

Criando seus primeiros Projetos Java no NetBeans

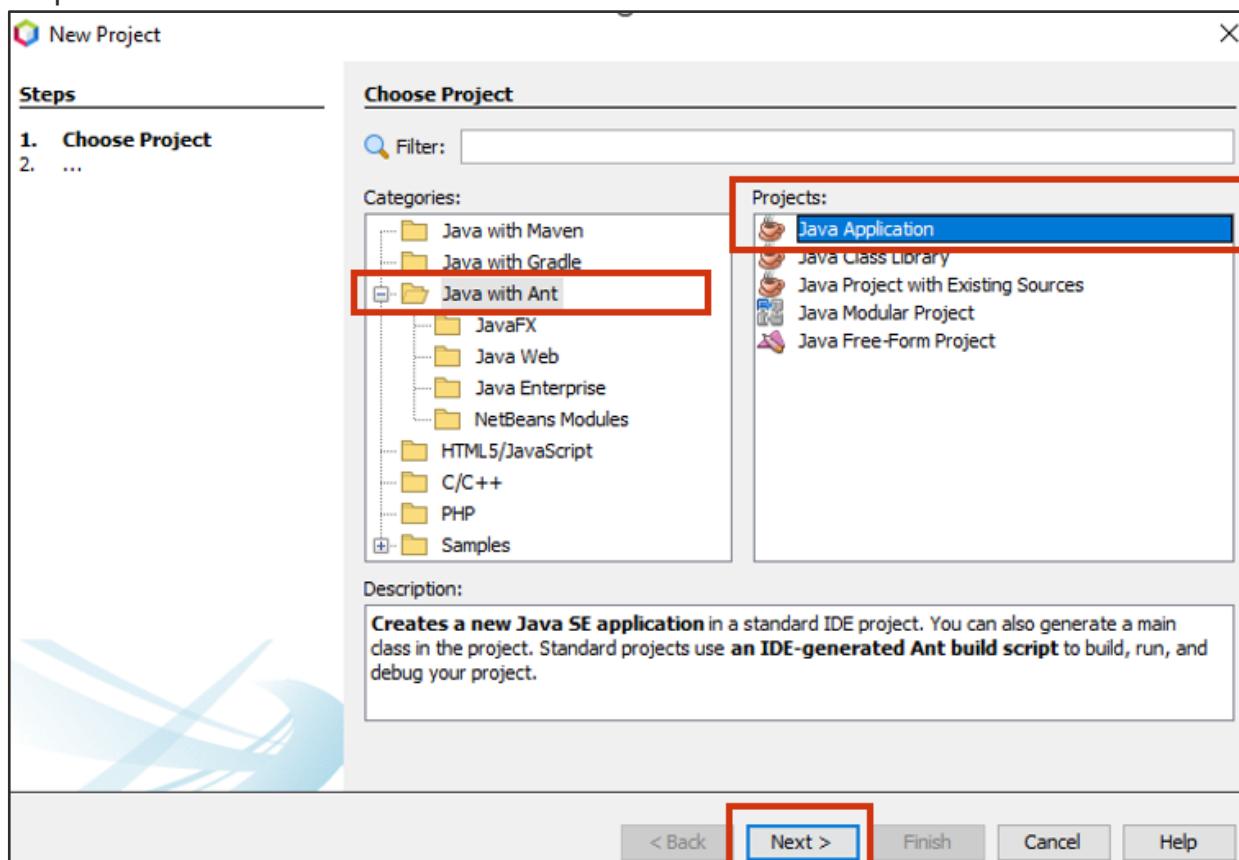
Parte 1: Criar um Projeto Java e adicionar um arquivo .java existente

Visão Geral

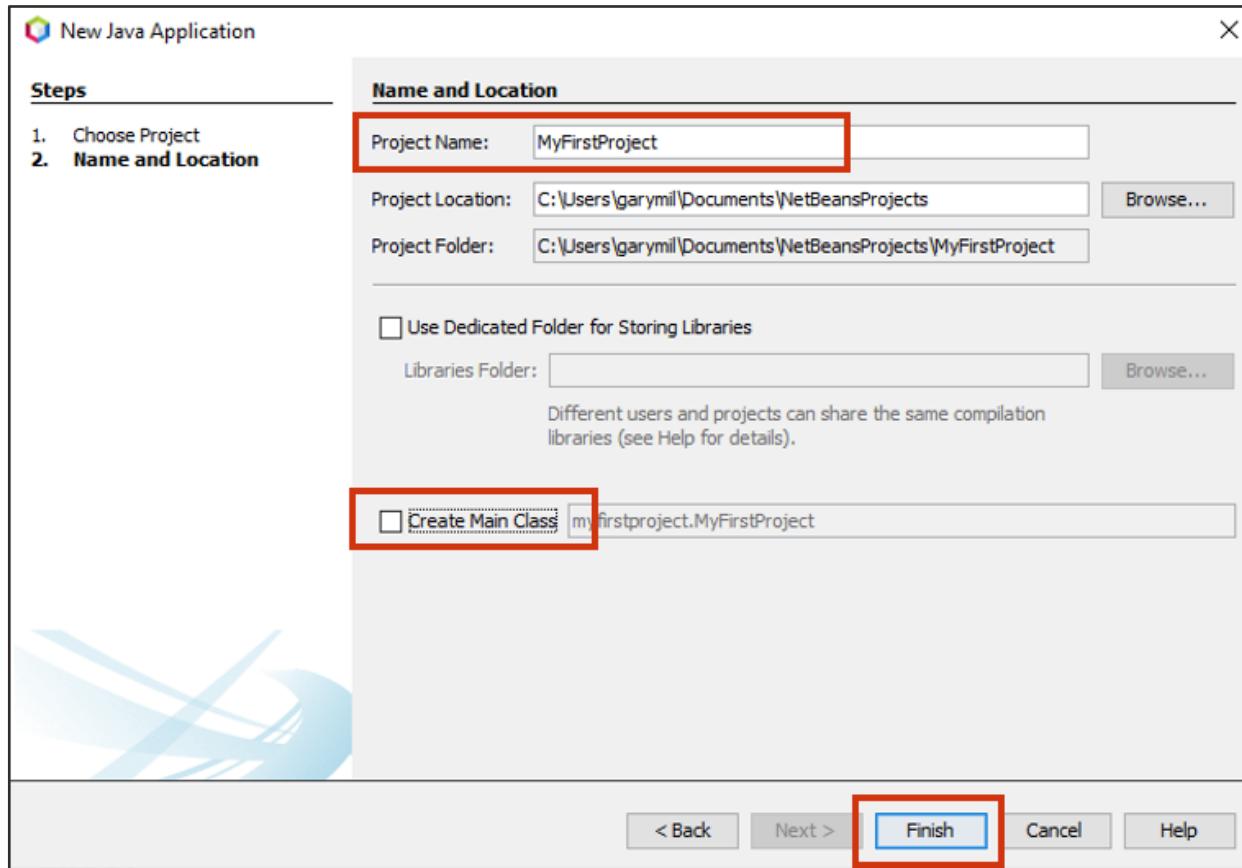
Nesta seção, você vai fazer download do arquivo **HelloWorld.zip**, criar um projeto no NetBeans e adicionar um arquivo **.java** existente ao projeto.

Tarefas

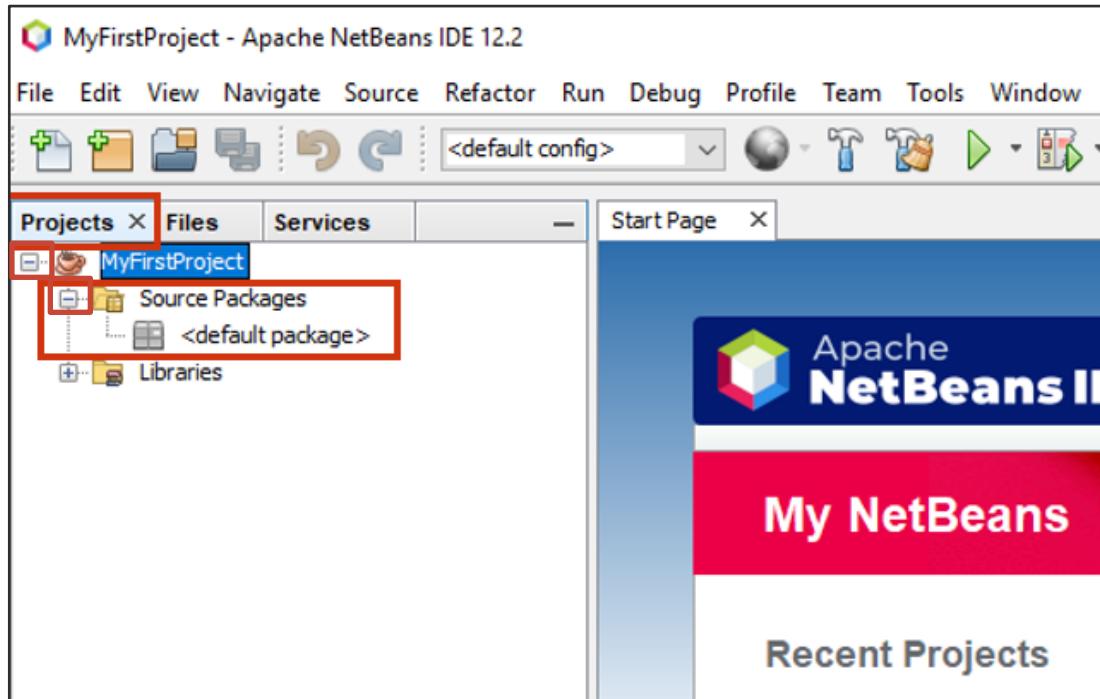
1. Faça download do arquivo zip **HelloWorld.zip** para esta prática. Extraia o arquivo zip, anotando o local.
2. Inicie o **NetBeans**.
3. Vá para **File > New Project** e selecione:
 - a. Categorias: **Java with Ant**
 - b. Projetos: **Java Application**
4. Clique em **Next >**.



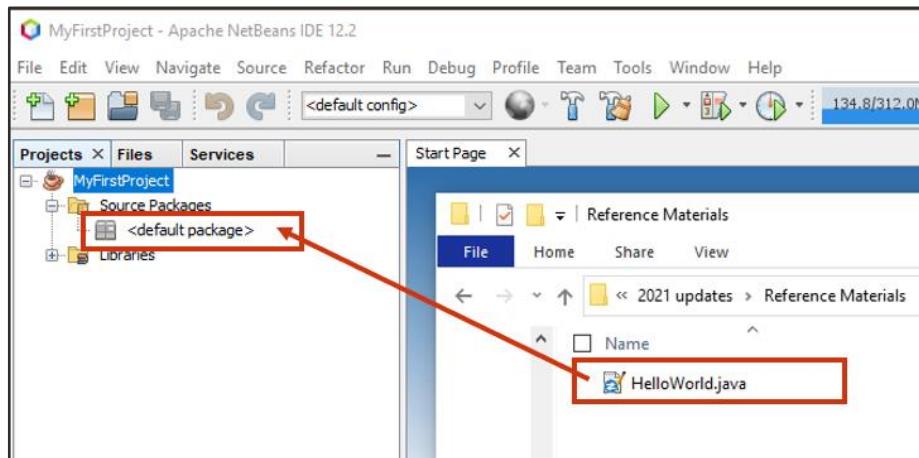
5. Atribua o nome **MyFirstProject** ao projeto. Desmarque a caixa de criação de classe principal, **Create Main Class**, e clique em **Finish**.



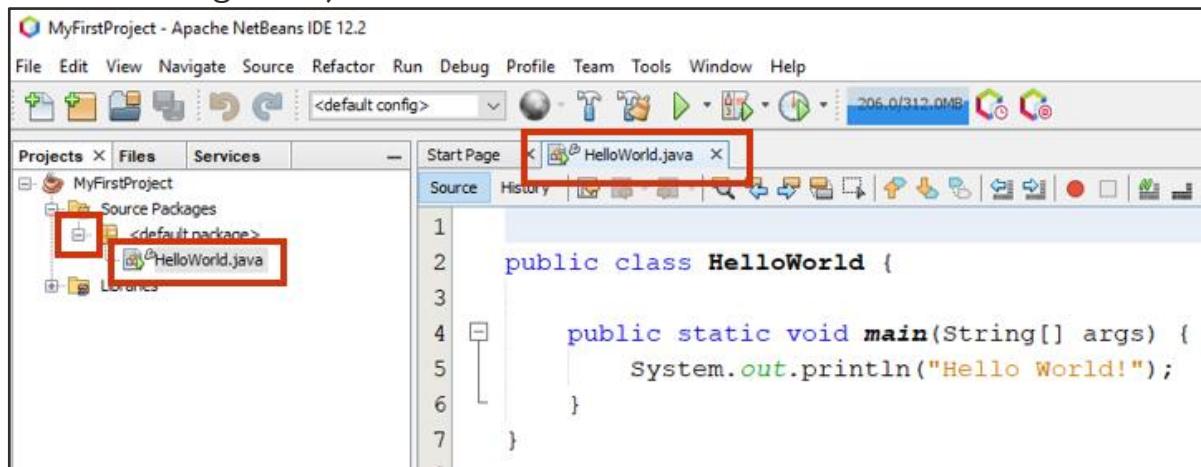
6. Na janela **Projects**, clique no **sinal de mais** para expandir o projeto e, novamente, no **sinal de mais** para expandir **Source Packages** a fim de que **<default package>** seja exibido.



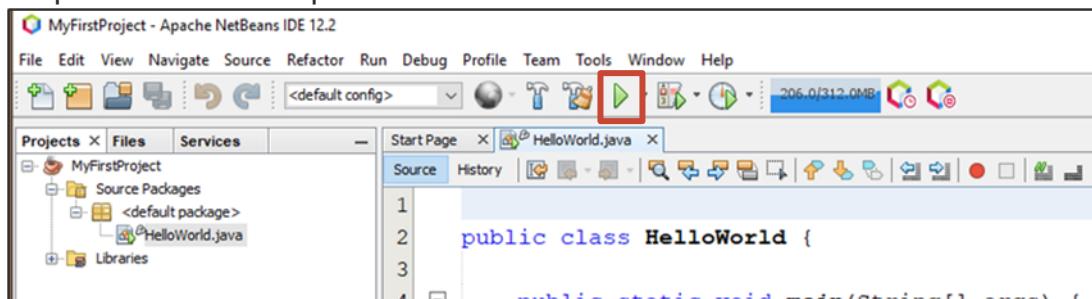
- Navegue até a pasta na qual você extraiu a pasta do arquivo **HelloWorld.zip** na etapa 1 e selecione o arquivo **HelloWorld.java**.
- Arraste o arquivo **HelloWorld.java** da pasta no seu dispositivo e solte-o em <**default package**>.



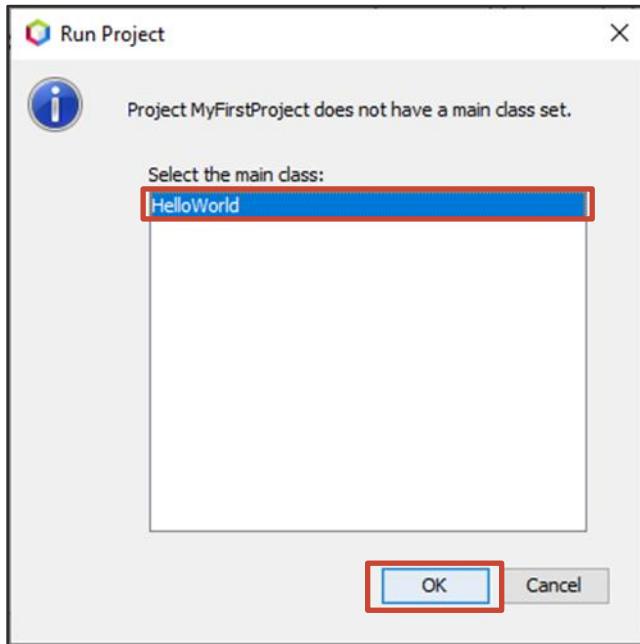
- Clique no **sinal de mais** ao lado de <**default package**> e você verá o arquivo **HelloWorld.java** no pacote. Clique duas vezes no arquivo **HelloWorld.java**. Ele será aberto em uma nova guia na janela do Code Editor.



- Clique no botão **Run** para testar.



11. Se for solicitado o preenchimento da opção **Select the main class**, selecione o padrão **HelloWorld** e clique em **OK**.



12. Na janela **Output** abaixo do Code Editor, a mensagem "**Hello World!**" será exibida

```
Output - MyFirstProject (run) ×
Created dir: C:\Users\garymil\Documents\NetBeansProjects\MyFirstProject\bin
Created dir: C:\Users\garymil\Documents\NetBeansProjects\MyFirstProject\lib
Created dir: C:\Users\garymil\Documents\NetBeansProjects\MyFirstProject\src
Compiling 1 source file to C:\Users\garymil\Documents\NetBeansProjects\MyFirstProject\bin
compile:
run:
Hello World!
BUILD SUCCESSFUL (total time: 2 seconds)
```

The screenshot shows the 'Output' window for the 'MyFirstProject' run configuration. It displays the build logs, including the creation of directories, compilation of one source file, and the execution of the program, which resulted in the output 'Hello World!'. The 'Hello World!' line is highlighted with a red border.

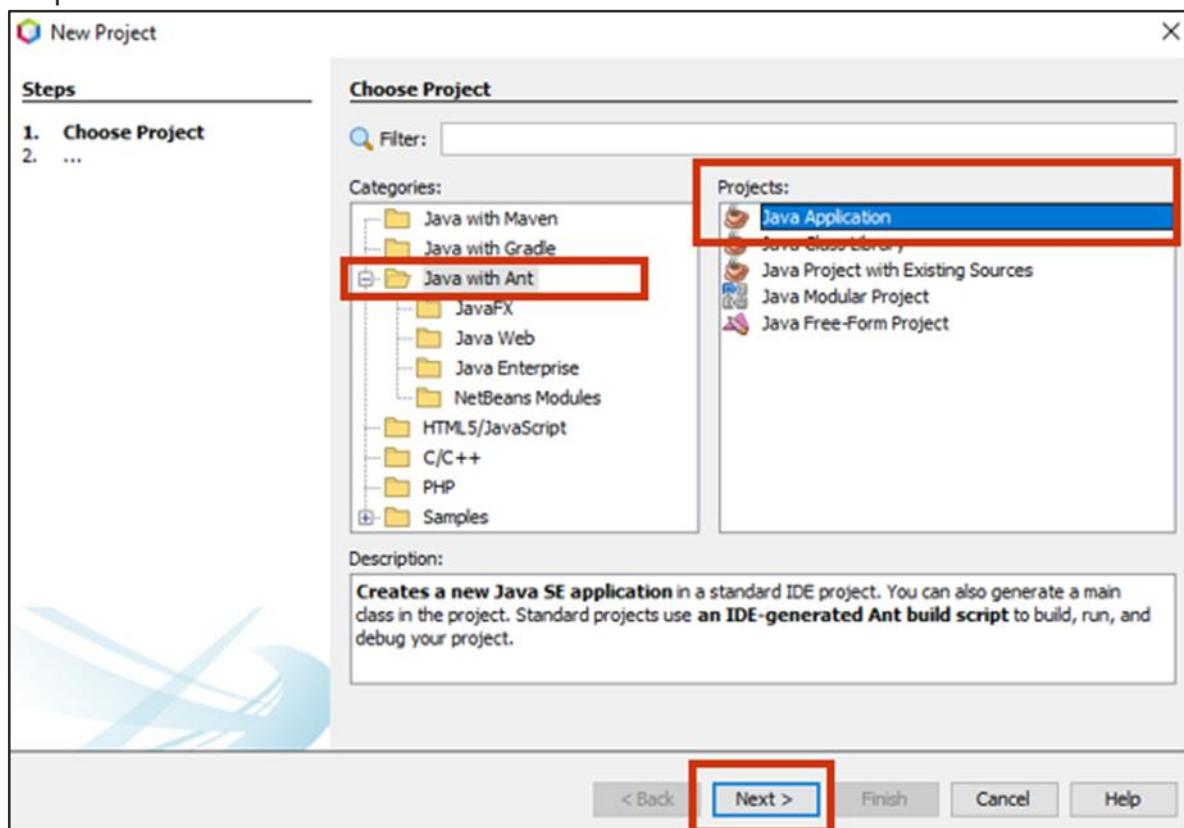
Parte 2: Criar um Projeto no NetBeans com uma classe principal

Visão Geral

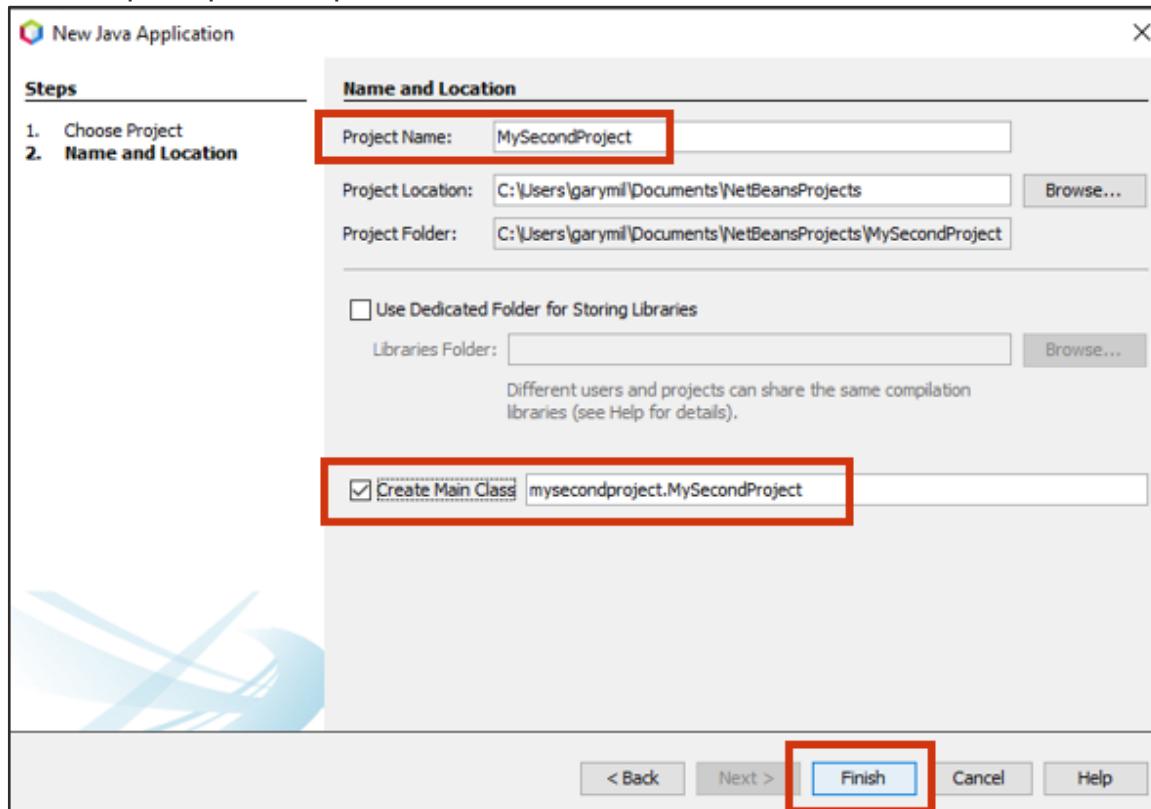
Nesta seção, você vai criar e executar um projeto java no NetBeans com uma classe Java principal.

Tarefas

1. Inicie o **NetBeans**, se ele ainda não estiver aberto.
2. Vá para **File > New Project** e selecione:
 - a. Categorias: **Java with Ant**
 - b. Projetos: **Java Application**
3. Clique em **Next >**.



4. Atribua o nome **MySecondProject** ao projeto. Marque a caixa **Create Main Class**, para criar a classe principal, e clique em **Finish**.



5. No Code Editor, localize o método principal da classe **MySecondProject** e informe a linha de código, como mostrado abaixo:

```
System.out.println("Hello again world");
```

```

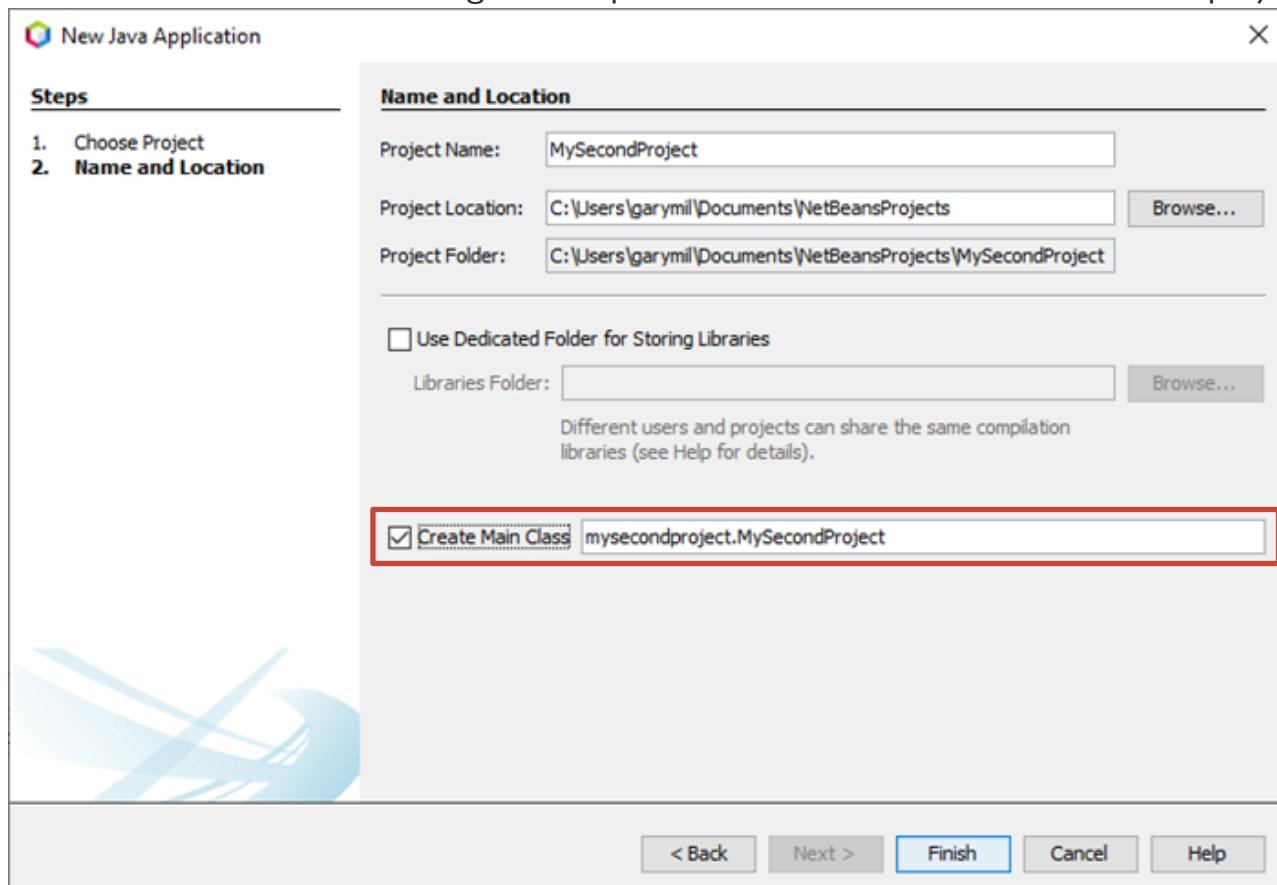
Start Page X HelloWorld.java X MySecondProject.java X
Source History | 
1
2 package mysecondproject;
3
4 public class MySecondProject {
5
6     /**
7      * @param args the command line arguments
8      */
9     public static void main(String[] args) {
10        // TODO code application logic here
11        System.out.println("Hello again world");
12    }
13 }
14

```

6. Clique no botão **Run** para testar. A mensagem "**Hello again world**" deve ser exibida no console **Output** na parte inferior do IDE.

The screenshot shows the NetBeans IDE's Output window titled "Output - MySecondProject (run)". The window displays the following text:
Created dir: C:\Users\garymil\Documents\Ne
Created dir: C:\Users\garymil\Documents\Ne
Created dir: C:\Users\garymil\Documents\Ne
Compiling 1 source file to C:\Users\garymil\Documents\Ne
compile:
run:
Hello again world
BUILD SUCCESSFUL (total time: 1 second)

Observe uma diferença importante entre o primeiro e o segundo projetos. Em **MySecondProject**, quando a caixa **Create main class** é marcada na Tarefa 4, um pacote nomeado também é criado. O NetBeans adiciona um nome sugerido de pacote e de classe com base no nome do projeto.



O formato do nome sugerido é **packagename.MainClassName**; neste caso, **mysecondproject.MySecondProject**. Neste ponto, é possível editar os nomes se desejar algo além do formato de denominação sugerido.

Na Parte 1, como você não criou uma Classe Principal, o pacote foi denominado **<default package>**. Se um pacote tiver um nome diferente de **<default package>**, observe que uma declaração de pacote deverá ser incluída como a primeira linha do código no arquivo **.java**. Se uma classe for criada em um pacote existente, o NetBeans adicionará automaticamente a declaração de pacote para você. No Code Editor do seu segundo projeto, role até o topo do código e você verá a declaração de pacote.

The screenshot shows a Java code editor with the following code:

```
1 package mysecondproject;
2
3
4 public class MySecondProject {
5
6     /**
7      * @param args the command line arguments
8      */
9     public static void main(String[] args) {
10         // TODO code application logic here
11         System.out.println("Hello again world");
12     }
13 }
14
```

The code editor has tabs for "Start Page", "HelloWorld.java", and "MySecondProject.java". The "Source" tab is selected. The "package" statement is highlighted with a red box. The code uses syntax highlighting with blue for packages, black for classes and methods, and orange for strings.

No seu primeiro projeto, como o pacote era **<default package>**, se você rolar até o topo do arquivo **.java** no Code Editor, verá que não há uma declaração de pacote.

Se você adicionar um arquivo **.java** existente a um pacote que já tenha sido nomeado, será preciso adicionar a declaração de pacote manualmente, como a primeira linha do código no arquivo **.java** usando o formato:

```
package packagename;
```