

Choose page language Search:

Home

Features

Plugins

Docs & Support

Community

Partners

HOME / [Docs & Support](#) / [NetBeans 6.0](#) / [Basic IDE Documentation](#)

NetBeans IDE Java Quick Start Tutorial

Welcome to NetBeans IDE!

This tutorial provides a very simple and quick introduction to the NetBeans IDE workflow by walking you through the creation of a simple "Hello World" Java console application. Once you are done with this tutorial, you will have a general knowledge of how to create, build, and run applications in the IDE.

This tutorial takes less than 10 minutes to complete.

After you finish this tutorial, you can move on to the learning trails, which are linked from the [Documentation, Training & Support](#) page. The learning trails provide comprehensive tutorials that highlight a wider range of IDE features and programming techniques for a variety of application types. If you do not want to do a "Hello World" application, you can skip this tutorial and jump straight to the learning trails.

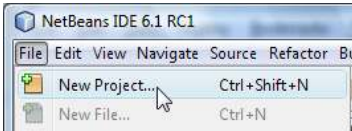
To follow this tutorial, you need the following software and resources.

Software or Resource	Version Required
NetBeans IDE	Version 6.1 or version 6.0
Java Development Kit (JDK)	Version 6 or version 5

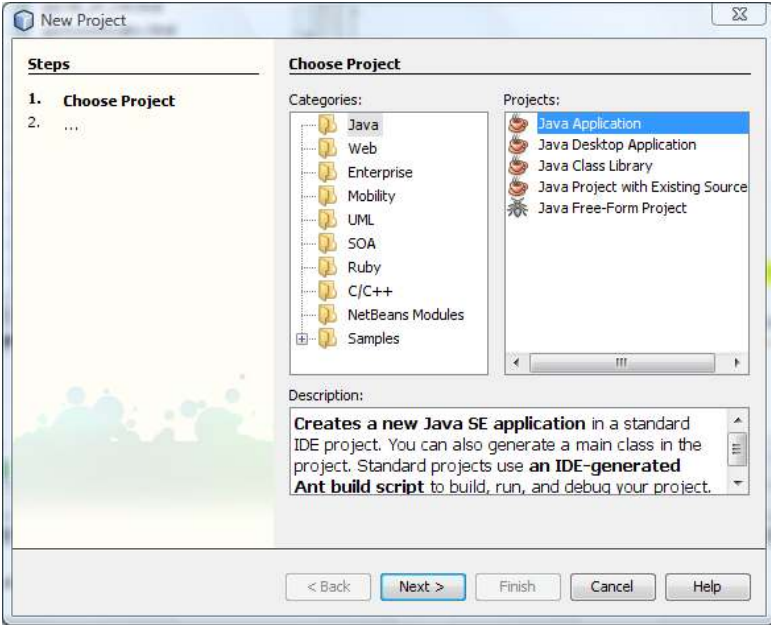
Setting Up the Project

To create an IDE project:


1. Start NetBeans IDE.
2. In the IDE, choose File > New Project (Ctrl-Shift-N), as shown in the figure below.




3. In the New Project wizard, expand the Java category and select Java Application as shown in the figure below. Then click Next.




4. In the Name and Location page of the wizard, do the following (as shown in the figure below):
 - In the Project Name field, type HelloWorldApp.
 - Leave the Use Dedicated Folder for Storing Libraries checkbox unselected. (If you are using NetBeans IDE 6.0, this option is not available.)
 - In the Create Main Class field, type helloworldapp.HelloWorldApp.
 - Leave the Set as Main Project checkbox selected.

PRINTABLE VERSION 

Download NetBeans IDE

 [Training](#)
[Developing Applications for the Java SE Platform](#)
[Java Programming Language](#)

 [Support](#)
[FAQs](#)
[Sun Microsystems Support Plans](#)

[Documentation](#)
[Basic Java Programming](#)
[Java GUIs](#)
[Java Web Applications](#)
[EJB & Web Service Applications](#)
[Mobile Applications](#)
[SOA Applications](#)
[UML Modelling](#)
[PHP Applications](#)
[Ruby Applications](#)
[C/C++ Applications](#)
[NetBeans Modules and Rich-Client Applications](#)

[Sample Applications](#)
[Demos and Screencasts](#)

[More](#)
[Contribute Documentation!](#)
[Docs for Earlier Releases](#)

Name and Location

Project Name:

Project Location:

Project Folder:

☐ Use Dedicated Folder for Storing Libraries

Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

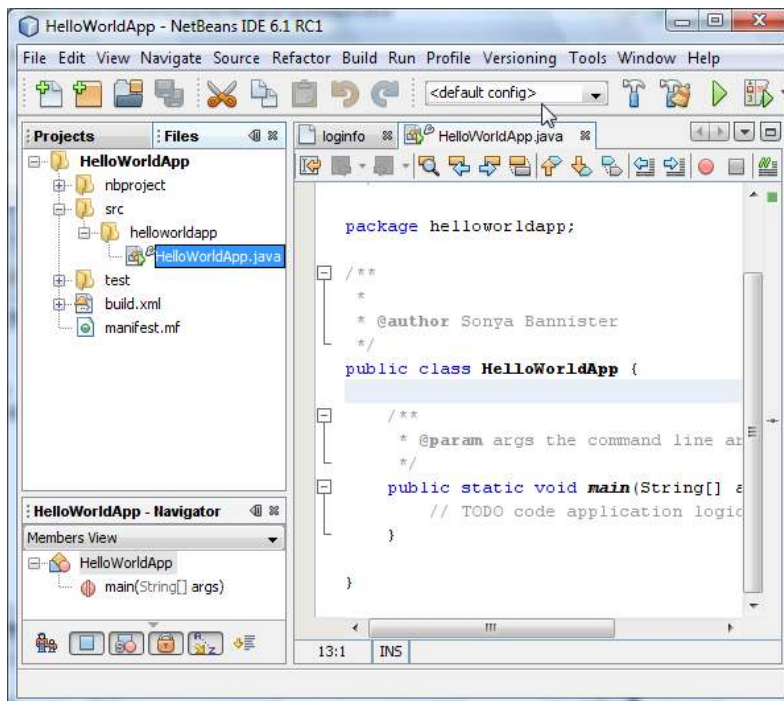
☒ Create Main Class

☒ Set as Main Project

5. Click Finish.

The project is created and opened in the IDE. You should see the following components:

- The Projects window, which contains a tree view of the components of the project, including source files, libraries that your code depends on, and so on.
- The Source Editor window with a file called `HelloWorldApp.java` open.
- The Navigator window, which you can use to quickly navigate between elements within the selected class.



Adding Code to the Generated Source File

Because you have left the Create Main Class checkbox selected in the New Project wizard, the IDE has created a skeleton class for you. You can add the "Hello World!" message to the skeleton code by replacing the line:

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

with the line:

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

Save the change by choosing File > Save.

The file should look something like the following:

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

package helloworldapp;

/**
 *
 * @author Sonya Bannister
 */
public class HelloWorldApp {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }

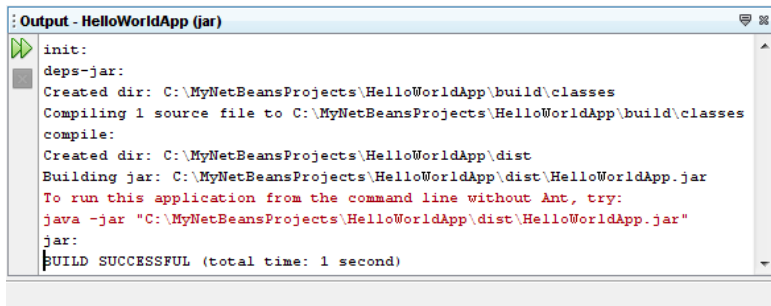
}
```

Compiling the Source File

To compile your source file, choose Build > Build Main Project (F11) from the IDE's main menu.

You can view the output of the build process by choosing Window > Output > Output.

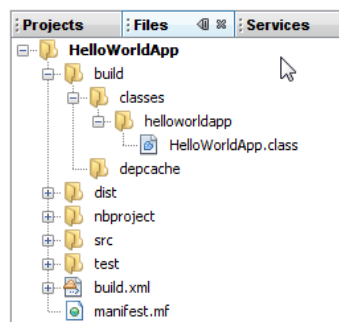
The Output window opens and displays output similar to what you see in the following figure.



If the build output concludes with the statement BUILD SUCCESSFUL, congratulations! You have successfully compiled your program!

If the build output concludes with the statement BUILD FAILED, you probably have a syntax error in your code. Errors are reported in the Output window as hyper-linked text. Click such a hyper-link to navigate to the source of an error. You can then fix the error and once again choose Build > Build Main Project.

When you build the project, the bytecode file HelloWorldApp.class is generated. You can see where the new file is generated by opening the Files window and expanding the Hello World App/build/classes/helloworldapp node as shown in the following figure.

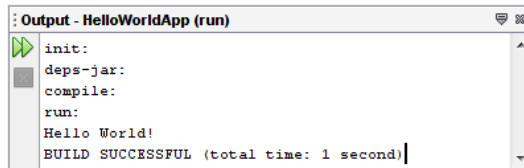


Now that you have built the project, you can run your program.

Running the Program

From the IDE's menu bar, choose Run > Run Main Project (F6).

The next figure shows what you should now see.



```
Output - HelloWorldApp (run)
init:
deps-jar:
compile:
run:
Hello World!
BUILD SUCCESSFUL (total time: 1 second)
```

Congratulations! Your program works!

You now know how to accomplish some of the most common programming tasks in the IDE.

[Send Us Your Feedback](#)

Next Steps

For a broader introduction to useful IDE features that are generally applicable to Java application development, see [Introduction to Developing General Java Applications](#).

To find information specific to the kind of applications you are developing, use the NetBeans IDE learning trail for that type of application. Each learning trail contains a series of tutorials and guides that range in scope from basic to advanced. The following learning trails are available:

- [Basic Java Programming](#)
- [Java GUI Applications](#)
- [Web Applications](#)
- [Java EE Applications](#)
- [Mobile Applications](#)
- [Ruby Applications](#)
- [SOA Applications](#)
- [UML Modeling](#)
- [NetBeans Plug-ins and Rich-Client Applications](#)
- [C/C++ Applications](#)

Bookmark this page

[Shop](#) [SiteMap](#) [About Us](#) [Contact](#) [Legal](#)

By use of this website, you agree to the [NetBeans Policies and Terms of Use](#)

Companion
Projects:



Sponsored by