

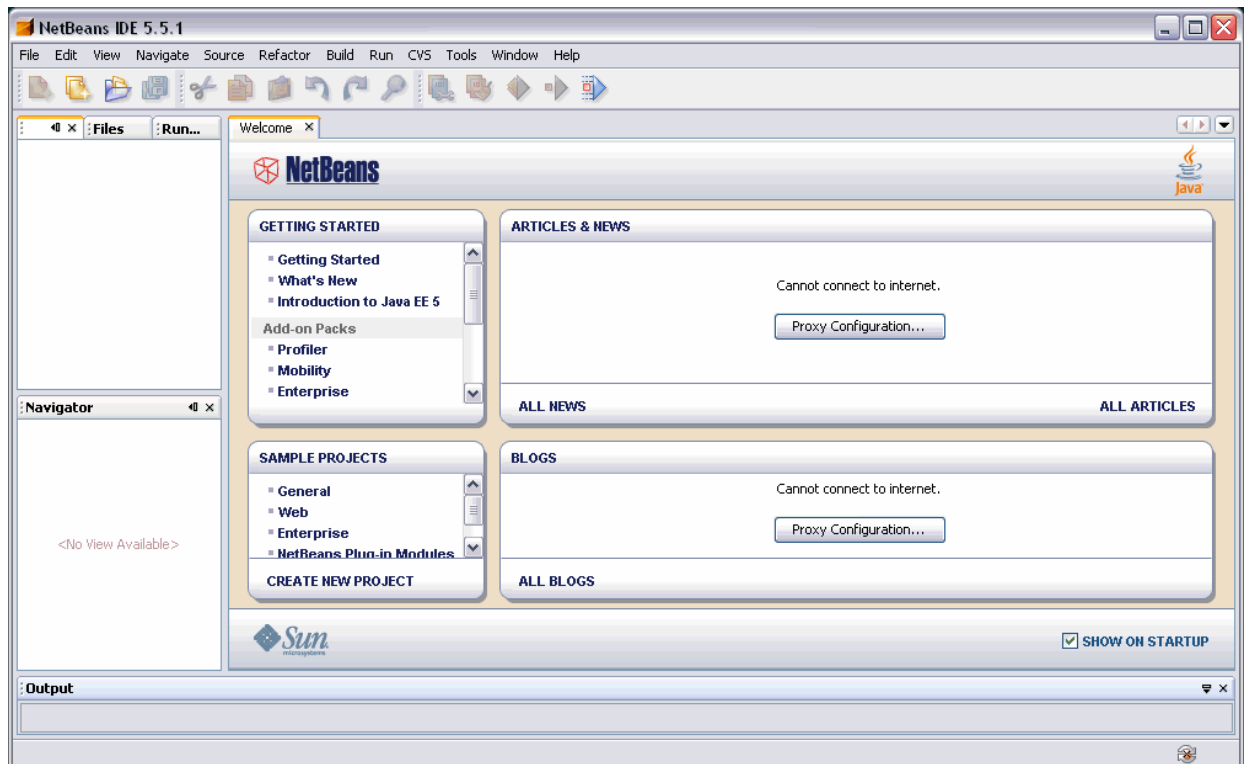
How to use the Sun *NetBeans* IDE for Java Application Development

Java application development is supported by many different tools. One of these tools is the *NetBeans* IDE (Integrated Development Environment). This very helpful development tool is available free of charge.

NetBeans is also a part of some versions of the Sun JDK (Java Development Kit). For more information about *NetBeans* please visit <http://www.netbeans.org/>.

The following sample was made with the *Sun J2SE Development Kit 5.0 Update 13 with NetBeans IDE 5.5.1 Bundle*. The download file was *jdk-1_5_0_13-nb-5_5_1-Win-ml.exe*. The Sun JDK 5 was installed on a Windows XP-based PC. Please see also *mHT-CPC1L-15.pdf: How to write a Hello World for the Java Runtime Environment (JRE)*.

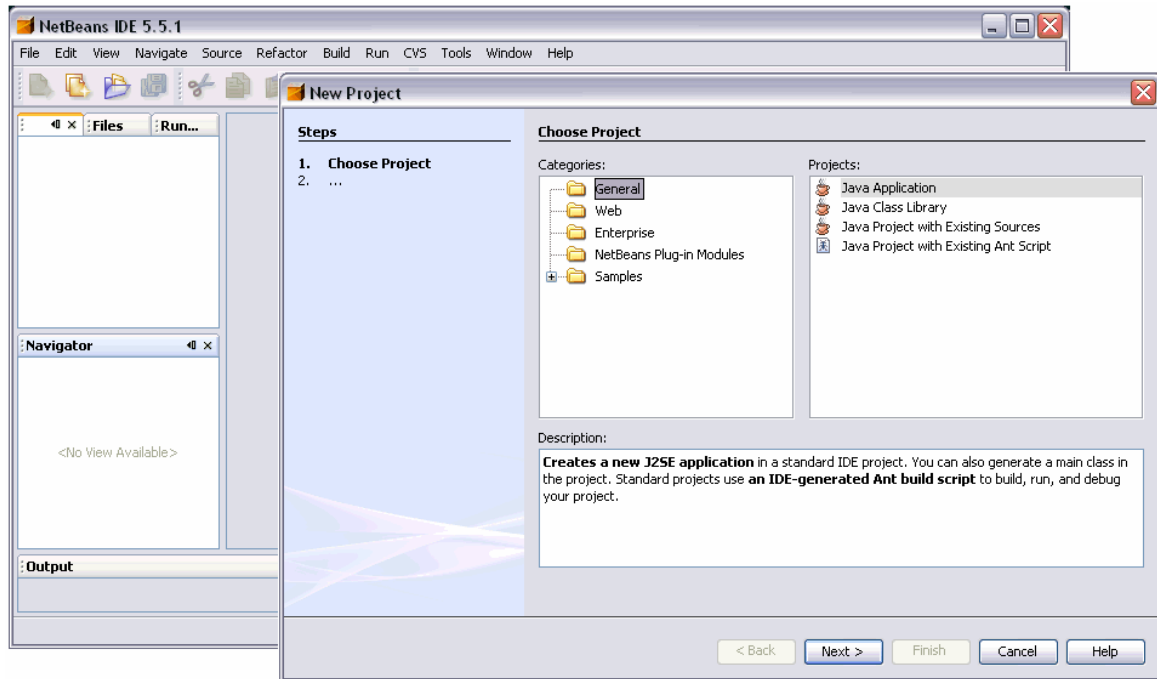
- **1. Step:** Please run your *NetBeans* IDE on your development PC. Most *NetBeans* installations leave an icon on the desktop. Just double-click this icon.



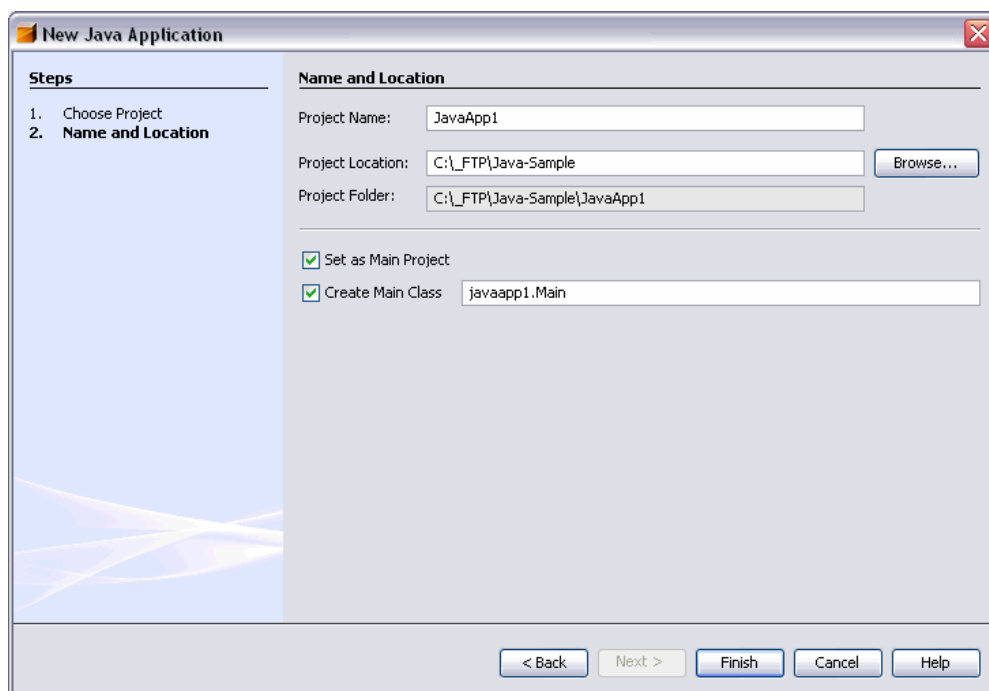
- **2. Step:** Close the *NetBeans* welcome window. Some parts of this window need an active Internet connection.

The Java development with *NetBeans* is also possible without an active Internet link of your development PC. In this case some help information not available.

- **3. Step:** Open the *NetBeans* *File* menu. Then select *New Project* and create a new Java project. The project category for this sample is *General*. The project type is *Java Application*. Then press the *Next* button.



- **4. Step:** Select a project name (e.g. *JavaApp1*) and a project location of your choice. Then press *Finish*.

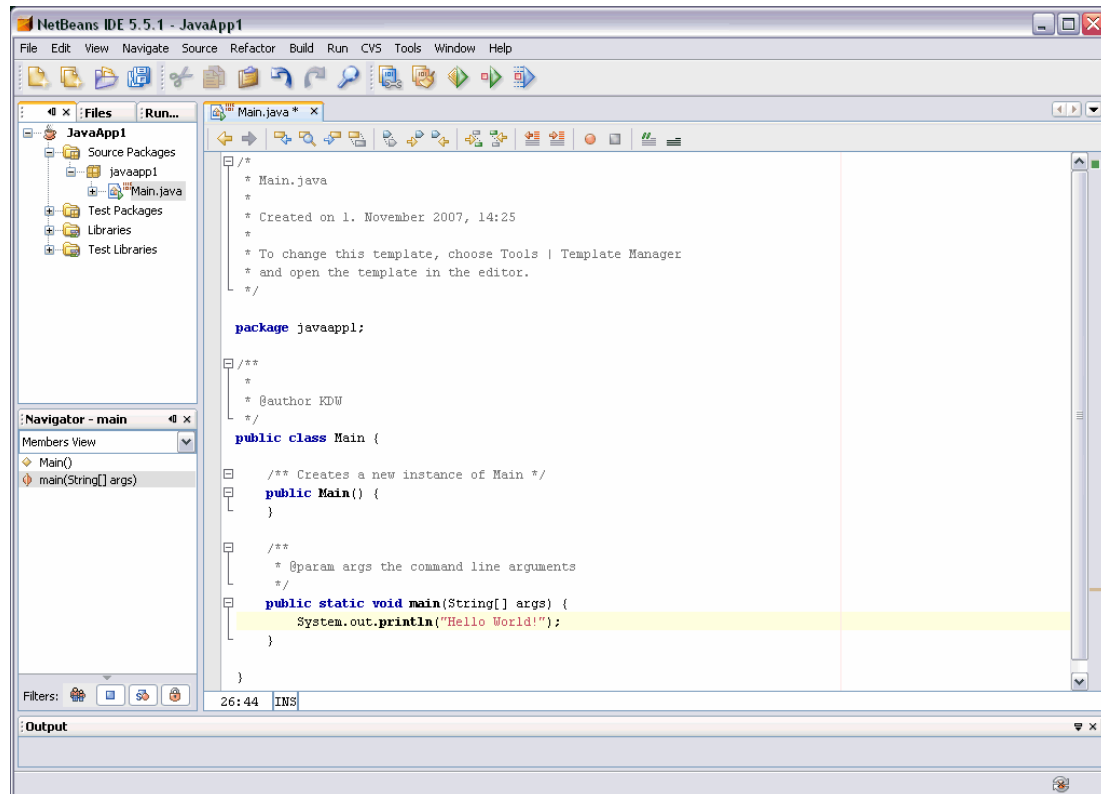


- **5. Step:** The *NetBeans* IDE creates the project environment and a Java source code file as a template. Please replace the line

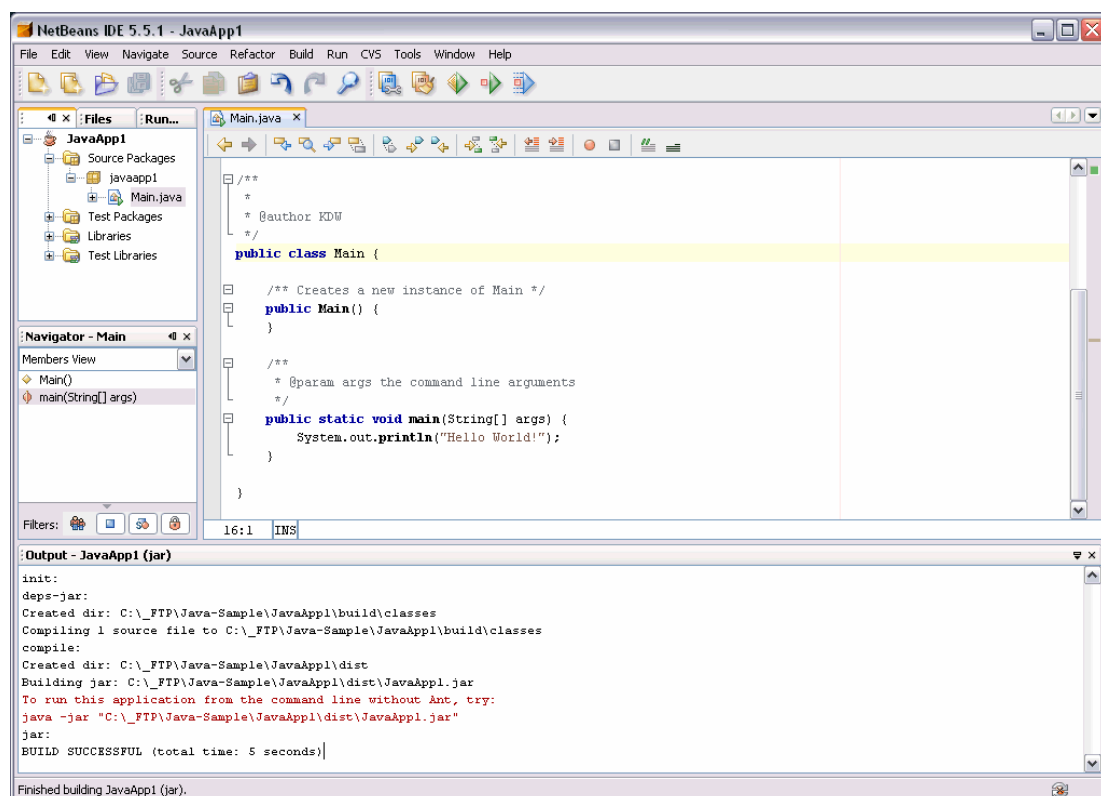
```
// TODO code application logic here
```

with

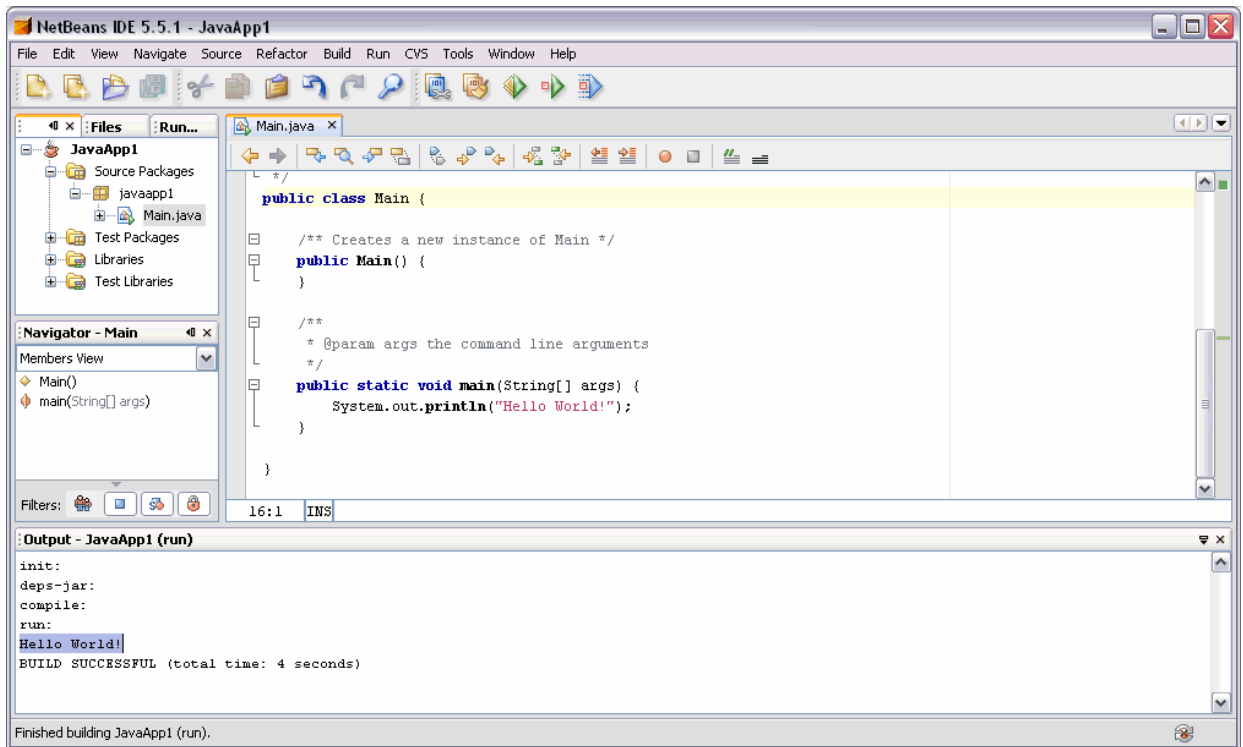
```
System.out.println("Hello World!");
```



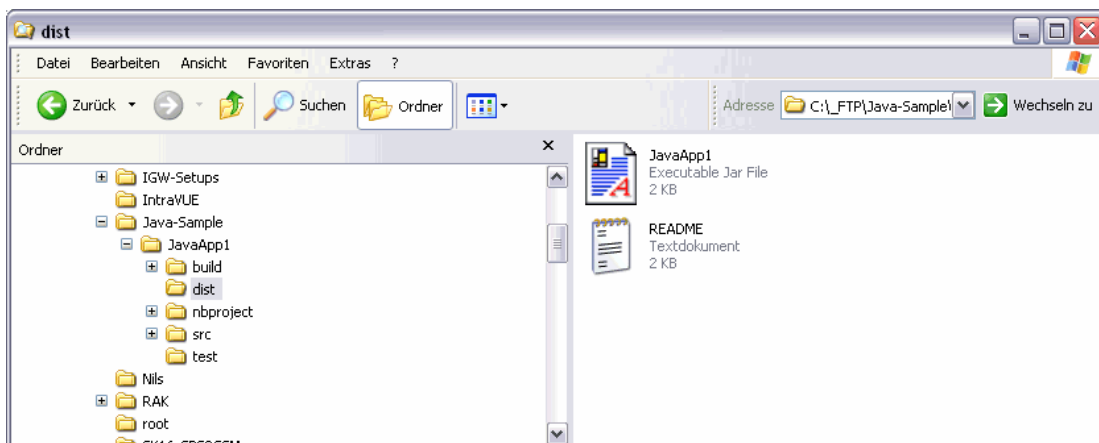
- **6. Step:** Open the *Build* menu and select *Build Main Project*. The IDE creates a JAR file for your project.



- **7. Step:** Open the *Run* menu and select the menu item *Run Main Project*. This starts the execution of your Java application within the *NetBeans* IDE. Watch the *Hello World!* output within the output window.



The executable JAR file is located in the *dist* directory of your project (in this sample the JAR file name is *JavaApp1.jar*). Please transfer this JAR file to the Com/PC.



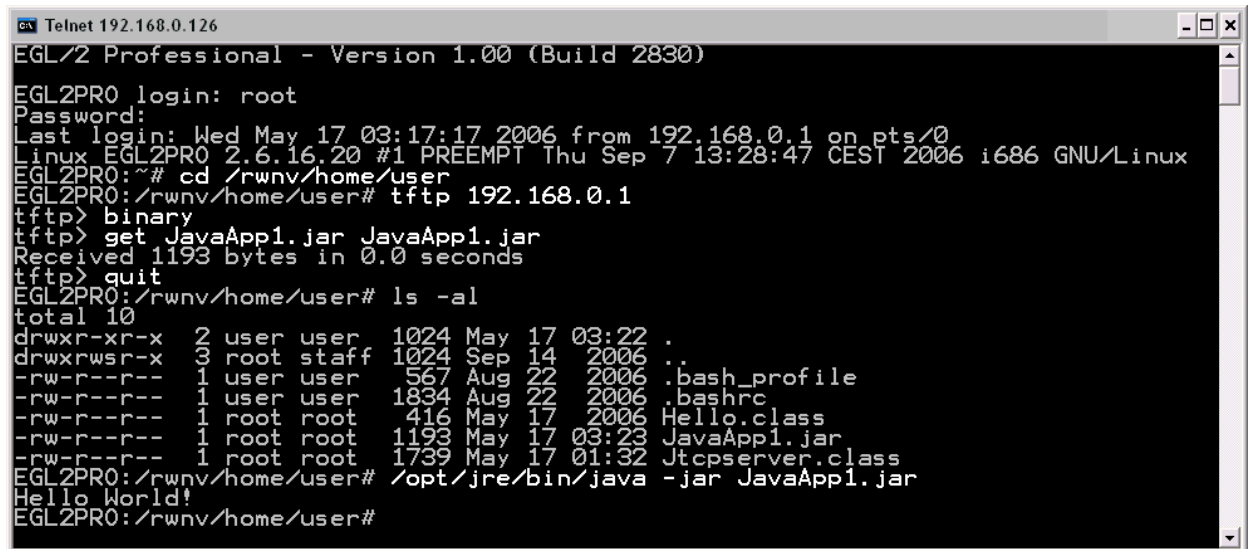
- **8. Step:** Use TFTP and a Telnet or SSH session for transferring the JAR file. Run the JAR file on the Com/PC. Use the following command sequence for this job:

```

cd /rwnv/home/user
tftp 192.168.0.1
binary
get JavaApp1.jar JavaApp1.jar

```

```
quit
/opt/jre/bin/java -jar JavaApp1.jar
```



```
Telnet 192.168.0.126
EGL/2 Professional - Version 1.00 (Build 2830)
EGL2PR0 login: root
Password:
Last login: Wed May 17 03:17:17 2006 from 192.168.0.1 on pts/0
Linux EGL2PR0 2.6.16.20 #1 PREEMPT Thu Sep 7 13:28:47 CEST 2006 i686 GNU/Linux
EGL2PR0:~# cd /rwnv/home/user
EGL2PR0:/rwnv/home/user# tftp 192.168.0.1
tftp> binary
tftp> get JavaApp1.jar JavaApp1.jar
Received 1193 bytes in 0.0 seconds
tftp> quit
EGL2PR0:/rwnv/home/user# ls -al
total 10
drwxr-xr-x  2 user user  1024 May 17 03:22 .
drwxrwsr-x  3 root staff 1024 Sep 14 2006 ..
-rw-r--r--  1 user user   567 Aug 22 2006 .bash_profile
-rw-r--r--  1 user user  1834 Aug 22 2006 .bashrc
-rw-r--r--  1 root root   416 May 17 2006 Hello.class
-rw-r--r--  1 root root  1193 May 17 03:23 JavaApp1.jar
-rw-r--r--  1 root root  1739 May 17 01:32 Jtcpserver.class
EGL2PR0:/rwnv/home/user# /opt/jre/bin/java -jar JavaApp1.jar
Hello World!
EGL2PR0:/rwnv/home/user#
```

Please note: A JAR file (or **J**ava **AR**chive) is used for aggregating many files into one. The JAR file is generally used to distribute Java classes and associated metadata. JAR files are typically started with a command line similar to:

```
java -jar foo.jar
```

or

```
java -jar "foo.jar"
```

This means, the JAR file is an executable for the JVM (Java Virtual Machine). A JAR file is a binary file and not a text file. Make sure that your file transfer operates in binary mode. Otherwise the execution with the Com/PC Java VM produces some unclear error messages.

That is all.