

Módulo 1

Introdução à Programação 1



Apêndice B

**Conhecendo seu ambiente de
programação (versão Windows XP)**

Versão 1.0 - Jan/2007

Autor

Florence Tiu Balagtas

Equipe

Joyce Avestro
 Florence Balagtas
 Rommel Feria
 Reginald Hutcherson
 Rebecca Ong
 John Paul Petines
 Sang Shin
 Raghavan Srinivas
 Matthew Thompson

Necessidades para os Exercícios**Sistemas Operacionais Suportados**

NetBeans IDE 5.5 para os seguintes sistemas operacionais:

- Microsoft Windows XP Professional SP2 ou superior
- Mac OS X 10.4.5 ou superior
- Red Hat Fedora Core 3
- Solaris™ 10 Operating System (SPARC® e x86/x64 Platform Edition)

NetBeans Enterprise Pack, poderá ser executado nas seguintes plataformas:

- Microsoft Windows 2000 Professional SP4
- Solaris™ 8 OS (SPARC e x86/x64 Platform Edition) e Solaris 9 OS (SPARC e x86/x64 Platform Edition)
- Várias outras distribuições Linux

Configuração Mínima de Hardware

Nota: IDE NetBeans com resolução de tela em 1024x768 pixel

Sistema Operacional	Processador	Memória	HD Livre
Microsoft Windows	500 MHz Intel Pentium III workstation ou equivalente	512 MB	850 MB
Linux	500 MHz Intel Pentium III workstation ou equivalente	512 MB	450 MB
Solaris OS (SPARC)	UltraSPARC II 450 MHz	512 MB	450 MB
Solaris OS (x86/x64 Platform Edition)	AMD Opteron 100 Série 1.8 GHz	512 MB	450 MB
Mac OS X	PowerPC G4	512 MB	450 MB

Configuração Recomendada de Hardware

Sistema Operacional	Processador	Memória	HD Livre
Microsoft Windows	1.4 GHz Intel Pentium III workstation ou equivalente	1 GB	1 GB
Linux	1.4 GHz Intel Pentium III workstation ou equivalente	1 GB	850 MB
Solaris OS (SPARC)	UltraSPARC IIIi 1 GHz	1 GB	850 MB
Solaris OS (x86/x64 Platform Edition)	AMD Opteron 100 Series 1.8 GHz	1 GB	850 MB
Mac OS X	PowerPC G5	1 GB	850 MB

Requerimentos de Software

NetBeans Enterprise Pack 5.5 executando sobre Java 2 Platform Standard Edition Development Kit 5.0 ou superior (JDK 5.0, versão 1.5.0_01 ou superior), contemplando a Java Runtime Environment, ferramentas de desenvolvimento para compilar, depurar, e executar aplicações escritas em linguagem Java. Sun Java System Application Server Platform Edition 9.

- Para **Solaris**, **Windows**, e **Linux**, os arquivos da JDK podem ser obtidos para sua plataforma em <http://java.sun.com/j2se/1.5.0/download.html>
- Para **Mac OS X**, Java 2 Platform Standard Edition (J2SE) 5.0 Release 4, pode ser obtida diretamente da Apple's Developer Connection, no endereço: <http://developer.apple.com/java> (é necessário registrar o download da JDK).

Para mais informações:

<http://www.netbeans.org/community/releases/55/relnotes.html>

Colaboradores que auxiliaram no processo de tradução e revisão

Alexandre Mori	Hugo Leonardo Malheiros Ferreira	Mauro Regis de Sousa Lima
Alexis da Rocha Silva	Ivan Nascimento Fonseca	Namor de Sá e Silva
Aline Sabbatini da Silva Alves	Jacqueline Susann Barbosa	Néres Chaves Rebouças
Allan Wojcik da Silva	Jader de Carvalho Belarmino	Nolyanne Peixoto Brasil Vieira
André Luiz Moreira	João Aurélio Telles da Rocha	Paulo Afonso Corrêa
Andro Márcio Correa Louredo	João Paulo Cirino Silva de Novais	Paulo José Lemos Costa
Antoniele de Assis Lima	João Vianney Barrozo Costa	Paulo Oliveira Sampaio Reis
Antonio Jose R. Alves Ramos	José Augusto Martins Nieviadonski	Pedro Antonio Pereira Miranda
Aurélio Soares Neto	José Leonardo Borges de Melo	Pedro Henrique Pereira de Andrade
Bruno da Silva Bonfim	José Ricardo Carneiro	Renato Alves Félix
Bruno dos Santos Miranda	Kleberth Bezerra G. dos Santos	Renato Barbosa da Silva
Bruno Ferreira Rodrigues	Lafaiete de Sá Guimarães	Reyderson Magela dos Reis
Carlos Alberto Vitorino de Almeida	Leandro Silva de Moraes	Ricardo Ferreira Rodrigues
Carlos Alexandre de Sene	Leonardo Leopoldo do Nascimento	Ricardo Ulrich Bomfim
Carlos André Noronha de Sousa	Leonardo Pereira dos Santos	Robson de Oliveira Cunha
Carlos Eduardo Veras Neves	Leonardo Rangel de Melo Filardi	Rodrigo Pereira Machado
Cleber Ferreira de Sousa	Lucas Mauricio Castro e Martins	Rodrigo Rosa Miranda Corrêa
Cleyton Artur Soares Urani	Luciana Rocha de Oliveira	Rodrigo Vaez
Cristiano Borges Ferreira	Luís Carlos André	Ronie Dotzlaw
Cristiano de Siqueira Pires	Luís Octávio Jorge V. Lima	Rosely Moreira de Jesus
Derlon Vandri Aliendres	Luiz Fernandes de Oliveira Junior	Seire Pareja
Fabiano Eduardo de Oliveira	Luiz Victor de Andrade Lima	Sergio Pomerancblum
Fábio Bombonato	Manoel Cotts de Queiroz	Silvio Sznifer
Fernando Antonio Mota Trinta	Marcello Sandi Pinheiro	Suzana da Costa Oliveira
Flávio Alves Gomes	Marcelo Ortolan Pazzetto	Tásio Vasconcelos da Silveira
Francisco das Chagas	Marco Aurélio Martins Bessa	Thiago Magela Rodrigues Dias
Francisco Marcio da Silva	Marcos Vinicius de Toledo	Tiago Gimenez Ribeiro
Gilson Moreno Costa	Maria Carolina Ferreira da Silva	Vanderlei Carvalho Rodrigues Pinto
Givailson de Souza Neves	Massimiliano Girolodi	Vanessa dos Santos Almeida
Gustavo Henrique Castellano	Mauricio Azevedo Gamarra	Vastí Mendes da Silva Rocha
Hebert Julio Gonçalves de Paula	Mauricio da Silva Marinho	Wagner Eliezer Roncoletta
Heraldo Conceição Domingues	Mauro Cardoso Mortoni	

Auxiliadores especiais

Revisão Geral do texto para os seguintes Países:

- **Brasil** – Tiago Flach
- **Guiné Bissau** – Alfredo Cá, Bunene Sisse e Buon Olossato Quebi – ONG Asas de Socorro

Coordenação do DFJUG

- **Daniel deOliveira** – JUGLeader responsável pelos acordos de parcerias
- **Luci Campos** - Idealizadora do DFJUG responsável pelo apoio social
- **Fernando Anselmo** - Coordenador responsável pelo processo de tradução e revisão, disponibilização dos materiais e inserção de novos módulos
- **Regina Mariani** - Coordenadora responsável pela parte jurídica
- **Rodrigo Nunes** - Coordenador responsável pela parte multimídia
- **Sérgio Gomes Veloso** - Coordenador responsável pelo ambiente JEDI™ (Moodle)

Agradecimento Especial

John Paul Petines – Criador da Iniciativa JEDI™

Rommel Faria – Criador da Iniciativa JEDI™

1. Objetivos

Neste apêndice, discutiremos como escrever, compilar e executar programas em Java. Existem duas maneiras para se fazer isto: a primeira é utilizando uma console e um editor de texto; e a segunda é utilizando o NetBeans, que é um **Integrated Development Environment (Ambiente Integrado de Desenvolvimento)** ou **IDE**.

Um IDE é um programa que contempla um **construtor de interface gráfica** (GUI builder), um editor de código ou texto, um compilador e um depurador de erros.

2. Primeiro Programa em Java

Antes de entrarmos em detalhes, vejamos o programa em Java que escreveremos:

```
public class Hello {  
    /**  
     * Meu primeiro programa em Java  
     */  
    public static void main(String[] args) {  
        //imprime o texto "Hello world" na tela  
        System.out.println("Hello world!");  
    }  
}
```

3. Utilizando um Editor de Texto e uma Console

Para este exemplo, utilizaremos o editor de texto "Notepad" (Bloco de notas do Windows) para editar o código-fonte em Java. Podem ser utilizados quaisquer outros editores de texto. Também será necessário abrir uma janela de console para compilar e executar o programa.

Passo 1: Para iniciar o Notepad, clique em Iniciar ⇒ Todos os Programas ⇒ Acessórios ⇒ Notepad.

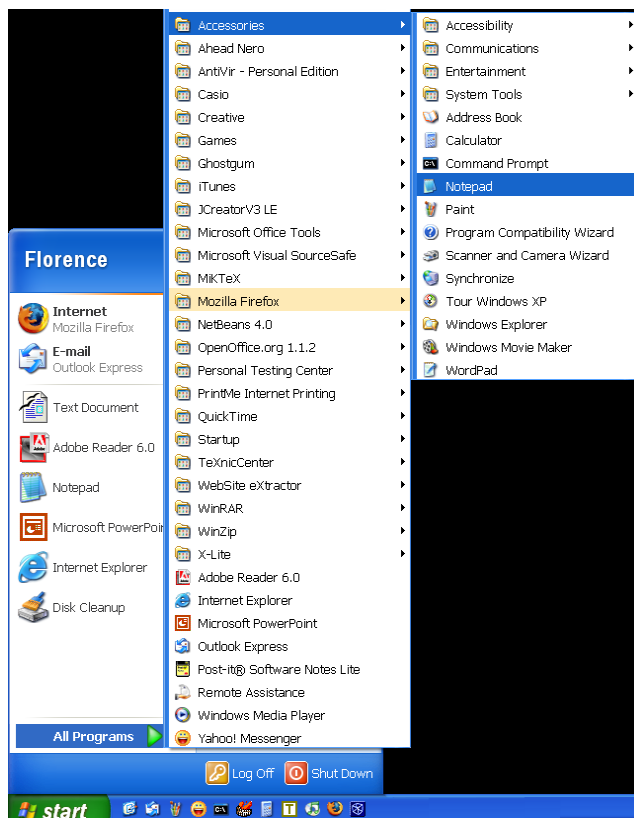


Figura 1: Clique em Start ⇒ All Programs ⇒ Accessories ⇒ Notepad

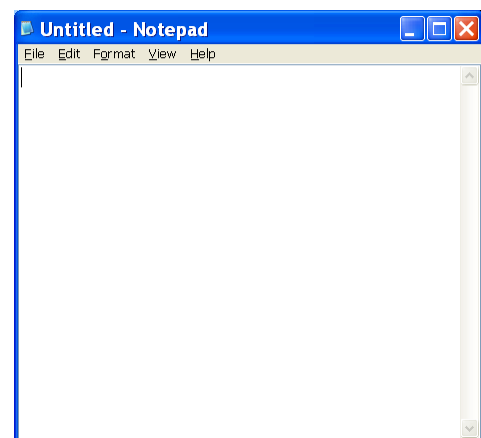


Figura 2: Programa Notepad

Passo 2: Para abrir uma janela de console, clique em Start ⇒ Programs ⇒ Accessories ⇒ Comand Prompt.

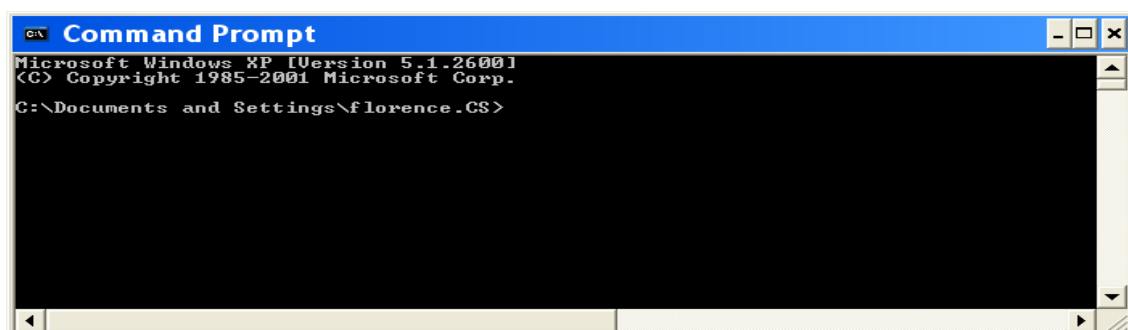
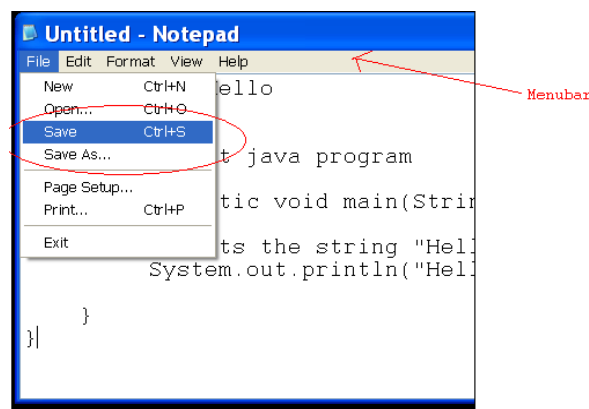


Figura 3: Janela de Console do Windows

Passo 3: Escreva o código-fonte, descrito na seção 2, no editor:

```
public class Hello
{
    /**
     * My first java program
     */
    public static void main(String[] args) {
        //prints the string "Hello world" on screen
        System.out.println("Hello world!");
    }
}
```

Passo 4: Salve o programa em um arquivo chamado "Hello.java" dentro de uma pasta chamada MYJAVAPROGRAMS. Selecione no menu a opção **File** na barra de menus e então selecione a opção **Save**.



Depois de executar o procedimento acima descrito, uma caixa de diálogo será mostrada, conforme a Figura 4.

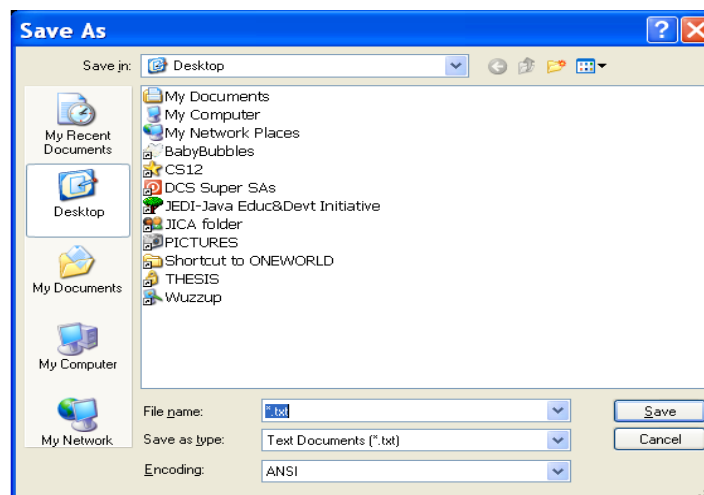


Figura 4: Esta caixa de diálogo aparece depois de clicar em File ⇒ Save

Pressione o botão **MY DOCUMENTS** para abrir a pasta My Documents onde salvaremos nossos programas em Java.

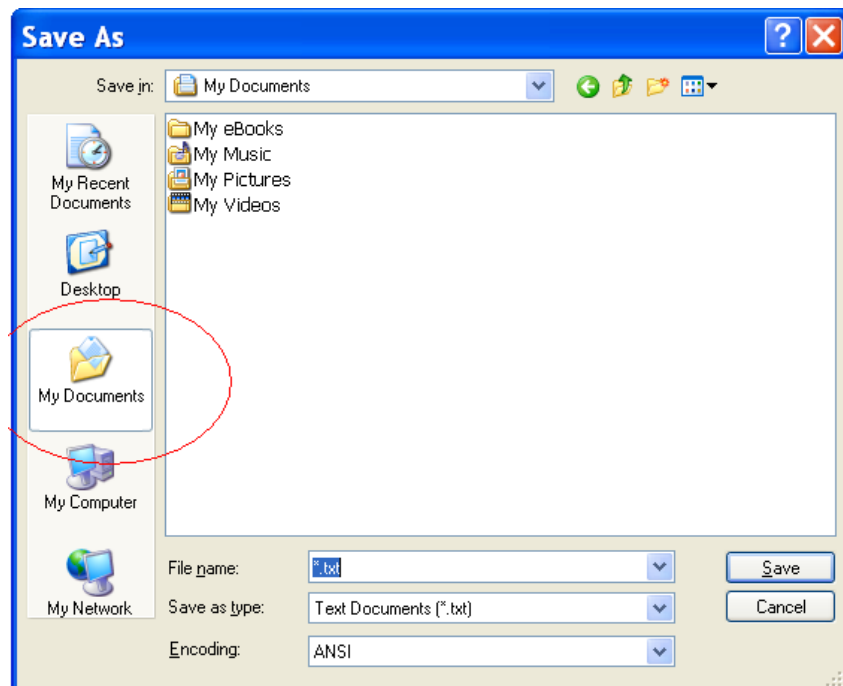


Figura 5: Selecione no botão circulado.

Criaremos uma nova pasta, dentro da pasta My Documents, onde salvaremos nossos programas. Iremos nomeá-la como MYJAVAPROGRAMS. Pressione o botão circulado conforme mostrado na Figura 6 para criar a pasta.

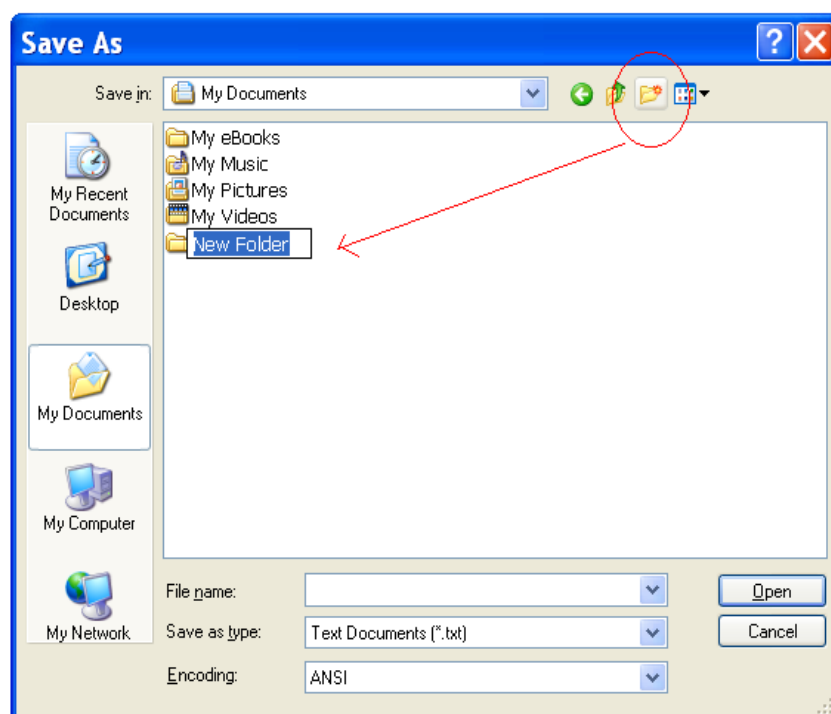
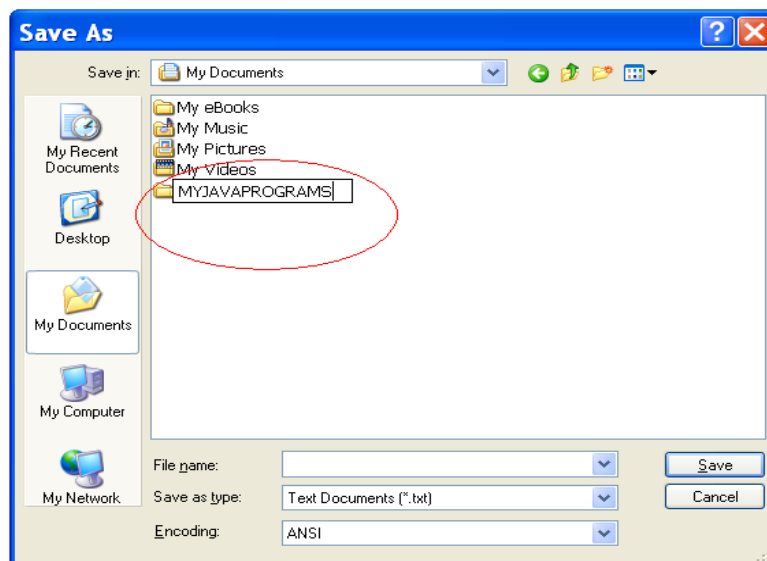
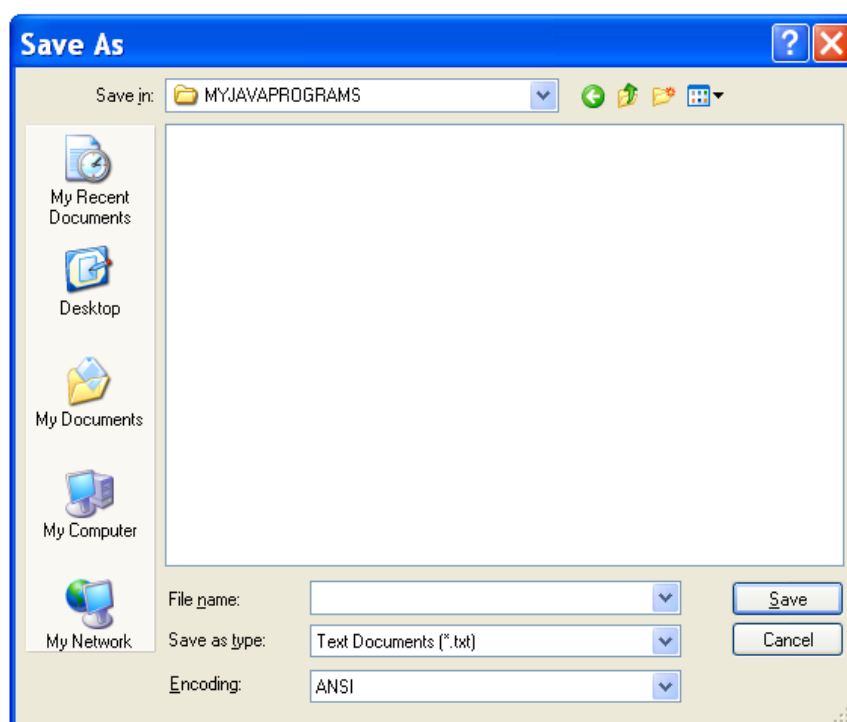


Figura 6: Clicar no botão circulado criará uma nova pasta.

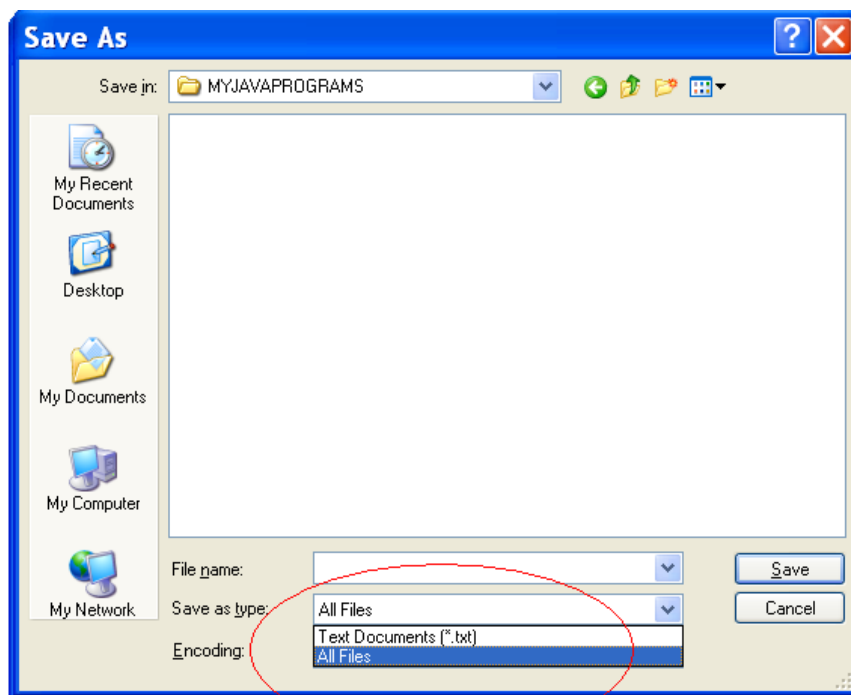
Após criar a pasta, digite o nome MYJAVAPROGRAMS, e então pressione a tecla ENTER.



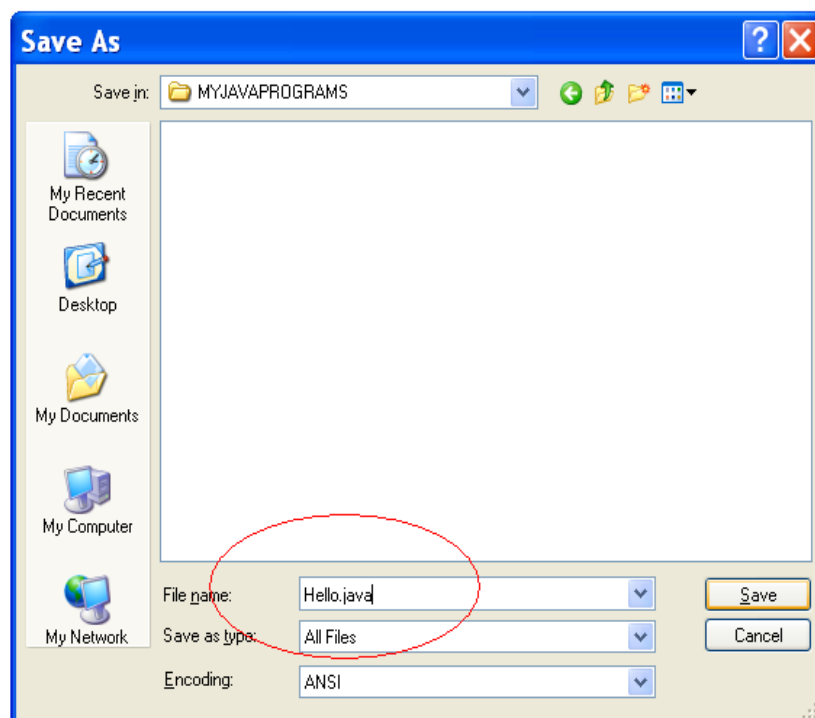
Criamos a pasta aonde salvaremos os arquivos, dê um duplo-clique nesta para abrí-la. Será mostrada uma janela como a figura seguinte. A pasta deverá estar vazia por ora, pois foi recém-criada e ainda não foi salvo nenhum arquivo.



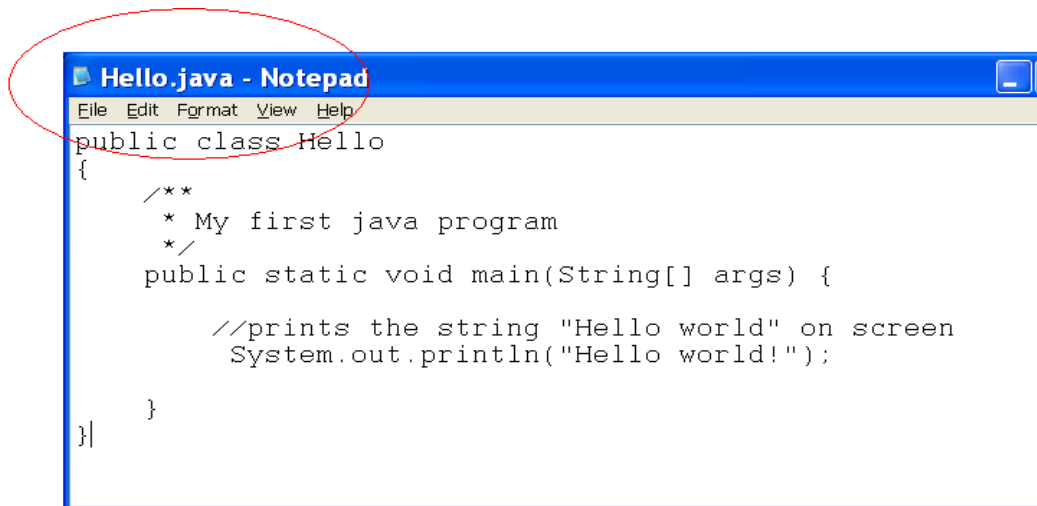
Selecione a caixa de seleção **Save as type**, a fim de que possamos escolher o tipo de arquivo que queremos salvar. Marque a opção **All Files**.



Digite "Hello.java" como nome do seu programa na caixa de texto **Filename**, e então pressione o botão **Save**.



Observe que o título da janela mudou de **Untitled - Notepad** para **Hello.java - Notepad**. Para efetuar alterações no arquivo, edite-o, e então salve-o novamente através das opções File ⇒ Save.

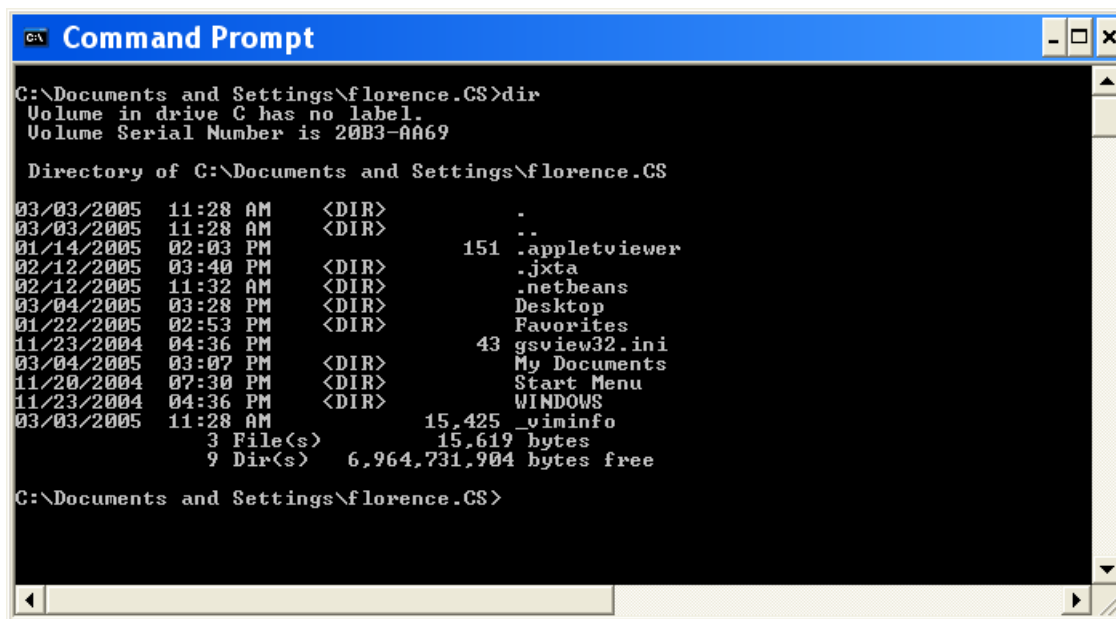


```

Hello.java - Notepad
File Edit Format View Help
public class Hello
{
    /**
     * My first java program
     */
    public static void main(String[] args) {
        //prints the string "Hello world" on screen
        System.out.println("Hello world!");
    }
}

```

Passo 5: Para compilar o programa, utilizaremos a janela de console aberta no **passo 2**. Normalmente, esta janela é iniciada no que chamamos de **home folder (pasta do usuário)**. Para ver uma lista do conteúdo desta pasta, digite **DIR** e pressione **Enter**. Será mostrada uma lista de arquivos e pastas que estão dentro desta.



```

Command Prompt
C:\Documents and Settings\florence.CS>dir
Volume in drive C has no label.
Volume Serial Number is 20B3-AA69

Directory of C:\Documents and Settings\florence.CS

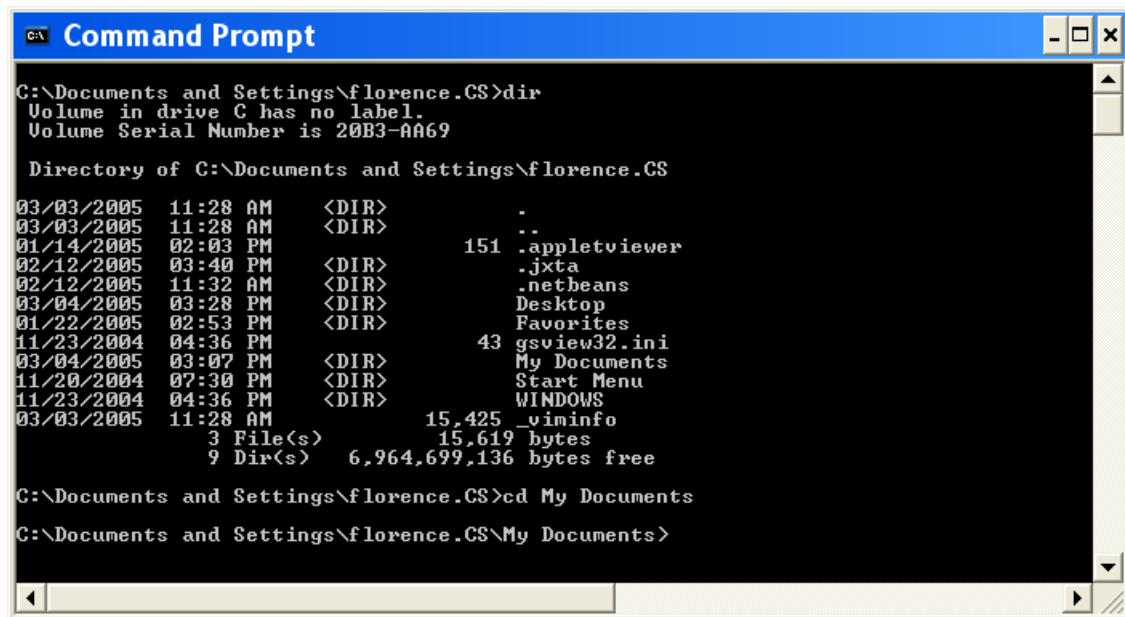
03/03/2005  11:28 AM    <DIR>        -
03/03/2005  11:28 AM    <DIR>        ..
01/14/2005  02:03 PM             151  .appletviewer
02/12/2005  03:40 PM    <DIR>        .jxta
02/12/2005  11:32 AM    <DIR>        .netbeans
03/04/2005  03:28 PM    <DIR>        Desktop
01/22/2005  02:53 PM    <DIR>        Favorites
11/23/2004  04:36 PM             43  gsview32.ini
03/04/2005  03:07 PM    <DIR>        My Documents
11/20/2004  07:30 PM    <DIR>        Start Menu
11/23/2004  04:36 PM    <DIR>        WINDOWS
03/03/2005  11:28 AM             15,425  _viminfo
               3 File(s)              15,619 bytes
               9 Dir(s)  6,964,731,904 bytes free

C:\Documents and Settings\florence.CS>

```

Figura 7: Lista de arquivos e pastas mostrados depois de executar o comando DIR.

Existe uma pasta chamada "My Documents" onde criamos a pasta MYJAVAPROGRAMS. Iremos para este diretório. Para entrar em um diretório, digite o comando: **cd [nome do diretório]**. O comando "cd" significa "Change Directory". Neste caso, já que o nome de nosso diretório é My Documents, digite: **cd My Documents** e pressione **Enter**.



```

C:\Documents and Settings\florence.CS>dir
Volume in drive C has no label.
Volume Serial Number is 20B3-AA69

Directory of C:\Documents and Settings\florence.CS

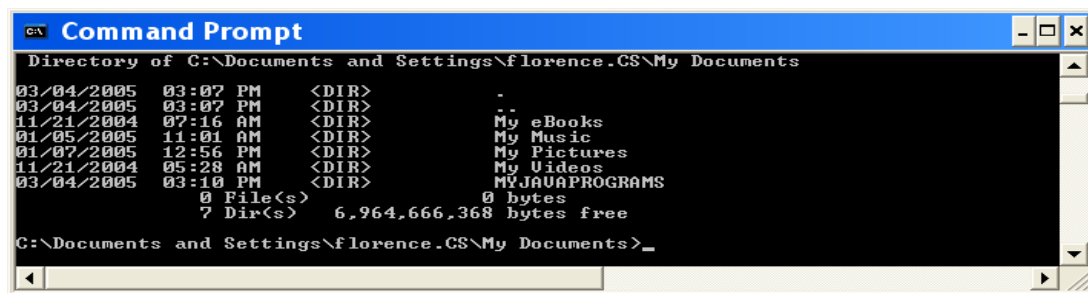
03/03/2005  11:28 AM  <DIR>      .
03/03/2005  11:28 AM  <DIR>      ..
01/14/2005  02:03 PM             151  .appletviewer
02/12/2005  03:40 PM             .jxta
02/12/2005  11:32 AM             <DIR>      .netbeans
03/04/2005  03:28 PM             <DIR>      Desktop
01/22/2005  02:53 PM             <DIR>      Favorites
11/23/2004  04:36 PM             43  gsvie32.ini
03/04/2005  03:07 PM             <DIR>      My Documents
11/20/2004  07:30 PM             <DIR>      Start Menu
11/23/2004  04:36 PM             <DIR>      WINDOWS
03/03/2005  11:28 AM      15,425  _vminfo
          3 File(s)         15,619 bytes
          9 Dir(s)    6,964,699,136 bytes free

C:\Documents and Settings\florence.CS>cd My Documents
C:\Documents and Settings\florence.CS\My Documents>

```

Figura 8: Dentro da pasta My Documents

Estamos dentro da pasta "My Documents", digite o comando "dir" novamente.



```

C:\Documents and Settings\florence.CS\My Documents>dir
Directory of C:\Documents and Settings\florence.CS\My Documents

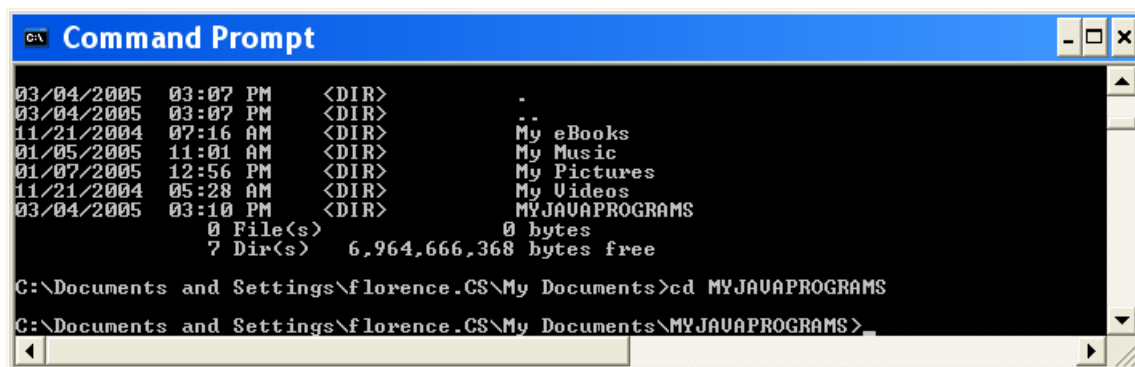
03/04/2005  03:07 PM  <DIR>      .
03/04/2005  03:07 PM  <DIR>      ..
11/21/2004  07:16 AM  <DIR>      My eBooks
01/05/2005  11:01 AM  <DIR>      My Music
01/07/2005  12:56 PM  <DIR>      My Pictures
11/21/2004  05:28 AM  <DIR>      My Videos
03/04/2005  03:10 PM  <DIR>      MYJAVAPROGRAMS
          0 File(s)              0 bytes
          7 Dir(s)    6,964,666,368 bytes free

C:\Documents and Settings\florence.CS\My Documents>_

```

Figura 9: O conteúdo de My Documents

Execute os mesmos passos descritos para entrar na pasta MYJAVAPROGRAMS.



```

C:\Documents and Settings\florence.CS\My Documents>dir
Directory of C:\Documents and Settings\florence.CS\My Documents

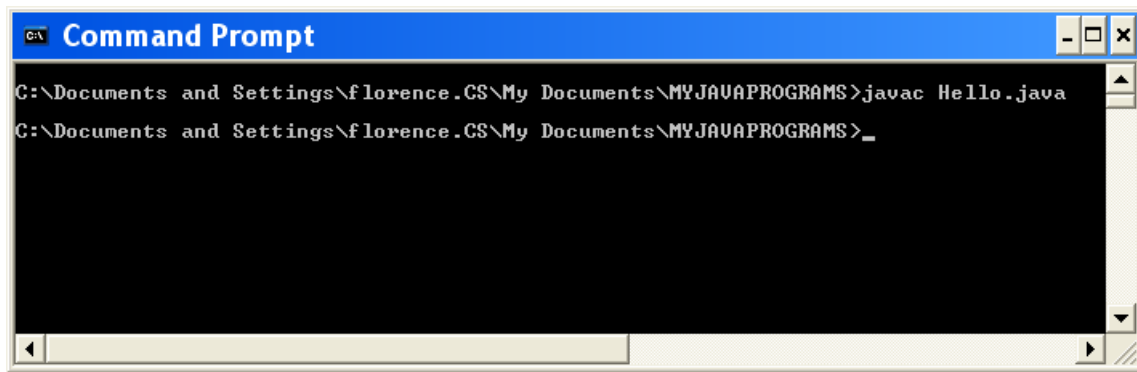
03/04/2005  03:07 PM  <DIR>      .
03/04/2005  03:07 PM  <DIR>      ..
11/21/2004  07:16 AM  <DIR>      My eBooks
01/05/2005  11:01 AM  <DIR>      My Music
01/07/2005  12:56 PM  <DIR>      My Pictures
11/21/2004  05:28 AM  <DIR>      My Videos
03/04/2005  03:10 PM  <DIR>      MYJAVAPROGRAMS
          0 File(s)              0 bytes
          7 Dir(s)    6,964,666,368 bytes free

C:\Documents and Settings\florence.CS\My Documents>cd MYJAVAPROGRAMS
C:\Documents and Settings\florence.CS\My Documents\MYJAVAPROGRAMS>_

```

Figura 10: Dentro da pasta MYJAVAPROGRAMS

Um vez nesta pasta onde seus programas estão localizados, iremos compilar o programa. Certifique-se de assegurar de que o arquivo está localizado nessa pasta. Para isso, execute o comando **dir** e verifique que realmente o arquivo se encontra nesta pasta. Para compilar um programa em Java, digitamos o comando: **javac [nomedoarquivo]**. Deste modo, digitamos: **javac Hello.java**.



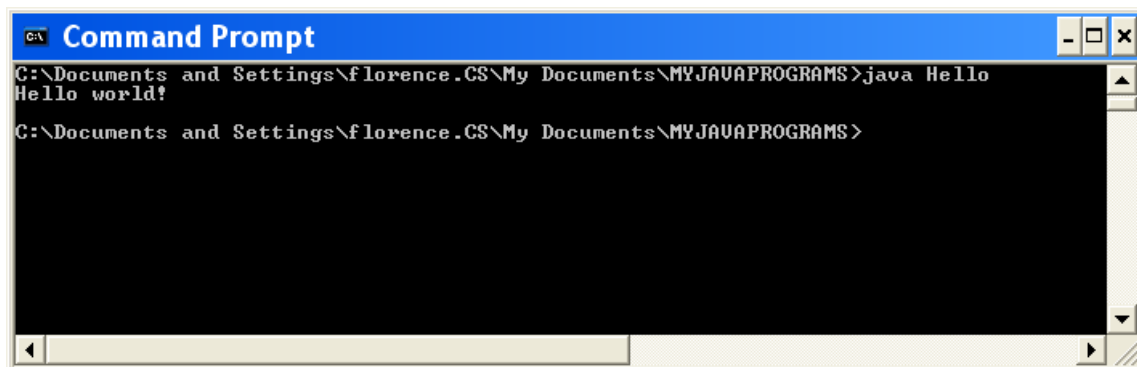
```
C:\Documents and Settings\florence.CS\My Documents\MYJAVAPROGRAMS>javac Hello.java
C:\Documents and Settings\florence.CS\My Documents\MYJAVAPROGRAMS>_
```

Figura 11: Compilar um programa utilizando comando **javac**

Durante o processo de compilação, o compilador **javac** cria um arquivo chamado **[nomedoarquivo].class**, neste caso, **Hello.class**, sendo o arquivo em linguagem de bytecode.

Passo 6: Para executar o programa, considerando que não existam erros durante a compilação, digite o comando: **java [nomedoarquivo]**, para nosso exemplo, digite: **java Hello**

Veja que na tela que executamos o programa em Java é mostrada a mensagem, "Hello world!".



```
C:\Documents and Settings\florence.CS\My Documents\MYJAVAPROGRAMS>java Hello
Hello world!
C:\Documents and Settings\florence.CS\My Documents\MYJAVAPROGRAMS>
```

Figura 12: Saída do programa

4. Configurando o caminho (Path)

Ao tentar executar o comando **javac** ou **java**, é mostrada a mensagem: **'javac' is not recognized as an internal or external command, operable program or batch file.** Isto significa que ou ainda não foi instalado o Java em seu sistema, ou é necessário configurar o caminho de pesquisa (path) onde os aplicativos de Java estão instalados. Dessa forma, seu sistema saberá aonde procurá-los.

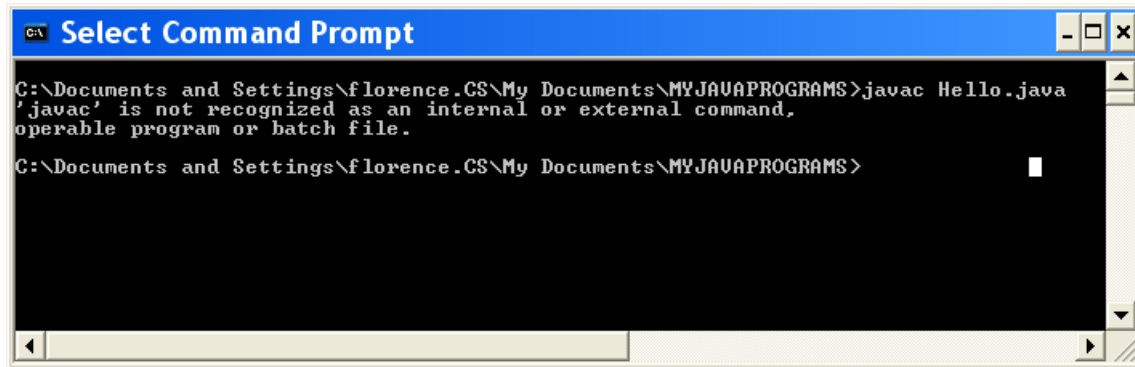


Figura 13: Sistema não reconhece o comando javac

Se o Java foi instalado corretamente em seu sistema, configure a variável **PATH** para apontar para a localização dos comandos Java. Para fazer isto, digite: **set PATH=C:\Program Files\Java\jdk1.5.0_01\bin**. Isto fará com que seu sistema procure pelos comandos na pasta **C:\Program Files\Java\jdk1.5.0_01\bin**, que é o padrão de localização dos arquivos Java quando de sua instalação. Depois de fazer isto, agora é possível utilizar os comandos Java.

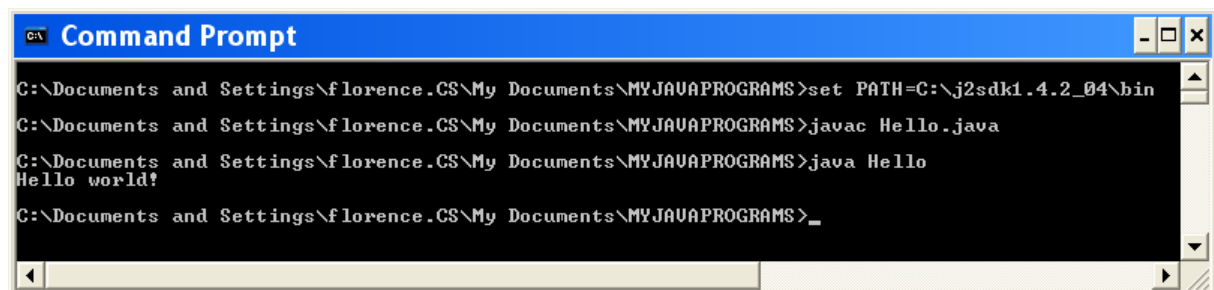
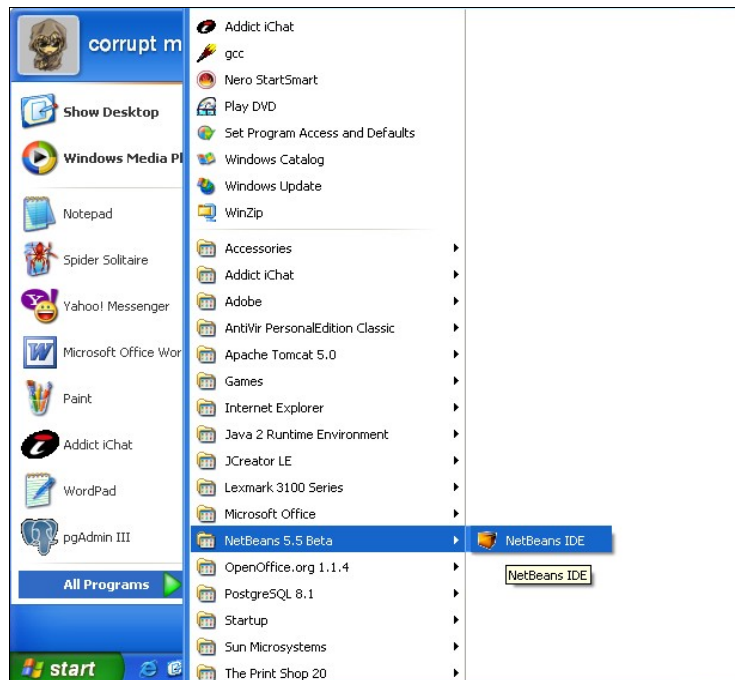


Figura 14: Configurando o caminho(path) e executando java

5. Utilizando NetBeans

Como já aprendemos a construção de programas de uma forma mais complicada, veremos como realizar todos estes processos descritos utilizando uma única aplicação. O **NetBeans**, é um Ambiente Integrado de Desenvolvimento (**Integrated Development Environment ou IDE**). Um IDE é um ambiente de programação, que possui um construtor de interface gráfica (GUI builder), um editor de código-fonte ou texto, um compilador e um depurador de erros.

Passo 1: Para executar NetBeans, selecione Start ⇒ All Programs ⇒ NetBeans 5.5 ⇒ NetBeans IDE



Depois de aberto o NetBeans, será mostrada uma janela similar a mostrada na Figura 15.

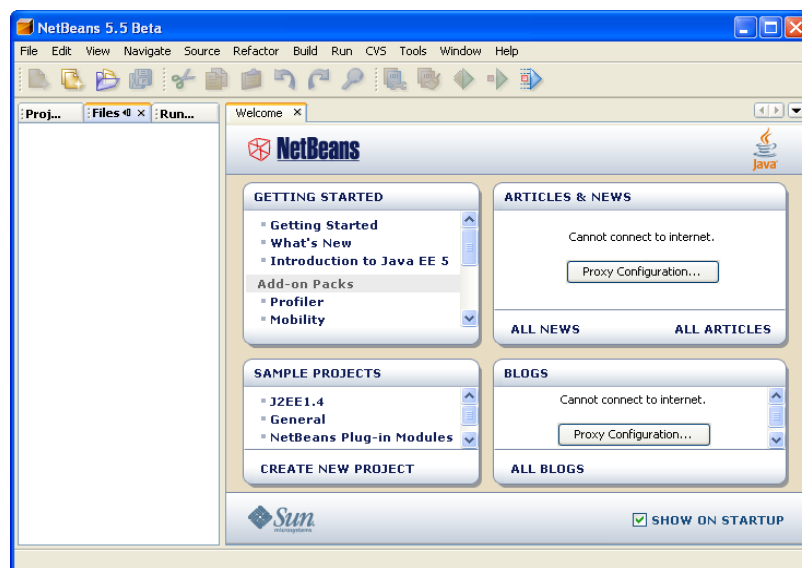
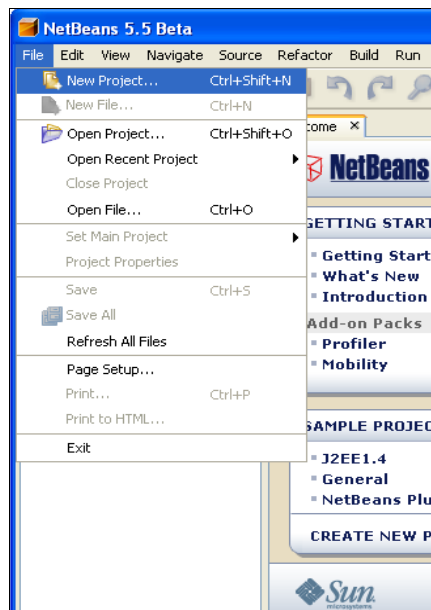
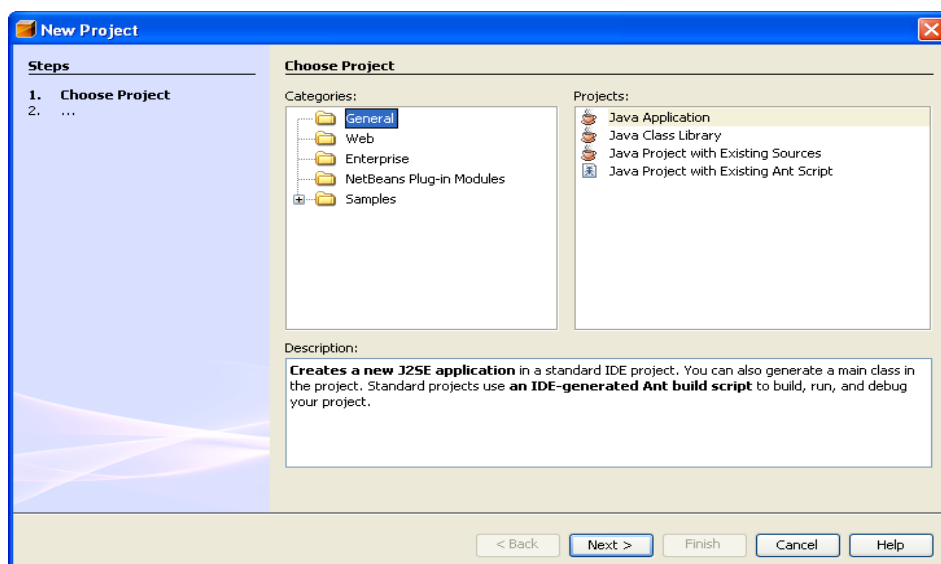


Figura 15: IDE NetBeans

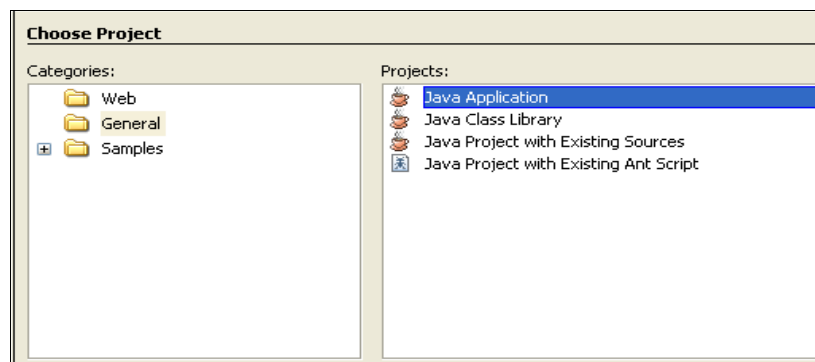
Passo 2: Vamos criar um projeto para o NetBeans. Selecione File ⇒ New Project.



Depois disto, uma caixa de diálogo New Project será mostrada.



Na opção **Categories** selecione **General** e em **Projects** selecione **Java Application** pressione o botão NEXT.



Uma caixa de diálogo de nova aplicação Java (New Java Application) será mostrada. Na caixa de texto do nome do projeto (Project Name), digite "HelloApplication".

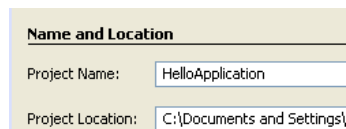
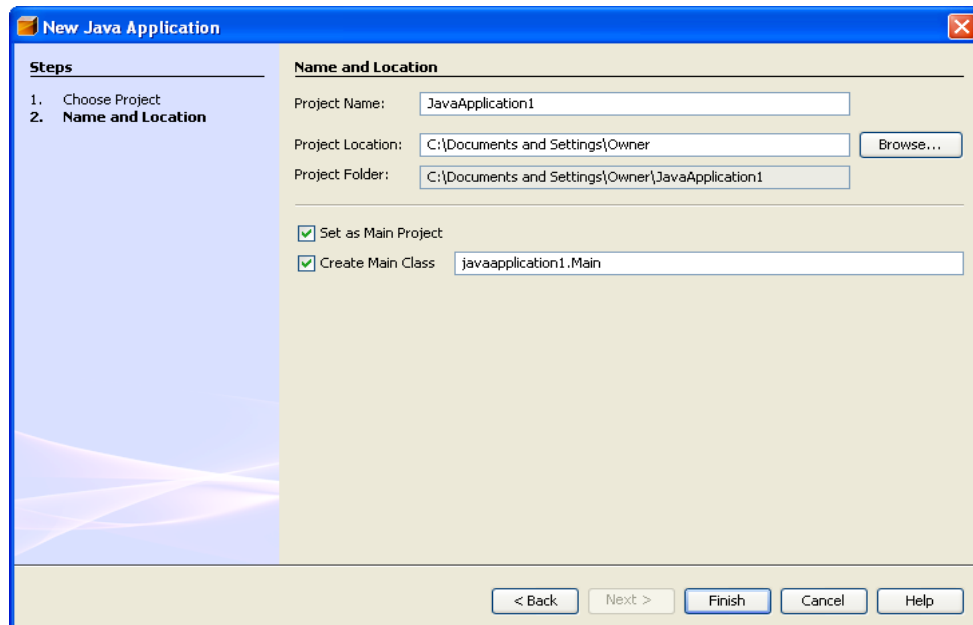
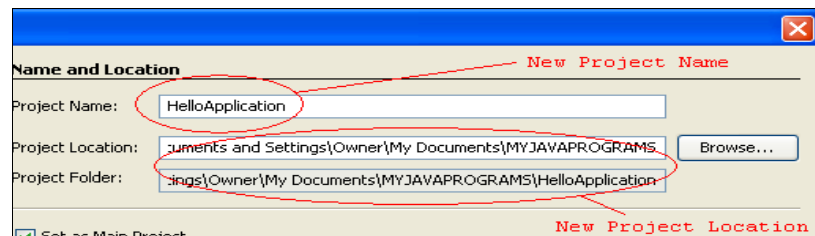
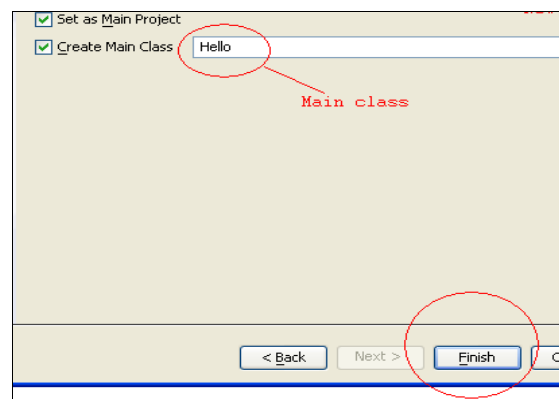


Figura 16: Altere Project Name

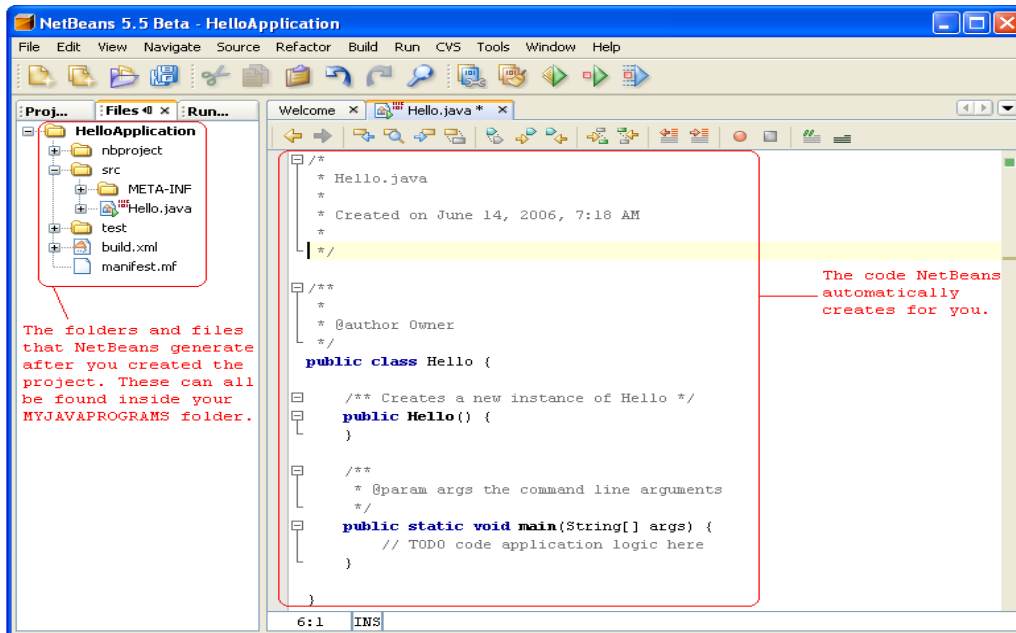
Altere a localização da aplicação (Application Location), pressionando no botão **Browse...** Siga os passos descritos na seção anterior para ir para a pasta MYJAVAPROGRAMS.



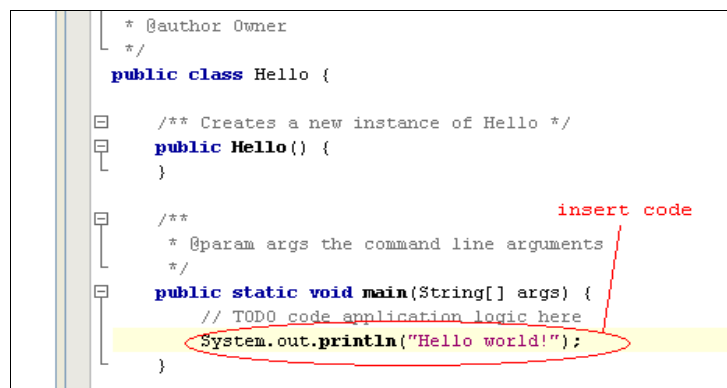
Por fim, na caixa de texto Create Main Class, digite Hello como a classe principal, e então clique no botão FINISH.



Passo 3: Antes de escrever o programa, vamos primeiro descrever a janela principal depois de criado o projeto. Como mostrado abaixo, NetBeans cria automaticamente o código básico para seu programa Java. Você pode então adicionar suas próprias declarações para o código gerado. No lado esquerdo da janela, temos uma lista de pastas e arquivos que o NetBeans gerou depois de criar o projeto. Isto é tudo que será encontrado na pasta MYJAVAPROGRAMS, que é onde está configurada a localização do projeto.



Modifique o código gerado pelo NetBeans ignorando as outras partes do programa. Insira a seguinte instrução: **System.out.println("Hello world!");**



Passo 4: Para compilar o programa, clique em Build ⇒ Build Main Project. Ou, utilize o botão de atalho para compilar a classe, conforme mostrado na Figura 17.

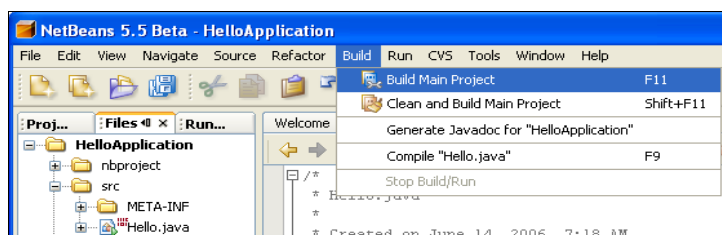


Figura 17: Botão de atalho para compilação do código

Se não houver nenhum erro em seu programa, você verá uma mensagem de "build successful"

na janela de saída.

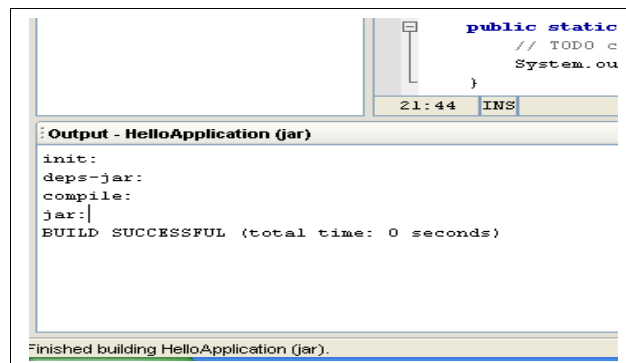


Figura 18: Janela de saída que está localizada abaixo da janela onde você escreve o código-fonte

Passo 5: Para executar o programa, clique em Run ⇒ Run Main Project. Ou utilize o botão de atalho.

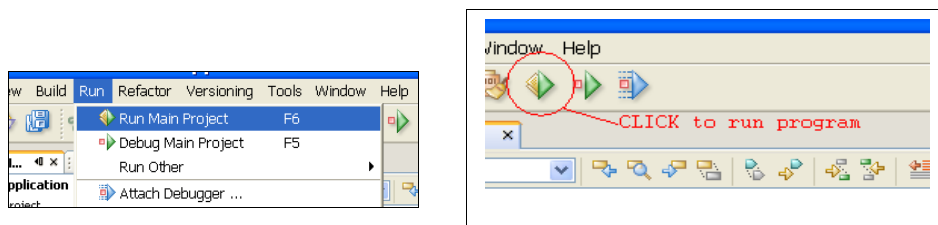


Figura 19: Botão de atalho para execução do programa

A saída do seu programa é mostrada na janela de saída(output window).

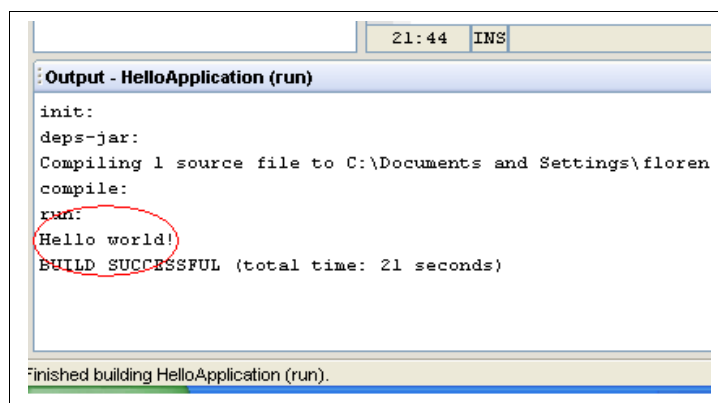


Figura 20: Saída de Hello.java

Parceiros que tornaram JEDI™ possível



Instituto CTS

Patrocinador do DFJUG.

Sun Microsystems

Fornecimento de servidor de dados para o armazenamento dos vídeo-aulas.

Java Research and Development Center da Universidade das Filipinas

Criador da Iniciativa JEDI™.

DFJUG

Detentor dos direitos do JEDI™ nos países de língua portuguesa.

Banco do Brasil

Disponibilização de seus *telecentros* para abrigar e difundir a Iniciativa JEDI™.

Politec

Suporte e apoio financeiro e logístico a todo o processo.

Borland

Apoio internacional para que possamos alcançar os outros países de língua portuguesa.

Instituto Gaudium/CNBB

Fornecimento da sua infra-estrutura de hardware de seus servidores para que os milhares de alunos possam acessar o material do curso simultaneamente.