

APRENDA A USAR **ARRAYLIST** EM UMA APLICAÇÃO **JAVA**

ROBERTA G. BRITTO

2020

APRESENTAÇÃO

QUEM CRIOU?

Esse material foi criado pela Profa. **Roberta Gondim Britto.**

Atualmente, a professora faz parte do quadro efetivo do **IFBA**, *campus* **Porto Seguro**, onde leciona nos cursos integrado e superior da área de **Computação**.

http://lattes.cnpq.br/5301957495717566

A QUEM SE DESTINA ?

A todos/as que desejam expandir seus conhecimentos na **Linguagem Java**.

O material foi criado partindo do princípio de que os/as interessados/as já possuam conhecimento em **Lógica de Programação** e também algum conhecimento em **Programação Orientada a Objetos** com Java.

NOTAS

Para seguir as etapas planejadas, neste material, é necessário que o/a estudante tenha o compilador Java instalado mais alguma *interface* de desenvolvimento.

Abaixo o *link* do compilador que utilizamos.
https://www.oracle.com/technetwork/pt/java/javase/downloads/index.html

Havendo dúvidas e/ou sugestões, entrem em contato conosco através do Instagram **@meninashardcore**.

VERSÃO 1.0

Proposta

Neste material, estamos propondo a criação de um programinha, desenvolvido na Linguagem Java, com a intenção de aprendermos a utilizar *arrays* dinâmicos em tempo de execução.

Trata-se de um material destinado àqueles/as que antes de tudo estejam predispostos/as a aprender. Contudo, ele foi elaborado pensando nas pessoas que já tenham um conhecimento básico da Linguagem de Programação Java ou uma boa base em Lógica de Programação.

Nesse sentido, cabe destacar a importância dos/as estudantes já possuírem algum conhecimento sobre conceitos relevantes da Programação Orientada a Objetos como, por exemplo, classes, objetos, métodos, atributos, modificadores, construtores, etc.

De todo modo, se o seu desejo é realmente aprender, seja muito bem-vindo/a. Independente do seu conhecimento em programação, nós estamos aqui para te auxiliar.

Um forte abraço,
Profa. Roberta

PLANEJAMENTO

O aprendizado foi estruturado em quatro etapas conforme descrição abaixo:

ETAPA 1

- Visão geral do programinha (*overview* da aplicação)
- Definição de *ArrayList*

ETAPA 2

- Criação do Projeto
- Estruturação das classes do Projeto
- Criação da Tela de *Interface* do Usuário (*JFrame Main*)

ETAPA 3

- Implementação das classes: *Task* e *Transaction*

ETAPA 4

- Inserção das funcionalidades na tela de *interface* do usuário (*class Main*)
- Construção do arquivo .jar

Overview da Aplicação

Nesta etapa, apresentamos a nossa proposta. Queremos que vocês aprendam a utilizar a estrutura de dados *ArrayList*. *ArrayList* é uma classe da Linguagem Java que implementa um *array* redimensionável de uma lista (*List*).

Para demonstrar o uso do *ArrayList*, desenvolveremos uma miniaplicação Java que se utiliza das quatro operações básicas de CRUD (*Create*, *Read*, *Update* e *Delete*) também conhecida por VEIA (Visualizar, Excluir, Inserir e Alterar).

Essas operações normalmente são aplicadas em Banco de Dados (BD's) relacionais, mas, neste material, nós as aplicamos em um *ArrayList*. Dessa forma, o usuário poderá inserir, alterar, excluir e visualizar informações registradas em um *ArrayList*.

Trata-se de uma aplicação para **registro de tarefas** que vão sendo armazenadas em um *array* dinâmico, ou seja, um *array* que vai crescendo ou diminuindo de tamanho enquanto o programa estiver sendo executado.

O exemplo foi feito para **tarefas**, mas poderíamos utilizar qualquer outra entidade que exige o cadastro de informações como clientes, produtos, fornecedores, etc.

Na Figura 1, apresentamos as informações da **tarefa** que serão adicionadas ao *ArrayList*, são elas: **título**, **data limite**, **descrição** e **nível de complexidade**. Além dessas, há também o **número identificador** que será gerado de forma automática pelo programa.

Overview da Aplicação

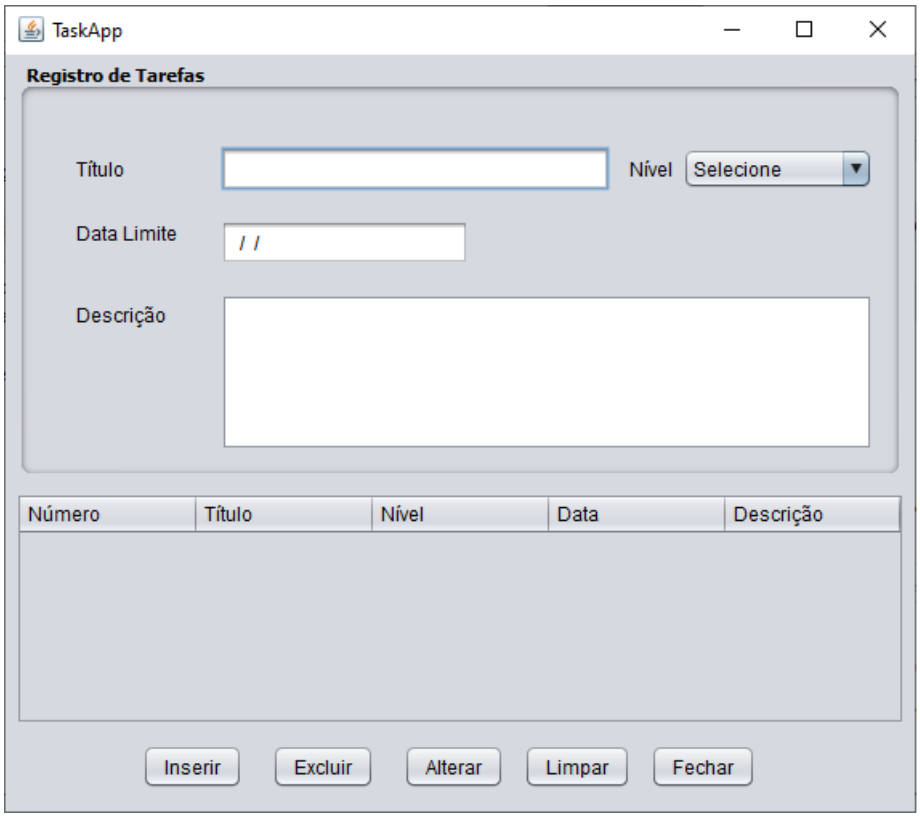


Figura 1. Tela de *Interface* do Usuário

Para registrar as tarefas, o usuário deve informar os dados solicitados e clicar no botão Inserir. Ao fazer isso, um número de tarefa é gerado automaticamente e a tarefa fica disponível para ser visualizada, excluída ou alterada. A Figura 2 mostra uma tarefa que foi adicionada ao *ArrayList*.

O programinha também oferece a opção LIMPAR, caso seja necessário limpar os valores dos atributos da tarefa que ficaram na área de Registro. Por fim, temos a opção FECHAR que encerra a nossa Aplicação.

Overview da Aplicação

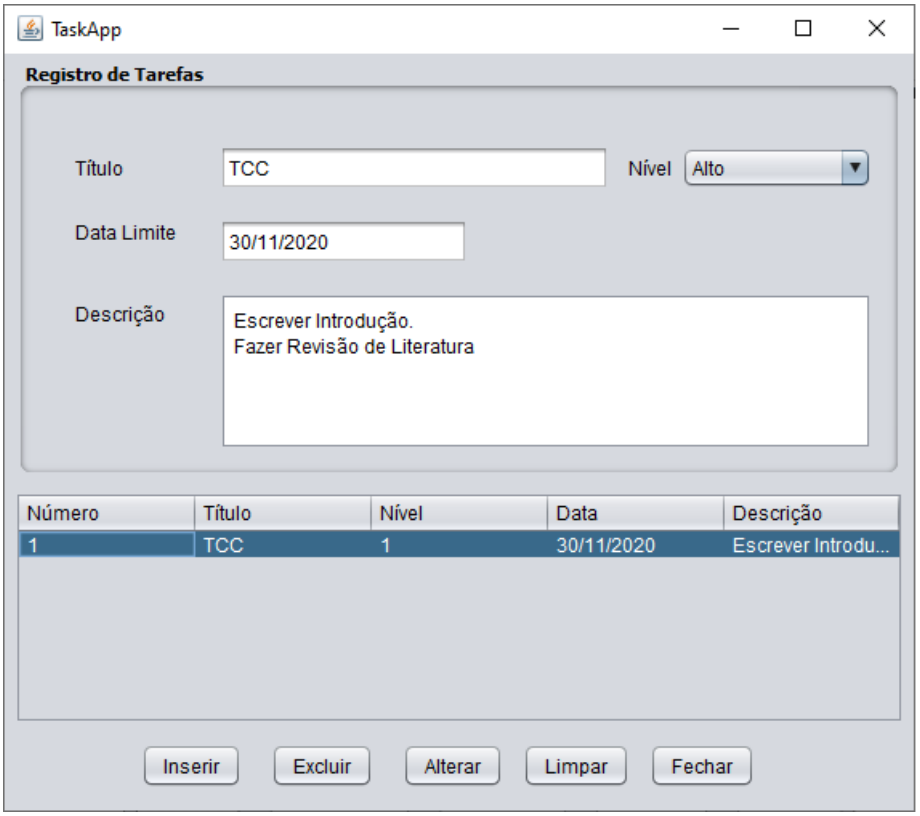


Figura 2. Tarefa adicionada ao *ArraylList*

Do mesmo modo que você inseriu a tarefa no *ArrayList*, você poderá também excluir a tarefa ou alterar os dados da tarefa que foi inserida.

Para tanto, será necessário visualizar a tarefa que deseja excluir ou alterar, na tabela abaixo da área de Registro, e clicar sobre a mesma.

A tarefa passará a ser visualizada na área de edição (identificada como área de Registro das Tarefas) e, então, você poderá deletar, clicando sobre o botão Excluir ou editar/alterar as informações, escolhendo a opção Alterar.

Lembrem-se de que os dados registrados em um *ArrayList* não são armazenados de forma permanente como, por exemplo, em um arquivo texto ou em um BD's.

Eles apenas permanecem no *array* enquanto o programa estiver sendo executado. Portanto, assim que o programa for encerrado, todas as informações contidas no *ArrayList* serão perdidas.

Caso necessitem armazená-las de forma permanente é preciso alterar o tipo de estrutura de armazenamento.

Na segunda versão deste material, vamos mostrar como armazenar esses dados em um arquivo permanente e logo em seguida em um BD's.

Se você ainda ficou com alguma dúvida ou queira enviar sugestões, entre em contato conosco pelo Instagram **@meninashardcore**.

Criação do Projeto

Nesta etapa, é necessário que vocês já estejam com o compilador Java instalado mais a *Interface* de Desenvolvimento (IDE) de sua preferência. Nós usamos a IDE NetBeans 8.2.

O primeiro passo que demonstraremos será a criação do Projeto. Então, com o NetBeans aberto clique no menu Arquivo e escolha a opção Novo Projeto. Você também pode usar as teclas de atalho **ctrl+shift+n** ou ir direto no ícone **Novo Projeto**.

Dentre as Categorias oferecidas escolha **Java** e dentre os Projetos oferecidos escolha **Aplicação Java** e clique em Próximo. Veja a Figura 3 abaixo.

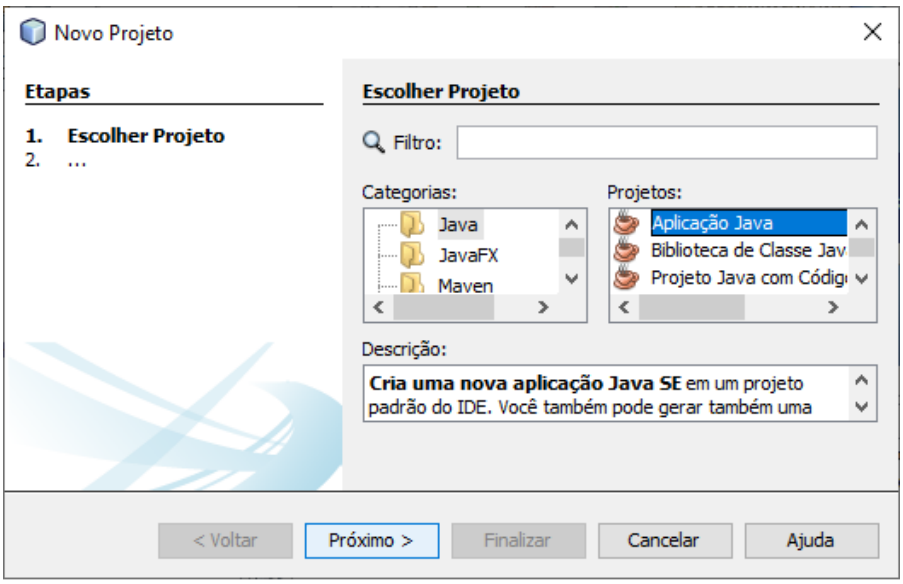


Figura 3. Criação do Novo Projeto

Ao clicar em Próximo, a janela "Novo Aplicação Java" é exibida (veja Figura 4).

Criação do Projeto

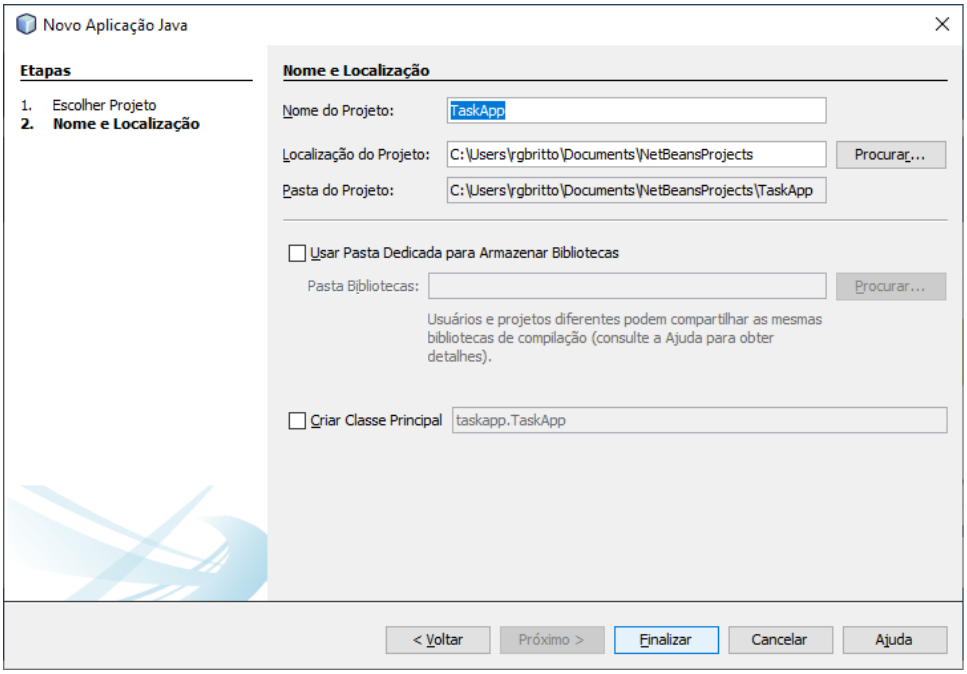


Figura 4. Tela de nome e localização do projeto

Nessa janela, você deve informar o nome do projeto e a localização. Aqui, pedimos que **desmarque** a opção 'Criar Classe Principal', pois criaremos a nossa própria classe Principal.

Depois que você informou o nome, a localização e desmarcou a opção solicitada, clique em Finalizar.

Nosso próximo passo será estruturarmos as classes do projeto. Por ser um programa bem simples, nós vamos estruturar as classes do projeto em um único pacote. Dessa forma, todas as classes criadas ficarão nesse mesmo pacote.