

cronapi.organizers.Operations.group

Grupo

O bloco de programação **Grupo** permite agrupar uma parte do código, **indicando o início e o fim de um bloco de código**, facilitando assim a legibilidade e visualização de cada responsabilidade da função.

Compatibilidade

 Servidor

Exemplo

Nos exemplos abaixo, apresentamos o bloco de programação **Grupo** e a forma de utilizá-lo. Este bloco de programação e o bloco **Comentário** permitem alteração do seu nome e a cor do bloco, isso acontece para que seja possível identificar facilmente um bloco de código específico que o usuário tenha agrupado. Para alterar o nome do bloco, clique diretamente sobre **Nome do Grupo**, para alterar a cor, clique no retângulo ao lado do ícone da paleta de cores.

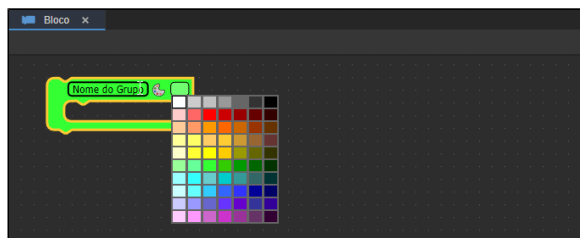


Figura 1 - Bloco de programação Grupo

A **figura 2** ilustra um exemplo de utilização do bloco. Perceba que a função **Operações** possui duas partes, uma que faz a soma de dois números e outra que faz a subtração, o bloco de programação **Grupo** está sendo utilizado nesta função para definir qual bloco de código é responsável pela soma e qual é responsável pela subtração.

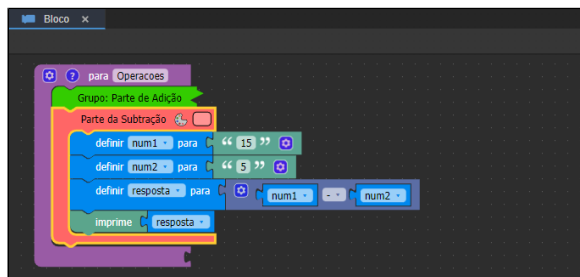


Figura 2 - Bloco de programação Grupo agrupando duas partes de uma função

O resultado da aplicação do bloco pode ser conferido na figura abaixo, para acessar essa aba basta ativar o **modo avançado** e clicar na aba **Código**. Os comentários em roxo definidos pelas duas barras roxas (//) são feitos pelo bloco de programação **Grupo**.

```

15 - /**
16 -  *
17 -  * Bloco de executar!
18 -  *
19 -  * @author Fábio Duarte Freitas
20 -  * @since 27/07/2022 11:23:39
21 -  */
22 - */
23 - public static Var Operacoes() throws Exception {
24 -     return new Callable<Var>() {
25 -
26 -         private Var num1 = Var.VAR_NULL;
27 -         private Var num2 = Var.VAR_NULL;
28 -         private Var resposta = Var.VAR_NULL;
29 -
30 -         public Var call() throws Exception {
31 -             // region Parte de Adição
32 -             num1 = Var.valueOf("5");
33 -             num2 = Var.valueOf("5");
34 -             resposta = cronapi.math.Operations.sum(num1,num2);
35 -             System.out.println(resposta.getObjectAsString());
36 -             // endregion Parte de Adição
37 -             // region Parte da Subtração
38 -             num1 = Var.valueOf("15");
39 -             num2 = Var.valueOf("5");
40 -             resposta = cronapi.math.Operations.subtract(num1,num2);
41 -             System.out.println(resposta.getObjectAsString());
42 -             // endregion Parte da Subtração
43 -             return Var.VAR_NULL;
44 -         }
45 -     }.call();
46 - }

```

Visualização Código

Figura 3 - Bloco de programação Grupo visualizado de forma high-code