# **Java Shell Tool (JShell)**

It is an interactive Java Shell tool, it allows us to execute Java code from the shell and shows output immediately. JShell is a REPL (Read Evaluate Print Loop) tool and run from the command line.

## **Advantages of JShell**

Jshell has reduced all the efforts that are required to run a Java program and test a business logic.

If we don't use Jshell, creating of Java program involves the following steps.

* Open editor and write program
* Save the program
* Compile the program
* Edit if any compile time error
* Run the program
* Edit if any runtime error
* Repeat the process

Jshell does not require above steps. We can evaluate statements, methods and classes, even can write hello program without creating class.

JShell helps you try out code and easily explore options as you develop your program. You can test individual statements, try out different variations of a method, and experiment with unfamiliar APIs within the JShell session.

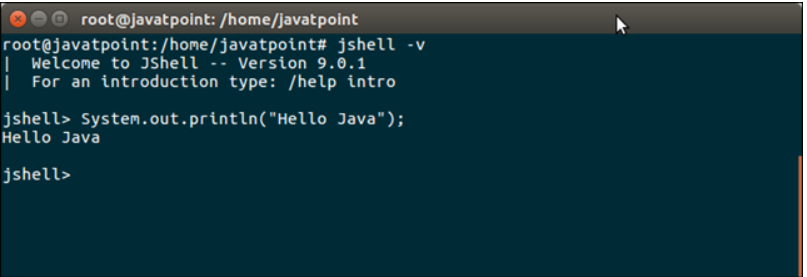
JShell doesn’t replace an IDE. As you develop your program, paste code into JShell to try it out, and then paste working code from JShell into your program editor or IDE.

### **How to Start JShell**

To start Jshell, first we must have installed Java 9 then open terminal in Linux or command prompt in windows and type **jshell ?v**. It will start jshell session and displays a welcome message to the console.

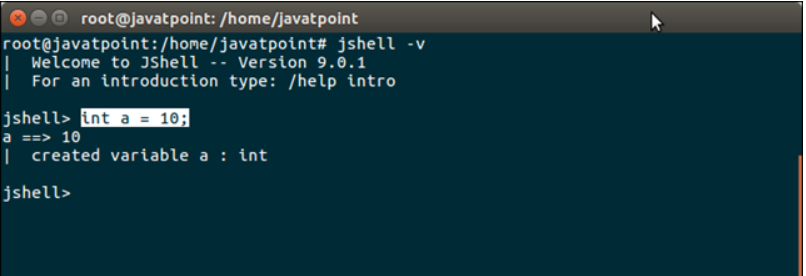
### **Hello Java Message**

To display a simple "Hello Java" message, write print command without creating class and hit enter.



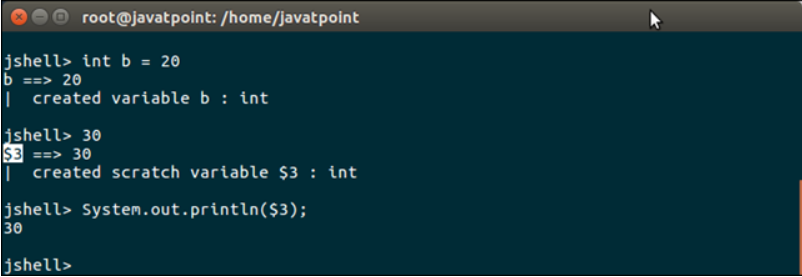
### **Variables**

We can declare variables and use anywhere throughout Jshell session. Let's create an integer variable. Semicolon (;) is optional.



### **Scratch Variables**

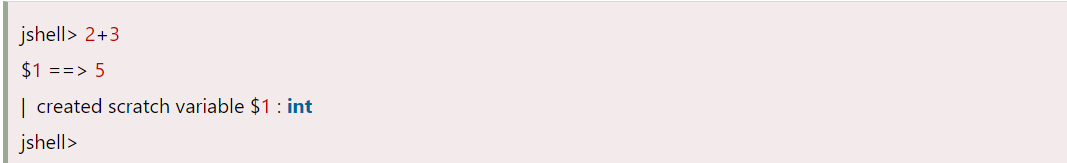
If we don't provide variable name, Java create implicit variable to store the value. These variables start with $ sign. We can use these variable by specifying implicit variable.



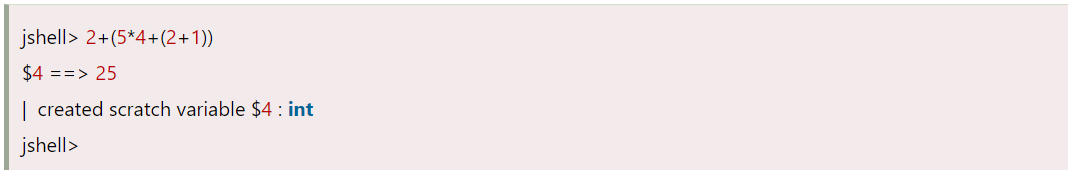
## **Expressions**

We can test any valid Java expression to get instant output. See, the following example.

**Adding two integers**

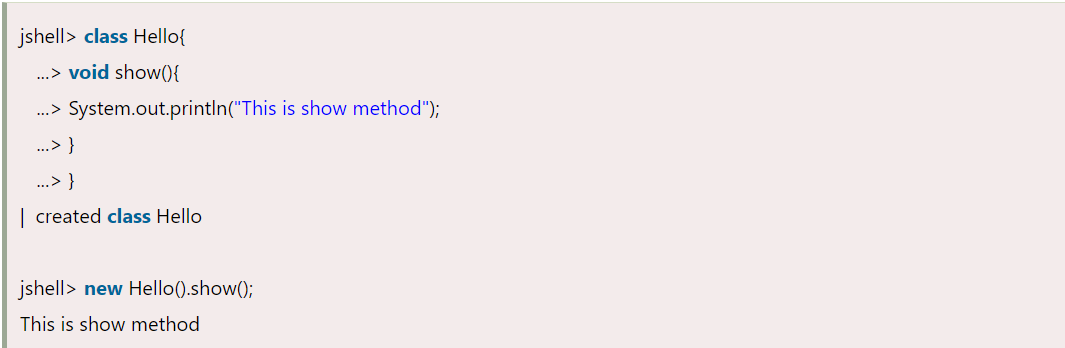


**Compound expression**



To create class, write source code for the class and call its method by creating object immediately. See the following example.

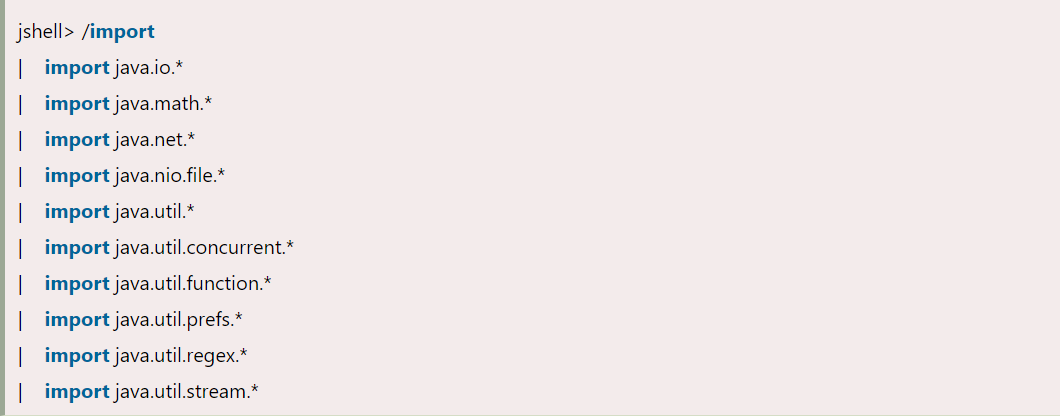
## **Class**



## **Package Imports**

By default, 10 packages are imported and can also be imported any package by using import statement.

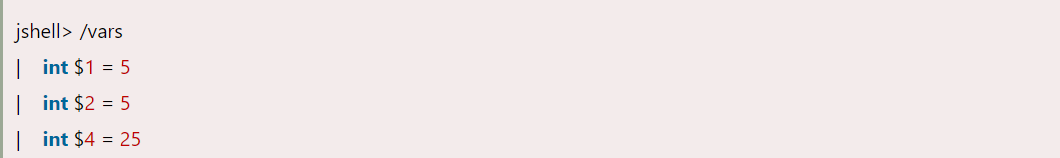
To see, default import packages, we can use following command.



## **Jshell Commands**

Jshell provides various useful commands that we can use to modify environment, manage code and to get code related information. Following are the useful information.

**Command /vars to show variables.**



**To get all written source code, use /list**



**https://docs.oracle.com/javase/9/jshell/introduction-jshell.htm#JSHEL-GUID-465BA4F5-E77D-456F-BCB7-D826AC1E18AE**