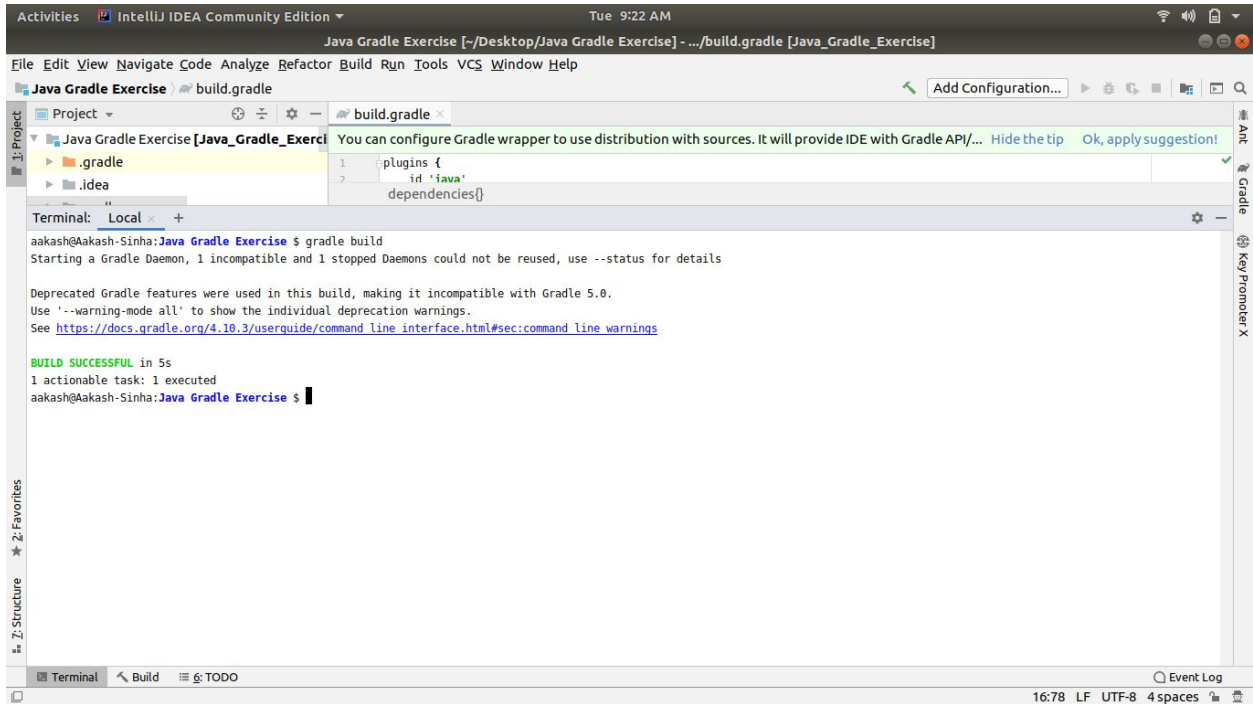


//Ques1: Add a gradle dependency and its related repository url.

```
dependencies {  
    testCompile group: 'junit', name: 'junit', version: '4.12'  
    compileOnly group: 'com.google.code.gson', name: 'gson', version: '2.8.6'  
}
```



//Ques2: 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"

Step-1

#Add the manifest in build.gradle

```
jar{
    manifest{
        attributes(
            "Main-Class": 'MyMain',
        )
    }
}
```

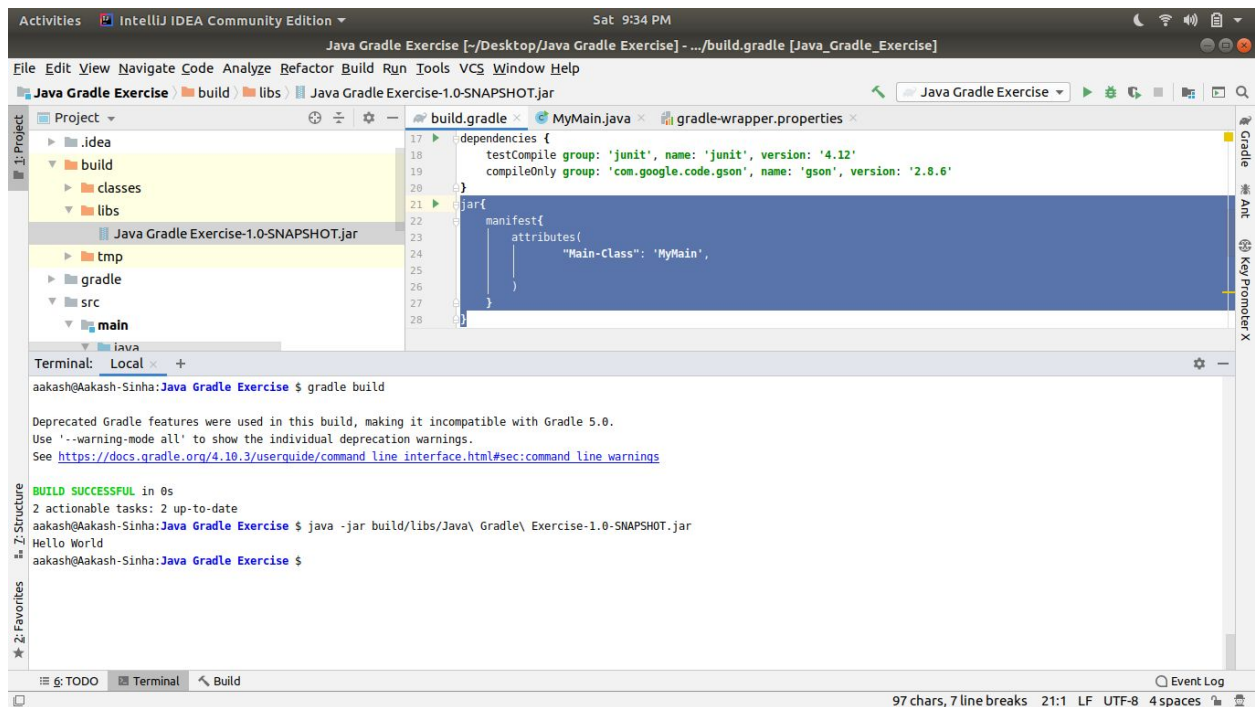
Step-2

#Build the gradle using command - gradle build

Step-3

#A jar file would be created - Run the file using the command

java -jar build/libs/Java\ Gradle\ Exercise-1.0-SNAPSHOT.jar



//Ques3: Differentiate between the different dependency scopes: compile, runtime, testCompile, testRuntime using different dependencies being defined in your build.gradle.

Compile: The compile scope is the default scope. We can use it when we have no special requirements for declaring a certain dependency.

Runtime: We use the runtime scope for dependencies that are not needed at compile time, like when we're compiling against an API and only need the implementation of that API at runtime.

testCompile: We can use the testCompile scope for dependencies that are only needed in tests compile time and that should not be available in production code.

testRuntime: We use the runRuntime scope for dependencies that are not needed at compile time, like when we're compiling against an API and only need the implementation of that API at runtime.

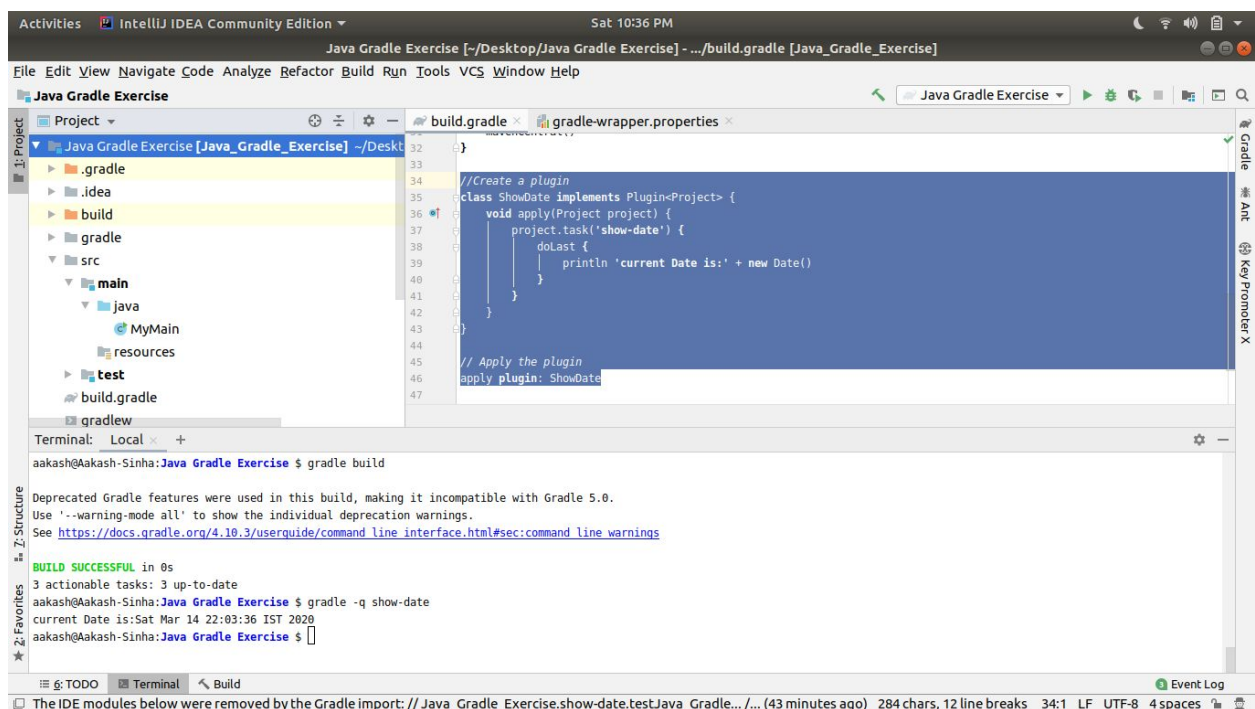
//Ques 4: Create a custom plugin which contains a custom task which prints the current date-time. Using that plugin in your project, execute that task after the jar task executes.

#Add the plugin to build.gradle

```
class ShowDate implements Plugin<Project> {  
    void apply(Project project) {  
        project.task('show-date') {  
            doLast {  
                println 'current Date is:' + new Date()  
            }  
        }  
    }  
}
```

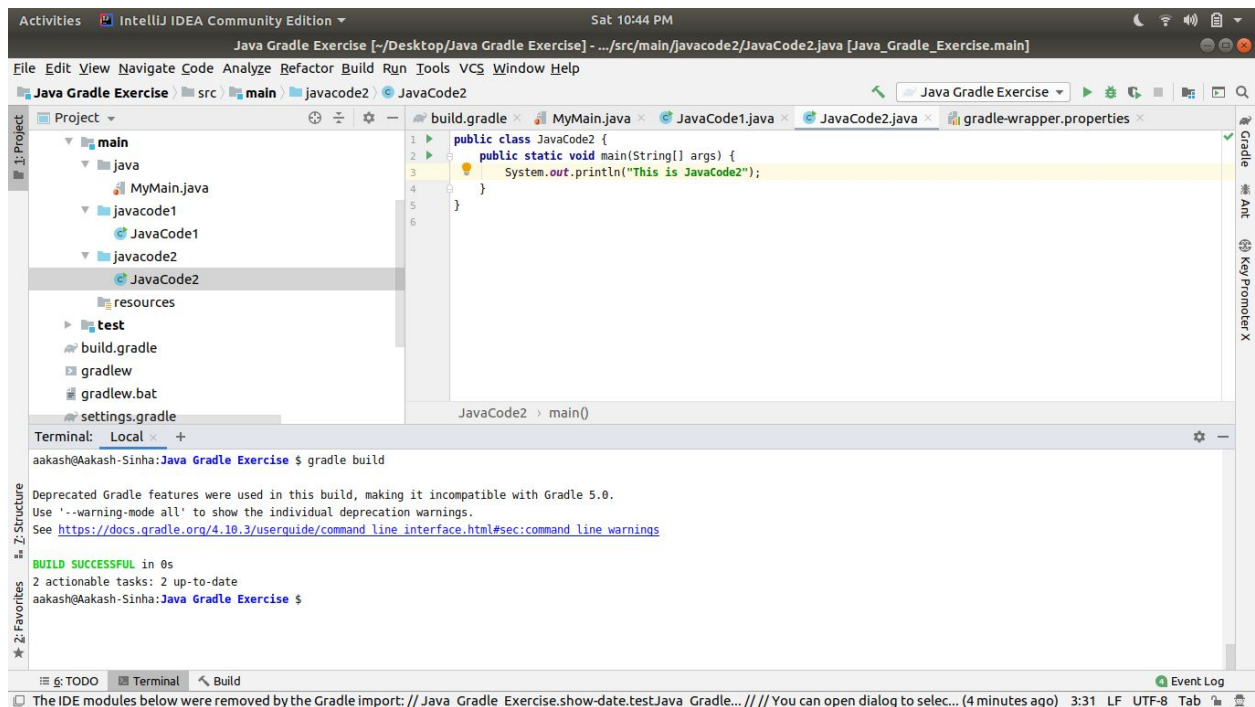
#Apply the plugin

apply plugin: ShowDate

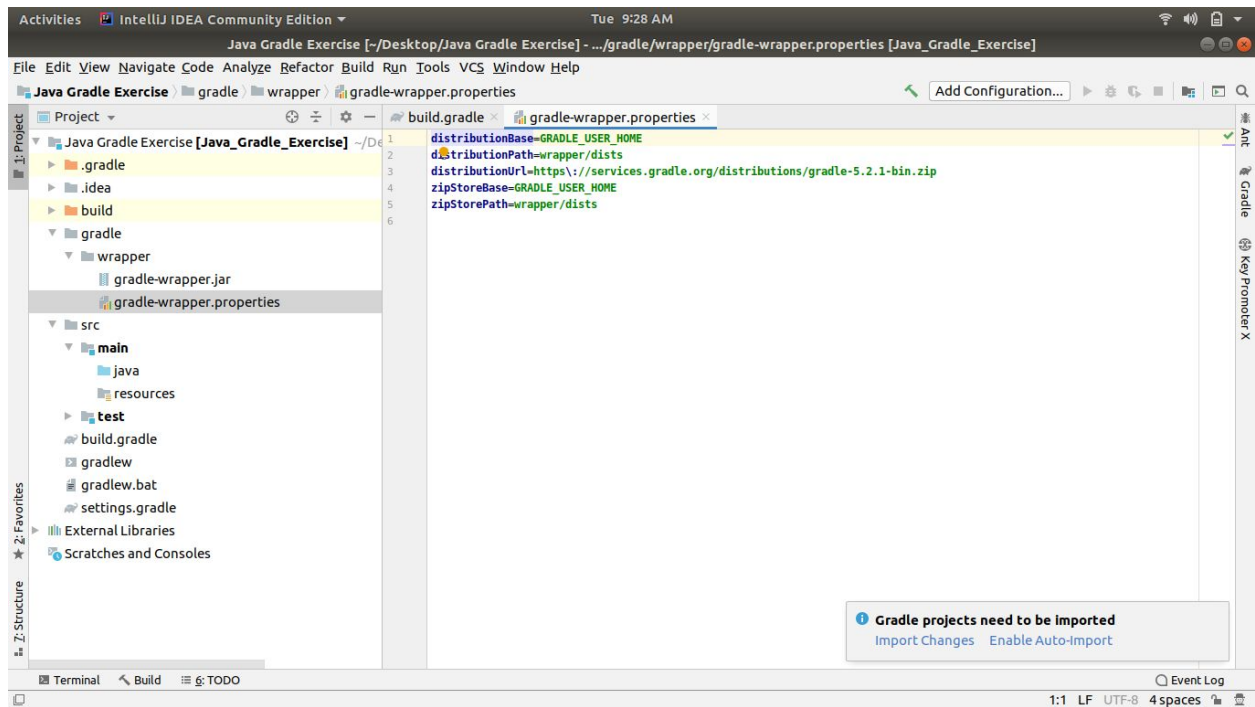


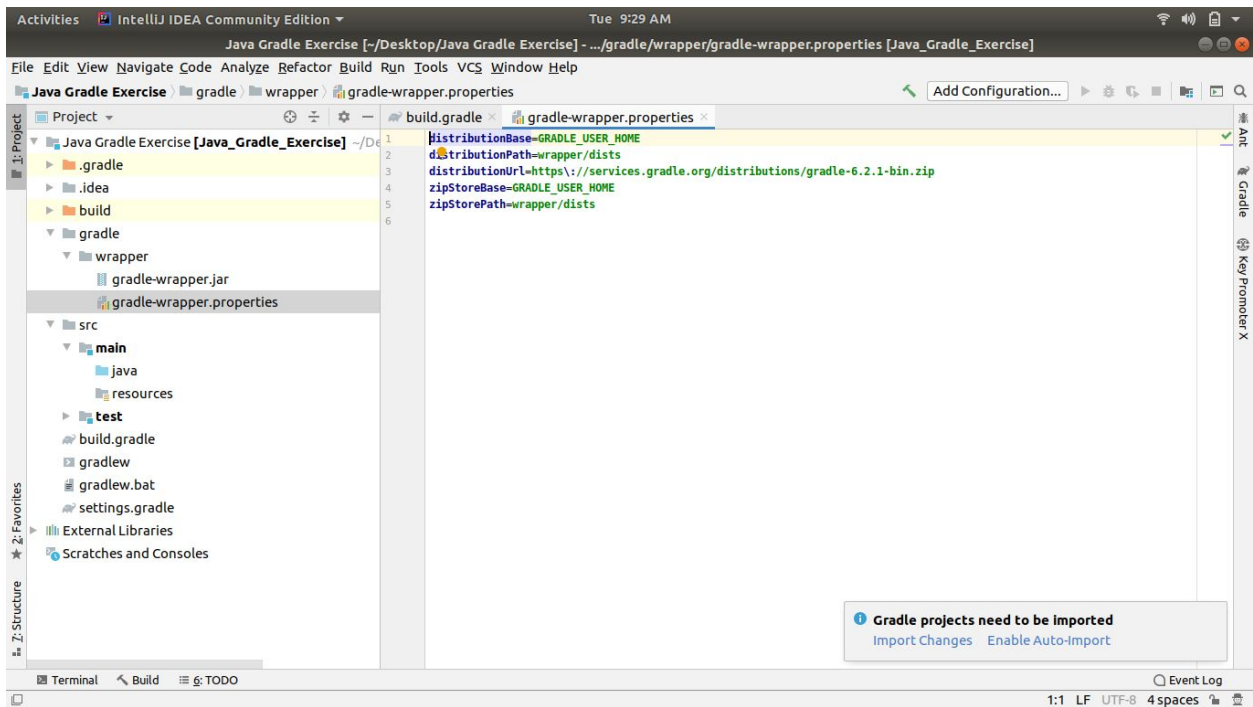
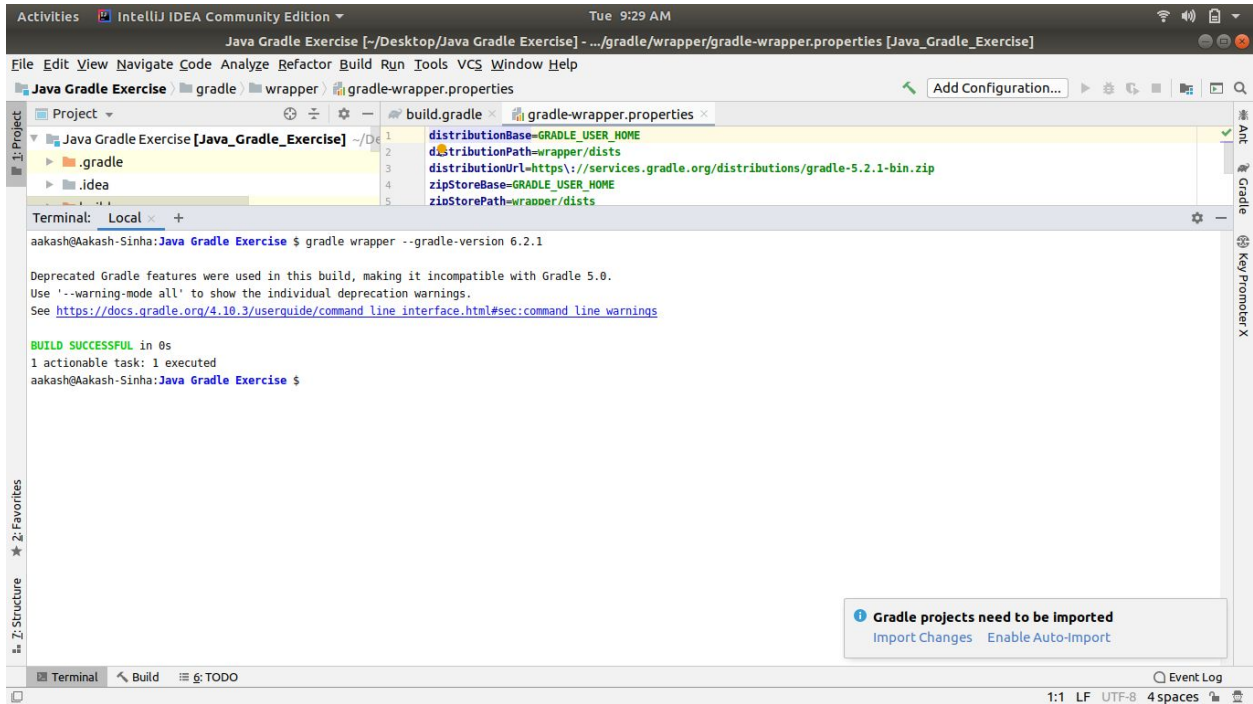
//Ques5: Instead of using default source set, use src/main/javaCode1, src/main/javaCode2 to be
//taken as code source. Make sure that the JAR created contains files from both the directories
//and not from src/main/java.

```
sourceSets{
main
{
java.srcDirs= ['src/main/JavaCode1','src/main/JavaCode2']
}
}
```

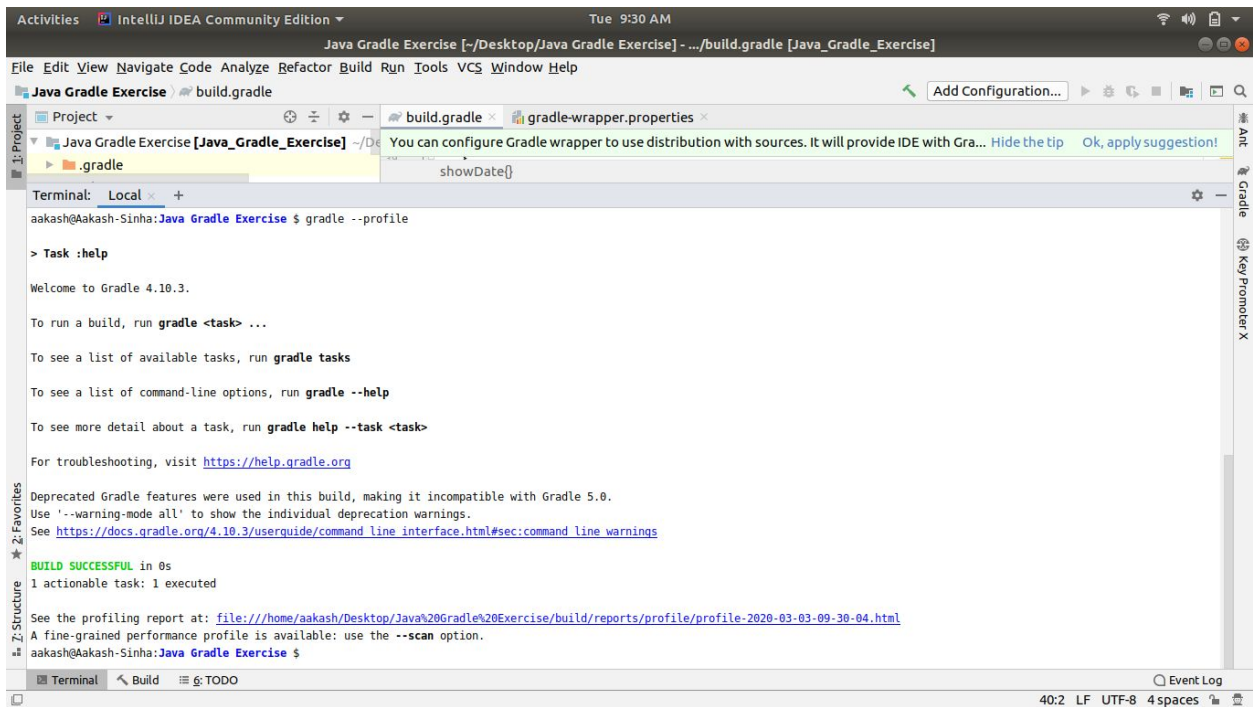


//Ques6: Override the Gradle Wrapper task to install a different version of gradle. Make sure
//that the task written in Q4 also executes with it.





//Ques7: Run the gradle profile command and attach the resulting files.



The screenshot shows the IntelliJ IDEA Community Edition interface. The top toolbar includes menus for File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, and Help. The main editor area displays the 'build.gradle' file with the content 'showDate{}'. A tooltip message states: 'You can configure Gradle wrapper to use distribution with sources. It will provide IDE with Gra... Hide the tip Ok, apply suggestion!'. The left sidebar shows the Project view with 'Java Gradle Exercise' and 'gradle' folders. The bottom panel contains the Terminal, Build, and TODO tabs. The Terminal tab is active, showing the command 'gradle --profile' and its output. The output includes task help information, deprecation warnings, and a successful build status.

```
aakash@Aakash-Sinha:Java Gradle Exercise $ gradle --profile

> Task :help

Welcome to Gradle 4.10.3.

To run a build, run gradle <task> ...

To see a list of available tasks, run gradle tasks

To see a list of command-line options, run gradle --help

To see more detail about a task, run gradle help --task <task>

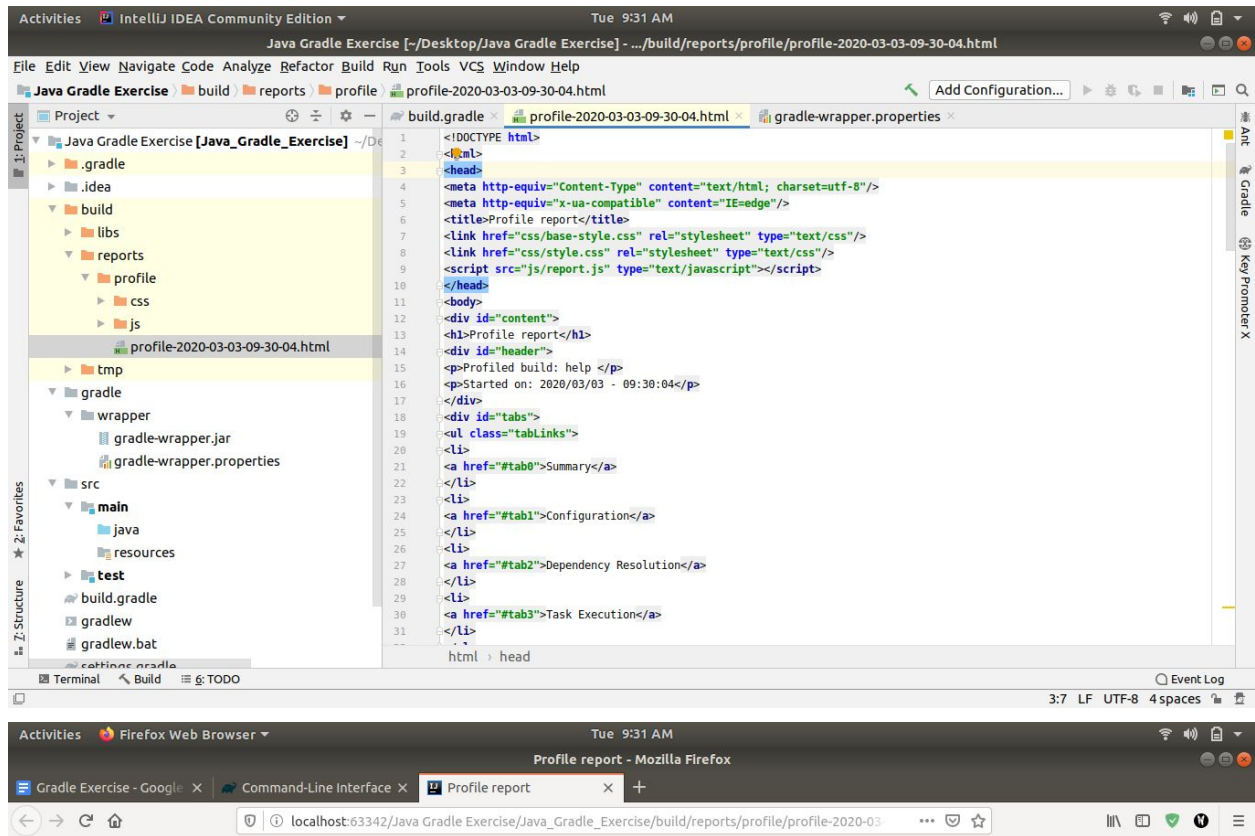
For troubleshooting, visit https://help.gradle.org

Deprecated Gradle features were used in this build, making it incompatible with Gradle 5.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/4.10.3/userguide/command\_line\_interface.html#sec:command\_line\_warnings

BUILD SUCCESSFUL in 0s
1 actionable task: 1 executed

See the profiling report at: file:///home/aakash/Desktop/Java%20Gradle%20Exercise/build/reports/profile/profile-2020-03-03-09-30-04.html
A fine-grained performance profile is available: use the --scan option.
aakash@Aakash-Sinha:Java Gradle Exercise $
```

At the bottom right of the terminal panel, the status bar shows '40:2 LF UTF-8 4 spaces'.



Profile report

Profiled build: help

Started on: 2020/03/03 - 09:30:04

Summary Configuration Dependency Resolution Task Execution

Description	Duration
Total Build Time	0.359s
Startup	0.329s
Settings and BuildSrc	0.001s
Loading Projects	0s
Configuring Projects	0.012s
Task Execution	0.001s

Generated by [Gradle 4.10.3](#) at 3 Mar, 2020 9:30:04 AM