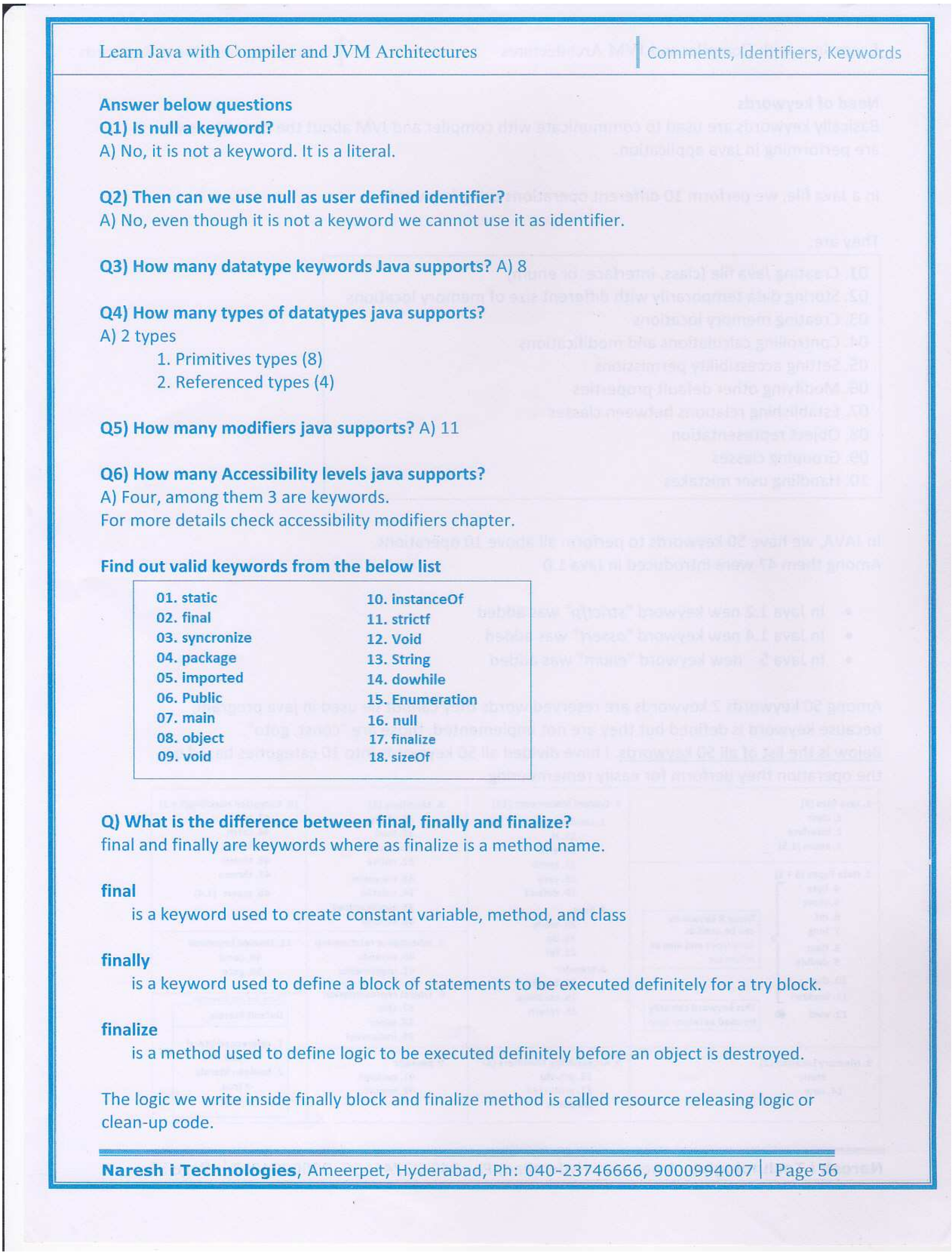


Answer below questions

Q1) Is null a keyword?
A) No, it is not a keyword. It is a literal.

Q2) Then can we use null as user defined identifier?
A) No, even though it is not a keyword we cannot use it as identifier.

Q3) How many datatype keywords Java supports? A) 8

Q4) How many types of datatypes java supports?
A) 2 types

1. Primitives types (8)
2. Referenced types (4)

Q5) How many modifiers java supports? A) 11

Q6) How many Accessibility levels java supports?
A) Four, among them 3 are keywords.
For more details check accessibility modifiers chapter.

Find out valid keywords from the below list

01. static
02. final
03. syncronize
04. package
05. imported
06. Public
07. main
08. object
09. void
10. instanceOf
11. strictf
12. Void
13. String
14. dowhile
15. Enumeration
16. null
17. finalize
18. sizeOf

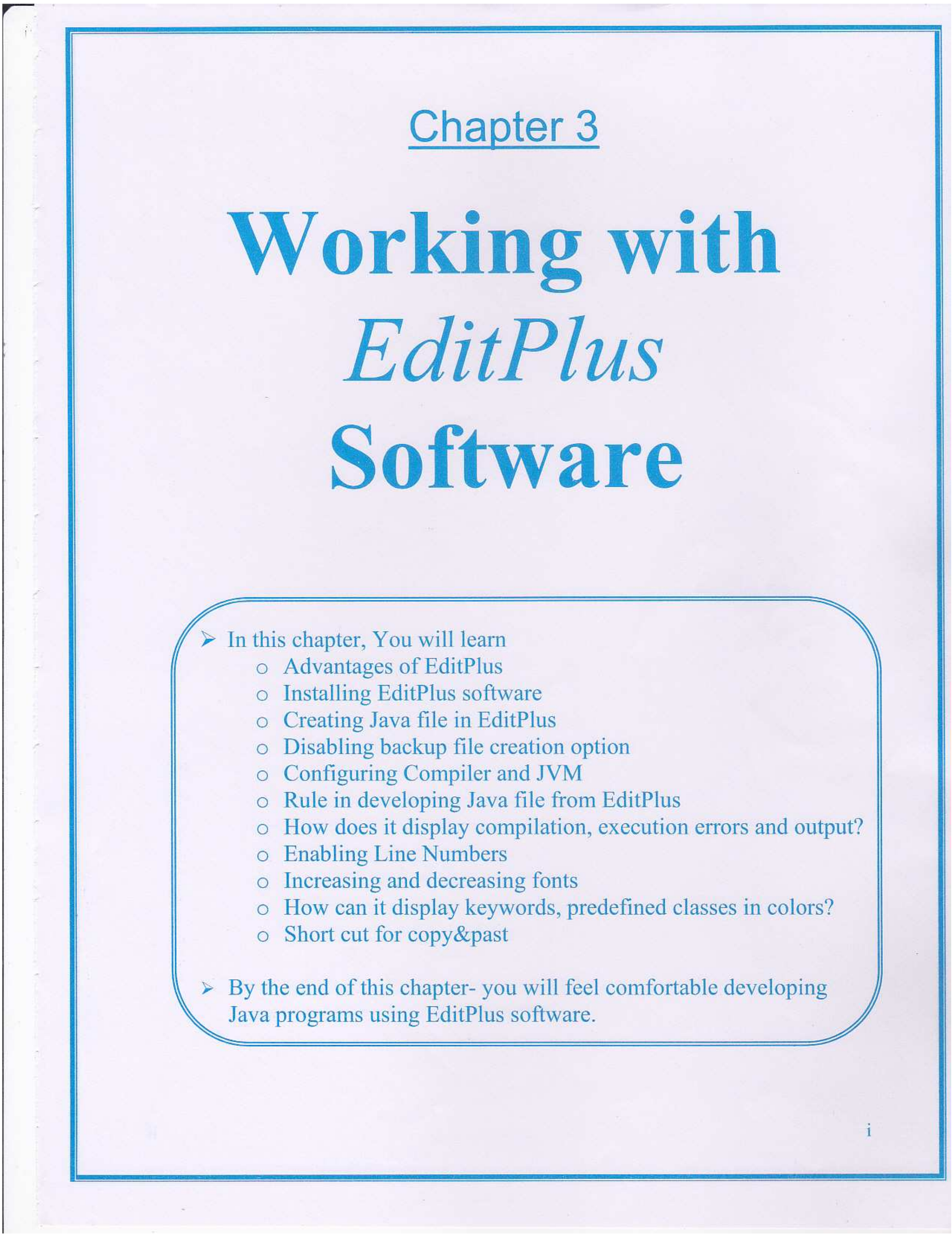
Q) What is the difference between final, finally and finalize?
final and finally are keywords where as finalize is a method name.

final
is a keyword used to create constant variable, method, and class

finally
is a keyword used to define a block of statements to be executed definitely for a try block.

finalize
is a method used to define logic to be executed definitely before an object is destroyed.

The logic we write inside finally block and finalize method is called resource releasing logic or clean-up code.



Chapter 3

Working with *EditPlus* Software

- In this chapter, You will learn
  - Advantages of EditPlus
  - Installing EditPlus software
  - Creating Java file in EditPlus
  - Disabling backup file creation option
  - Configuring Compiler and JVM
  - Rule in developing Java file from EditPlus
  - How does it display compilation, execution errors and output?
  - Enabling Line Numbers
  - Increasing and decreasing fonts
  - How can it display keywords, predefined classes in colors?
  - Short cut for copy&past
- By the end of this chapter- you will feel comfortable developing Java programs using EditPlus software.

Working with *EditPlus* Software

EditPlus is an editor software that supports developing Java, C, C++, HTML, PHP, Perl programs including a pain text file. While it can serve as a good Notepad replacement, it also offers many powerful features for Web page authors and programmers.

Advantages of EditPlus

EditPlus software provides more features then notepad. It provides below features

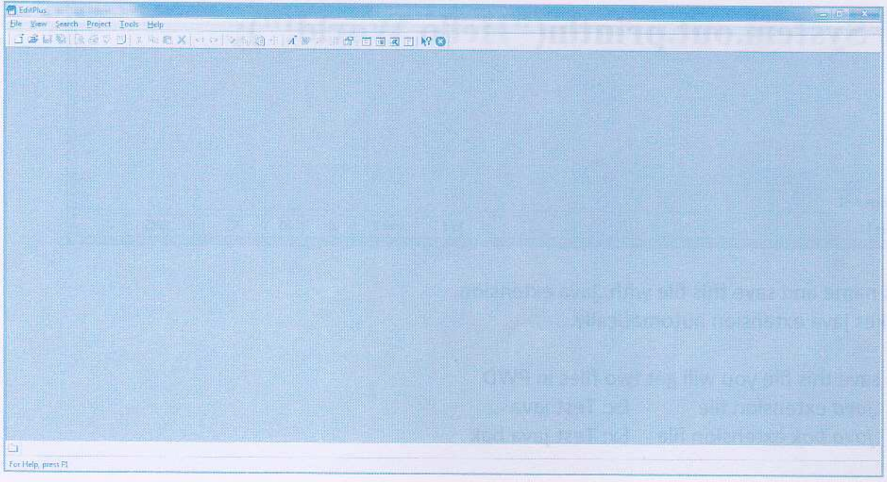
1. It has colorful screen, It shows
   a. *keywords* in **blue** color
   b. *predefined* classes in **red** color
   c. *Strings* and *literals* in **pink** color
2. It automatically manages *code indentation*.
3. Can add ".java" extension automatically
4. We can *compile* and *execute* Java program directly from Editplus editor
5. It also has auto save option
6. Can display line numbers

Installing *EditPlus* software

It is a trail version software you can download it from its home page "http://www.editplus.com/"

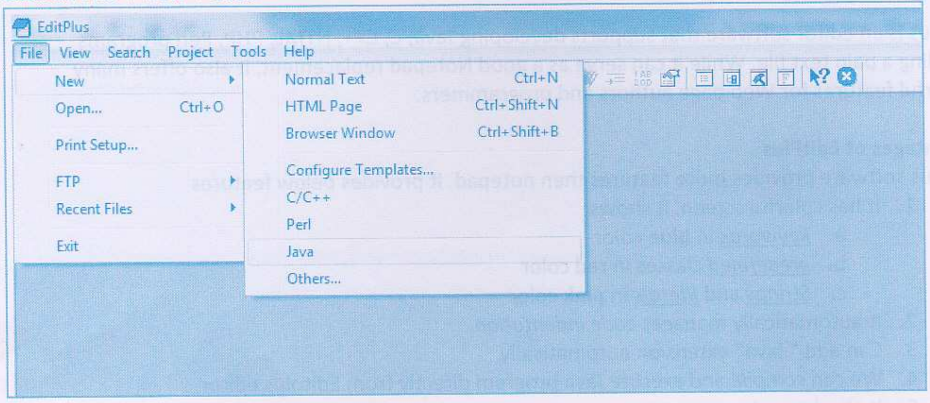
- You will be downloading "*epp331.exe*" file.
- Double click this exe file, its installation will be started.
- Click on Finish button at last window that shows Editplus installation is completed.

After installation you will find EditPlus shortcut image on desktop.
Double click on it, below window is opened

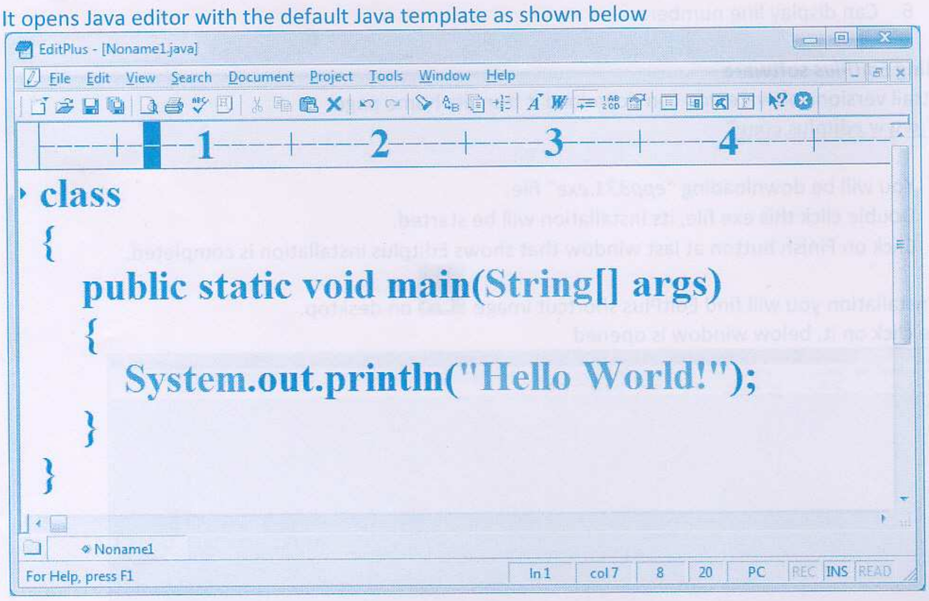


Creating a Java file in Editplus

Click on File -> new -> Java



It opens Java editor with the default Java template as shown below



```
class
{
    public static void main(String[] args)
    {
        System.out.println("Hello World!");
    }
}
```

Enter class name and save this file with .java extension.
Note: It takes java extension automatically.

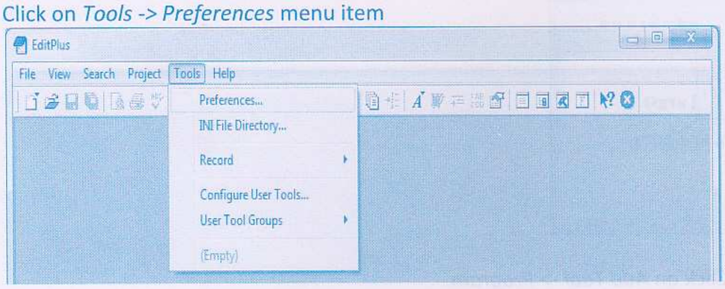
When you save this file you will get two files in PWD

1. *.java* extension file Ex: Test.java
2. *.java.bak* extension file Ex: Test.java.bak

Disabling backup file creation option

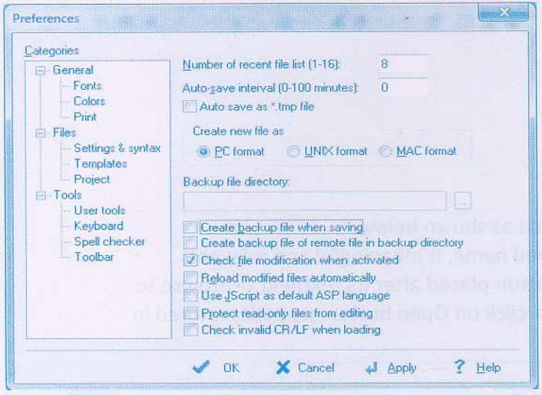
To disable back up file creation follow below steps

Click on *Tools -> Preferences* menu item



Below window is opened,

Click on *Files -> uncheck the checkbox "Create backup file when saving"*



Number of recent file list (1-16): 8

Auto-save interval (0-100 minutes): 0

□ Auto save as *.tmp file

Create new file as

● PC format ○ UNIX format ○ MAC format

Backup file directory:

□ Create backup file when saving

□ Create backup file of remote file in backup directory

☑ Check file modification when activated

□ Reload modified files automatically

□ Use JScript as default ASP language

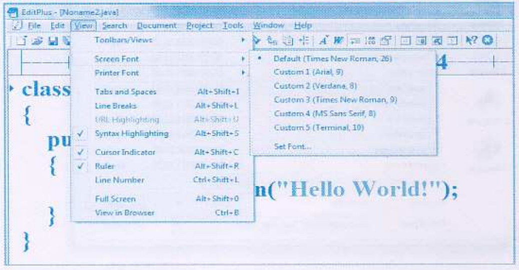
□ Protect read-only files from editing

□ Check invalid CR/LF when loading

Click Ok button to save this change.

Increasing and decreasing fonts

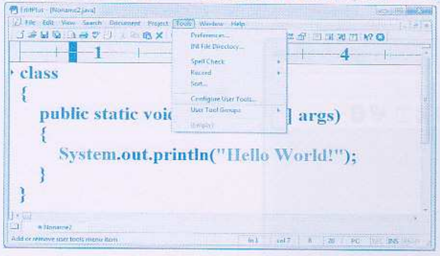
Open Java editor -> Click on *View -> Screen Font -> Set Font* menu item.



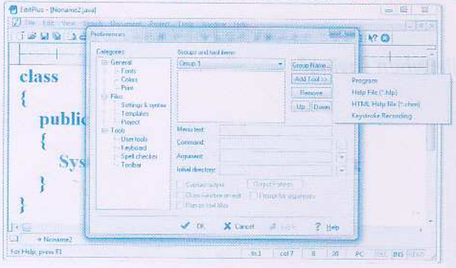
It opens a window, set your desired font -> click Ok button to save changes.

Configuring *javac* and *java* tools to compile and execute Java classes directly from Editplus:

Open Java Editor -> Click on *Tools Menu* -> *Configure User Tools Menu item*


```
class
{
    public static voi            ] args)
    {
        System.out.println("Hello World!");
    }
}
```

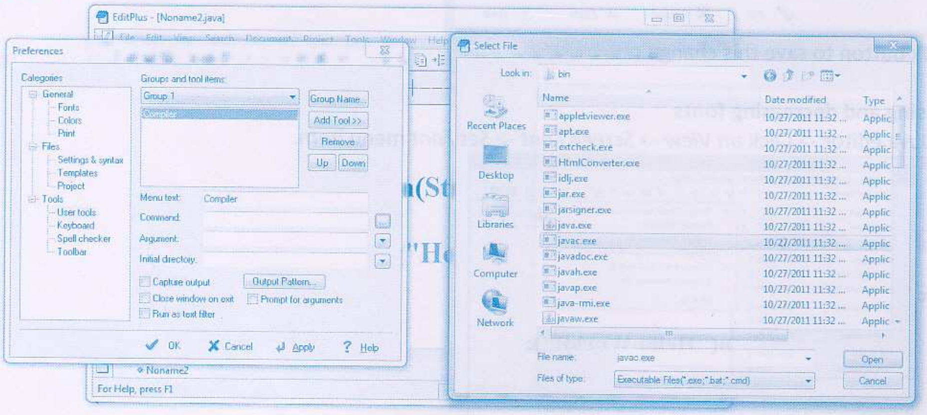
Below window is opened. Click on *Add Tool* -> *Program*



All above four text fields are enabled, enter text as shown below

Enter **Menu Text: Compiler** (can be user defined name, it meaningful and relevant)

To select *javac* command path, click on the button placed after its textfield -> browse to C:\jdk1.6.0_29\bin folder -> select *javac.exe* -> click on Open button. Its path is stored in textfield


Menu text: Compiler

Look in: bin

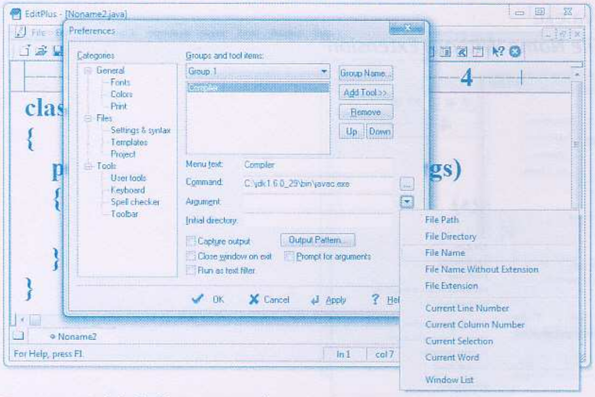
Name	Date modified	Type
appletviewer.exe	10/27/2011 11:32 ...	Applic
apt.exe	10/27/2011 11:32 ...	Applic
extcheck.exe	10/27/2011 11:32 ...	Applic
HtmlConverter.exe	10/27/2011 11:32 ...	Applic
idlj.exe	10/27/2011 11:32 ...	Applic
jar.exe	10/27/2011 11:32 ...	Applic
jarsigner.exe	10/27/2011 11:32 ...	Applic
java.exe	10/27/2011 11:32 ...	Applic
javac.exe	10/27/2011 11:32 ...	Applic
javadoc.exe	10/27/2011 11:32 ...	Applic
javah.exe	10/27/2011 11:32 ...	Applic
javap.exe	10/27/2011 11:32 ...	Applic
java-rmi.exe	10/27/2011 11:32 ...	Applic
javaw.exe	10/27/2011 11:32 ...	Applic

File name: javac.exe

Files of type: Executable Files(*.exe;*.bat;*.cmd)

To enter **Argument** value,

Click on *dropdown* button that is placed after its textfield -> Click on *File Name*



Groups and tool items: Group 1 — Compiler

Menu text: Compiler

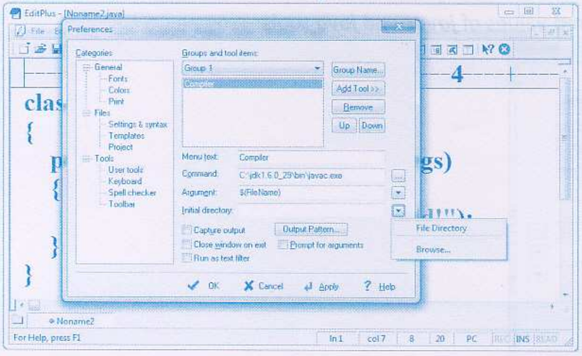
Command: C:\jdk1.6.0_29\bin\javac.exe

Argument:

Initial directory:

To enter **Initial directory** value

Click on *dropdown* button that is placed after its textfield -> Click on *File Directory*



Groups and tool items: Group 1 — Compiler

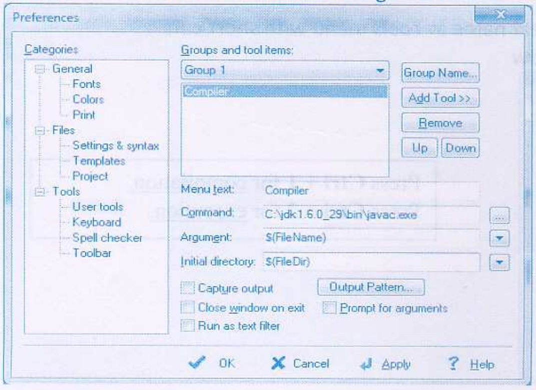
Menu text: Compiler

Command: C:\jdk1.6.0_29\bin\javac.exe

Argument: $(FileName)

Initial directory:

Below is the final window with all configured values



Groups and tool items: Group 1 — Compiler

Menu text: Compiler

Command: C:\jdk1.6.0_29\bin\javac.exe

Argument: $(FileName)

Initial directory: $(FileDir)

□ Capture output

□ Close window on exit

□ Prompt for arguments

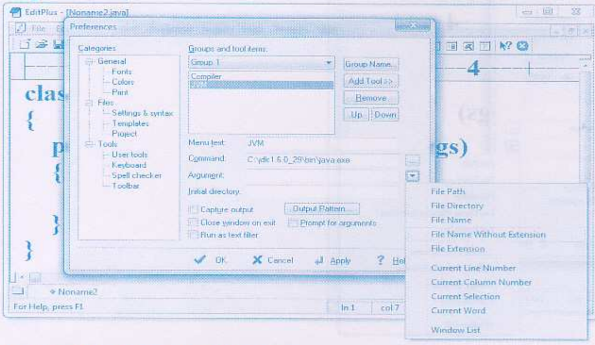
□ Run as text filter

Follow the same above procedure to configure *java* tool

Menu Text: JVM

Command: C:\jdk1.6.0.29\bin\java.exe

Argument: your should choose *File Name Without Extension*

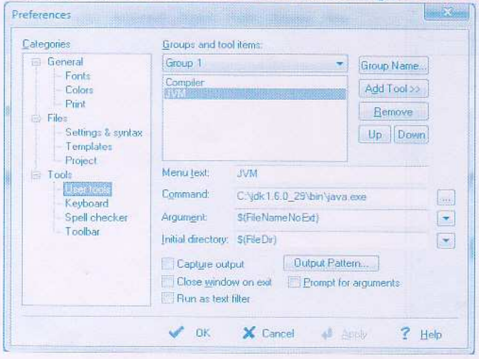


Menu text: JVM

Command: C:\jdk1.6.0_29\bin\java.exe

Initial Directory: File Directory

Below is the final window with all configured values of ***javac*** and ***java*** tools



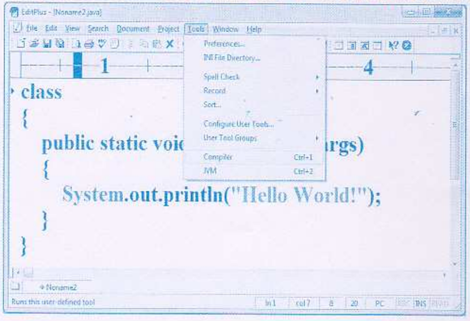
Menu text: JVM

Command: C:\jdk1.6.0_29\bin\java.exe

Argument: $(FileNameNoExt)

Initial directory: $(FileDir)

Now you can find these two tools with *Menu text* name in *Tools* menu with short cuts **Compiler:** Ctrl +1 & **JVM:** Ctrl + 2 as shown below



```
class
{
    public static voi                    rgs)
    {
        System.out.println("Hello World!");
    }
}
```

Press **Ctrl + 1** for compilation.
Press **Ctrl + 2** for execution.

Rule in developing Java file from EditPlus
To execute Java program from editplus, the Java file name must be same as class name even though class is not public. Else you will get exception "*java.lang.NoClassDefFoudError*"

Why this rule?
Recollect the configuration done for *java* tool,
We set **Argument** value as "*FileName Without Extension*".

So editplus software executes *java* command by passing filename.

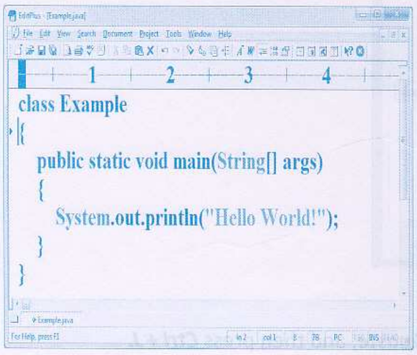
For Example
If we create class with name ***A*** and java file name with name ***Example.java***.

Editplus Software
Compiles this java file as
javac Example.java
Executes the class as
java Example

So in this case JVM throws "*java.lang.NoClassDefFoundError*: Example" as there is no ***Example.class***, instead we have ***A.class***.

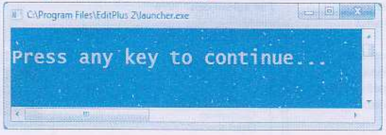
Note:
This rule is common for all editor softwares those support in-build compilation and execution

How does it display compilation, execution errors and output?
For compilation and execution it also uses command prompt. So all compilation, execution errors and output is displayed on command prompt windows as shown below.


EditPlus - [Example.java]
File Edit View Search Document Project Tools Window Help
Example.java
For Help, press F1

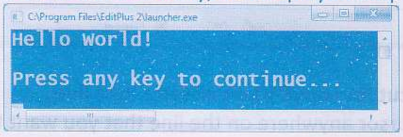
```
class Example
{
    public static void main(String[] args)
    {
        System.out.println("Hello World!");
    }
}
```

Compiled successfully



```
Press any key to continue...
```

Executed successfully, and displayed output

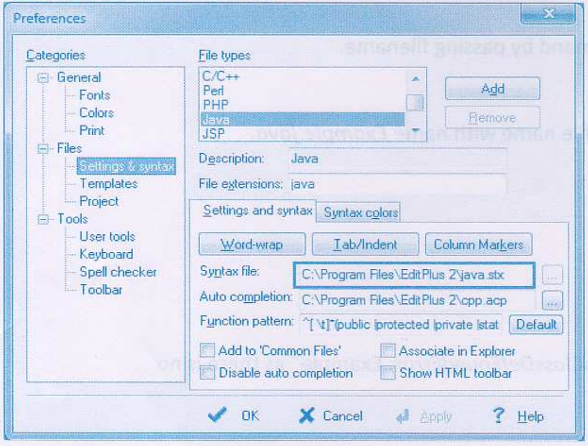


```
Hello World!
Press any key to continue...
```

To enable Line number
Open Java editor -> Click on View -> Line Number menu item.
Short-Cut: Ctrl + shift + L

Q) How can it display keywords, predefined classes in blue and red colors?
All predefined class names are saved in file, that file path is
C:\Program Files\EditPlus 2\java.stx

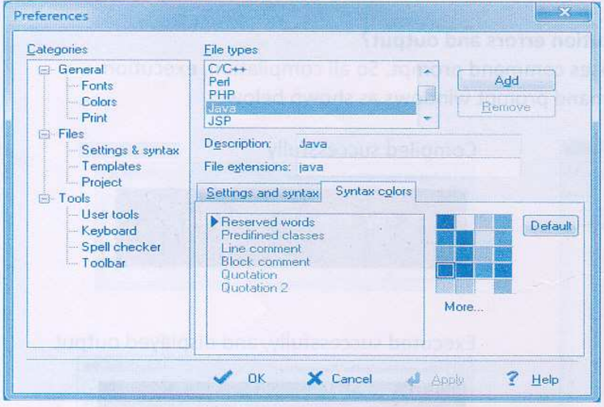
In Editplus, this file path is configured in the below window
Click on *Tools -> Preferences* menu item ->
Click on *Files -> Settings & syntax*



Description: Java
File extensions: java
Syntax file: C:\Program Files\EditPlus 2\java.stx
Auto completion: C:\Program Files\EditPlus 2\cpp.acp
Function pattern: ^[\t]*(public |protected |private |stat
□ Add to 'Common Files' □ Associate in Explorer
□ Disable auto completion □ Show HTML toolbar

If you want highlight any other predefined or user defined classes, you must add those class names in this file.

Click on ***Syntax colors*** tab, you will be shown default configured colors



Description: Java
File extensions: java

Short- cut for copy&past existed single line:
Place cursor anywhere on the line that you want to ***copy&paste***, and then press ***Ctrl + J.***