

Dependency Management using Gradle

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

```
plugins {  
    id 'java'  
}  
  
group 'org.example'  
version '1.0-SNAPSHOT'  
sourceCompatibility = 1.8  
repositories {  
    mavenCentral()  
    maven{  
        url "https://mvnrepository.com/artifact/com.google.code.gson/gson"  
    }  
}  
  
dependencies {  
    testCompile group: 'junit', name: 'junit', version: '4.12'  
    compileOnly group: 'com.google.code.gson', name: 'gson', version: '2.8.6'  
}
```

OUTPUT



The screenshot shows an IDE window with several tabs: 'build.gradle', 'A.java', 'B.java', 'profile-2020-03-02-16-22-37.html', and 'Ques2.java'. A green tip banner at the top reads: 'You can configure Gradle wrapper to use distribution with sources. It will provide IDE with... Hide the tip'. The 'build.gradle' file is open and shows the same code as the previous block, with syntax highlighting. The code is as follows:

```
1 plugins {  
2     id 'java'  
3 }  
4  
5 group 'org.example'  
6 version '1.0-SNAPSHOT'  
7  
8 sourceCompatibility = 1.8  
9  
10  
11 repositories {  
12     mavenCentral()  
13     maven{  
14         url "https://mvnrepository.com/artifact/com.google.code.gson/gson"  
15     }  
16 }  
17  
18 dependencies {  
19     testCompile group: 'junit', name: 'junit', version: '4.12'  
20     compileOnly group: 'com.google.code.gson', name: 'gson', version: '2.8.6'  
21 }  
22  
23
```

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"

```
plugins {
    id 'java'
}

group 'org.example'
version '1.0-SNAPSHOT'

sourceCompatibility = 1.8

sourceSets{
    main{
        java {
            srcDirs = ['src/main/java']
        }
    }
}

jar{
    manifest{
        attributes(
            "Main-Class": 'Ques2',
            "Class-Path": configurations.compile.collect { it.getName() }.join(' '),
        )
    }
}

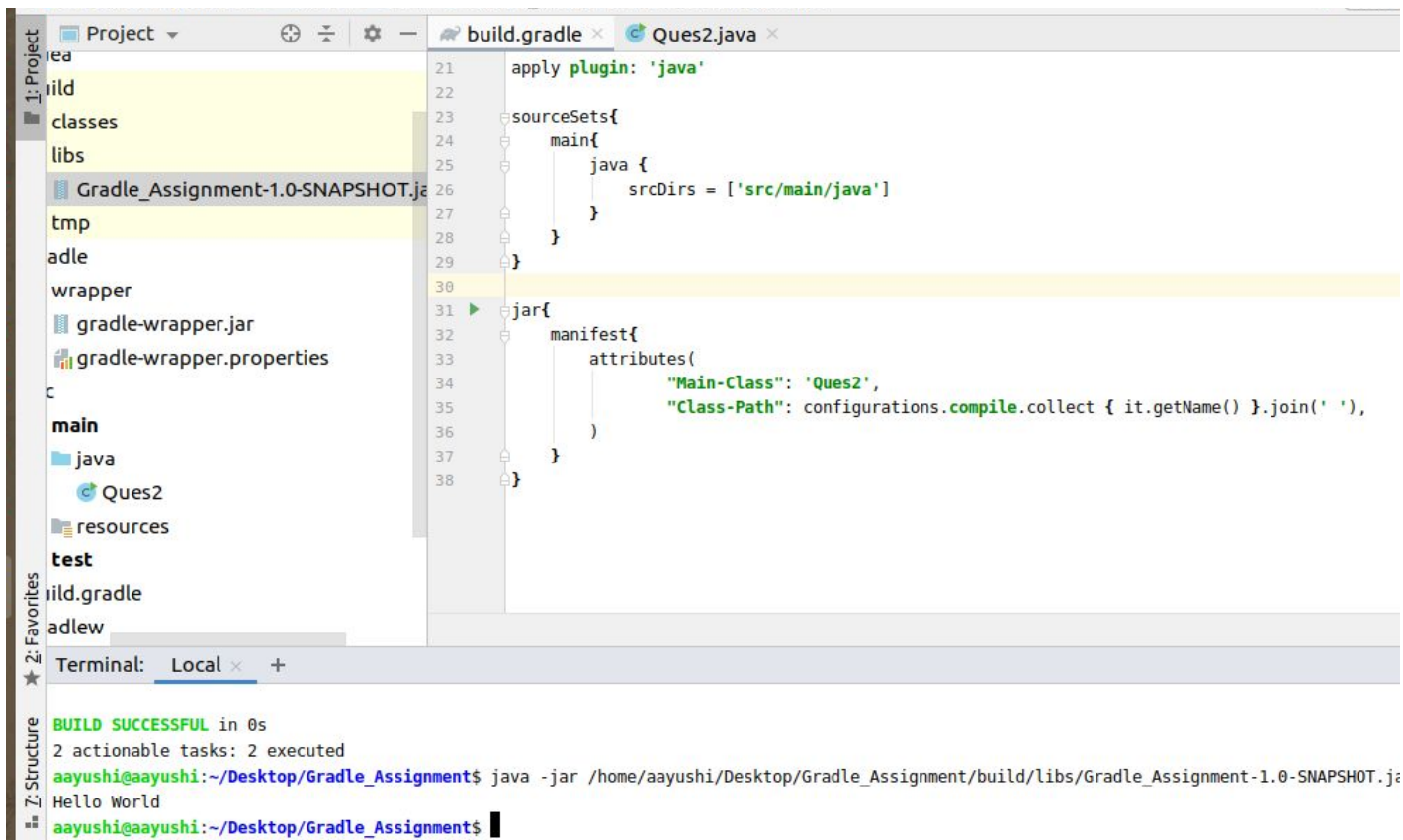
repositories {
    mavenCentral()
    maven{
        url "https://mvnrepository.com/artifact/com.google.code.gson/gson"
    }
}

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

OUTPUT



```
1  plugins {
2      id 'java'
3  }
4
5  group 'org.example'
6  version '1.0-SNAPSHOT'
7
8  sourceCompatibility = 1.8
9
10
11 sourceSets{
12     main{
13         java {
14             srcDirs = ['src/main/java']
15         }
16     }
17 }
18
19 jar{
20     manifest{
21         attributes(
22             "Main-Class": 'Ques2',
23             "Class-Path": configurations.compile.collect { it.getName() }.join(' '),
24         )
25     }
26 }
```



```
21  apply plugin: 'java'
22
23  sourceSets{
24      main{
25          java {
26              srcDirs = ['src/main/java']
27          }
28      }
29  }
30
31  jar{
32      manifest{
33          attributes(
34              "Main-Class": 'Ques2',
35              "Class-Path": configurations.compile.collect { it.getName() }.join(' '),
36          )
37      }
38  }
```

Terminal: Local

```
BUILD SUCCESSFUL in 0s
2 actionable tasks: 2 executed
aayushi@aayushi:~/Desktop/Gradle_Assignment$ java -jar /home/aayushi/Desktop/Gradle_Assignment/build/libs/Gradle_Assignment-1.0-SNAPSHOT.jar
Hello World
aayushi@aayushi:~/Desktop/Gradle_Assignments$
```

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

Ans -

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.

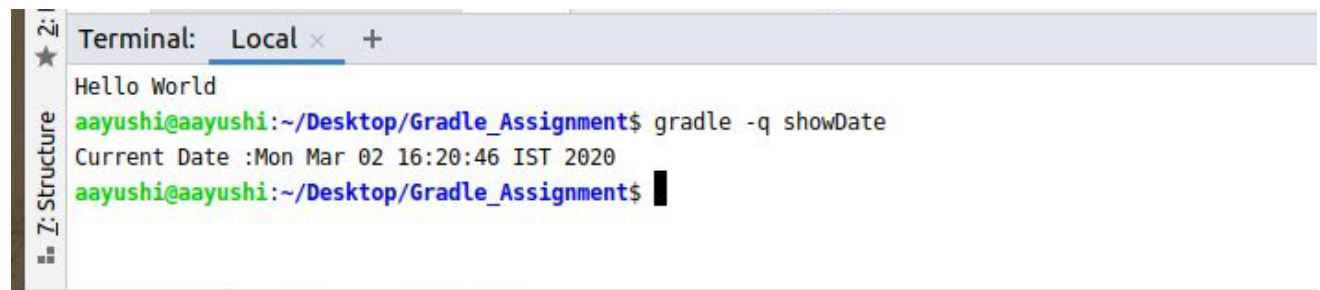
testRuntime: We use the runRuntime scope for dependencies that are not needed at compile time, like when we're compiling against an API and only need the implementation of that API at runtime.

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.

```
task showDate{
    dependsOn(build)
    doLast {
        println 'Current Date :' + new Date();
    }
}
```

OUTPUT

```
task showDate{
    dependsOn(build)
    doLast {
        println 'Current Date :' + new Date();
    }
}
```



The screenshot shows a terminal window with the title 'Terminal: Local x +'. The first line of output is 'Hello World'. The second line shows the command 'aayushi@aayushi:~/Desktop/Gradle_Assignment\$ gradle -q showDate' being executed. The third line shows the output 'Current Date :Mon Mar 02 16:20:46 IST 2020'. The prompt 'aayushi@aayushi:~/Desktop/Gradle_Assignment\$' is visible on the fourth line.

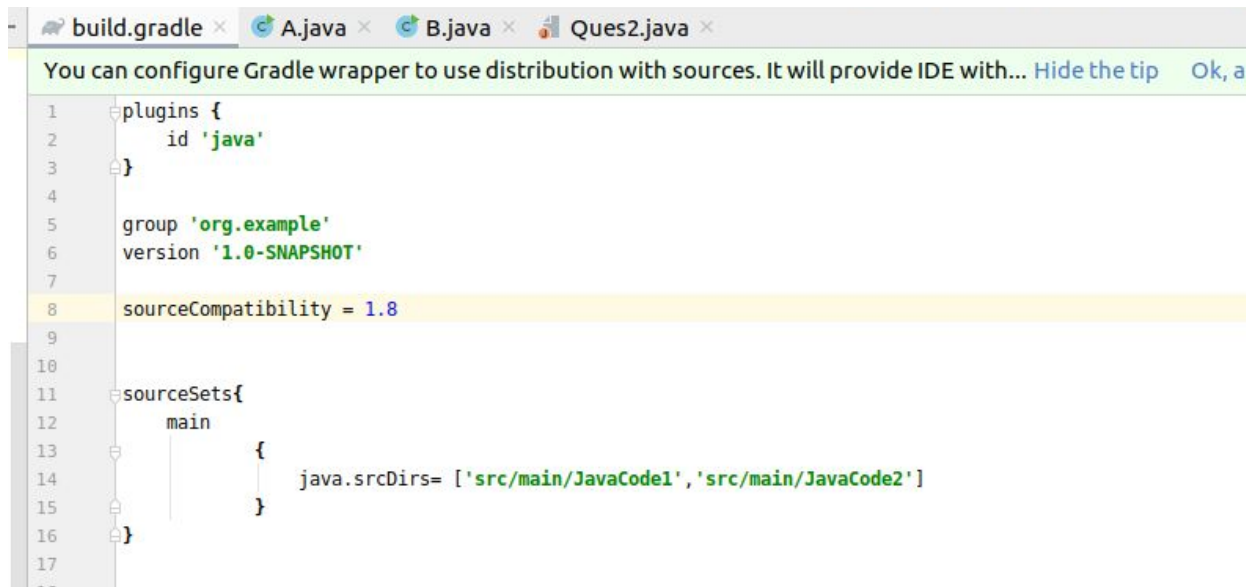
```
Terminal: Local x +
Hello World
aayushi@aayushi:~/Desktop/Gradle_Assignment$ gradle -q showDate
Current Date :Mon Mar 02 16:20:46 IST 2020
aayushi@aayushi:~/Desktop/Gradle_Assignment$
```

5. 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`.

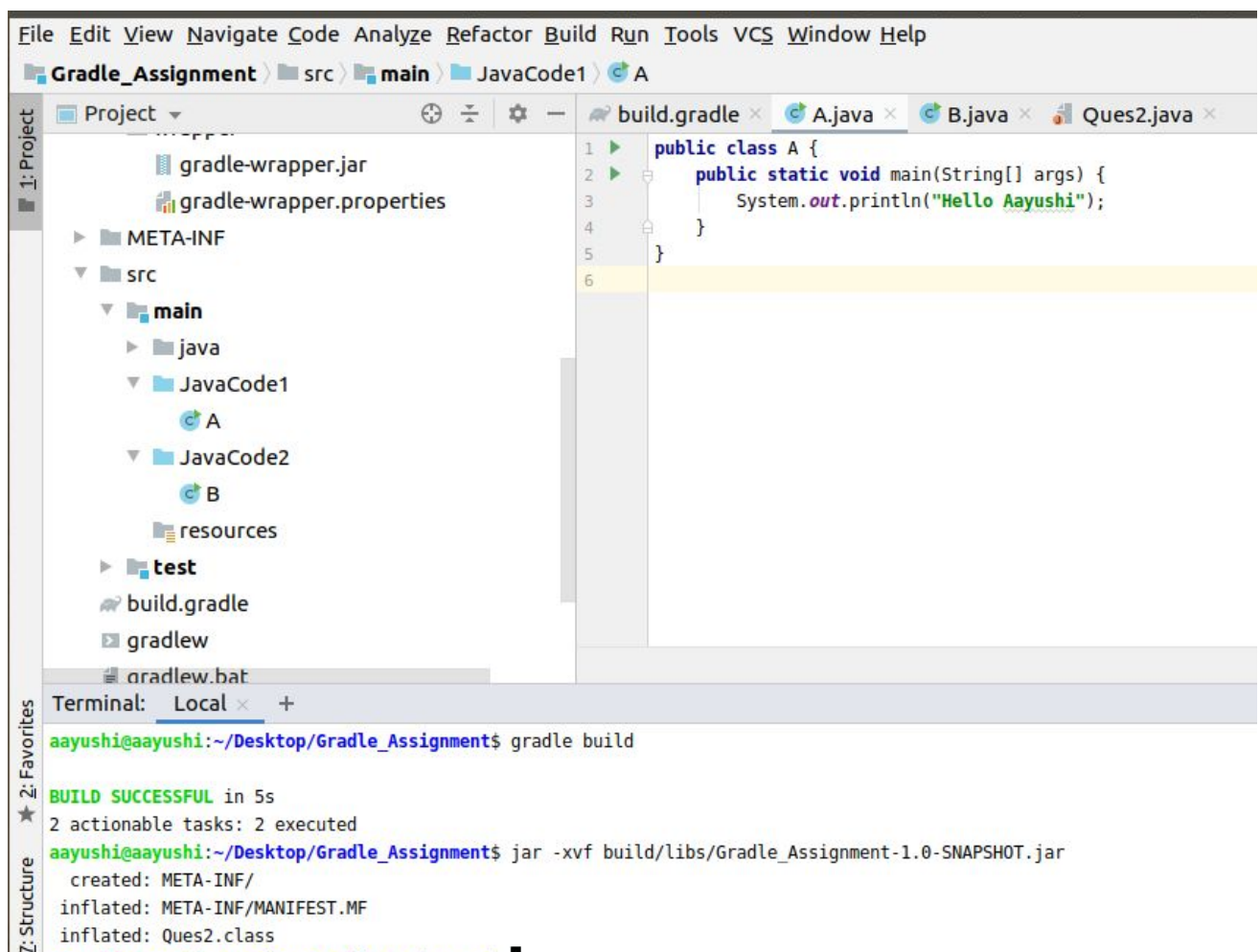
CODE

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

OUTPUT



```
1 plugins {
2     id 'java'
3 }
4
5 group 'org.example'
6 version '1.0-SNAPSHOT'
7
8 sourceCompatibility = 1.8
9
10
11 sourceSets {
12     main {
13         {
14             java.srcDirs = ['src/main/JavaCode1', 'src/main/JavaCode2']
15         }
16     }
17 }
18
```

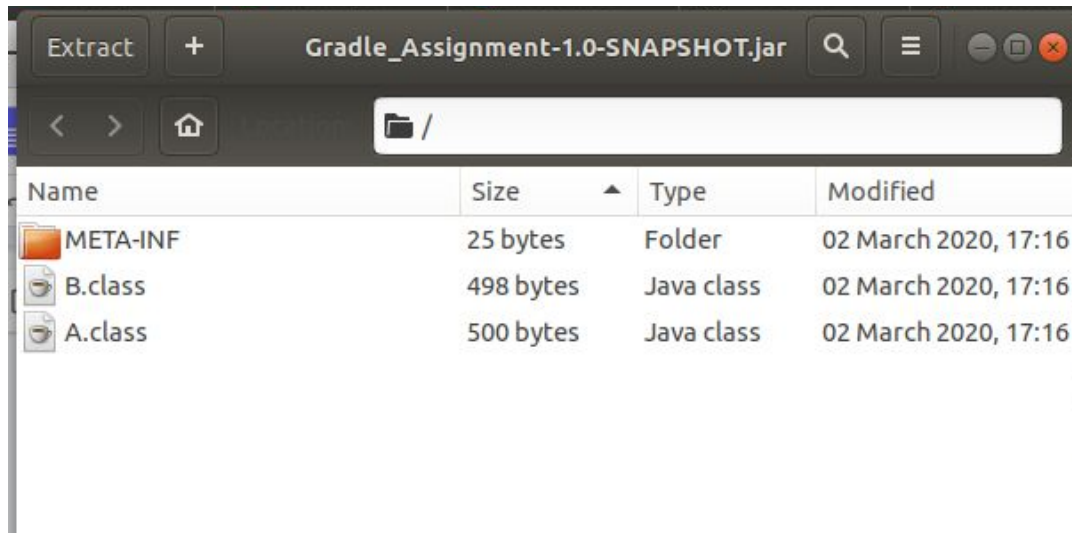


```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Gradle_Assignment > src > main > JavaCode1 > A

1 public class A {
2     public static void main(String[] args) {
3         System.out.println("Hello Aayushi");
4     }
5 }
6

Terminal: Local x +
aayushi@aayushi:~/Desktop/Gradle_Assignment$ gradle build

BUILD SUCCESSFUL in 5s
2 actionable tasks: 2 executed
aayushi@aayushi:~/Desktop/Gradle_Assignment$ jar -xvf build/libs/Gradle_Assignment-1.0-SNAPSHOT.jar
created: META-INF/
inflated: META-INF/MANIFEST.MF
inflated: Ques2.class
```



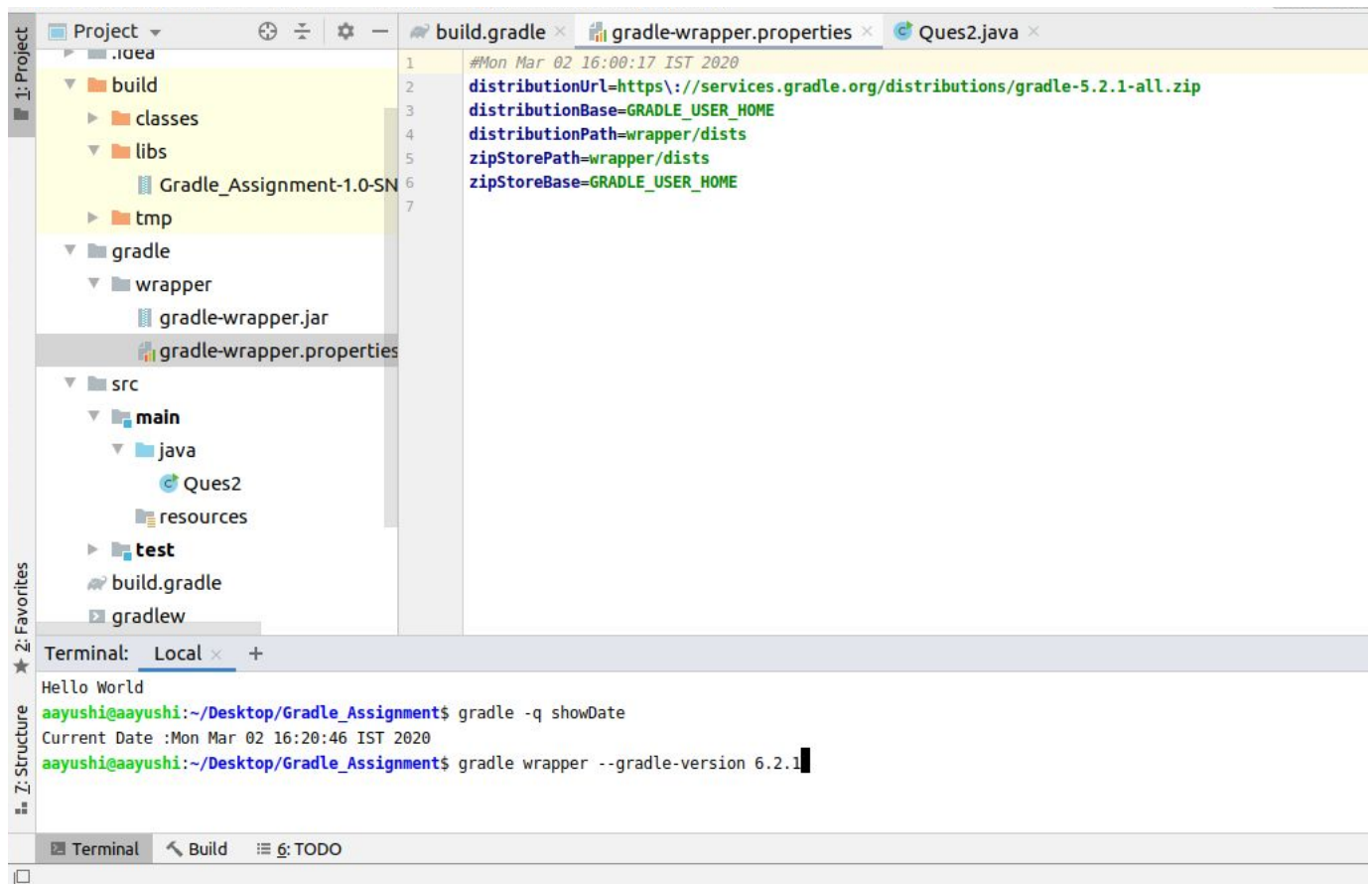
**6. Override the Gradle Wrapper task to install a different version of gradle.
Make sure that the task written in Q4 also executes with it.**

Previous Version: gradle-5.2.1

Gradle wrapper --gradle-version 6.2.1

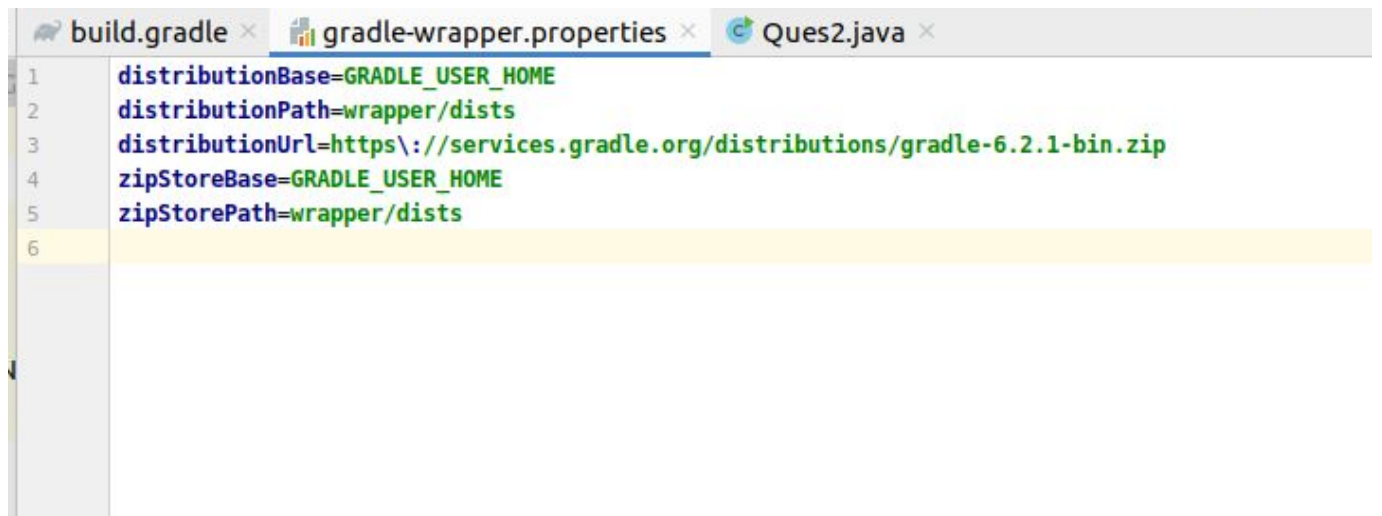
New Version: gradle-6.2.1

OUTPUT



The screenshot shows an IDE interface with a project structure on the left and a terminal window at the bottom. The project structure includes a `build` directory with `classes`, `libs`, and `tmp` subdirectories. The `libs` directory contains `Gradle_Assignment-1.0-SN`. The `gradle` directory contains a `wrapper` subdirectory with `gradle-wrapper.jar` and `gradle-wrapper.properties`. The `src` directory contains a `main` subdirectory with a `java` subdirectory containing `Ques2` and a `resources` subdirectory. The `test` directory is also present. The terminal window shows the following output:

```
Hello World
aayushi@aayushi:~/Desktop/Gradle_Assignment$ gradle -q showDate
Current Date :Mon Mar 02 16:20:46 IST 2020
aayushi@aayushi:~/Desktop/Gradle_Assignment$ gradle wrapper --gradle-version 6.2.1
```



The screenshot shows the `gradle-wrapper.properties` file in the IDE. The file contains the following properties:

```
1 distributionBase=GRADLE_USER_HOME
2 distributionPath=wrapper/dists
3 distributionUrl=https\://services.gradle.org/distributions/gradle-6.2.1-bin.zip
4 zipStoreBase=GRADLE_USER_HOME
5 zipStorePath=wrapper/dists
6
```


7. Run the gradle profile command and attach the resulting files.

CODE

Gradle --profile

Profile-2020-03-02-16-22-37.html

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<meta http-equiv="x-ua-compatible" content="IE=edge"/>
<title>Profile report</title>
<link href="css/base-style.css" rel="stylesheet" type="text/css"/>
<link href="css/style.css" rel="stylesheet" type="text/css"/>
<script src="js/report.js" type="text/javascript"></script>
</head>
<body>
<div id="content">
<h1>Profile report</h1>
<div id="header">
<p>Profiled build: help </p>
<p>Started on: 2020/03/02 - 16:22:37</p>
</div>
<div id="tabs">
<ul class="tabLinks">
<li>
<a href="#tab0">Summary</a>
</li>
<li>
<a href="#tab1">Configuration</a>
</li>
<li>
<a href="#tab2">Dependency Resolution</a>
</li>
<li>
<a href="#tab3">Task Execution</a>
</li>
</ul>
<div class="tab" id="tab0">
<h2>Summary</h2>
<table>
<thead>
```

```
<tr>
<th>Description</th>
<th class="numeric">Duration</th>
</tr>
</thead>
<tr>
<td>Total Build Time</td>
<td class="numeric">0.336s</td>
</tr>
<tr>
<td>Startup</td>
<td class="numeric">0.315s</td>
</tr>
<tr>
<td>Settings and BuildSrc</td>
<td class="numeric">0.001s</td>
</tr>
<tr>
<td>Loading Projects</td>
<td class="numeric">0.001s</td>
</tr>
<tr>
<td>Configuring Projects</td>
<td class="numeric">0.010s</td>
</tr>
<tr>
<td>Task Execution</td>
<td class="numeric">0.001s</td>
</tr>
</table>
</div>
<div class="tab" id="tab1">
<h2>Configuration</h2>
<table>
<thead>
<tr>
<th>Project</th>
<th class="numeric">Duration</th>
</tr>
</thead>
<tr>
<td>All projects</td>
<td class="numeric">0.010s</td>
```

```

</tr>
<tr>
<td>:</td>
<td class="numeric">0.010s</td>
</tr>
</table>
</div>
<div class="tab" id="tab2">
<h2>Dependency Resolution</h2>
<table>
<thead>
<tr>
<th>Dependencies</th>
<th class="numeric">Duration</th>
</tr>
</thead>
<tr>
<td>All dependencies</td>
<td class="numeric">0s</td>
</tr>
<tr>
<td>:compile</td>
<td class="numeric">0s</td>
</tr>
</table>
</div>
<div class="tab" id="tab3">
<h2>Task Execution</h2>
<table>
<thead>
<tr>
<th>Task</th>
<th class="numeric">Duration</th>
<th>Result</th>
</tr>
</thead>
<tr>
<td>:</td>
<td class="numeric">0.001s</td>
<td>(total)</td>
</tr>
<tr>
<td class="indentPath">:help</td>

```

```
<td class="numeric">0.001s</td>
<td></td>
</tr>
</table>
</div>
</div>
<div id="footer">
<p>
<div>
<label class="hidden" id="label-for-line-wrapping-toggle" for="line-wrapping-toggle">Wrap lines
<input id="line-wrapping-toggle" type="checkbox" autocomplete="off"/>
</label>
</div>Generated by
<a href="http://www.gradle.org">Gradle 4.10.3</a> at 2 Mar, 2020 4:22:37 PM</p>
</div>
</div>
</body>
</html>
```

OUTPUT

```
■ ■ Z: Struct BUILD SUCCESSFUL in 0s
1 actionable task: 1 executed
aayushi@aayushi:~/Desktop/Gradle_Assignment$ gradle --profile
```

```
build.gradle x  gradle-wrapper.properties x  profile-2020-03-02-16-22-37.html x  Ques2.java x
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
5  <meta http-equiv="x-ua-compatible" content="IE=edge"/>
6  <title>Profile report</title>
7  <link href="css/base-style.css" rel="stylesheet" type="text/css"/>
8  <link href="css/style.css" rel="stylesheet" type="text/css"/>
9  <script src="js/report.js" type="text/javascript"></script>
10 </head>
11 <body>
12 <div id="content">
13 <h1>Profile report</h1>
14 <div id="header">
15 <p>Profiled build: help </p>
16 <p>Started on: 2020/03/02 - 16:22:37</p>
17 </div>
18 <div id="tabs">
19 <ul class="tabLinks">
20 <li>
21 <a href="#tab0">Summary</a>
22 </li>
23 <li>
```

localhost:63342/Gradle_Assignment/build/reports/profile/profile-2020-03-02-16-22-37.html?_ijt=vvvsnf7i

Profile report

Profiled build: help

Started on: 2020/03/02 - 16:22:37

Summary

Configuration

Dependency Resolution

Task Execution

Description	Duration
Total Build Time	0.336s
Startup	0.315s
Settings and BuildSrc	0.001s
Loading Projects	0.001s
Configuring Projects	0.010s
Task Execution	0.001s

Generated by [Gradle 4.10.3](#) at 2 Mar, 2020 4:22:37 PM