Steps To Prepare First Java Application

Steps to prepare First Java Application:

1) Install Java Software
2) Select Java Editor
3) Write Java Program
4) Save Java File
5) Compile Java File
6) Execute Java Application

1) Install Java Software:

1) Download jdk-8-windows-i586.exe file from internet.
2) Double click on jdk-8-windows-i586.exe file
3) Click on "Yes" button.
4) Click On "Next" button.
5) Change JDK installation location from "C:\Program Files(x86)\Java\jdk1.8.0" to "C:\Java\jdk1.8.0" by clicking on "Change" button and "OK" button.
6) Click on "Next" button.
7) Change JRE installation location from "C:\Program Files(x86)\Java\jre8" to "C:\Java\jre8" by clicking on "Change" button and "OK" button.

h) Click on "Next" button.

a. Click on "Close" button.

After installation of JAVA software, we have to set "path" environment variable to the location where all JDK commands are existed that is "C:\Java\JDK1.8.0\bin" inorder to make available all JAVA commands to Operating System.

On command Prompt:
set path=C:\Java\JDK1.8.0\bin;

If we provide "path" set up like above on the command prompt then this set up is available upto the present command prompt only, it is not available to all the command prompts.

If we want to set "path" environment variable permanently then we have to use the following steps.

1) Right Click on "Computers" or "This PC" on desktop .
2) Select "Properties".
3) Select "Advanced System Settings" hyper link.
4) Click on "Advanced" Tab[Bydefault Selected].
5) Click on "Environment VAriables.." button.
6) Goto User Variables part and click on "New" button.

7) Provide the following details.
variable name: path
variable value: C:\Java\jdk1.8.0\bin";
8) Click on "Ok" button.
9) Click on "OK" button.
10) Click on "OK" button.

Q)If we set all three versions[JAVA6,JAVA7,JAVA8] to the "path" environment variable then which version will come to the command prompt?

If set all JAVA6, JAVA7, JAVA8 versions to "path" environment variable then Command prompt will take the java version which we provided as first one to the path environment variable in the order.

EX:
path=C:\Java\jdk1.6.0\bin;C:\Java\jdk1.7.0\bin;C:\Java\jdk1.8.0\bin;
On Command Prompt:

D:\javaapps>java -version --> Enter
Java - Version: JDK1.6.0

If we want to switch java from one version to another version in simpified manner as per the requirement then we have to use batch files.

a)Take a text file.
b)provide the required "path" command.
c)Save file with "file_Name.bat" .
d)Open Command prompt and goto the location where bat files are saved.
e)write bat file name and click on enter button.
D:\java9\ java6.bat
set path=C:\Java\jdk1.6.0\bin;

D:\java9\ java7.bat
set path=C:\Java\jdk1.7.0\bin;

D:\java9\ java8.bat
set path=C:\Java\jdk1.8.0\bin;

On Command Prompt:
D:\java9>java6.bat ---> Enter button
--- we will get java6 setup----

D:\java9>java7.bat ---> Enter button

--- we will get java7 setup----

D:\java9>java8.bat ---> Enter button
--- we will get java8 setup----

2) Select Java Editor:

Editor is a software, it will provide very good env to write java programs and to save java programs in our system.

EX: Notepad, Notepadplus, Editplus,....

Note: In real time application development, it is not suggestible to use Editors, it is always suggestible to use IDEs[Integrated Development Environment].

EX: Eclipse, MyEclipse, Netbeans,.....

3) Write Java Program:

To write java program we have to use some prederfined library provided by JAVA API[Application Programming Interface].

EX
D:\javaapps\ Test.java

```
1)  class Test {
2)      public static void main(String[] args) {
3)          System.out.println("First Java Application");
4)     }
5)  }
```

4) Save Java File:

To save java file in our system, we have to follow the following two conditions.

1.If the present java file contains any public element[class, abstract class, interface, enum] then we must save java file with public element name only.If we voilate this condition then compiler will rise an error.

2.If no public element is identified in our java file then it is possible to save java file with any name like abc.java or xyz.java, but, it is suggestible to save java file with main() method class name.

Q)Is it possible to provide more than one public class with in a single java file?

No, it is not possible to provide more than one public class with in a single java file, because, if we provide more than one class as public then we must save that java file with more than one name, it is not possible in all the operating systems.

EX1: File Name: abc.java

```
1)  class FirstApp {
2)          public static void main(String[] args) {
3)                  System.out.println("First Java Application");
4)          }
5)  }
```

Status: No Compilation Error, but not suggestible.

EX2:File Name: FirstApp.java

```
1)  class FirstApp {
2)          public static void main(String[] args) {
3)                  System.out.println("First Java Application");
4)          }
5)  }
```

Status: No COmpilation Error, it is suggestible.

EX3:File Name: FirstApp.java

```
1)  public class A {
2)  }
3)  class FirstApp {
4)          public static void main(String[] args) {
5)                  System.out.println("First Java Application");
6)          }
7)  }
```

Status: Compilation Error.

EX4:File Name: A.java

```
1)  public class A {
2)  }
3)  class FirstApp {
4)          public static void main(String[] args) {
5)                  System.out.println("First Java Application");
6)          }
7)  }
```

Status: No Compilation Error.

EX5: File Name: A.java

```
1)  public class A {
2)  }
3)  public class B {
4)  }
5)  class FirstApp {
6)          public static void main(String[] args) {
7)                  System.out.println("First Java Application");
8)          }
9)  }
```

Status: Compilation Error

5)Compile Java File:

1.The main purpose of compiling JAVA file is to convert JAVA programme from High Level Representation to LowLevelRepresentation.
2.To Check compilation errors.
To compile JAVA file,we have to use the following command on command prompt from the location where JAVA File is Saved

javac File_Name.java

EX: d:\java9>javac FirstApp.java

If we use the above command on command prompt then operating System will perform the following actions.

1.Operating System will take "javac" command from command prompt and search for it at its predefined commands lists and at the locations referred by "path" environment variable.

2.If the required "javac" program is not available at the above two locations then operating System will give a message on command prompt.
"javac can not be recognized as an internal command,external command or operable program and a batch file"

NOTE:To make available all JDK tools like javac,java...to the operating System we have to set "path" environment variable to
"c:\java\jdk1.7.0\bin" location.

D:\java9>set path=C:\Java\jdk1.7.0\bin;

3.If the required "javac" program is identified at "c:\java\jdk1.7.0\bin" location through "path" environment variable then operating System will execute "javac" program and activate "Java Compiler" software.
4.When Java Complier Software is activated then Java Complier will perform the following tasks.

a)Compiler will take java File name from command prompt.
b)Compiler will search java file at current location.
c)If the required java file is not available at current location then compiler will provide the following message on command prompt.

javac : file not found : FirstApp.java

d)If the required java file is available at current location then compiler will start compilation from starting point to ending point of the java file.

e)In compilation process,if any syntax violations are identified then compiler will generate error messages on command prompt.

f)In compilation process,if no syntax errors are identified then compiler will generate .class files at the same current location.

NOTE:Generating no of .class files is completely depending on the no of classes,no of abstract classes,no of interfaces,no of enums and no of inner classes which we used in the present java File

If we want to compile java file from current location and if we want to send the generated .class Files to some other target location then we have to use the following command on command prompt

javac -d target_location File_Name.java

EX:
d:\java9>javac -d c:\abc FirstApp.java
If we use package declaration statement in java file and if we want to store the generated .class files by creating directory structure w.r.t the package name in compilation process then we have to use
"-d" option along with "javac" command.

EX: File Name :D:\java9\FirstApp.java

```
1)  package com.durgasoft.core;
2)  enum E {
3)  }
4)  interface I {
5)  }
6)  abstract class A {
7)  }
8)  class B {
9)      class C {
10)     }
11) }
12) class FirstApp {
13)         public static void main(String[] args) {
14)         }
15) }
```

D:\java9>javac -d c:\abc FirstApp.java
Compiler will compile FirstApp.java File from "D:\java9" location and it will generate E.class, I.class, A.class, B.class, B$C.class, FirstApp.class files at the specified target location "C:\abc" by creating directory structure w.r.t the package name "com.durgasoft.core".

If we want to compile all the java files which are available at current location then we have to use the following command.

D:\java9>javac *.java

If we want to compile all the java files which are prefixed with a word then we have to use the following command.

D:\java9>javac Employee*.java

If we want to compile all the java files which are postfixed with a common word then we have to use the following command.

D:\java9>javac *Address.java

If we want to compile all the java Files which contain particular word then we have to use the following command
D:\java9>javac *Account*.java
D:\java9>javac -d C:\abc *Account*.java

6) Execute JAVA Application:

If we want to execute java program then we have to use the following command prompt.

java Main_Class_Name

EX:
d:\java9>java FirstApp

If we use the above command on command prompt then operating System will execute "java" operable program at "C:\java\jdk1.7.0\bin",with this,JVM software will be activated and JVM will perform the following actions.

JVM will take Main_Class name from command prompt.

JVM will search for Main_Class at current location,at Java predefined library and at the locations referred by "classpath" environment variable.

If the required Main_Class .class file is not available at all the above locations then JVM will provide the following.
JAVA6: java.lang.NoClassDefFoundError:FirstApp
JAVA7: Error:Could not find or load main class FirstApp

NOTE:If main class .class file is available at some other location then to make available main class.class file to JVM we have to set "classpath" environment varaible.

D:\java7>set classpath=E:\XYZ;

If the required main class .class file is available at either of the above locations then JVM will load main class bytecode to the memory by using "Class Loaders".

After loading main class bytecode to the memory,JVM will search for main() method.

If main() method is not available at main class byteCode then JVM will provide the following.

JAVA6: java.lang.NoSuchMethodError:main

JAVA7: Error:Main method not found in class FirstApp,please define main method as:
public static void main(String args[])

If main() method is available at main class bytecode then JVM will access main() method by creating a thread called as "Main thread".

JVM will access main() method to start application execution by using main thread,when main thread reached to the ending point of main() method then main thread will be in destroyed/dead state.

When main Thread is in dead state then JVM will stop all of its internal processes and JVM will go to ShutDown mode.