

Aula 2 – Lógica de Programação

O laço while: o que é, como usar e calculando PA e PG com while

Nos últimos artigos aprendemos vários operadores e atalhos utilizados pelos programadores Java, como ++, +=, *= dentre outros.

Está na hora de colocá-los em ação e aprender um dos comandos mais importantes em qualquer linguagem de programação: os laços

Nesse artigo, explicaremos o que é e como usar o laço while.

Como usar o laço while em Java

While, do inglês 'enquanto' é um laço de repetição. Cada ciclo de repetição é chamado de iteração.

Sua sintaxe é a seguinte:

```
while ( condição ){  
    códigos  
}
```

Ou seja, enquanto (while) a condição for verdadeira, os códigos entre chaves serão repetidos.

Mas qual a vantagem de ter um código repetido? Isso não parece absurdo?

Bom, quando pensamos em repetido, pensamos em algo estático, mas não isso o que acontece.

É aí que entra os operadores de incremento, decremento, atribuição e outros.

Nas linhas de código, geralmente as coisas vão 'mudando' a condição e à cada iteração a condição é testada.

Se a condição retornar um valor lógico verdadeiro, o código entre chaves é executado.

Caso o valor retornado seja falso, o laço while é então terminado e o programa Java segue normalmente.

Vamos ver um exemplo simples, que mostra os números de 1 até 10.

Contando até 10 em Java, com o laço while

O programa é simples.

Definimos uma variável inteira, 'count'.

Enquanto essa variável for menor que 10, seu valor será imprimido e será incrementado.

Ou seja será, será impresso 1, 2, 3, ..., 9, 10...opa!

Quando chegar em 11, a condição dentro do laço não será mais verdadeira, e o laço terminará!

Teste:

```
public class contando {  
    public static void main(String[] args) {  
        int count=1;  
  
        while(count<=10){  
            System.out.println(count);  
            count++;  
        }  
    }  
}
```

Fazendo um PA (progressão aritmética) com o laço while, em Java

O programa a seguir mostra os termos 'an' de uma PA, de termo inicial 'inicial' e razão 'razao', até um valor máximo 'valor_max'. A diferença deste programa para o anterior é que os termos de uma PA crescem através da adição da razão ao termo anterior.

Em programação, poderíamos representar isso assim: **$an = an + razao$**

Porém, como vimos alguns operadores em Java, faremos isso: **$an += razao$**

Veja o resultado:

```
public class pa {  
    public static void main(String[] args) {  
        int inicial=1,  
            razao=3,  
            an=inicial,  
            valor_max=20;  
  
        System.out.printf("Elementos da PA, de valor inicial %d e razão %d, menores que %d\n", inicial, razao,  
            valor_max );  
        while(an<=valor_max){  
            System.out.println(an);  
            an += razao;  
        }  
    }  
}
```

Fazendo um PG (progressão geométrica) com o laço while, em Java

O programa a seguir mostra os termos 'gn' de uma PG, de termo inicial 'inicial' e quociente 'quociente', até um valor máximo 'valor_max'. Os produtos de uma PG, progressão geométrica, crescem através do produto do termo anterior com o quociente.

Em programação, poderíamos representar isso assim: $gn = gn * razao$

Porém, como vimos alguns operadores em Java, faremos isso: $gn *= razao$

Veja o resultado:

```
public class pg {  
    public static void main(String[] args) {  
        int inicial=1,  
            quociente=2,  
            gn=inicial,  
            valor_max=32;  
  
        System.out.printf("Elementos da PG, de valor inicial %d e razão %d, menores que %d\n", inicial, quociente,  
valor_max );  
        while(gn<=valor_max){  
            System.out.println(gn);  
            gn *= quociente;  
        }  
    }  
}
```

Links para consulta:

<https://www.udemy.com/logica-de-programacao-csharp/>

<https://www.youtube.com/watch?v=kCGAGS3e8cY>

Pratique os exercícios 1114, 1115, 1117, 1118, 1131, 1134, 1154, 1159 no link abaixo:

<https://www.urionlinejudge.com.br/judge/pt/login>