

Fundamentos de Java

2-14: Métodos e Classes de Java

Atividades Práticas

Objetivos da Lição:

- Descrever um método, uma classe e uma instância
- Descrever um cenário no qual uma estrutura de controle IF seria usada
- Descrever um cenário no qual uma estrutura de controle WHILE seria usada
- Reconhecer a sintaxe para um método, uma classe, uma função e um procedimento
- Descrever entrada e saída

Vocabulário:

Identifique a palavra do vocabulário para cada definição a seguir.

	Instruções que permitem selecionar e executar blocos específicos de código enquanto ignoram outras seções.
	Objeto de uma classe.
	Especificação, como um projeto ou um padrão, e conjunto de instruções de como construir algo.
	Código que envia uma mensagem para um objeto solicitando que ele execute uma ação.
	Estrutura de controle que permite executar seções específicas do código uma quantidade de vezes.

Tente/solucione:

1. No Alice 3, você criará uma animação de uma criança se exercitando. Você definirá variáveis e definirá uma instrução de controle while. Você solicitará informações do usuário final e fará com que a criança diga algo (similar à criação de saída).
Nota: Se você já criou a primeira parte da animação "Criança se Exercitando" em uma atividade anterior, abra essa animação e vá para a **etapa 8** destas instruções.
 - a. Crie um mundo com uma criança. Salve o projeto como "Criança se Exercitando".
 - b. Faça com que a criança acene. Atribua o valor do aceno a "Estou feliz".
 - c. Faça com que a criança diga "Gostaria de me exercitar hoje."
 - d. Faça com que a criança faça vários exercícios (alongamentos laterais, alongamentos frontais, saltos). Depois do exercício, faça com que a criança pare e diga: "Acabei de me exercitar."
 - e. Altere o código para que antes de a criança se exercitar, você declare uma variável do tipo Inteiro denominada numSets. Defina o valor padrão como 3. Salve sua animação.
 - f. Use esse valor para controlar quantos grupos de exercícios a criança faz.
 - g. Faça com que a criança acene e diga adeus no final da animação.

- h. Modifique a variável numSets de modo que ela pergunte ao usuário quantos grupos de exercícios a criança deve fazer. (Dica: use a função `getIntegerfromUser`.)
 - i. Usando um loop faça com que a criança diga o número de grupos restantes. Quando restar um grupo, a criança deverá dizer "Último grupo." (Dica: use uma instrução de controle IF.)
 - j. Salve sua animação.
2. Incremente a animação anterior da atividade nº 1 acima e adicione outras duas crianças com variáveis diferentes (numset, numset2, numset3) para controlar o número de grupos. Salve sua animação.
3. Se você olhar o código do programa de exercícios, poderá ver que há muitas duplicações no `myFirstMethod`.
 - a. Crie um procedimento chamado `exercício` e mova o bloco de código que tem um loop `while` e a instrução "Acabei de me exercitar" do `myFirstMethod` para o novo procedimento de exercício. Faça isso por meio da área de transferência.
 - b. Crie um parâmetro denominado `numSets` do tipo `wholenumber`
 - c. Adicione uma variável de local ao procedimento de exercício nominada `conjuntos` do tipo `wholenumber` e defina seu valor equivalente a `numsets`.
 - d. Altere as instâncias de `numsets` no código para `conjuntos`
 - e. Substitua a linha que diz `numsets = numsets - 1` para usar a variável de conjuntos.
 - f. Salve sua animação.
4. Implemente seu novo método de exercício
 - a. Adicione uma instrução `doTogether` abaixo da seção da declaração da variável em `myFirstMethod`
 - b. Adicione uma chamada de exercício para cada criança usando a variável associada (`numset`, `numset2`, `numset3`)
 - c. Desative o código existente que faz com que as crianças se exercitem.
 - d. Execute sua animação para testá-la.
 - e. Exclua o código que não for mais necessário.
 - f. Salve sua animação.
5. No Alice 3, crie uma animação que use as estruturas de controle IF e WHILE para controlar o movimento de um objeto.
6. Preencha a seguinte Folha de Revisão de Sintaxe de Java:

Constructo	Sintaxe
Sintaxe IF ELSE	
Sintaxe WHILE	