sabe antecipadamente quantas vezes deseja repetir um bloco de código. Em Python:

```
for i in range(5):  
    print(i)
```

While Loop

Usado quando você deseja repetir um bloco de código enquanto uma condição for verdadeira. Em Python:

```
count = 0
while count < 5:
    print(count)
    count += 1
```

Do-While Loop (Não disponível em Python)

Este tipo de loop executa um bloco de código pelo menos uma vez e, em seguida, verifica a condição para continuar a execução.

Em Python, não existe uma estrutura de loop do-while incorporada como em algumas outras linguagens de programação, como C, C++, C#, etc. No entanto, você pode simular um loop do-while usando um loop while tradicional com uma condição que sempre seja verdadeira na primeira iteração e, em seguida, usar uma instrução break para sair do loop quando a condição desejada não for mais atendida.

Aqui está um exemplo de como simular um loop do-while em Python:

```
while True:
    print("Este é o bloco do loop do-while simulado.")

    continuar = input("Deseja continuar? (S para sim, qualquer outra tecla para sair): ").strip().lower()

    if continuar != 's':
        break
```

Em C#, por exemplo, que possui essa estrutura de repetição incorporada ficaria dessa forma:

```
using System;

class Program
{
    static void Main()
    {
        int contador = 0;

        do
        {
            Console.WriteLine($"Este é um loop do-while. Contador: {contador}");
            contador++;
        }
        while (contador < 5);
    }
}
```

```
}
```

Loop Aninhado

É possível usar loops dentro de outros loops para realizar tarefas complexas ou percorrer matrizes multidimensionais. Em Python:

```
for i in range(3):  
    for j in range(2):  
        print(i, j)
```

Loop Infinito

Um loop que executa indefinidamente até que seja explicitamente interrompido. Cuidado ao usar loops infinitos, pois eles podem causar travamentos. Em Python:

```
while True:  
    print("Isso é um loop infinito")
```

As estruturas de repetição e iteração são elementos fundamentais na programação e são essenciais para controlar o fluxo de um programa, permitindo a automação de tarefas repetitivas e a manipulação de dados em coleções.

Integrantes da Squad:

| Beatriz Souza | [Bruno Soares](#) | [Leanderson Lima](#) | [Rebeca Borges](#) | Sara Cruz |