

# Lição 8



## Argumentos de linha de comando

# Objetivos

Ao final desta lição, o estudante será capaz de:

- Utilizar o argumento de linha de comando
- Receber dados enviados pelo usuário utilizando os argumentos de linha de comando
- Aprender como passar argumentos para os programas no NetBeans



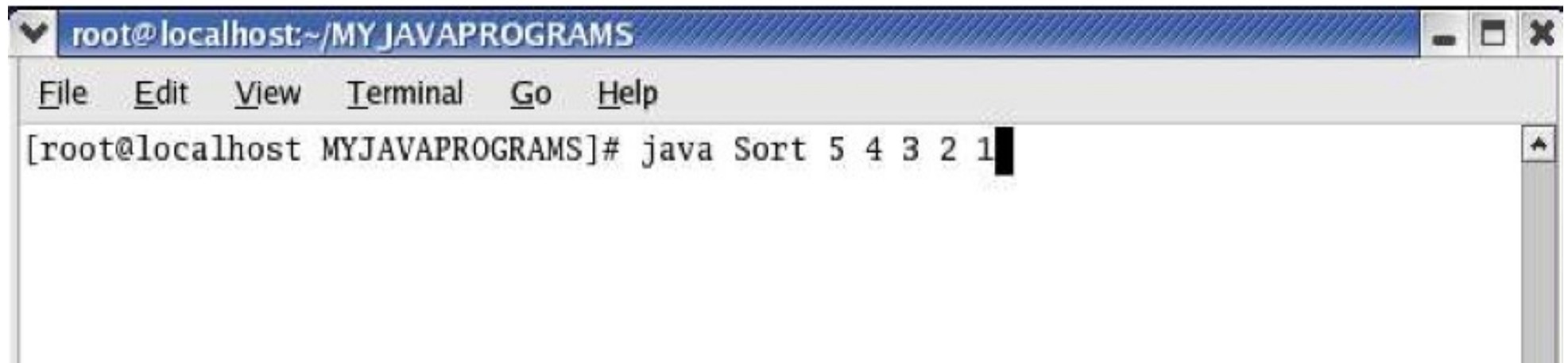
# Argumentos de linha de comando

- Uma aplicação em Java aceita qualquer quantidade de argumentos passados pela linha de comando
- Argumentos de linha de comando permitem ao usuário modificar a operação de uma aplicação a partir de sua inicialização
- O usuário digita os argumentos de linha de comando quando está solicitando a execução da aplicação



# Argumentos de linha de comando

- Suponha a existência de uma aplicação Java, chamada **Sort**, que ordena cinco números



A screenshot of a terminal window. The title bar reads 'root@localhost:~/MY\_JAVAPROGRAMS'. The menu bar includes 'File', 'Edit', 'View', 'Terminal', 'Go', and 'Help'. The command prompt shows '[root@localhost MY\_JAVAPROGRAMS]# java Sort 5 4 3 2 1' with a cursor at the end of the line.

- Os argumentos são separados por espaço



# Argumentos de linha de comando

- Na linguagem Java, quando uma aplicação é invocada, o argumento de linha de comando é repassado para o método main da aplicação por um Array de String

```
public static void main(String[] args)
```

Cada elemento do array contém um dos argumentos passados



# Argumentos de linha de comando

- Dado o exemplo anterior, onde executamos:

```
java Sort 5 4 3 2 1
```

os argumentos são armazenados no array `args`, vindo da declaração do método `main`, do seguinte modo:

`args[0] = 5`

`args[1] = 4`

`args[2] = 3`

`args[3] = 2`

`args[4] = 1`



# Argumentos de linha de comando

- Mostrar os argumentos na tela:

```
1 public class CommandLineSample {  
2     public static void main( String[] args ) {  
3         for(int i=0; i<args.length; i++){  
4             System.out.println( args[i] );  
5         }  
6     }  
7 }
```



# Argumentos de linha de comando

- Se o programa precisa manipular argumento de linha de comando numérico, então, converte-se o argumento do tipo String, assim como "34", para um número

```
int firstArg = 0;

if (args.length > 0) {
    firstArg = Integer.parseInt(args[0]);
}
```



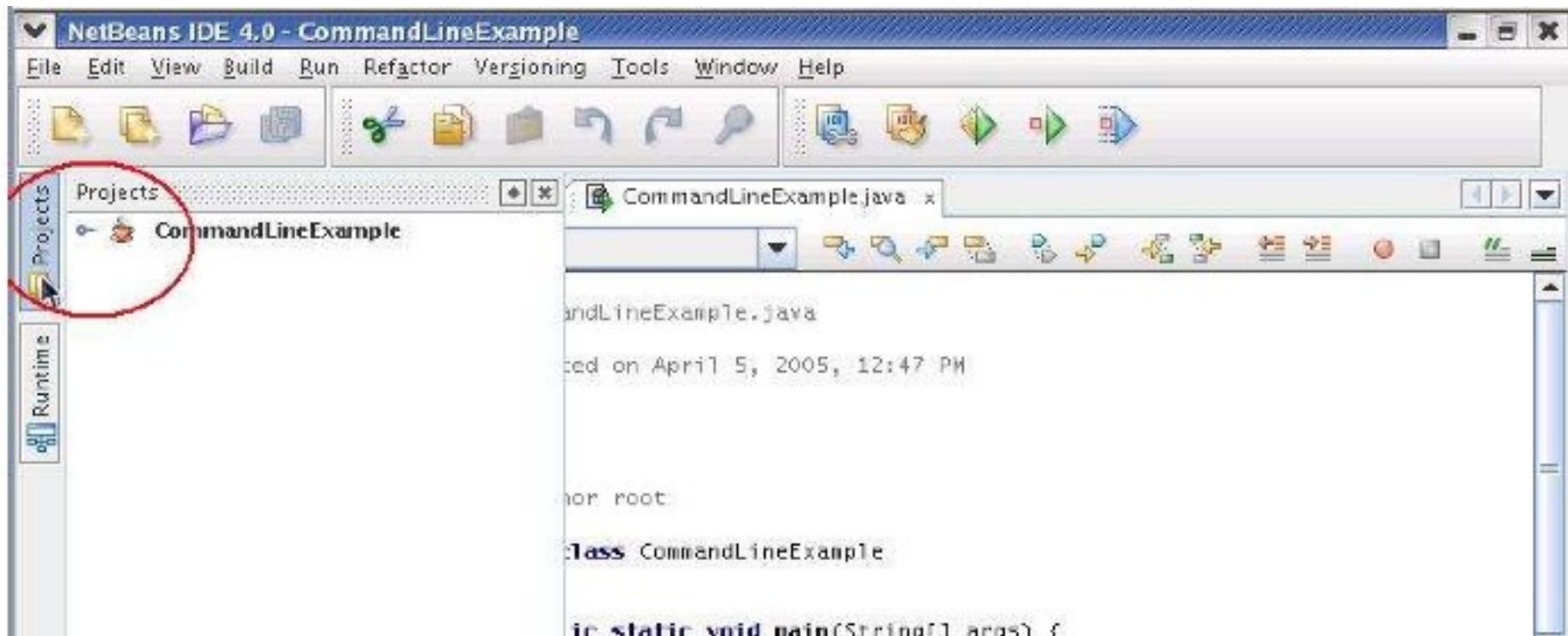


# Argumentos de linha de comando no NetBeans

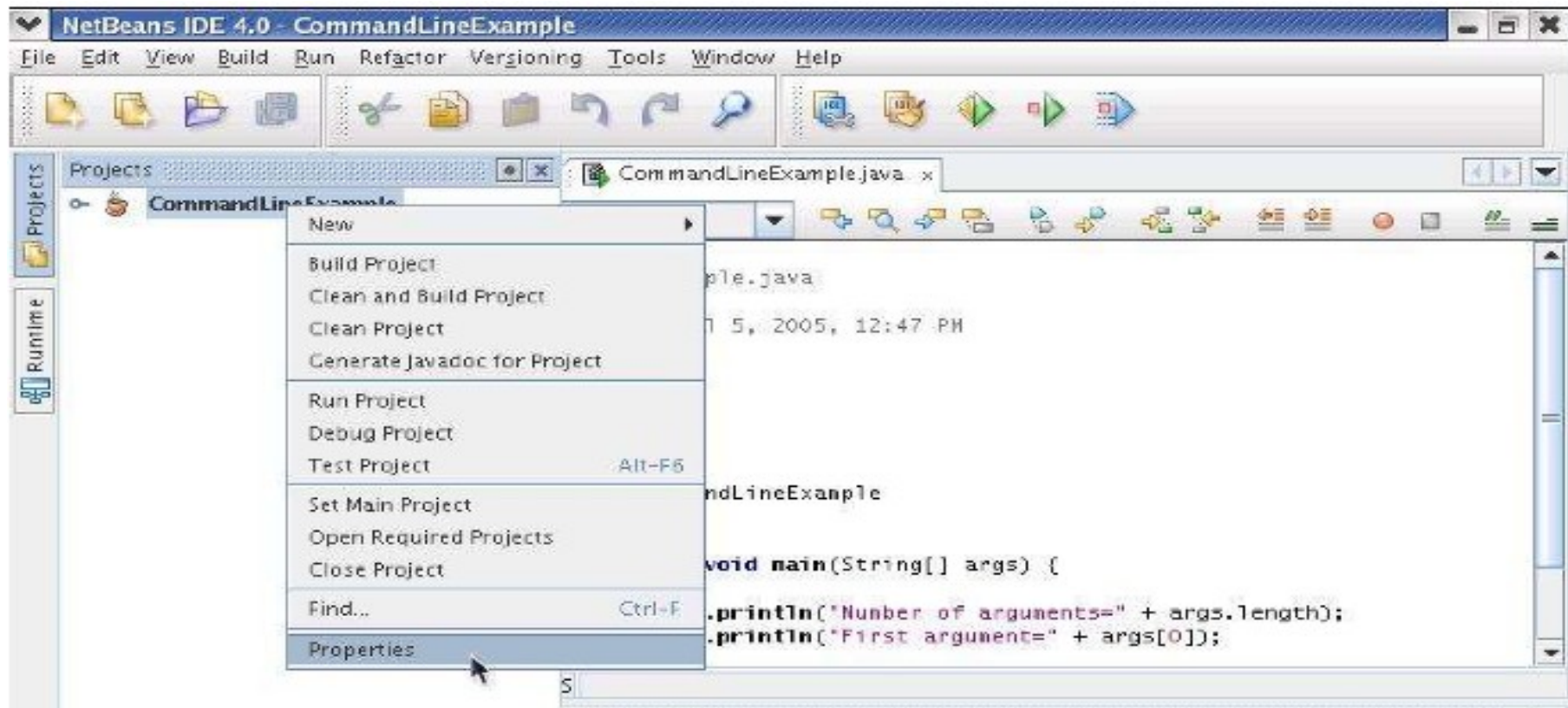
- Como podemos passar argumentos de linha de comando no NetBeans?
- Assumindo a existência de um projeto, siga os passos a seguir para passar argumentos de linha de comando no NetBeans



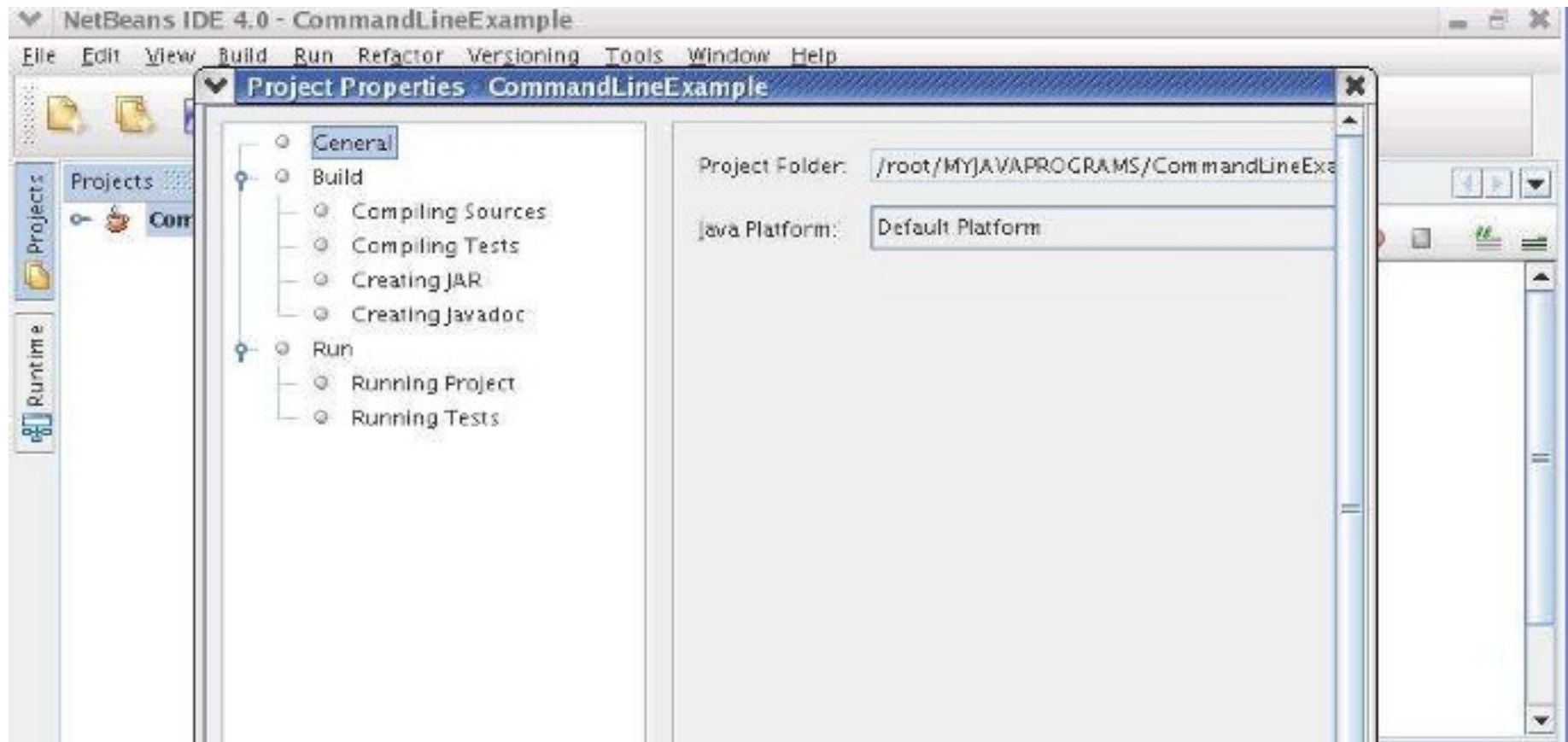
# Argumentos de linha de comando no NetBeans



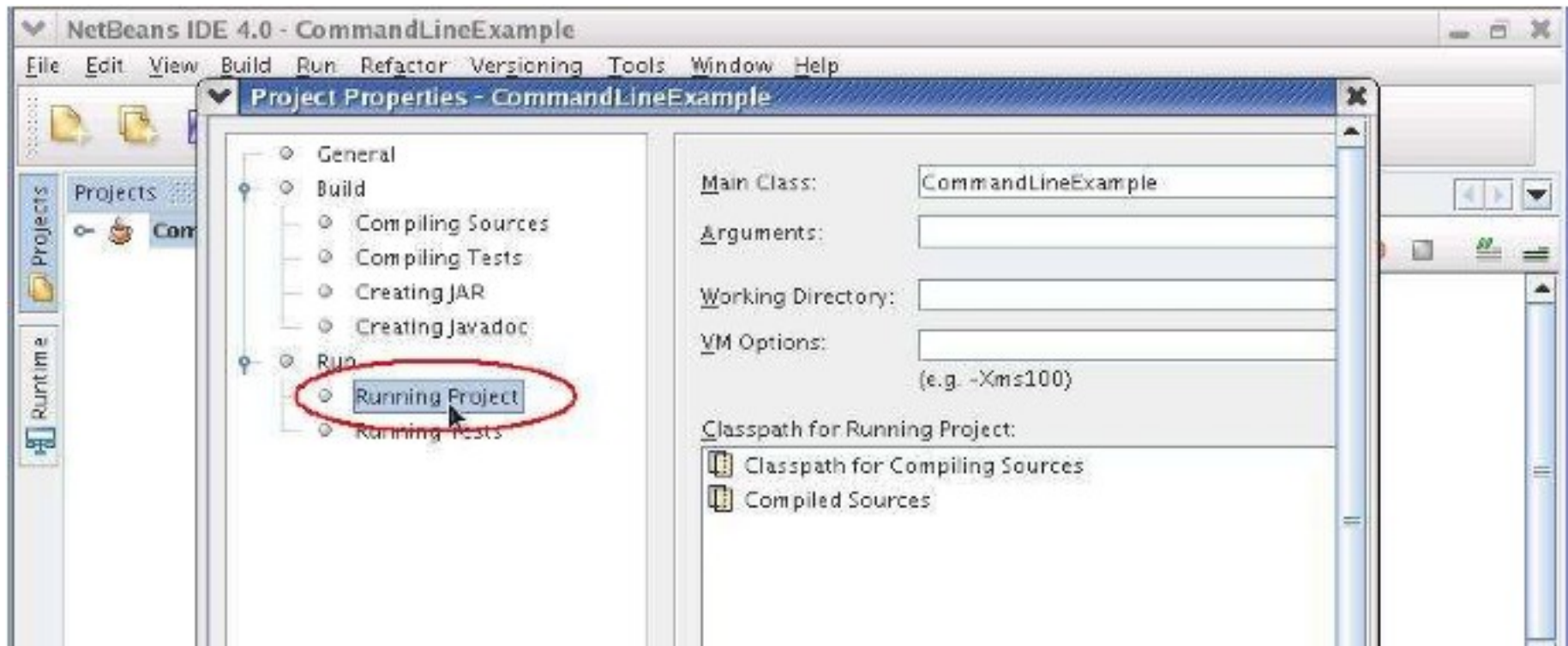
# Argumentos de linha de comando no NetBeans



# Argumentos de linha de comando no NetBeans

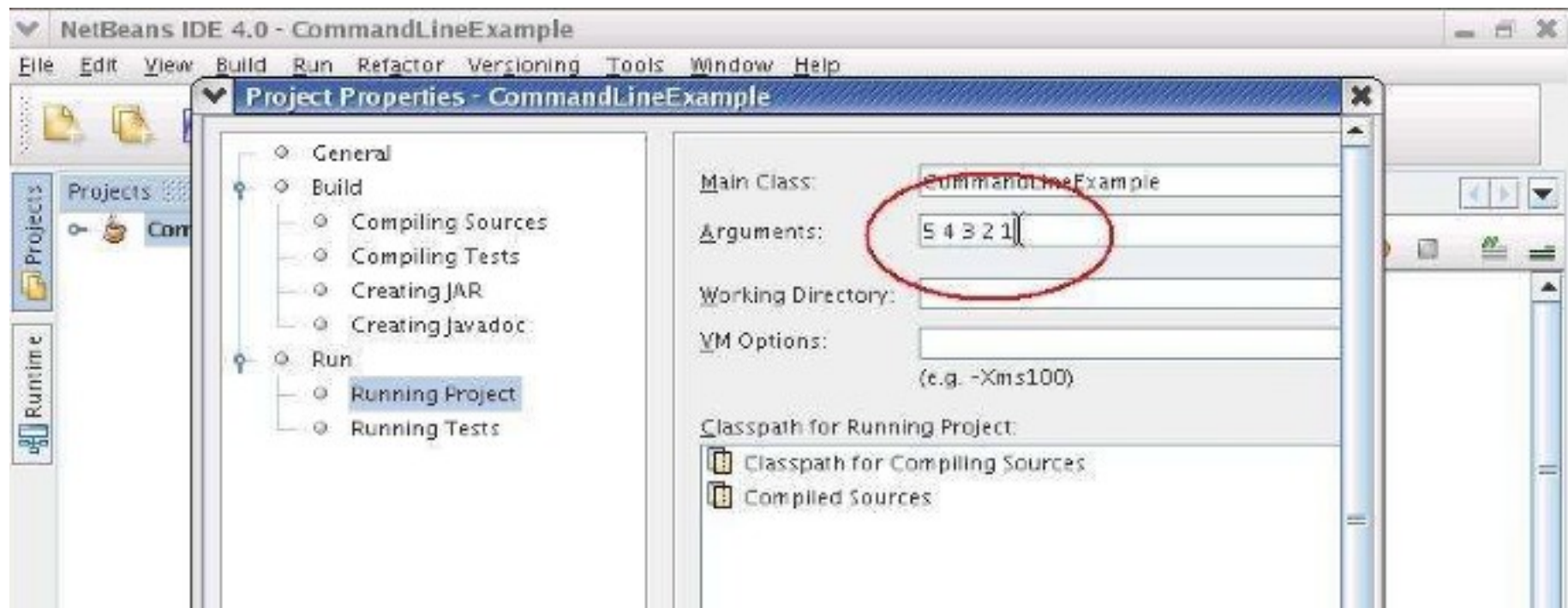


# Argumentos de linha de comando no NetBeans

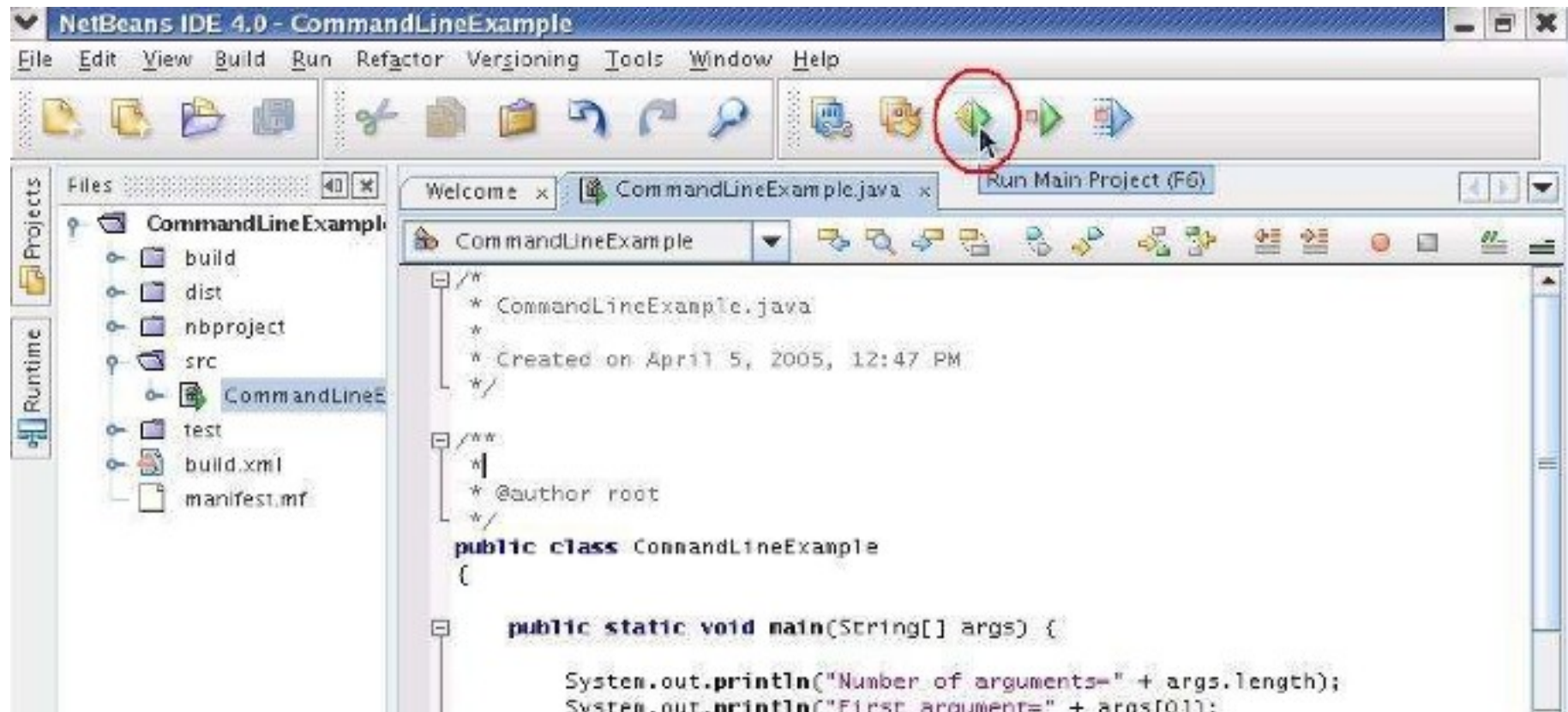




# Argumentos de linha de comando no NetBeans



# Argumentos de linha de comando no NetBeans



# Sumário

- Argumentos de linha de comando
  - Como acessar os argumentos
  - Como converter String em inteiro usando o método `Integer.parseInt`
  - Como passar argumentos de linha de comando no NetBeans



# Parceiros

- Os seguintes parceiros tornaram JEDI possível em Língua Portuguesa:

