

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DO RIO GRANDE DO NORTE
CAMPUS JOÃO CÂMARA

AMBIENTE DE DESENVOLVIMENTO NETBEANS

Nickerson Fonseca Ferreira
nickerson.ferreira@ifrn.edu.br

Ambientes de Desenvolvimento

O QUE SÃO ??

QUEM SÃO ??

PRA QUE SERVEM ??

ONDE ENCONTRAMOS ??

QUAL IREMOS UTILIZAR ??

SEXTA NO GLOBO REPÓRTER!!



O que são ambientes de desenvolvimento

3

- São ferramentas que dão apoio ao desenvolvimento de software.
- Seu principal objetivo é agilizar o processo de desenvolvimento.
- Possui como principais características e ferramentas:
 - Editor
 - Compilador
 - Depurador
 - Geração de códigos
 - Testes
 - Distribuição
 - Refatoração

Quem são ??

4

- Os principais ambientes de desenvolvimento são:



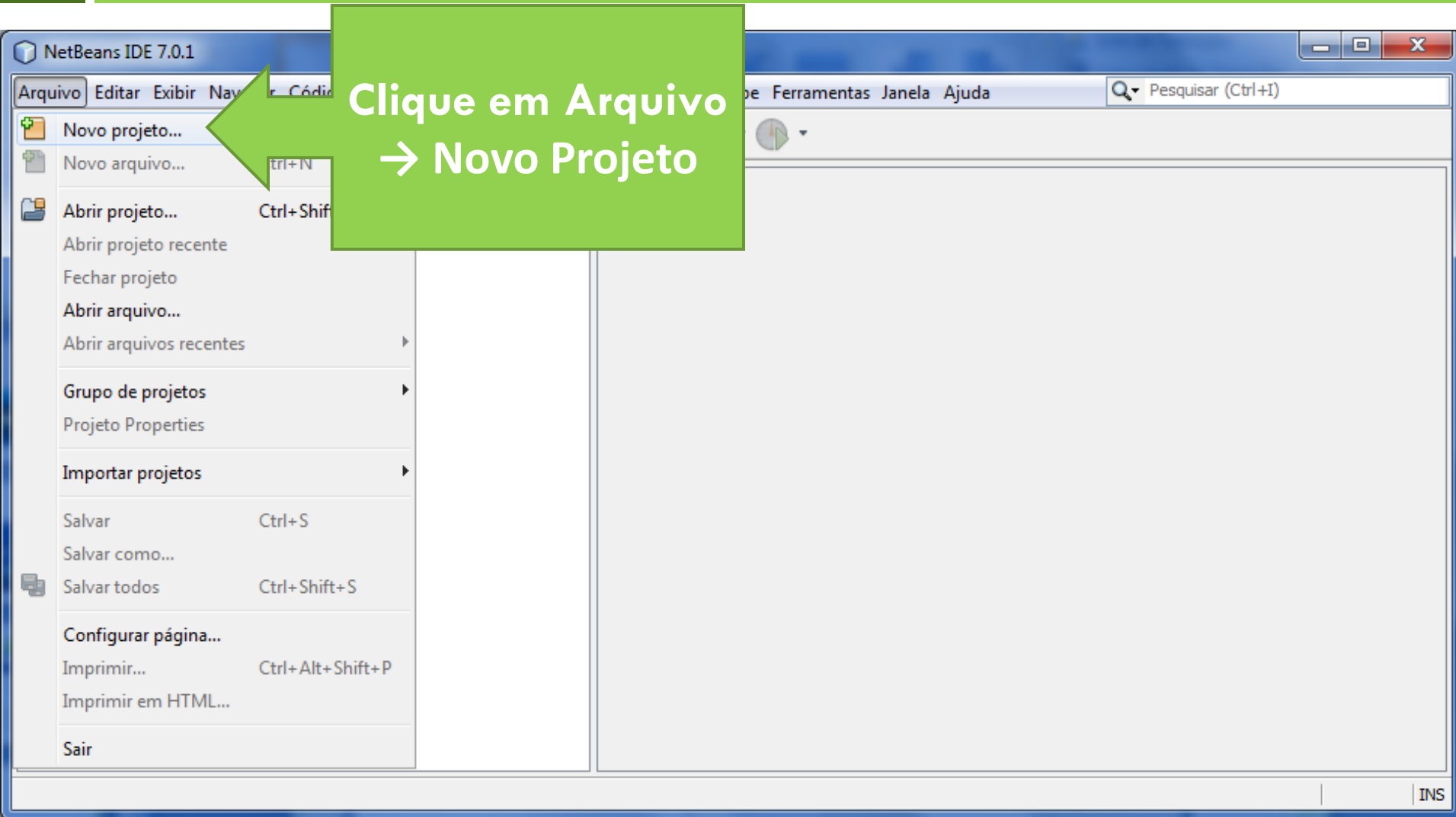
Netbeans

5

- ❑ GRATUITO!! E de código aberto.
- ❑ Multiplataforma: Windows, Linux, MacOS...
- ❑ Suporte a várias linguagens: Java, C/C++, PHP...
- ❑ Disponível em várias línguas, inclusive em português.

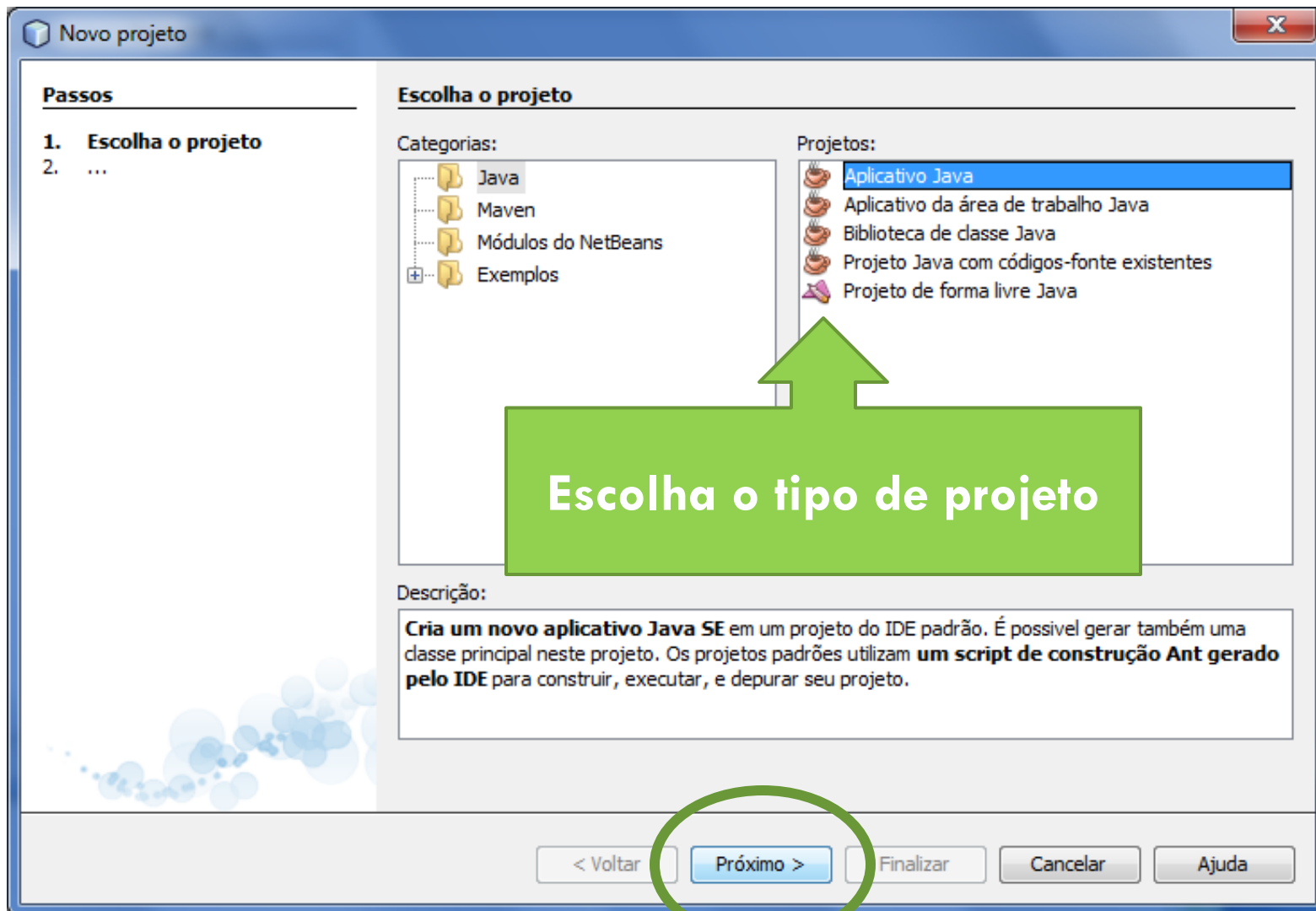
Criando um projeto

6



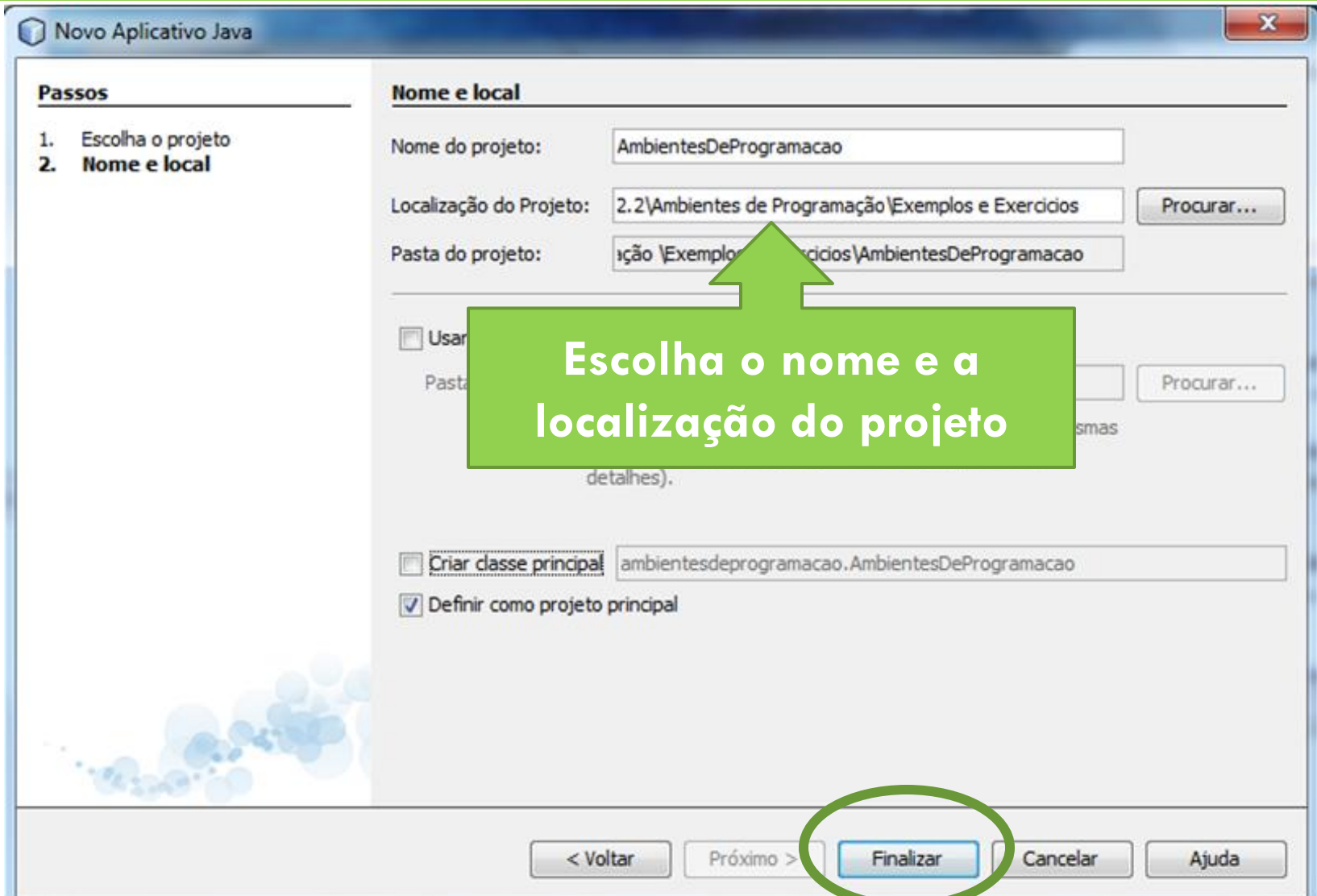
Criando um projeto

7



Criando um projeto

8



Criando um projeto

9

The image shows the NetBeans IDE 7.0.1 interface. The main window displays a Java project named 'ProjetoJava1'. The 'ProjetoJava1.java' file is open in the editor, showing the following code:

```
1 /*  
2  * To ch  
3  * and o  
4  */  
5 package  
6  
7 /**  
8  *  
9  * @author  
10 */  
11 public class  
12  
13 /**  
14  * @param a  
15  */  
16 public static void  
17 // TODO code s  
18 }  
19  
20
```

The 'Tarefas' (Tasks) window at the bottom shows a task with the description 'TODO code application logic here' located in 'ProjetoJava1.../...ava1/ProjetoJava1.java'.

A callout bubble highlights the 'AmbientesDeProgramacao' (Development Environments) settings. The settings are as follows:

Descrição	Arquivo	Localização
TODO code application logic here	ProjetoJava1...	...ava1/ProjetoJava1.java

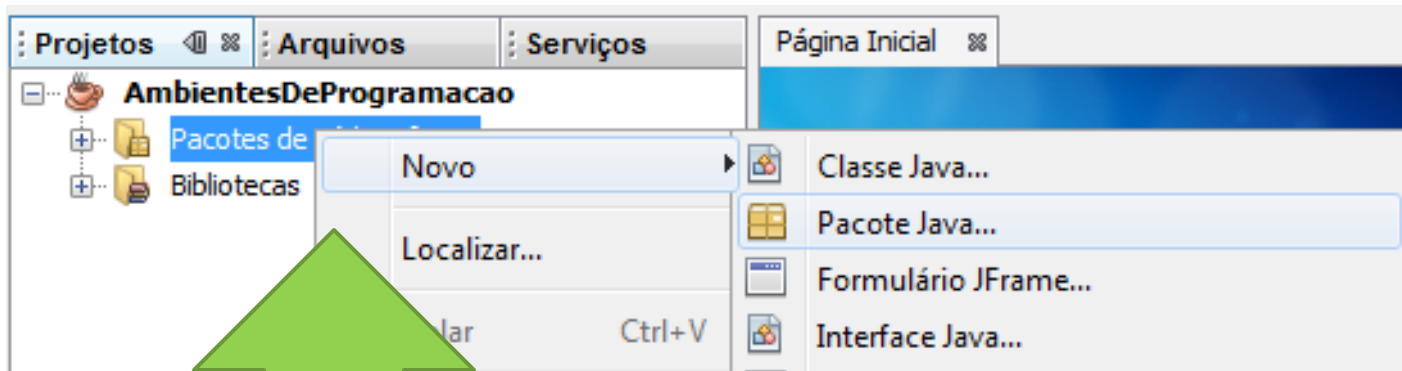
The status bar at the bottom indicates 'TODO: 1 em todos os projetos abertos' (1 TODO in all open projects).

Organização dos projetos

- ❑ Um projeto possui uma ou várias classes.
- ❑ Essas classes estão separadas em pacotes, para organizar melhor o projeto.
- ❑ Cada pacote possui classes de uma determinada finalidade. Ex: `br.edu.ifrn.gui`
- ❑ O Netbeans vai criar subpastas para cada nível do pacote. Ex: `NomeProjeto/src/br/edu/ifrn/gui`
- ❑ Por padrão, a nomenclatura dos pacotes utiliza o domínio reverso. Ex: `br.edu.ifrn.aula01`

Criando um pacote

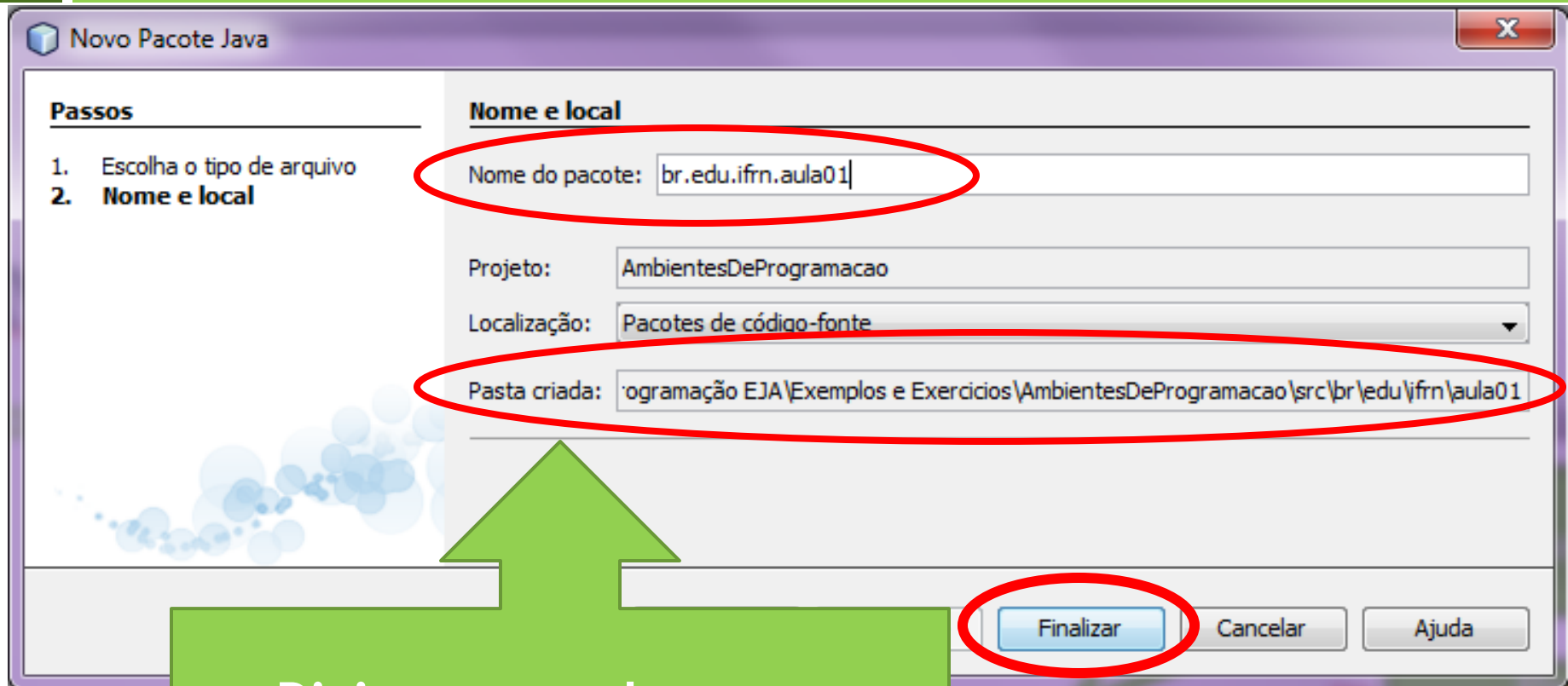
11



**Na aba Projetos, selecione
Pacotes de código-fonte
utilizando o botão direito do
mouse → Novo → Pacote Java**

Criando um pacote

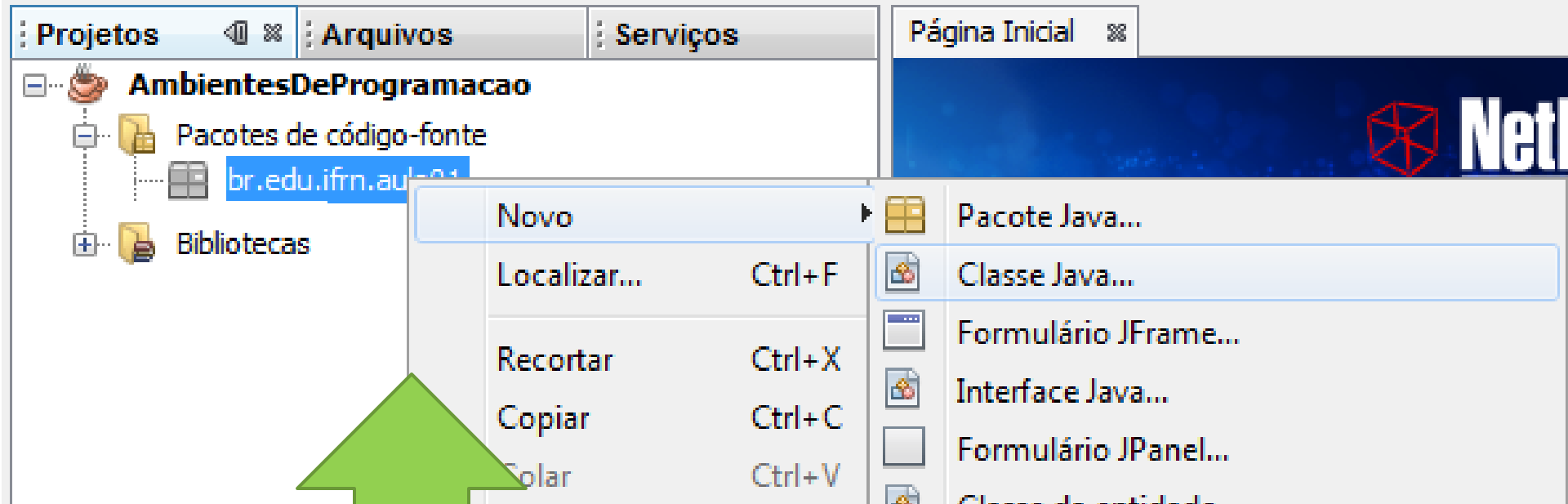
12



**Digite o nome do pacote.
Perceba a pasta que o NetBeans
cria para o seu pacote.**

Criando uma classe

13



Selecione o pacote e clique com o botão direito. Selecione Novo → Classe Java...

Criando uma classe

14

Novo Classe Java

Passos

1. Escolha o tipo de arquivo
2. **Nome e local**

Nome e local

Nome da classe: OlaMundo|

Projeto: AmbientesDeProgramacao

Localização: Pacotes de código fonte

Pacote: br.edu.ifrn.aula01

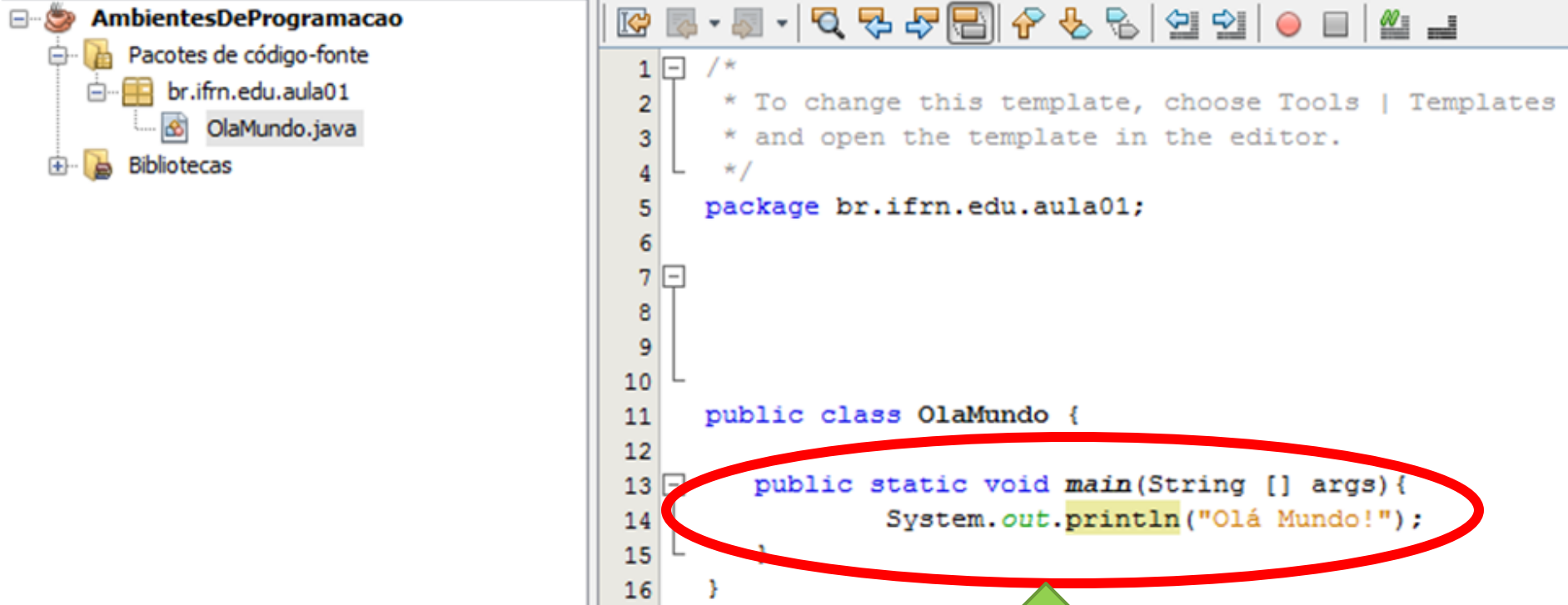
Arquivo criado: xemplos e Exercicios\AmbientesDeProgramacao\src\br\edu\ifrn\aula01\OlaMundo.java

< Voltar Próximo > **Finalizar** Cancelar Ajuda

**Digite o nome da classe.
Perceba o pacote e a pasta onde
a classe será criada.**

Executando a classe criada

15



The screenshot shows an IDE interface. On the left, a project tree under 'AmbientesDeProgramacao' contains a source code package 'br.ifrn.edu.aula01' with a file 'OlaMundo.java'. The main editor displays the following Java code:

```
1  /*
2  * To change this template, choose Tools | Templates
3  * and open the template in the editor.
4  */
5  package br.ifrn.edu.aula01;
6
7
8
9
10
11 public class OlaMundo {
12
13     public static void main(String [] args){
14         System.out.println("Olá Mundo!");
15     }
16 }
```

The code block for the `main` method (lines 13-15) is circled in red. A green arrow points from a text box below to this circled code.

Para executar uma classe Java é necessário implementar o método main.

Encapsulando atributos

16

- Uma classe possui atributos e métodos.
- A melhor forma de proteger o conteúdo dos atributos é realizando o encapsulamento.

```
public class ContaCorrente {  
    private float saldo;  
  
    public float getSaldo() {  
        return saldo;  
    }  
  
    public void setSaldo(float novoSaldo) {  
        saldo = novoSaldo;  
    }  
}
```

Encapsulando atributos

17

- Se uma classe possuir muitos atributos, a criação dos métodos get e set se torna uma tarefa demorada.

```
package teste;
```

```
/**  
 *  
 * @author Nickerson  
 */
```

```
public class Contato {
```

```
    private String nome;  
    private String telefone;  
    private String endereco;  
    private String email;
```

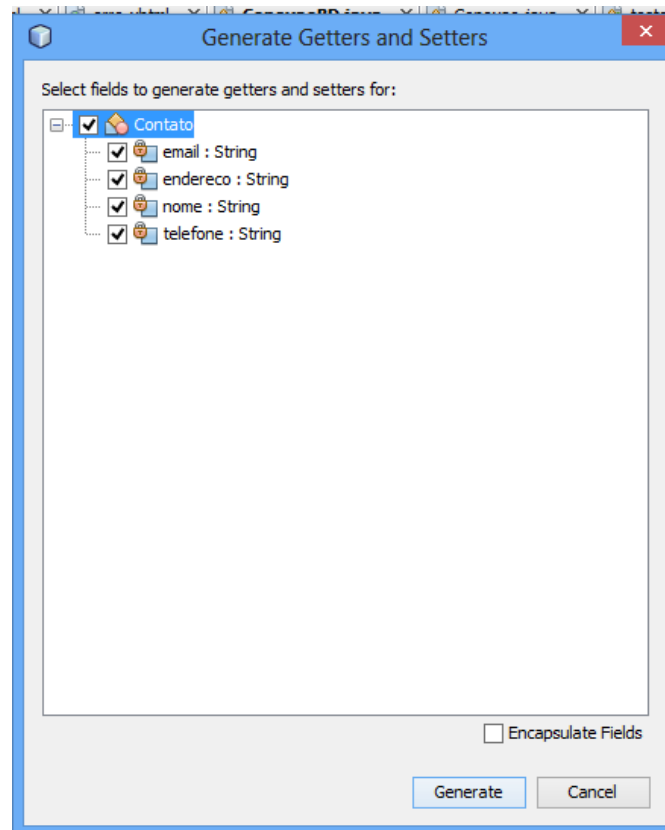
Navigate	
Show Javadoc	Alt+F1
Find Usages	Alt+F7
Call Hierarchy	
Insert Code...	Alt+Insert
Fix Imports	Ctrl+Shift+I
Refactor	
Format	Alt+Shift+F
Run File	Shift+F6
Debug File	Ctrl+Shift+F5
Test File	Ctrl+F6

Generate
Constructor...
Logger...
Getter...
Setter...
Getter and Setter...
equals() and hashCode()...
toString()...
Delegate Method...
Override Method...
Add Property...
Call Web Service Operation...
Generate REST Client...

Encapsulando atributos

18

- ❑ Selecione para quais atributos serão criados os métodos e clique em gerar.



Exercícios - I

19

- ❑ Abra o Netbeans e crie um novo projeto Java chamado Exercicios.
- ❑ Dentro do Pacote de código-fonte, crie um novo pacote: `br.edu.ifrn.aula01`
- ❑ Dentro do pacote `br.edu.ifrn.aula01` adicione uma nova classe chamada Contato.
- ❑ Adicione os atributos na classe: nome, telefone, endereço e e-mail.
- ❑ Crie os métodos get e set para todos os atributos.

Exercícios - II

- ❑ Crie uma classe Agenda dentro do pacote `br.edu.ifrn.aula01`.
- ❑ Essa classe deve ter um atributo chamado `contatos` do tipo vetor de `Contatos` que irá armazenar objetos da classe `Contato`.
- ❑ Crie um método para adicionar um `Contato` no vetor `contatos`.
- ❑ Crie outro método para buscar um contato pelo nome de um contato e que retorne o objeto `Contato` referente.