

Gradle Assignment

Instructions :

Following parameters are to be considered for exercise submission:

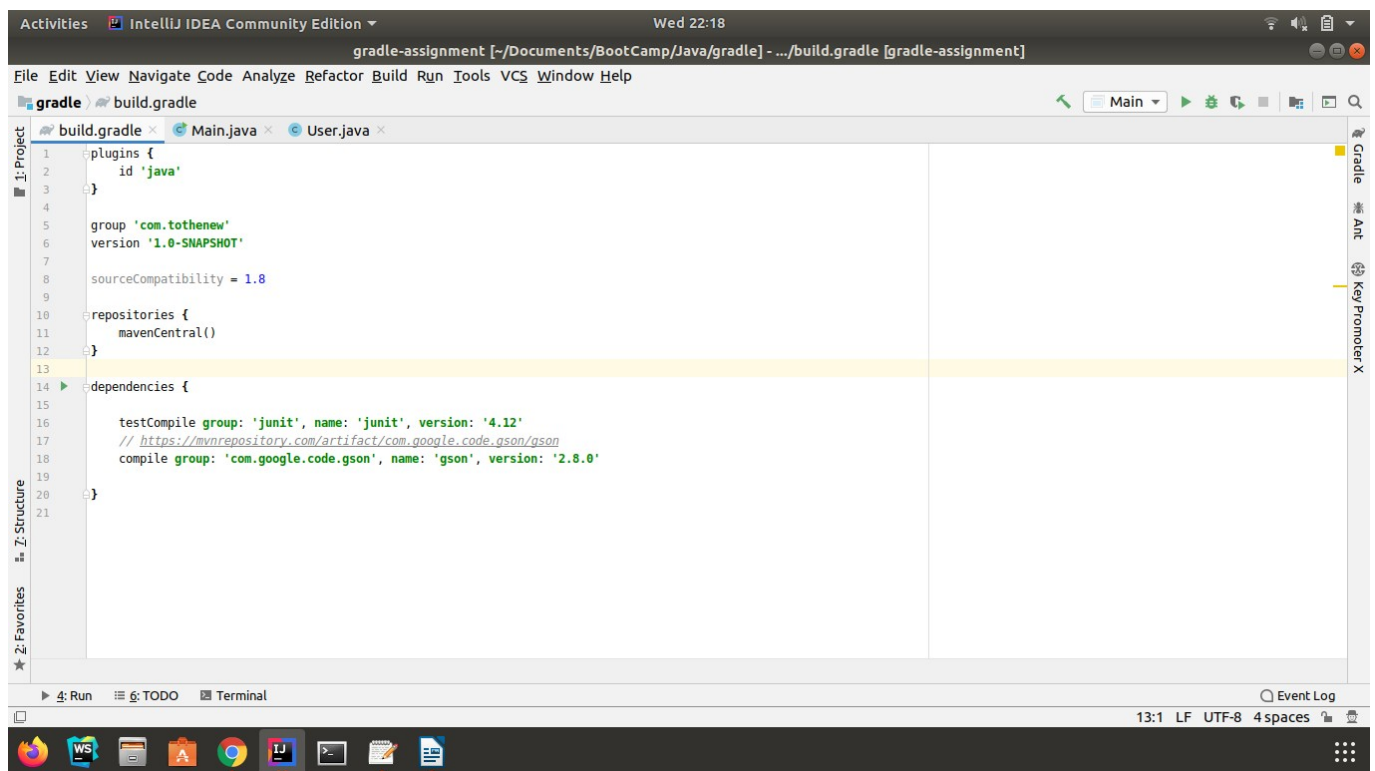
- Complete project has to be submitted.
- .gradle, .idea, build directories should be excluded
- A text file containing the relevant URLs which are referenced and any other documentation.

Questions :

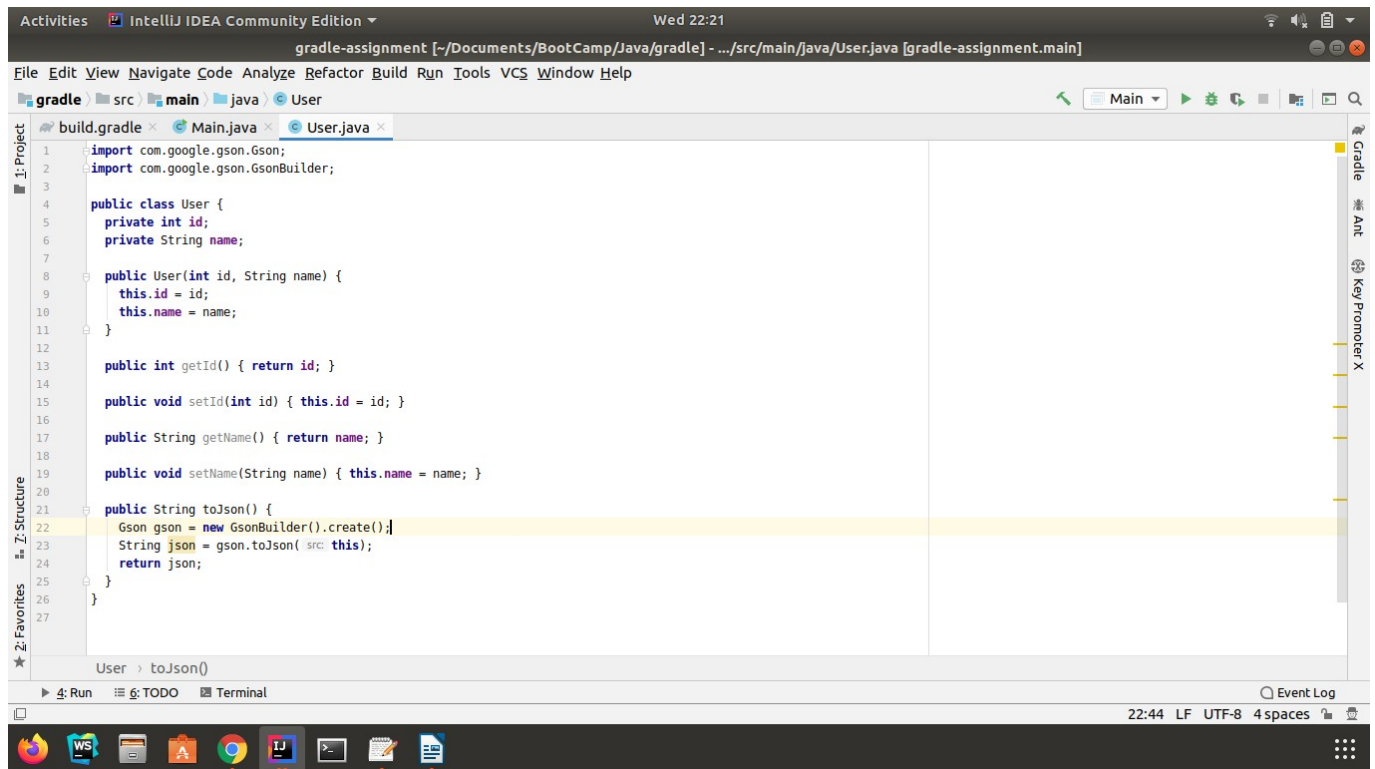
1) Add a gradle dependency and its related repository url.

A dependency can be added in Gradle by mentioning it in the build.gradle file inside the 'dependencies' section.

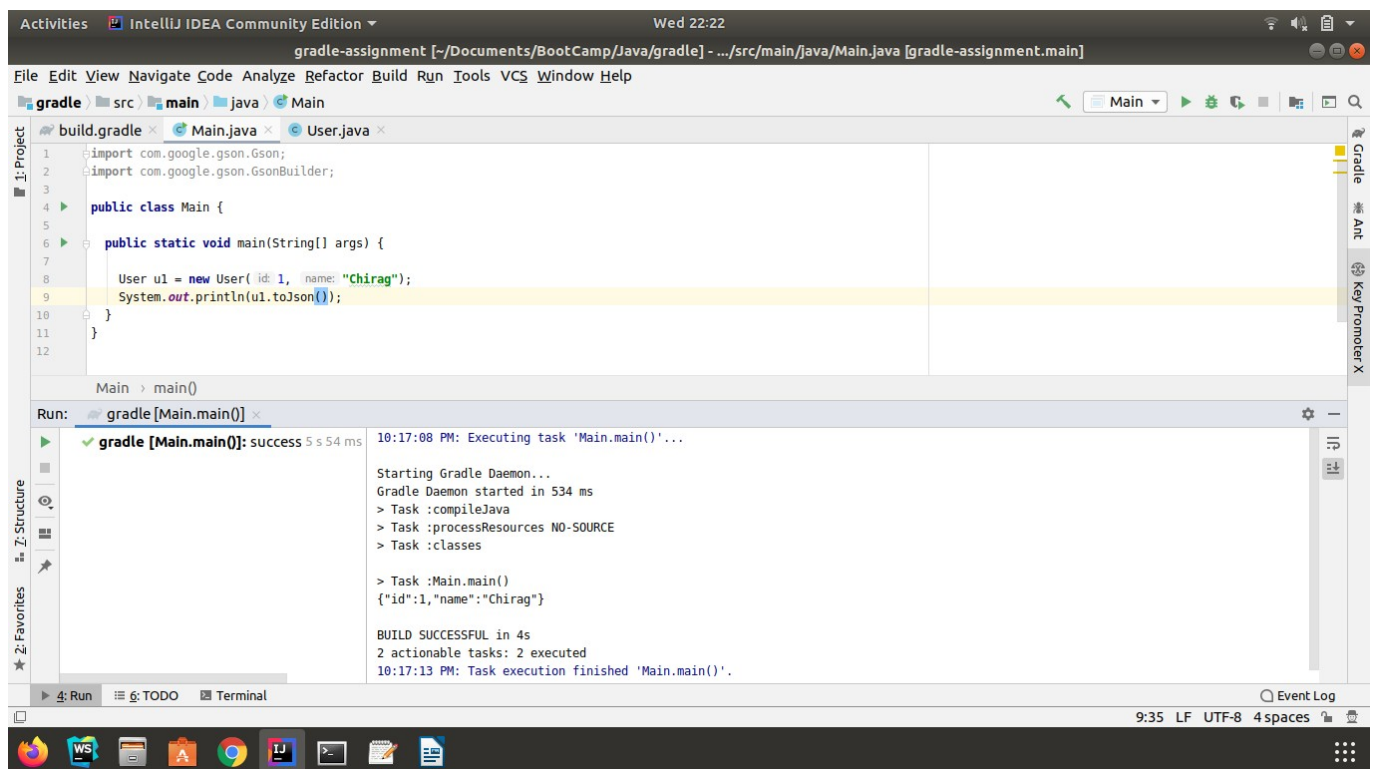
build.gradle



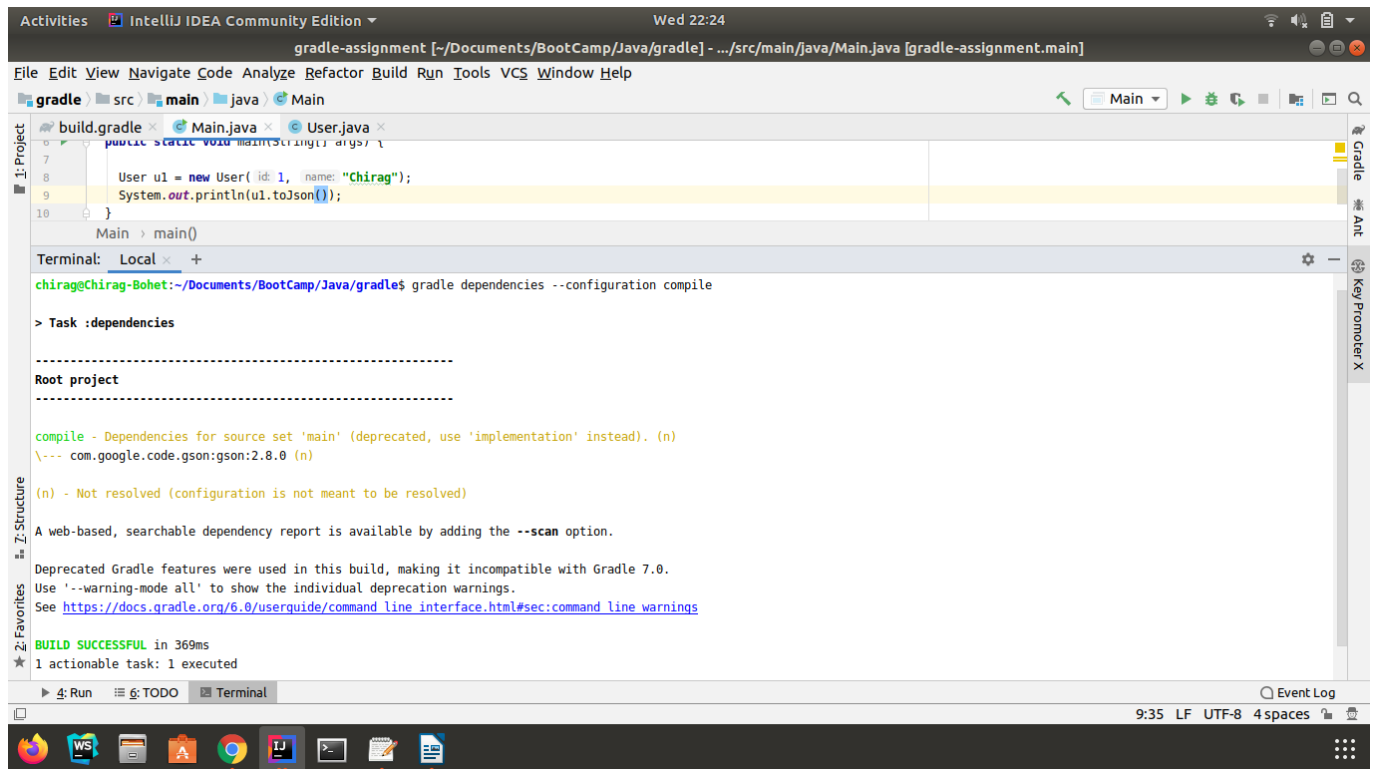
User.java



Main.java



Output of : 'gradle dependencies --configuration compile'



The screenshot shows the IntelliJ IDEA Community Edition interface. The top toolbar includes buttons for Run, Build, and other IDE functions. The main editor displays the `Main.java` file with the following code:

```
1 public static void main(String[] args) {  
2     // ...  
3     User u1 = new User(1, "Chirag");  
4     System.out.println(u1.toJson());  
5 }  
6  
7 Main > main()
```

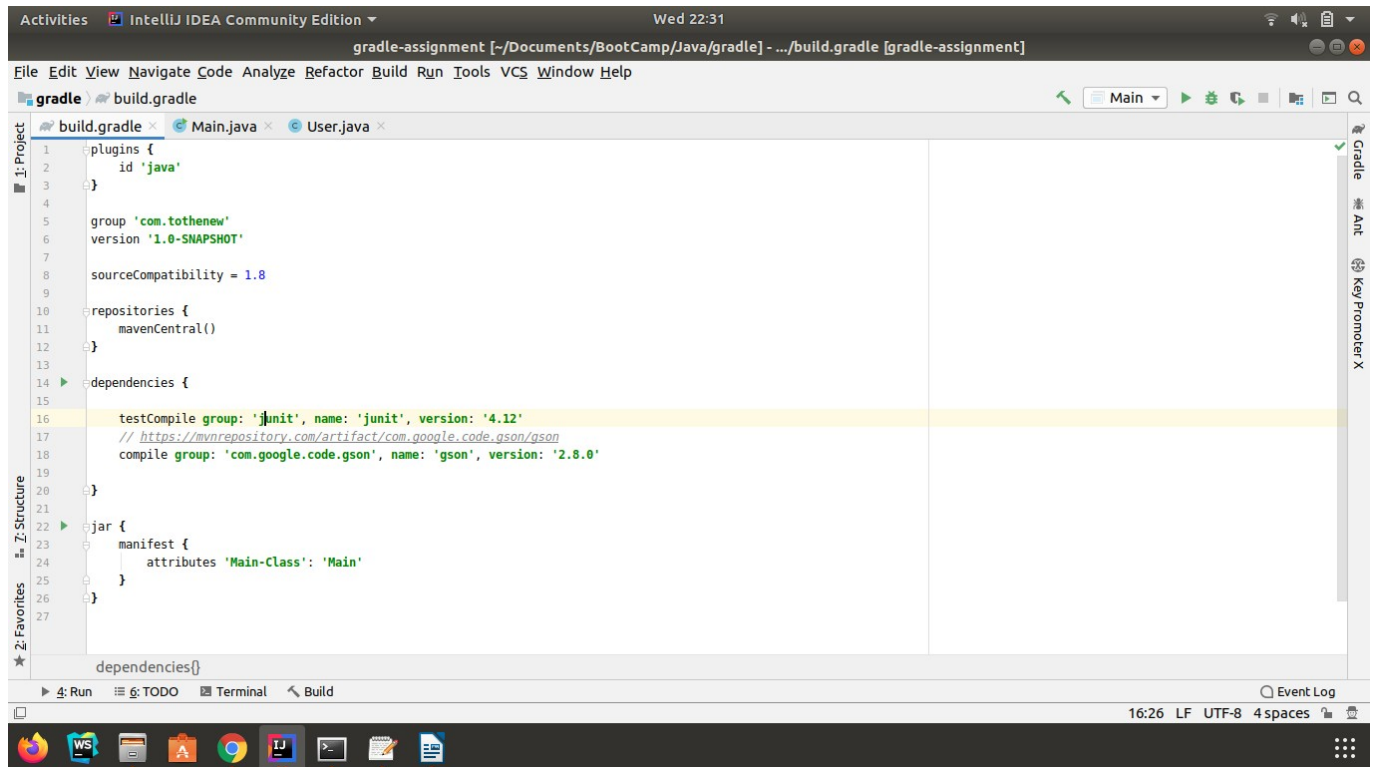
The terminal window at the bottom shows the output of the command `gradle dependencies --configuration compile`:

```
chirag@Chirag-Bohet:~/Documents/BootCamp/Java/gradle$ gradle dependencies --configuration compile  
  
> Task :dependencies  
  
-----  
Root project  
-----  
  
compile - Dependencies for source set 'main' (deprecated, use 'implementation' instead). (n)  
\\--- com.google.code.gson:gson:2.8.0 (n)  
  
(n) - Not resolved (configuration is not meant to be resolved)  
  
A web-based, searchable dependency report is available by adding the --scan option.  
  
Deprecated Gradle features were used in this build, making it incompatible with Gradle 7.0.  
Use '--warning-mode all' to show the individual deprecation warnings.  
See https://docs.gradle.org/6.0/userguide/command\_line\_interface.html#sec:command\_line\_warnings  
  
BUILD SUCCESSFUL in 369ms  
1 actionable task: 1 executed
```

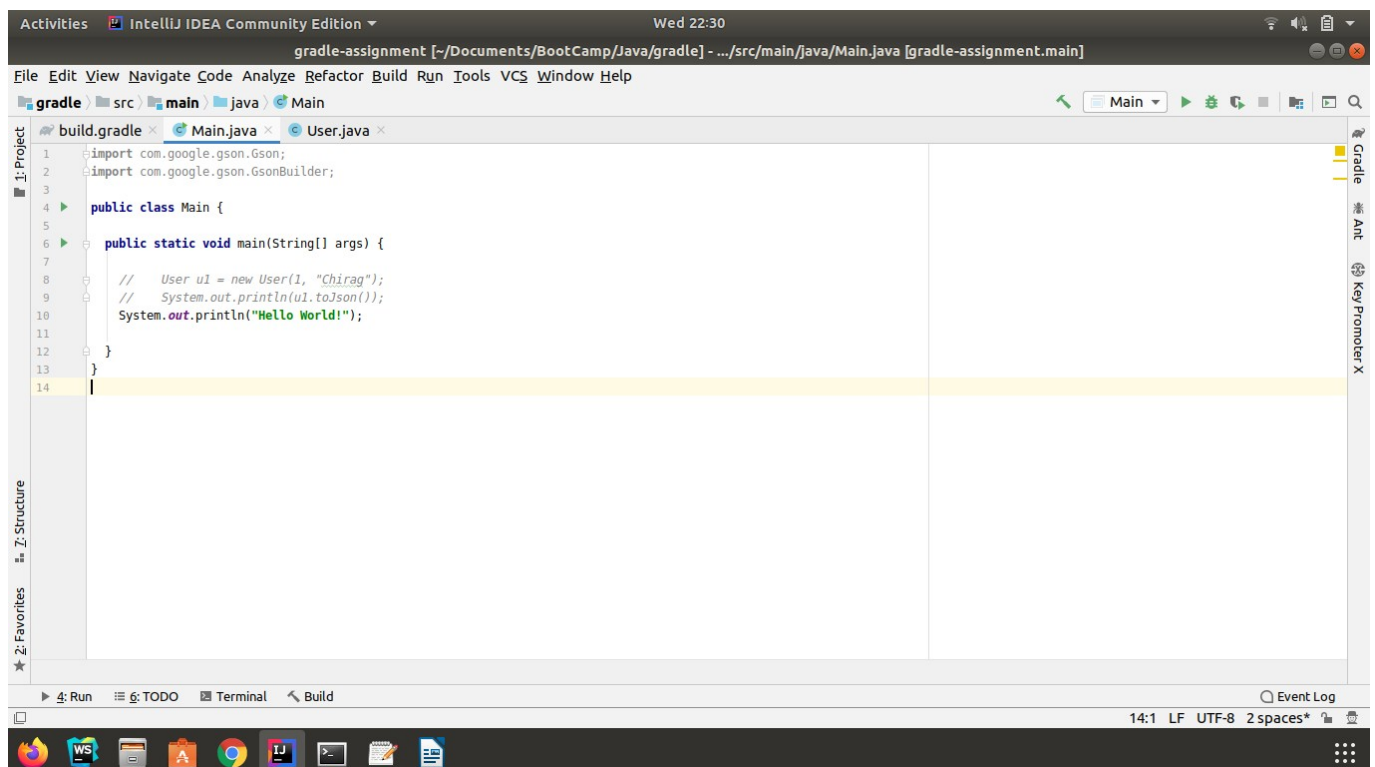
The bottom status bar shows the time as 9:35, the file encoding as UTF-8, and 4 spaces for indentation.

2) Using java plugin, make changes in the manifest to make the jar executable. Using `java -jar JAR_NAME`, the output should be printed as "Hello World"

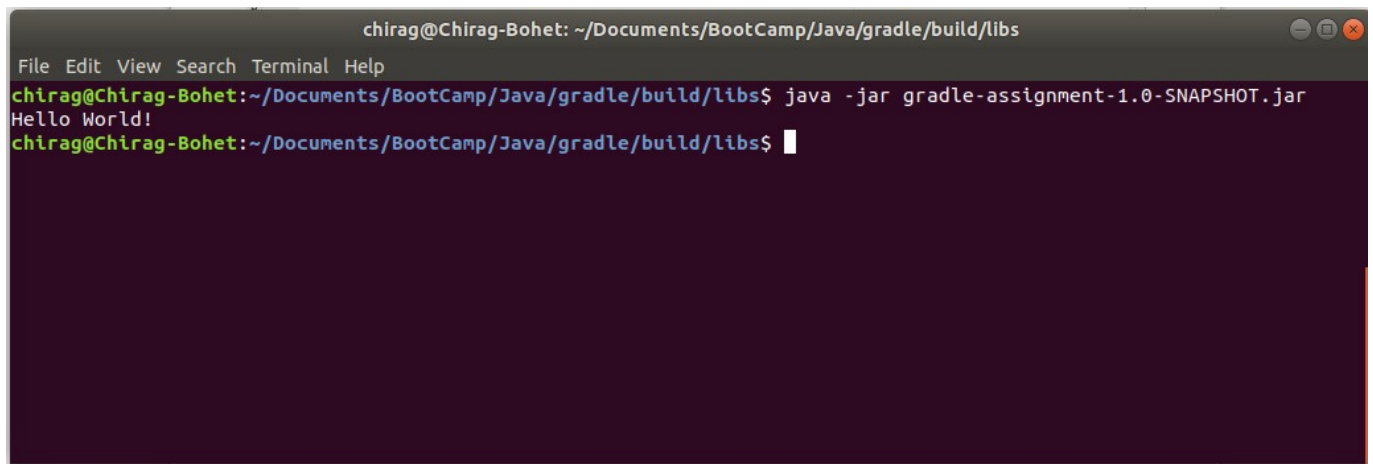
build.gradle



Main.java



Output of : 'java -jar gradle-assignment-1.0-SNAPSHOT.jar'

A screenshot of a terminal window with a dark background. The title bar at the top reads 'chirag@Chirag-Bohet: ~/Documents/BootCamp/Java/gradle/build/libs'. Below the title bar is a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal shows a prompt 'chirag@Chirag-Bohet:~/Documents/BootCamp/Java/gradle/build/libs\$' followed by the command 'java -jar gradle-assignment-1.0-SNAPSHOT.jar'. The output of the command is 'Hello World!'. The prompt is followed by a cursor.

```
chirag@Chirag-Bohet: ~/Documents/BootCamp/Java/gradle/build/libs
File Edit View Search Terminal Help
chirag@Chirag-Bohet:~/Documents/BootCamp/Java/gradle/build/libs$ java -jar gradle-assignment-1.0-SNAPSHOT.jar
Hello World!
chirag@Chirag-Bohet:~/Documents/BootCamp/Java/gradle/build/libs$
```

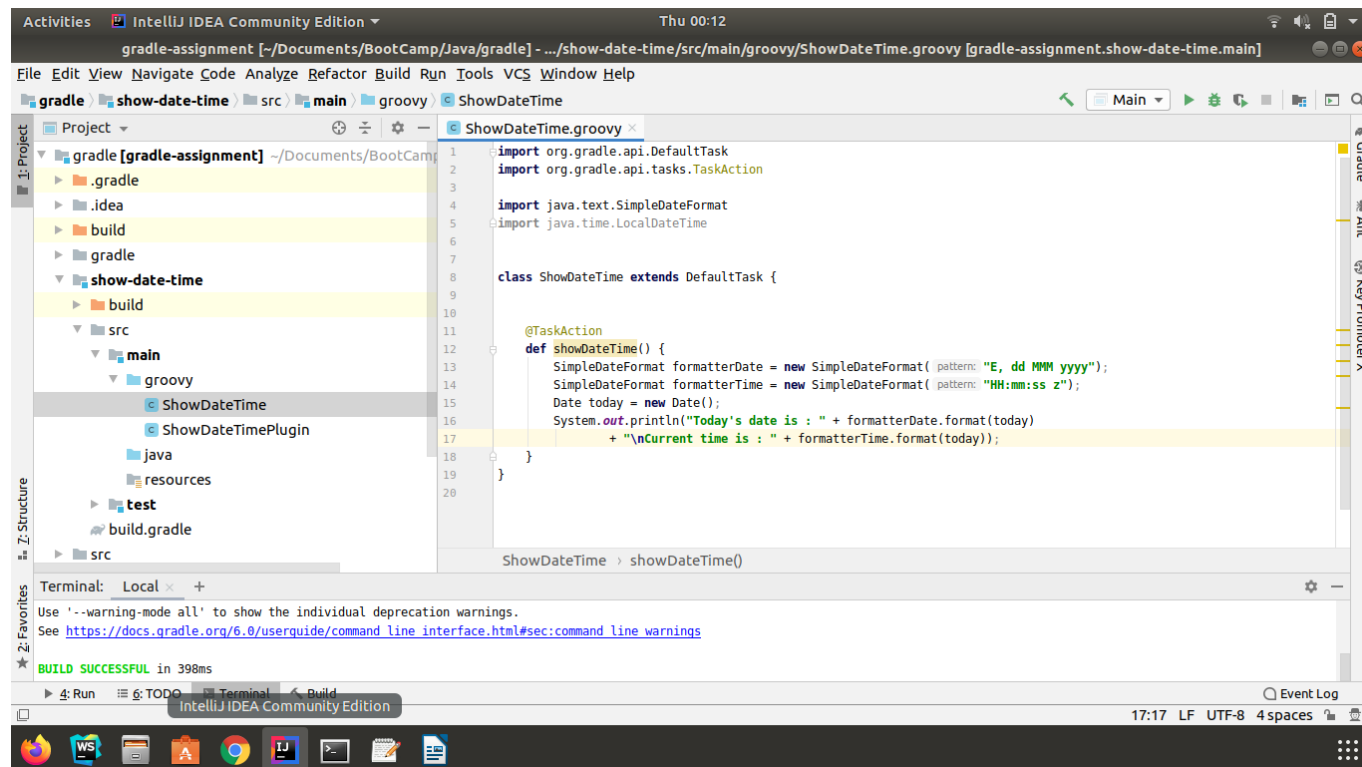
resources : <https://stackoverflow.com/questions/21721119/creating-runnable-jar-with-gradle>

3) Differentiate between the different dependency scopes: compile, runtime, testCompile, testRuntime using different dependencies being defined in your build.gradle.

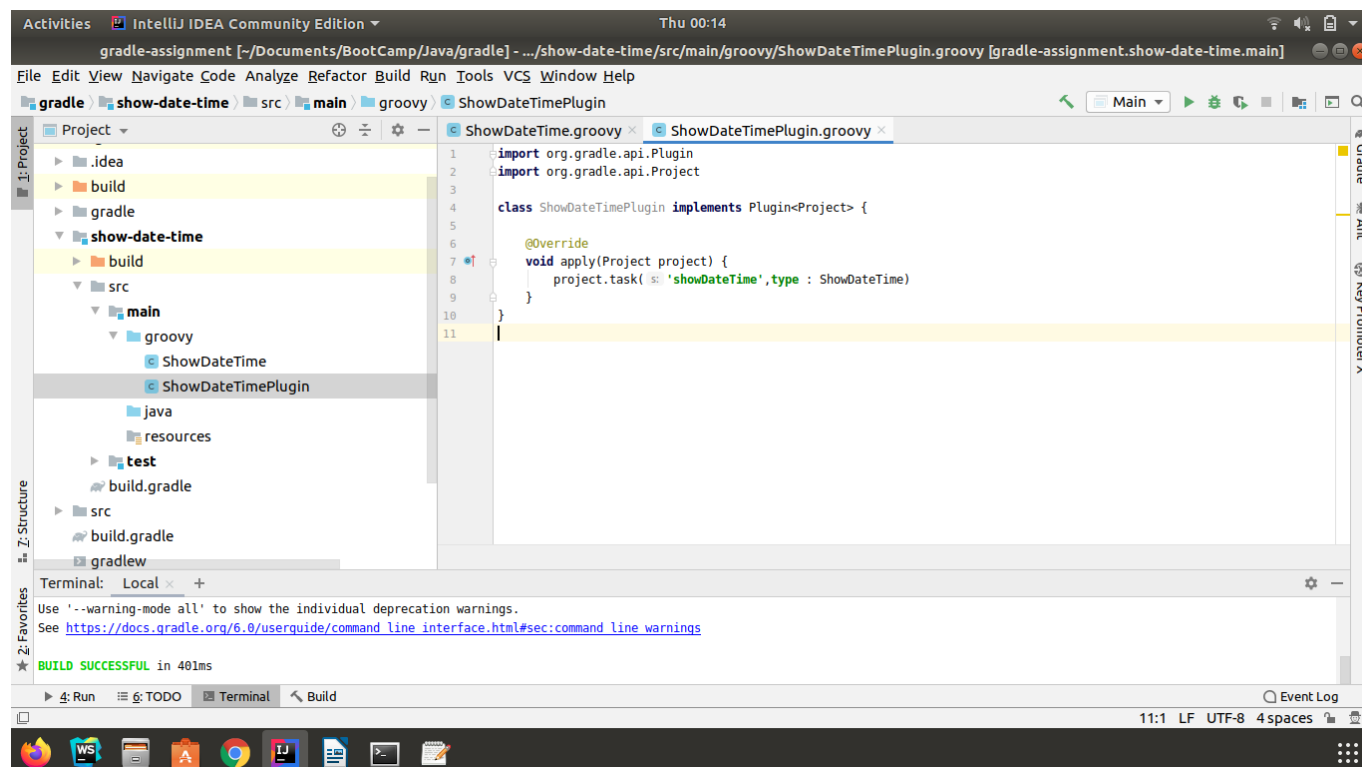
https://docs.gradle.org/5.3.1/userguide/java_plugin.html#sec:java_plugin_and_dependency_management

4) Create a custom plugin which contains a custom task which prints the current date-time.

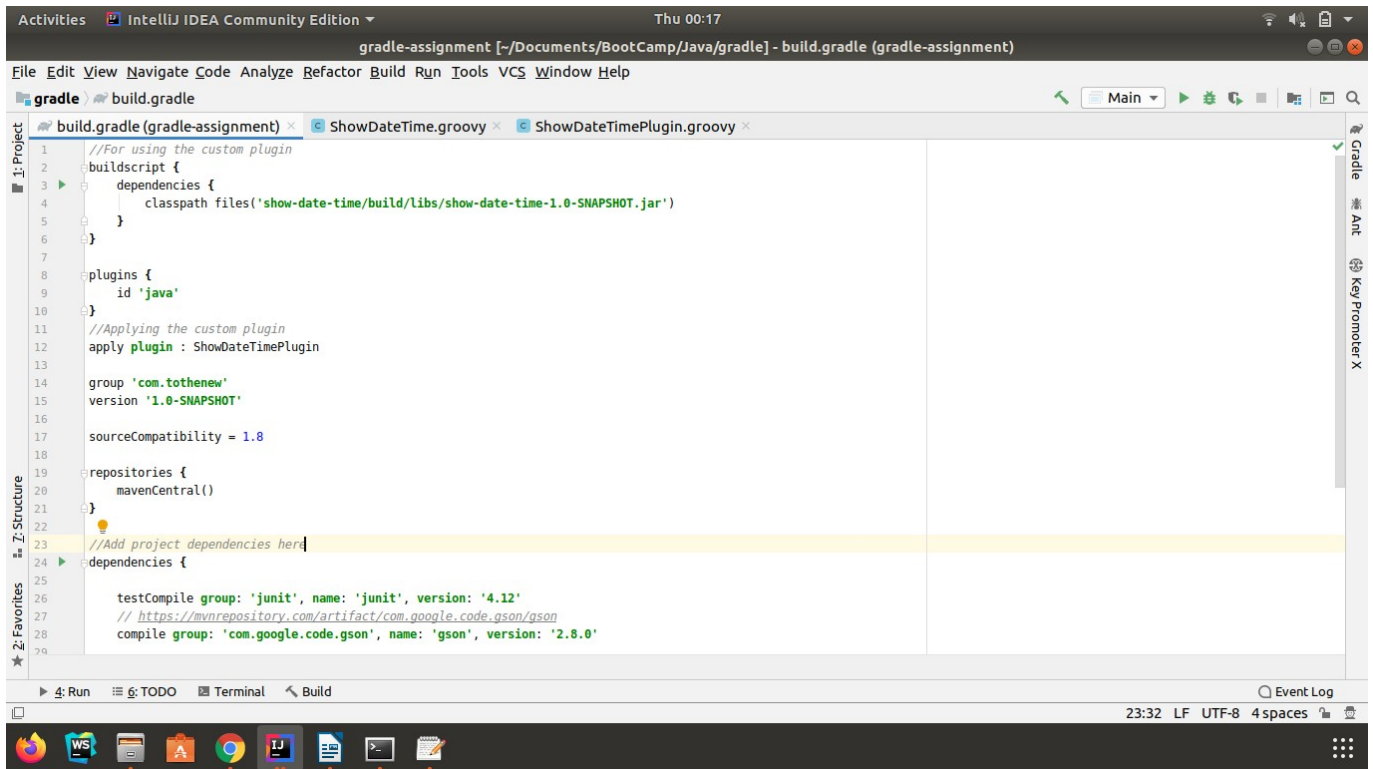
ShowDateTime.groovy



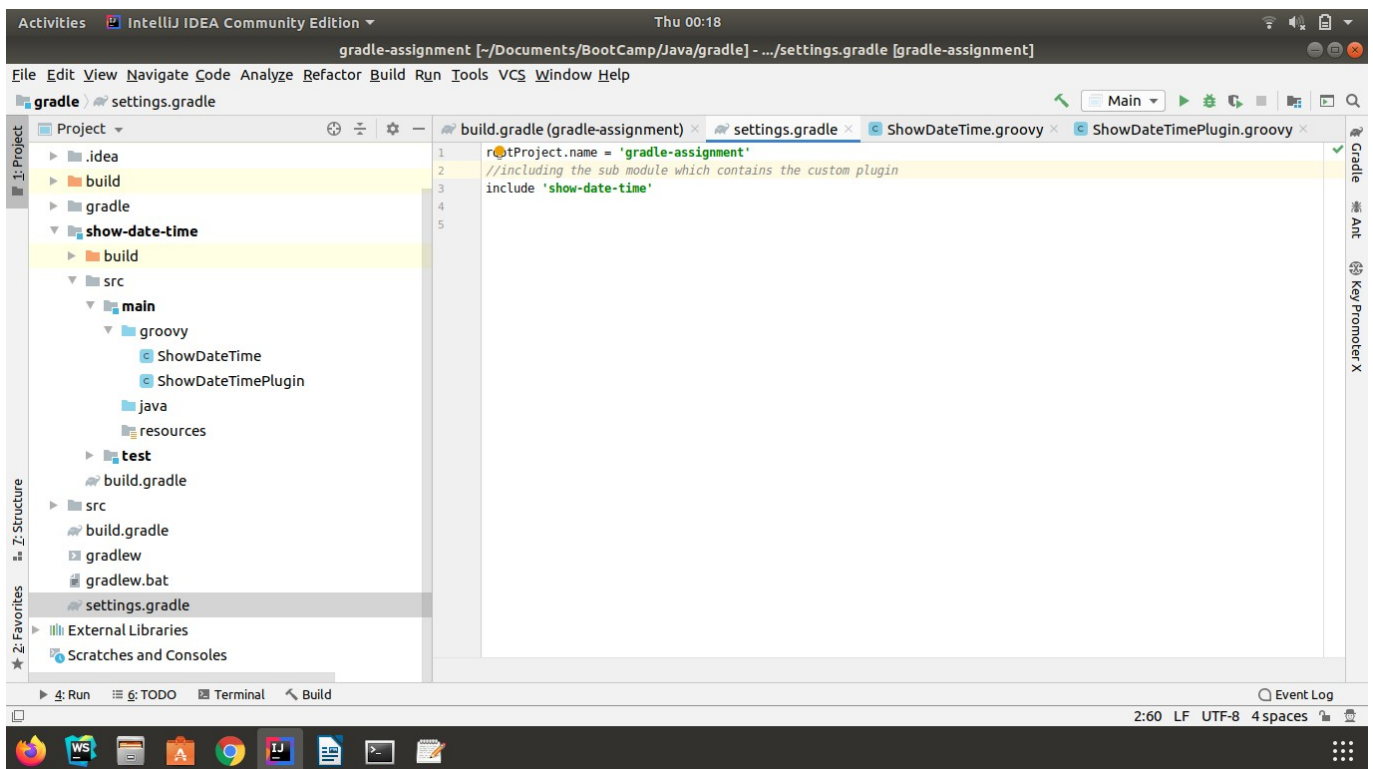
ShowDateTimePlugin.groovy



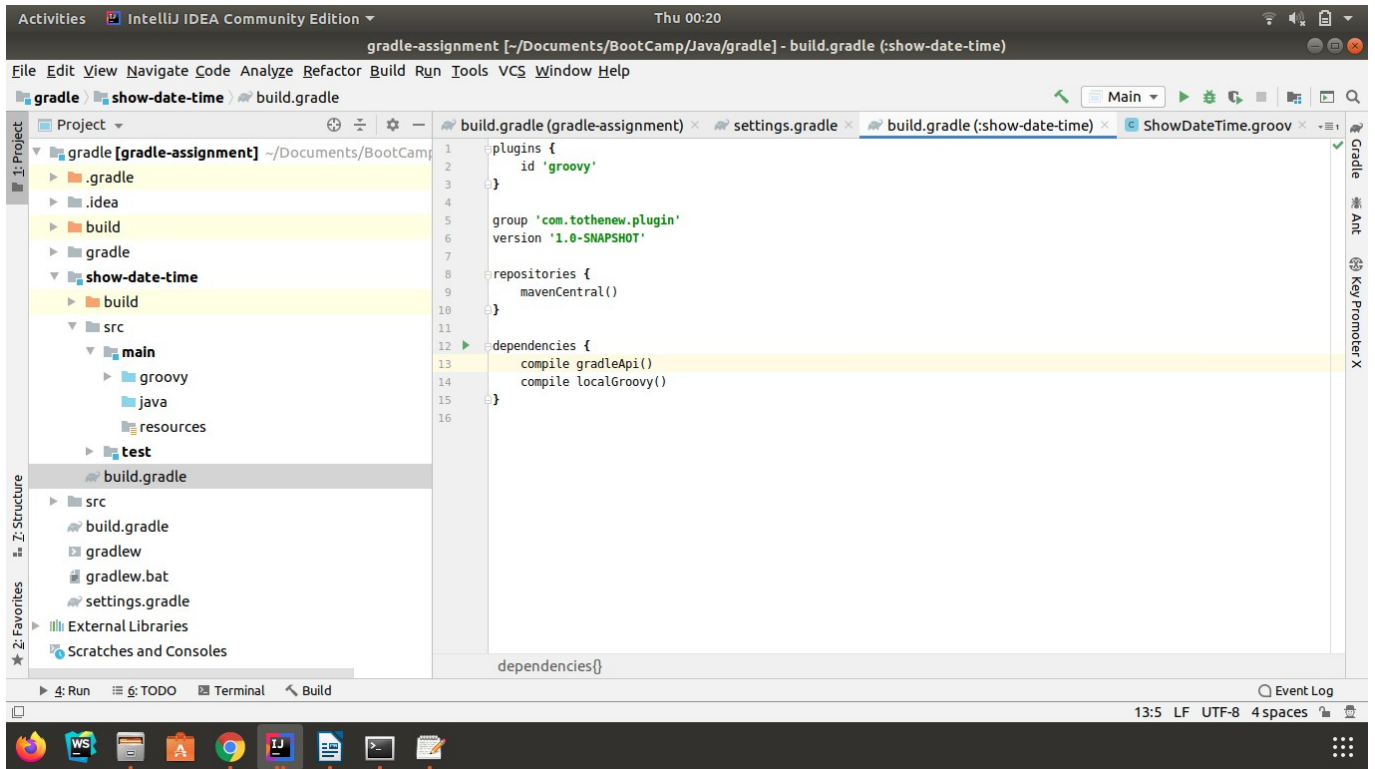
build.gradle (root project)



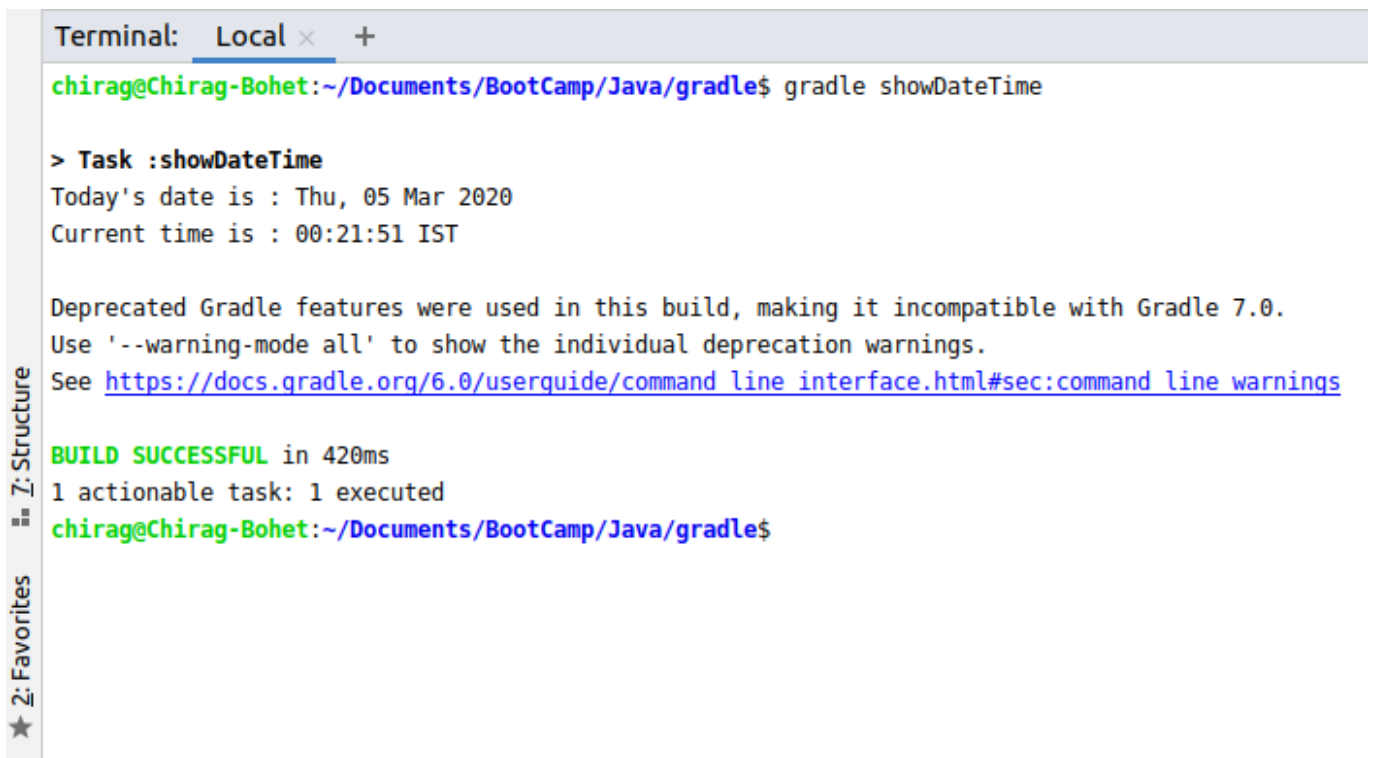
settings.gradle (root project)



build.gradle (plugin module)

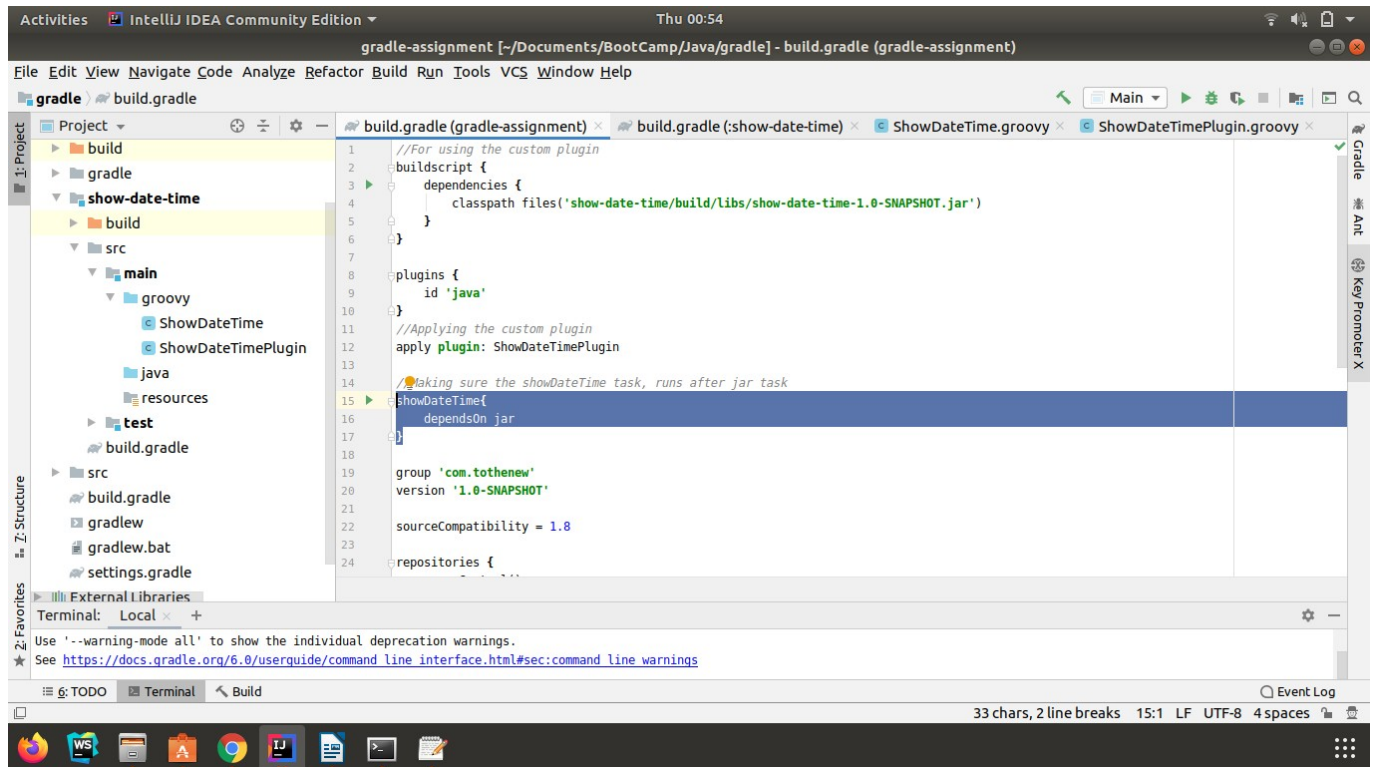


Output of : gradle showDateTime



5) Using that plugin in your project, execute that task after the jar task executes.

build.gradle



Output of : gradle showDateTime --console verbose

```
Terminal: Local x +
chirag@Chirag-Bohet:~/Documents/BootCamp/Java/gradle$ gradle showDateTime --console verbose
> Task :compileJava UP-TO-DATE
> Task :processResources NO-SOURCE
> Task :classes UP-TO-DATE
> Task :jar UP-TO-DATE

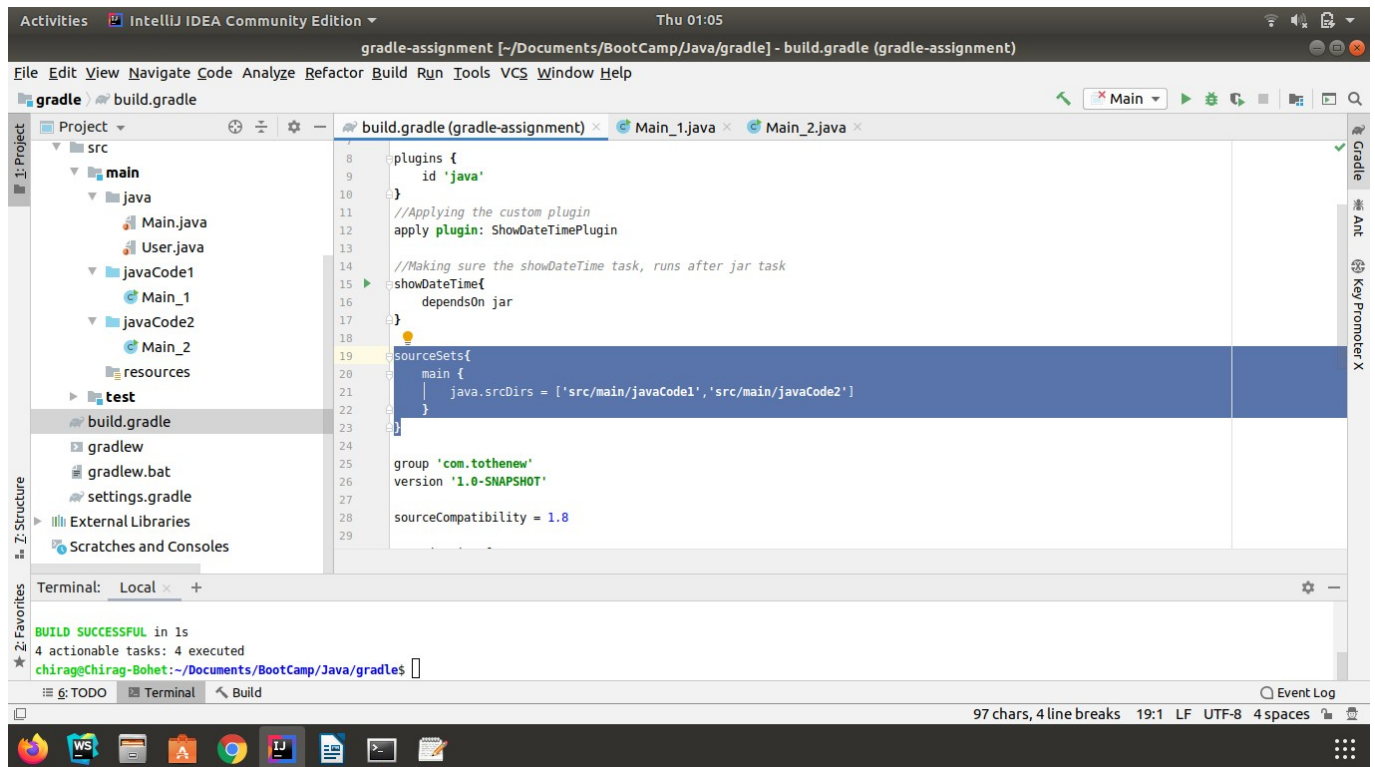
> Task :showDateTime
Today's date is : Thu, 05 Mar 2020
Current time is : 00:55:40 IST

Deprecated Gradle features were used in this build, making it incompatible with Gradle 7.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/6.0/userguide/command\_line\_interface.html#sec:command\_line\_warnings

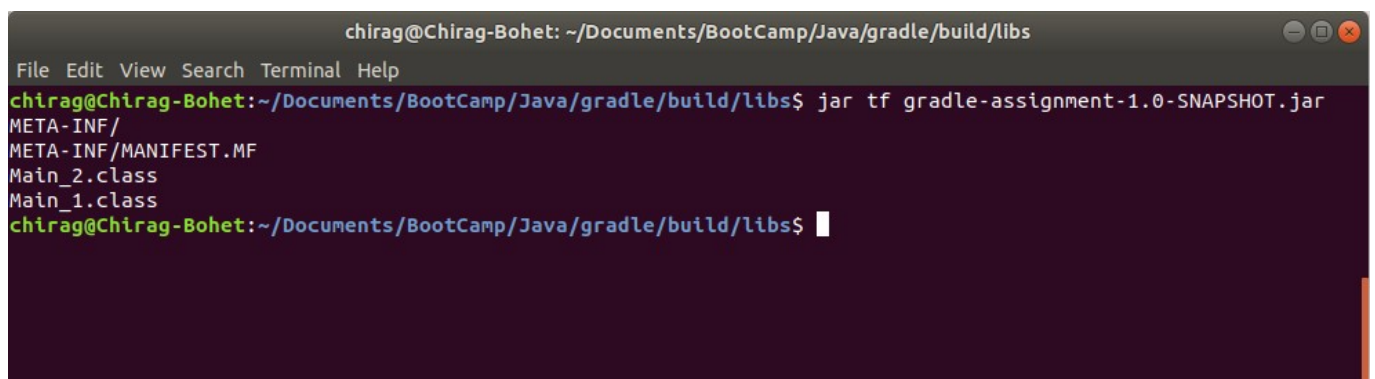
BUILD SUCCESSFUL in 394ms
3 actionable tasks: 1 executed, 2 up-to-date
```

6) Instead of using default source set, use src/main/javaCode1, src/main/javaCode2 to be taken as code source. Make sure that the JAR created contains files from both the directories and not from src/main/java.

Build.gradle

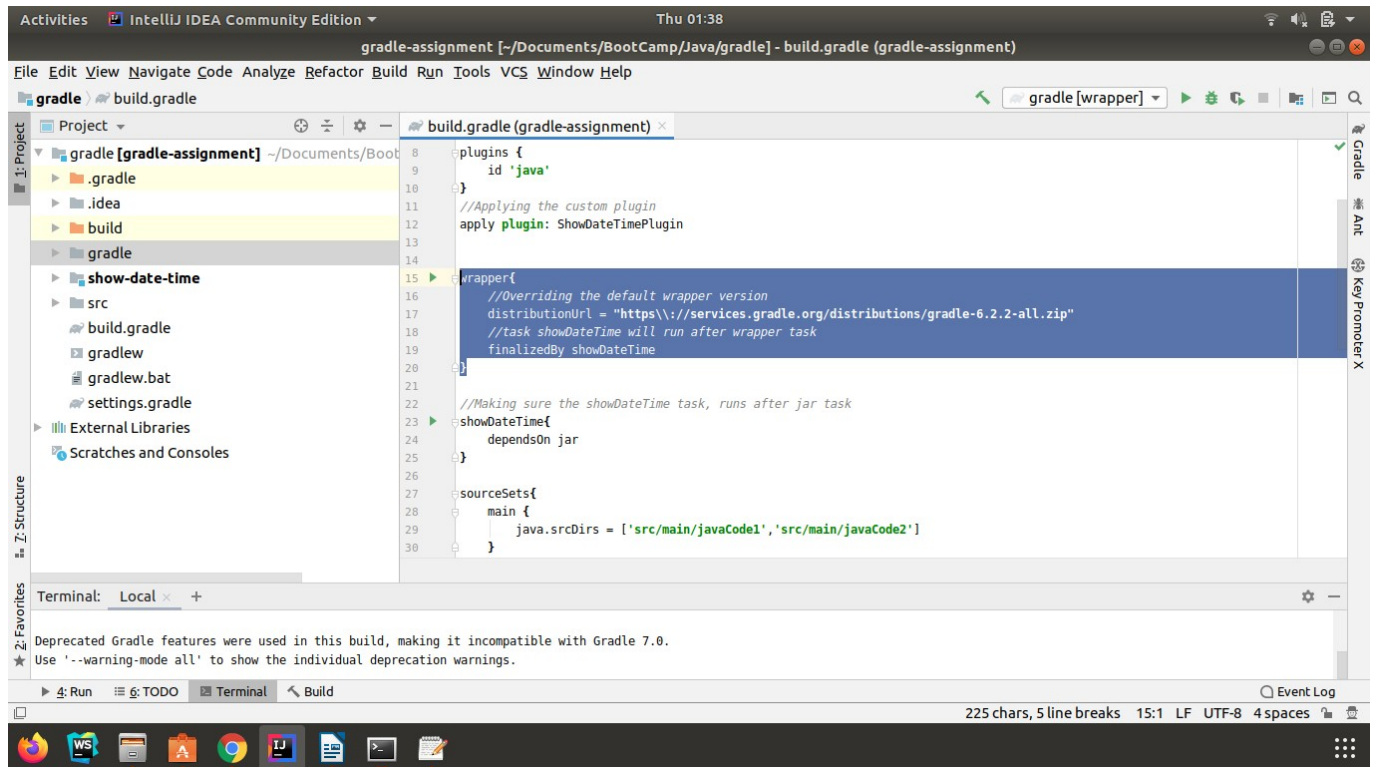


Output of : jar tf gradle-assignment-1.0-SNAPSHOT.jar

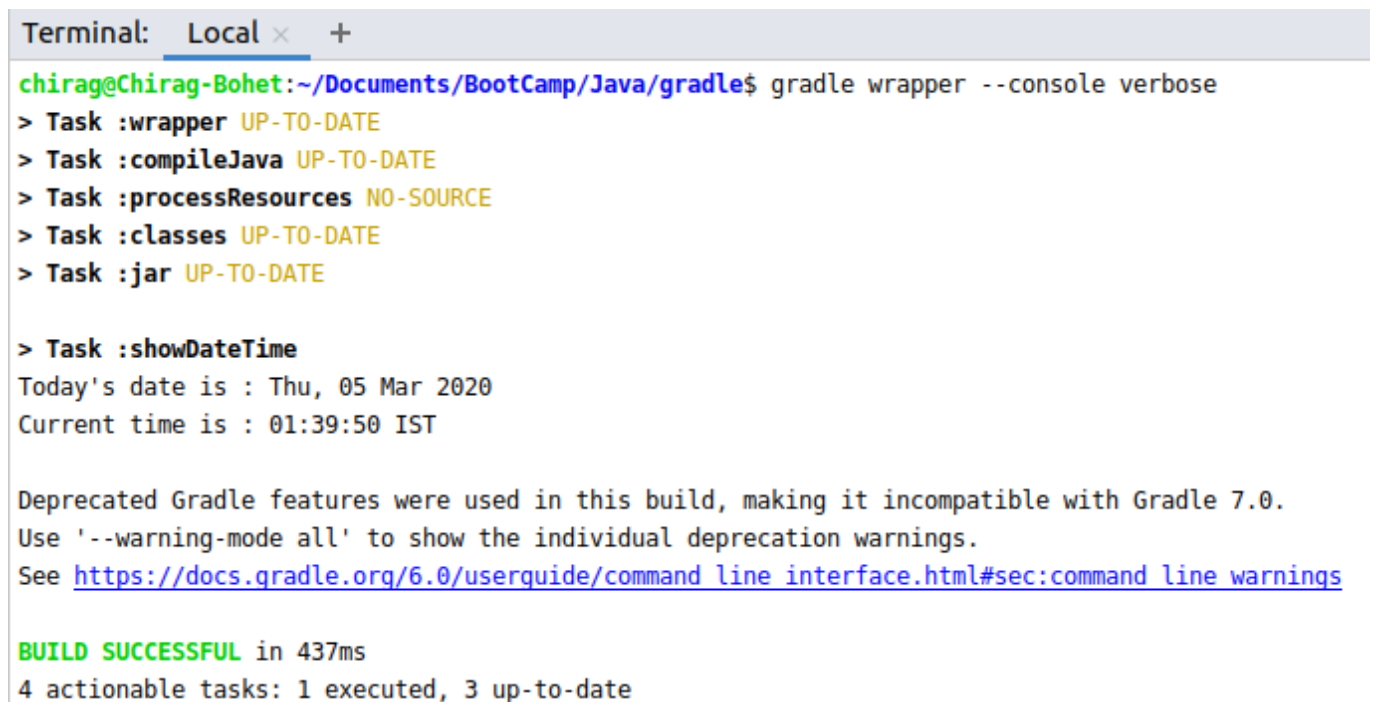


7) Override the Gradle Wrapper task to install a different version of gradle. Make sure that the task written in Q4 also executes with it.

build.gradle



output of : gradle wrapper --console verbose



8) Run the gradle profile command and attach the resulting files.

Profiling output can be seen at this link :

<https://github.com/chiragbohet/ttn-java-bootcamp/tree/gradle/gradle/profile>