Gatling setup with IntelliJ

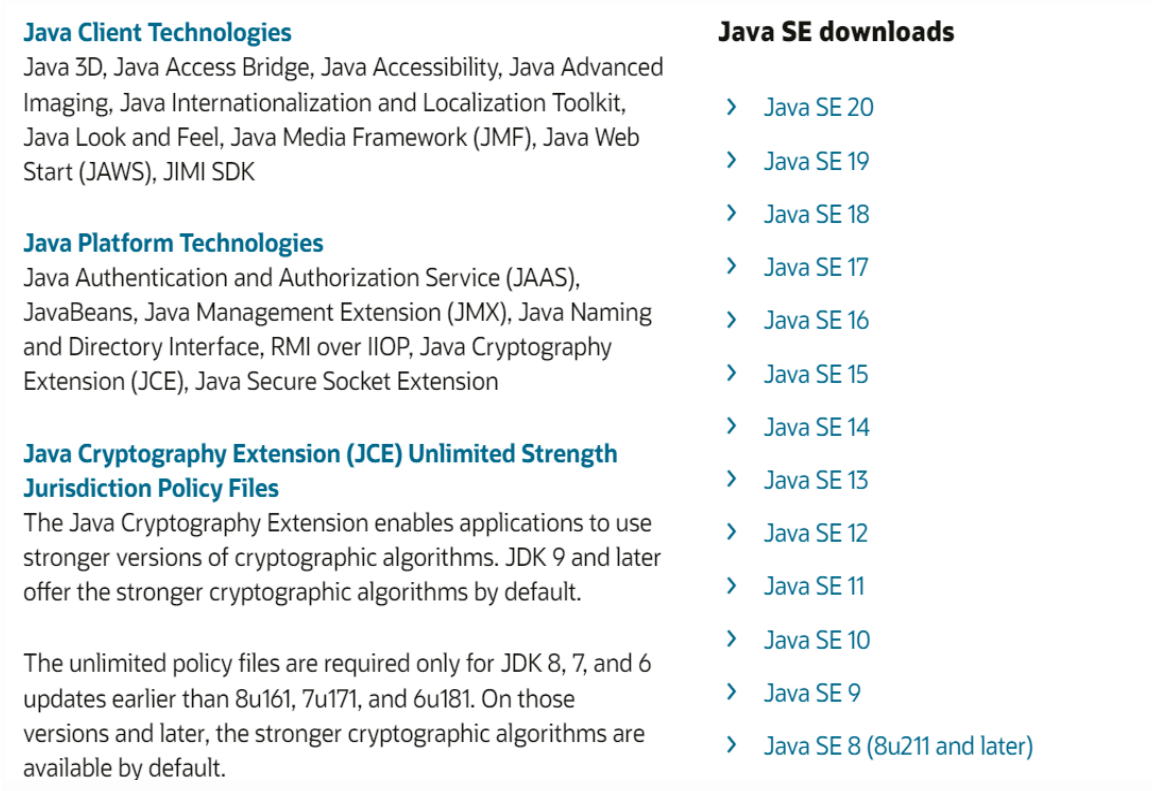
Overview:
To set up the gatling framework in IntelliJ, we need to install Java, Maven, Gatling, IntelliJ IDE. For the version related queries, you can refer to the below link:
https://gatling.io/docs/gatling/reference/current/whats_new/3.9/#collapsable-1db06ac3bc5ace23ad37eec7de5c484d

Supported versions across languages:

Gatling Version	Scala Version	Java Version
3.5x	2.12.x	8,11,14
3.4x	2.12.x	8,11,14
3.3x	2.12.x	8,11
3.2x	2.12.x	8,11
3.1x	2.12.x	8,11

Java

1. Open the Java download page.
2. Choose the latest stable version from the Java SE archive page. (Recommended to download above version 1.8)



Java Client Technologies
Java 3D, Java Access Bridge, Java Accessibility, Java Advanced Imaging, Java Internationalization and Localization Toolkit, Java Look and Feel, Java Media Framework (JMF), Java Web Start (JAWS), JIMI SDK

Java Platform Technologies
Java Authentication and Authorization Service (JAAS), JavaBeans, Java Management Extension (JMX), Java Naming and Directory Interface, RMI over IIOP, Java Cryptography Extension (JCE), Java Secure Socket Extension

Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files
The Java Cryptography Extension enables applications to use stronger versions of cryptographic algorithms. JDK 9 and later offer the stronger cryptographic algorithms by default.

The unlimited policy files are required only for JDK 8, 7, and 6 updates earlier than 8u161, 7u171, and 6u181. On those versions and later, the stronger cryptographic algorithms are available by default.

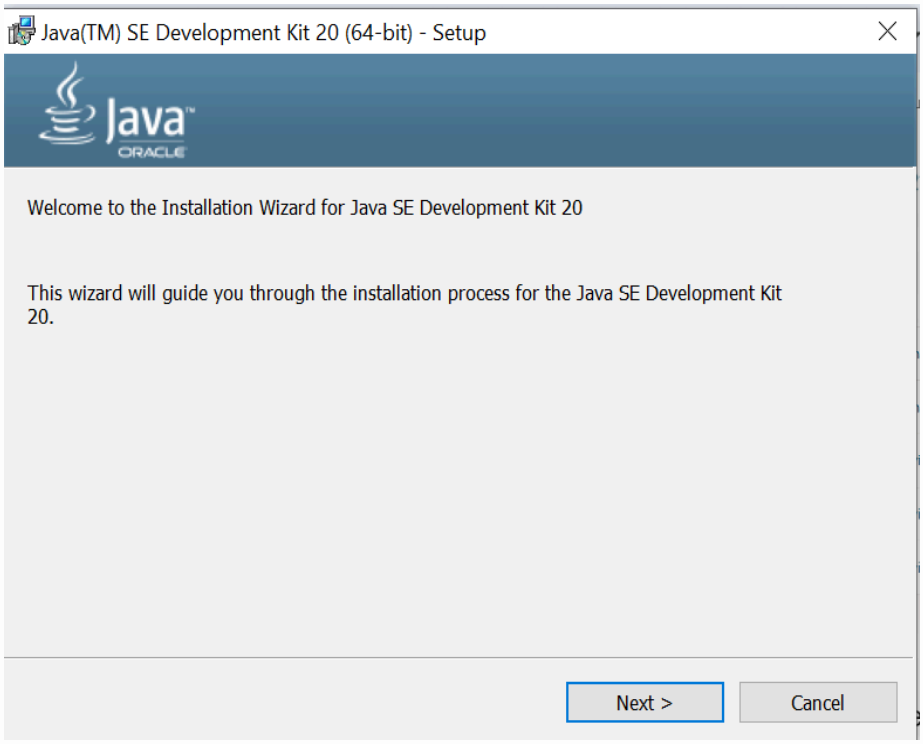
Java SE downloads

- Java SE 20
- Java SE 19
- Java SE 18
- Java SE 17
- Java SE 16
- Java SE 15
- Java SE 14
- Java SE 13
- Java SE 12
- Java SE 11
- Java SE 10
- Java SE 9
- Java SE 8 (8u211 and later)

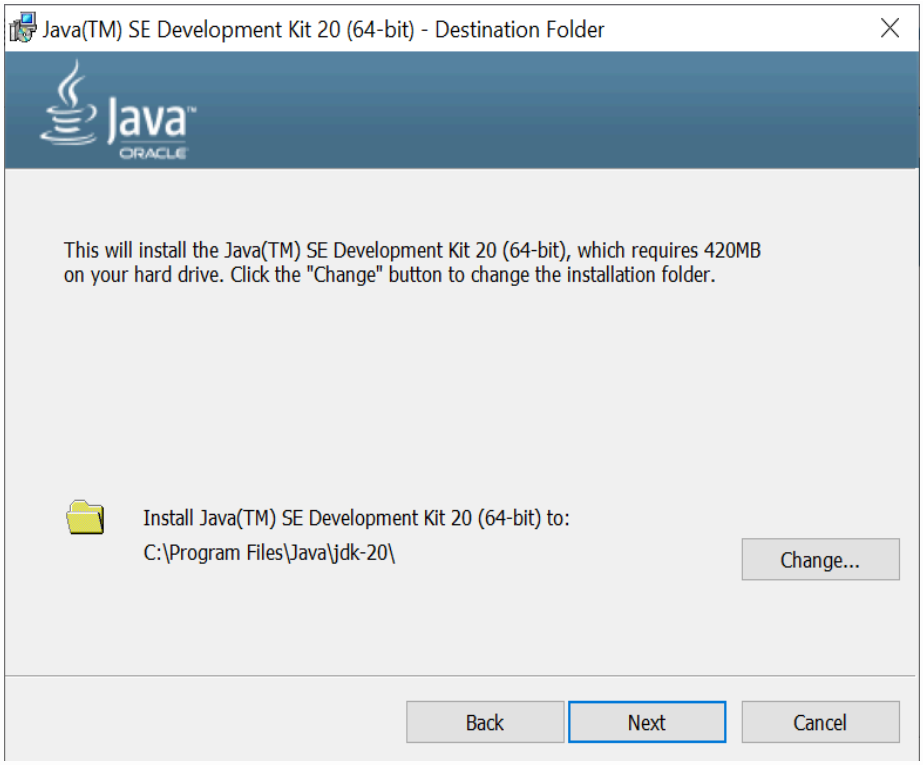
3. Download the 64-bit installer.

macOS x64 Compressed Archive	179.21 MB	https://download.oracle.com/java/20/archive/jdk-20_macos-x64_bin.tar.gz (sha256)
macOS x64 DMG Installer	178.53 MB	https://download.oracle.com/java/20/archive/jdk-20_macos-x64_bin.dmg (sha256)
Windows x64 Compressed Archive	180.80 MB	https://download.oracle.com/java/20/archive/jdk-20_windows-x64_bin.zip (sha256)
Windows x64 Installer	159.94 MB	https://download.oracle.com/java/20/archive/jdk-20_windows-x64_bin.exe (sha256)
Windows x64 msi Installer	158.72 MB	https://download.oracle.com/java/20/archive/jdk-20_windows-x64_bin.msi (sha256)

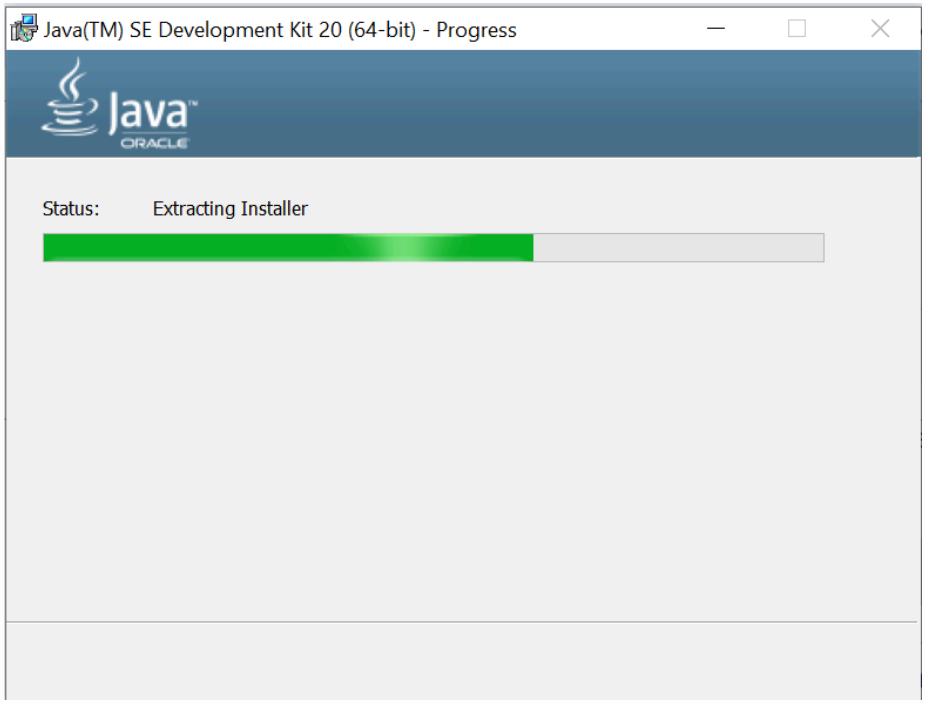
4. Open the installer and click **next**.

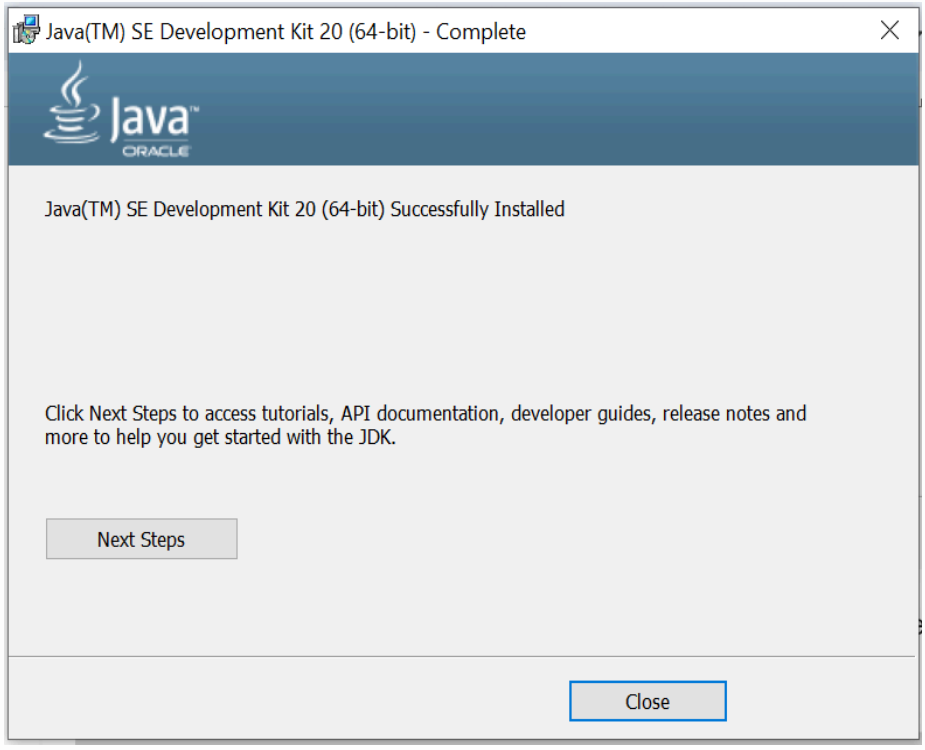


5. Change the destination path according to your preference. (It is preferable to install in the default path)

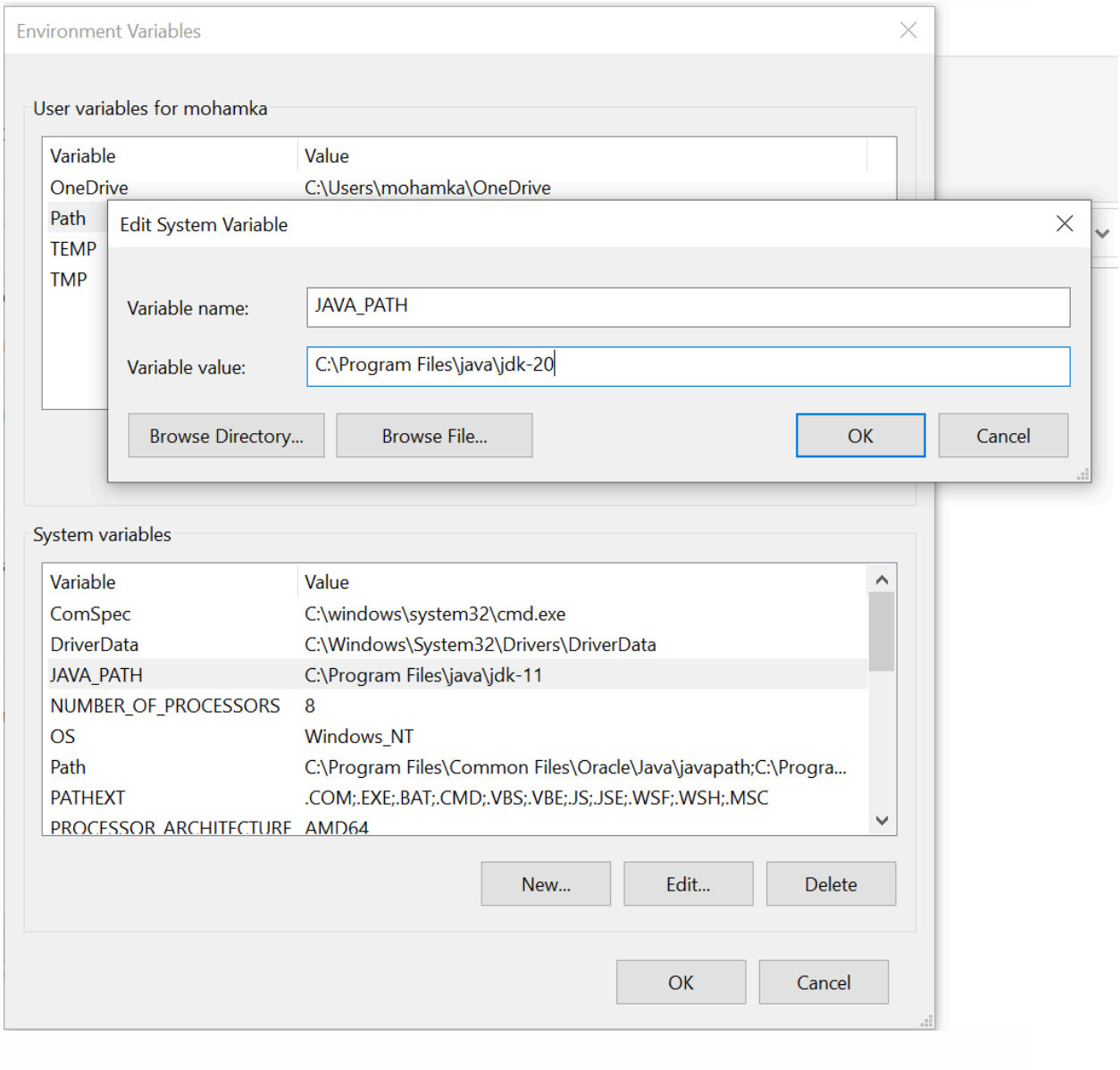


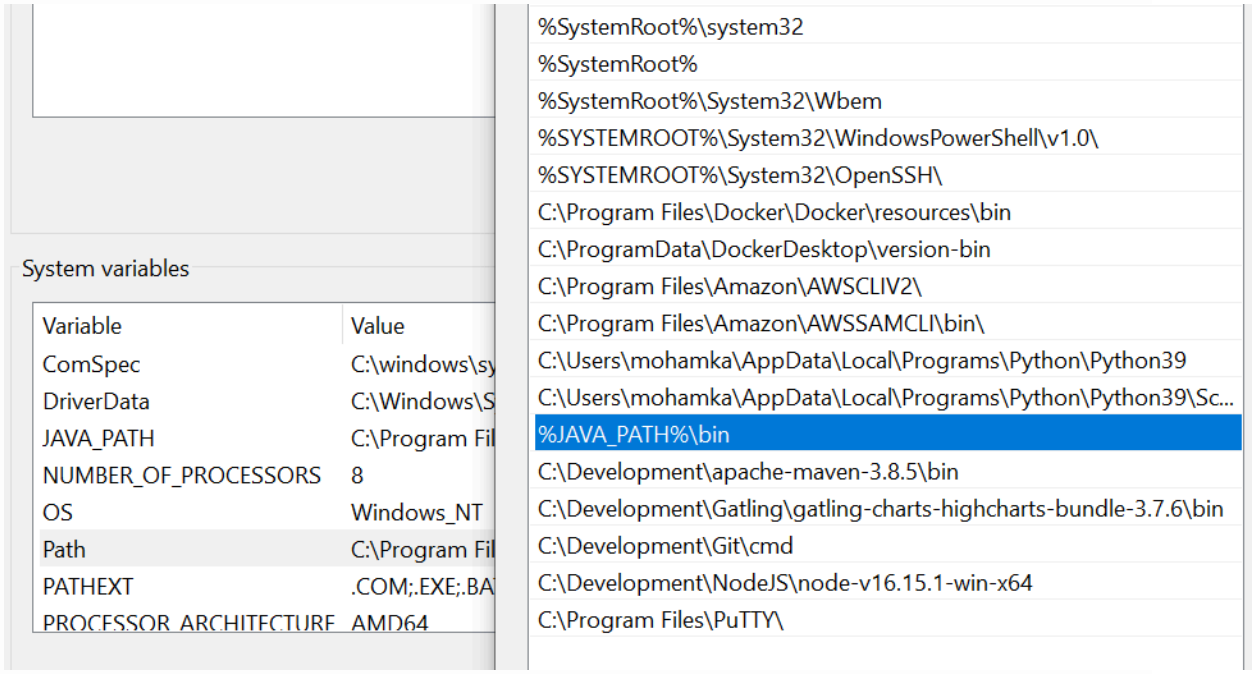
6. After successful installation, close the application.



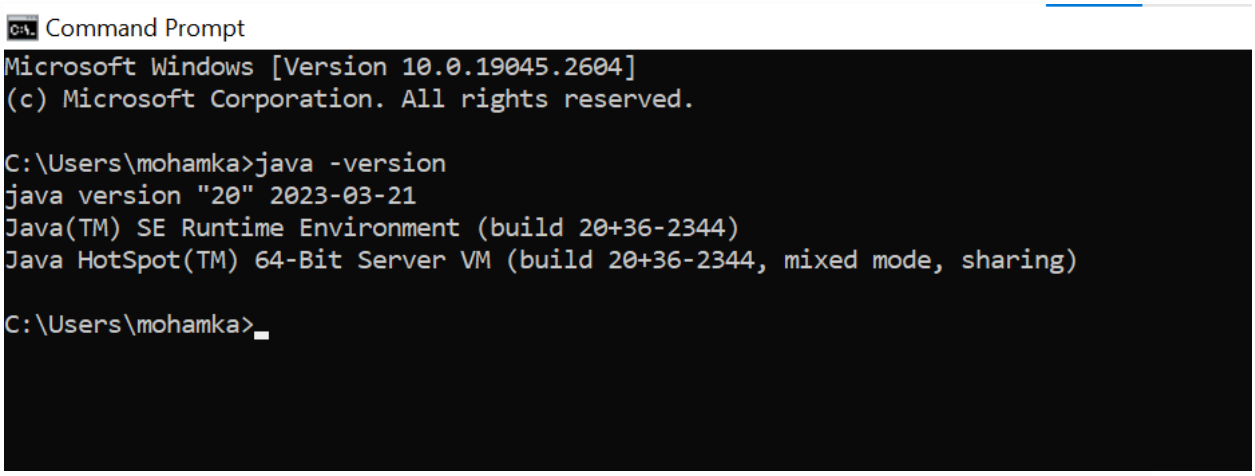


7. Set environment variable to access it through CMD.


Environment Variables
User variables for mohamka
Variable
Value
OneDrive
C:\Users\mohamka\OneDrive
Path
TEMP
TMP
Edit System Variable
Variable name:
JAVA_PATH
Variable value:
C:\Program Files\java\jdk-20
Browse Directory...
Browse File...
OK
Cancel
System variables
Variable
Value
ComSpec
C:\windows\system32\cmd.exe
DriverData
C:\Windows\System32\Drivers\DriverData
JAVA_PATH
C:\Program Files\java\jdk-11
NUMBER_OF_PROCESSORS
8
OS
Windows_NT
Path
C:\Program Files\Common Files\Oracle\Java\javapath;C:\Progra...
PATHEXT
.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE
AMD64
New...
Edit...
Delete
OK
Cancel

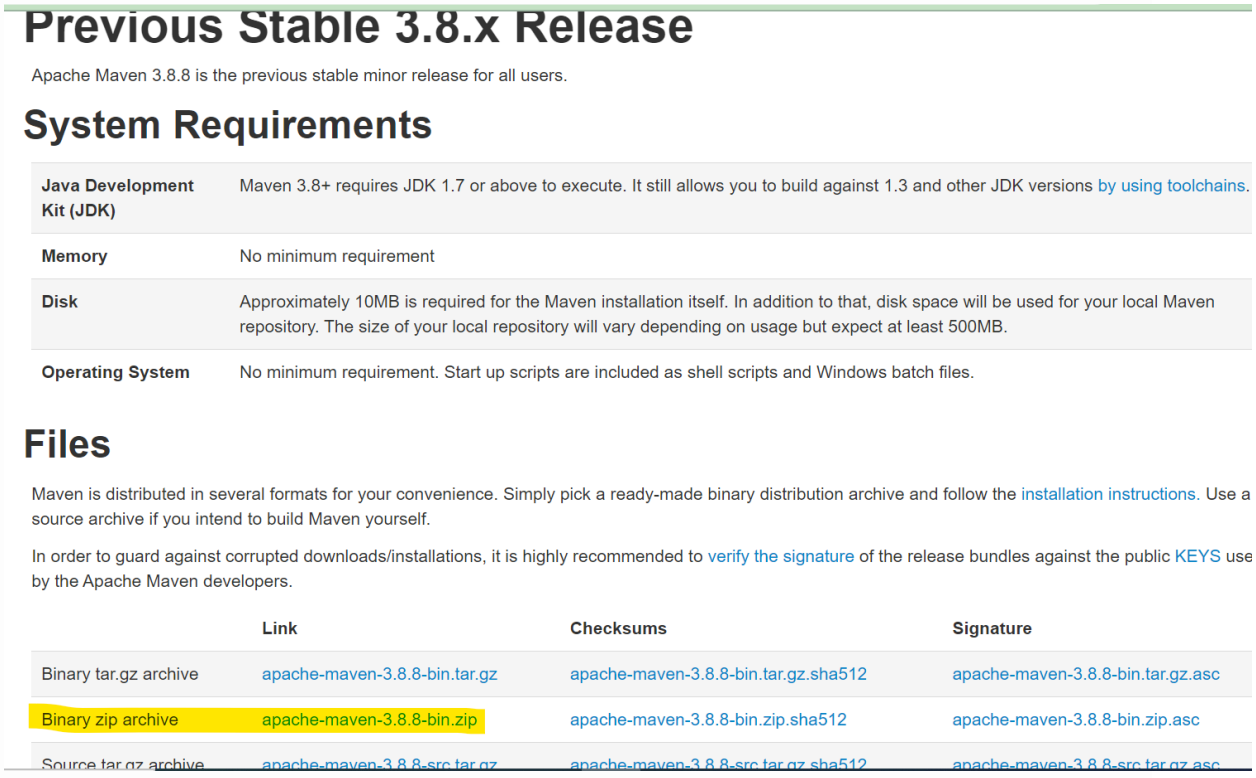


8. Open CMD and check java is accessible.



Maven:

1. Go to the apache maven website.
2. Download the binary zip of the previous stable version.



Previous Stable 3.8.x Release

Apache Maven 3.8.8 is the previous stable minor release for all users.

System Requirements

Java Development Kit (JDK)	Maven 3.8+ requires JDK 1.7 or above to execute. It still allows you to build against 1.3 and other JDK versions by using toolchains.
Memory	No minimum requirement
Disk	Approximately 10MB is required for the Maven installation itself. In addition to that, disk space will be used for your local Maven repository. The size of your local repository will vary depending on usage but expect at least 500MB.
Operating System	No minimum requirement. Start up scripts are included as shell scripts and Windows batch files.

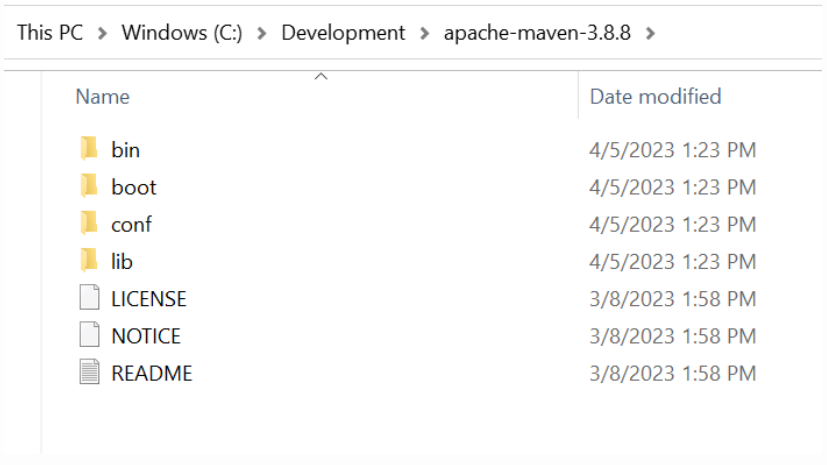
Files

Maven is distributed in several formats for your convenience. Simply pick a ready-made binary distribution archive and follow the installation instructions. Use a source archive if you intend to build Maven yourself.

In order to guard against corrupted downloads/installations, it is highly recommended to verify the signature of the release bundles against the public KEYS use by the Apache Maven developers.

	Link	Checksums	Signature
Binary tar.gz archive	apache-maven-3.8.8-bin.tar.gz	apache-maven-3.8.8-bin.tar.gz.sha512	apache-maven-3.8.8-bin.tar.gz.asc
Binary zip archive	apache-maven-3.8.8-bin.zip	apache-maven-3.8.8-bin.zip.sha512	apache-maven-3.8.8-bin.zip.asc
Source tar.gz archive	apache-maven-3.8.8-src.tar.gz	apache-maven-3.8.8-src.tar.gz.sha512	apache-maven-3.8.8-src.tar.gz.asc

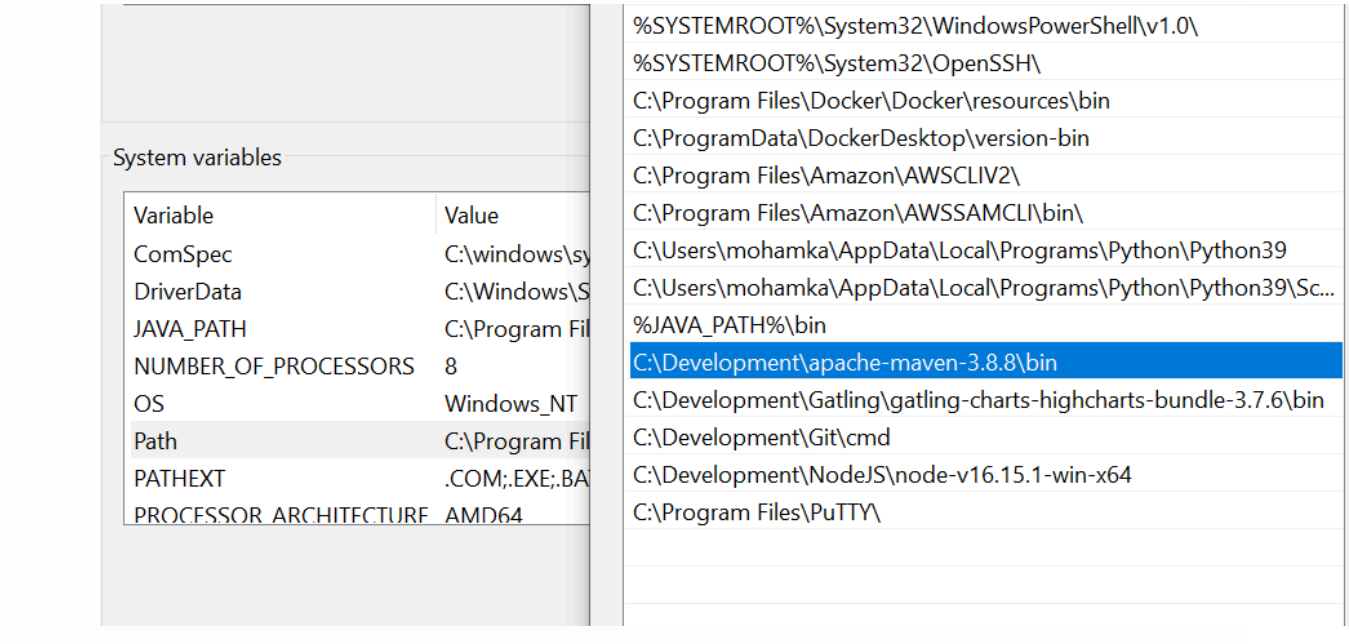
3. Unzip it and put it in the desired location.



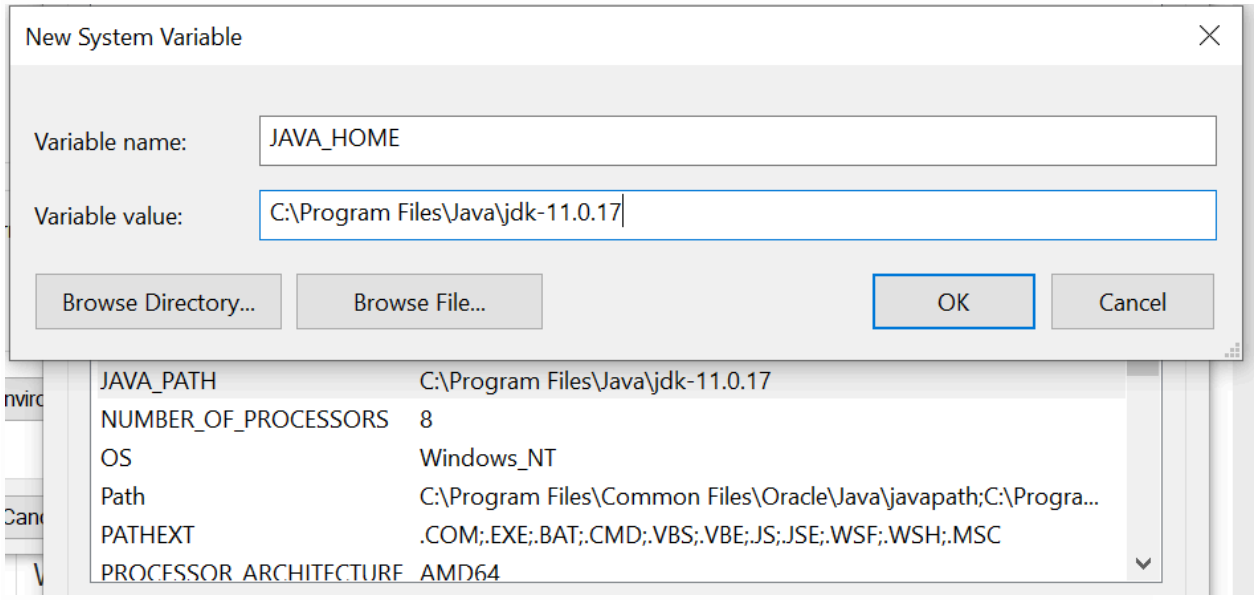
This PC › Windows (C:) › Development › apache-maven-3.8.8 ›

Name	Date modified
bin	4/5/2023 1:23 PM
boot	4/5/2023 1:23 PM
conf	4/5/2023 1:23 PM
lib	4/5/2023 1:23 PM
LICENSE	3/8/2023 1:58 PM
NOTICE	3/8/2023 1:58 PM
README	3/8/2023 1:58 PM

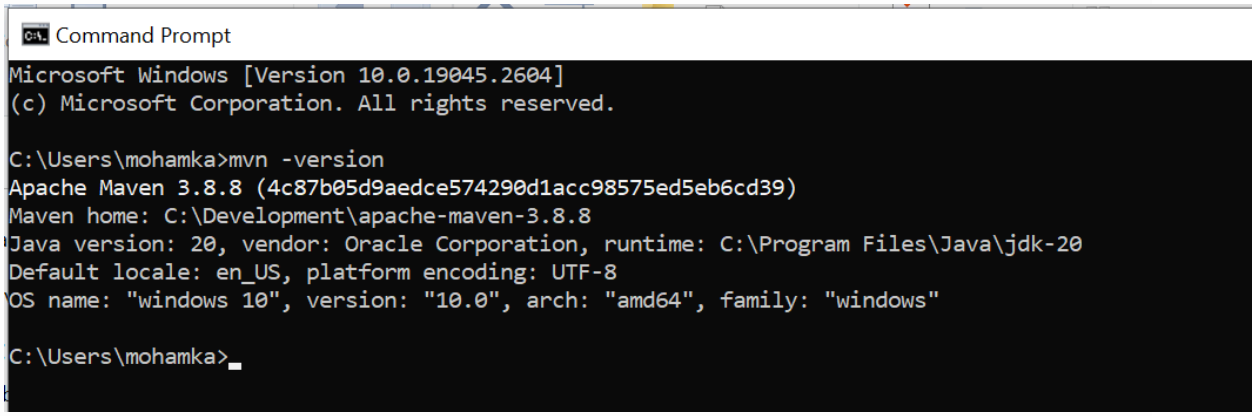
4. Add the path to an environment variable.



5. Set the **JAVA_HOME** environment variable to point to the base directory location where Java is installed on your machine.

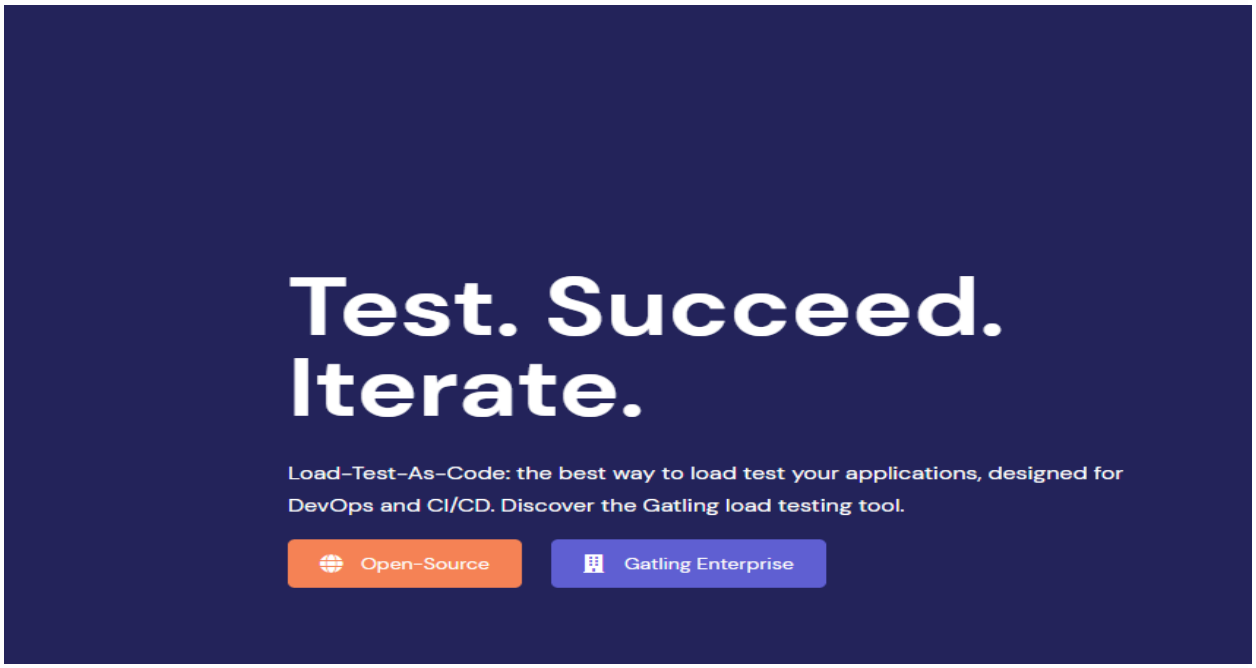


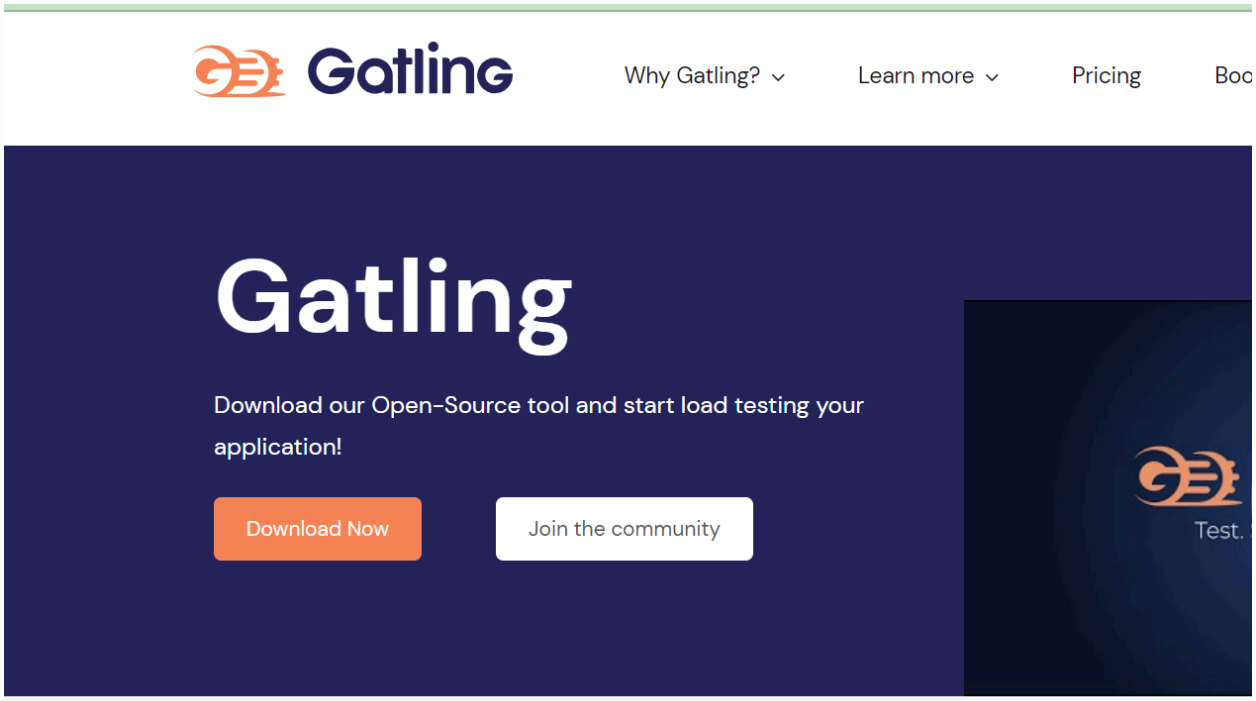
6. Open CMD and check maven is setup correctly.



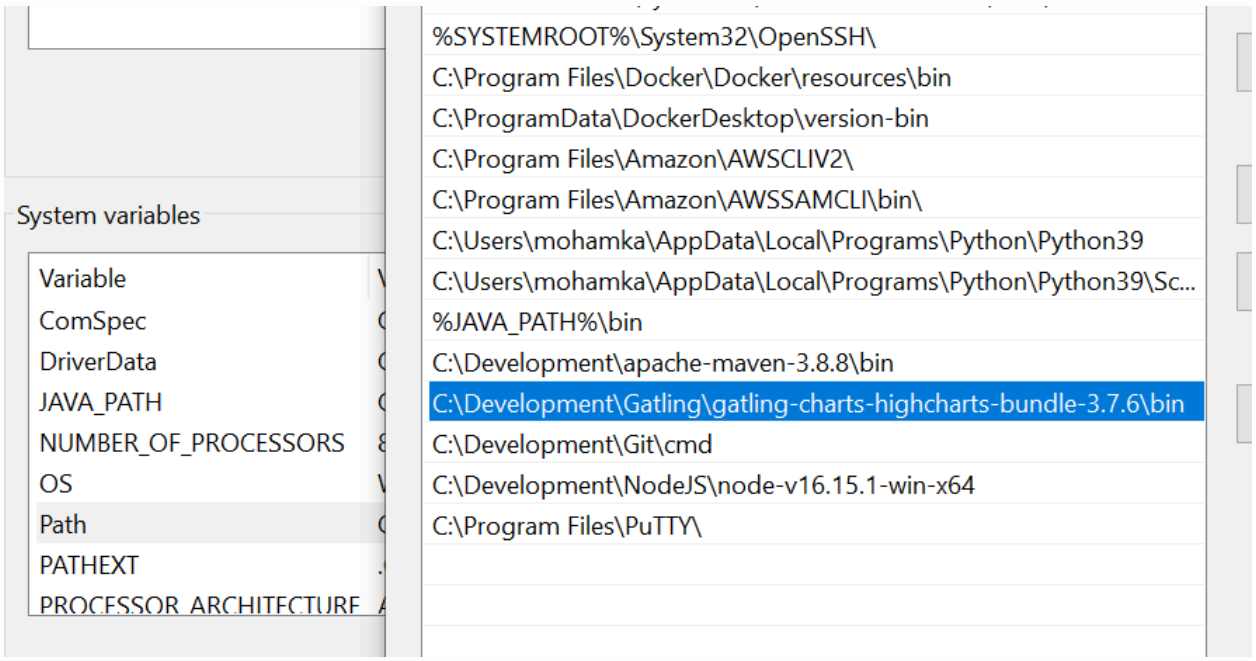
Gatling(Community Edition):

1. Go to the gatling website and download the community edition.





2. Extract it and put it in a desired location.
3. Add the path to the environment variable.



IntelliJ:

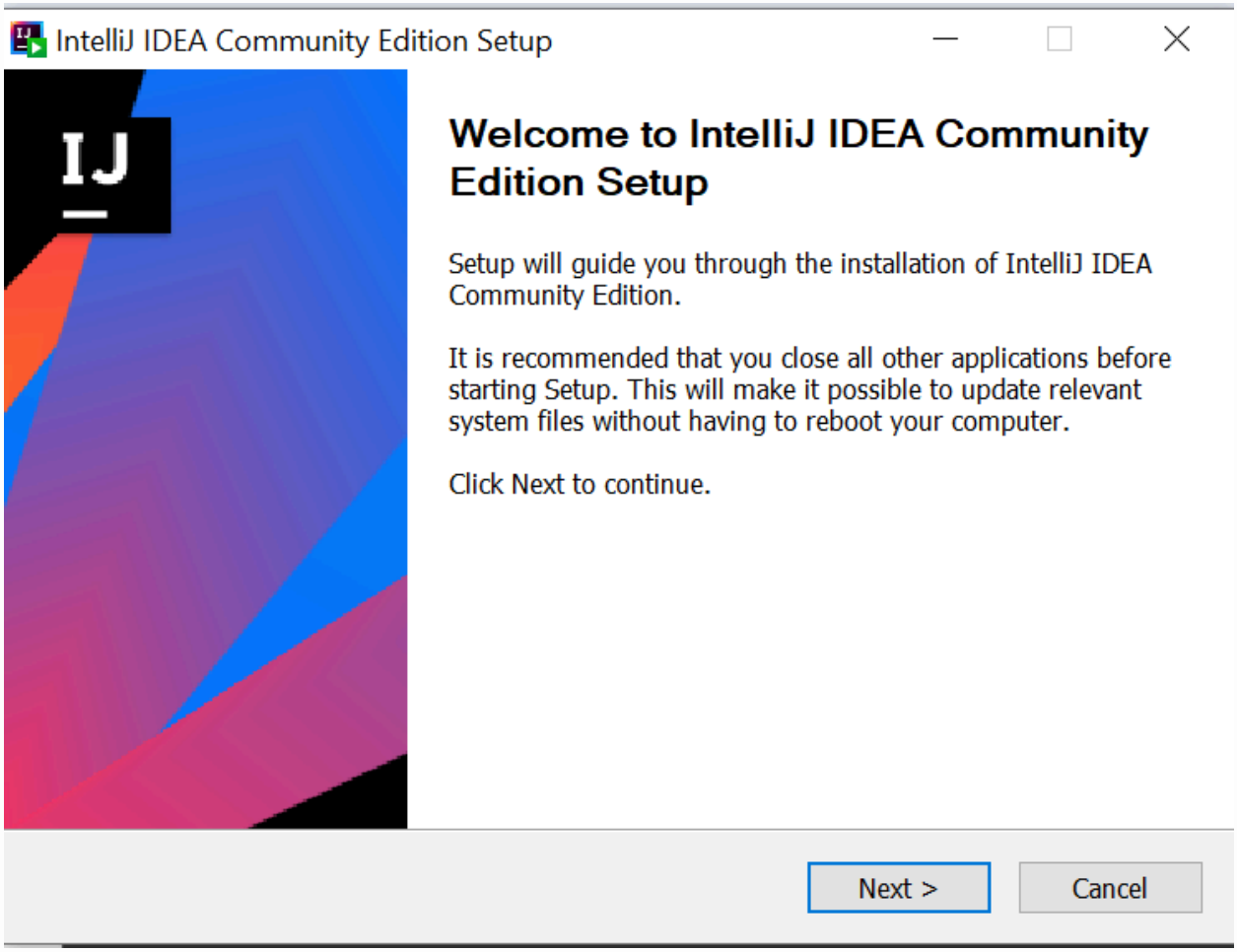
IntelliJ IDEA is an Integrated Development Environment(IDE) by JetBrains. It attempts to integrate all of the development tools that you might need into one single place.

Please download the Community Edition.

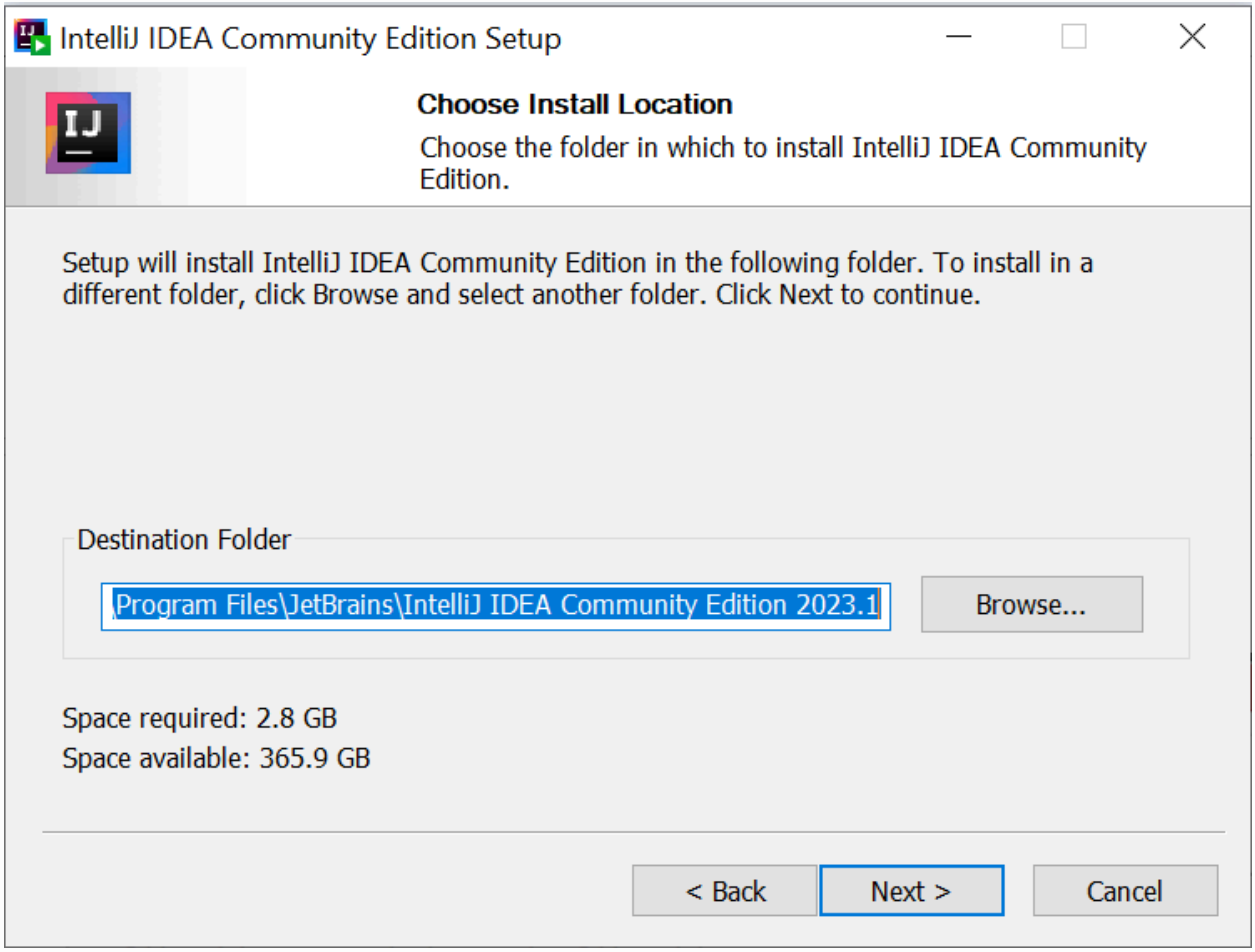
Windows:

After downloading, follow the below installation steps:

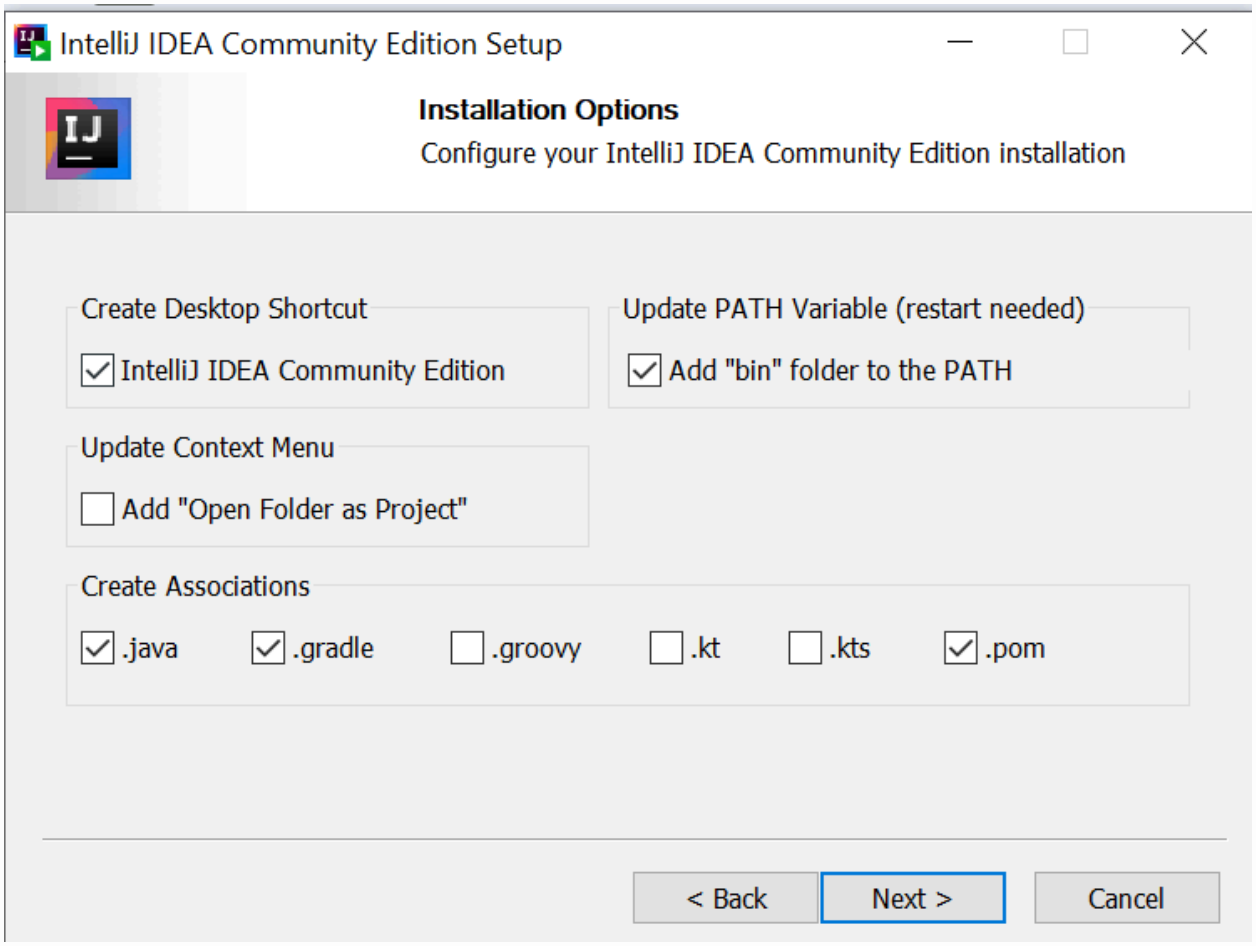
1. Open the .exe and choose next



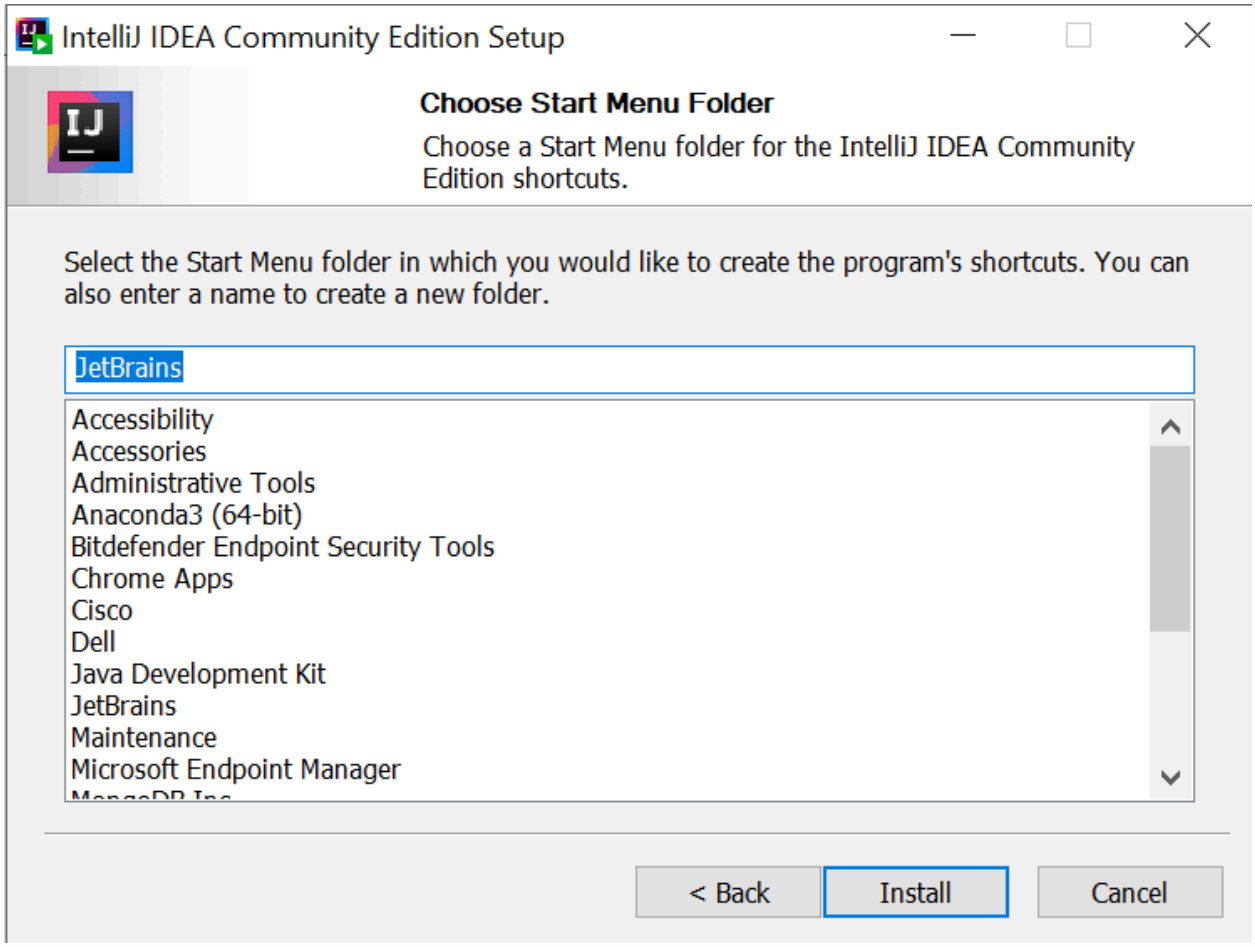
2. Change the destination path according to your preference.

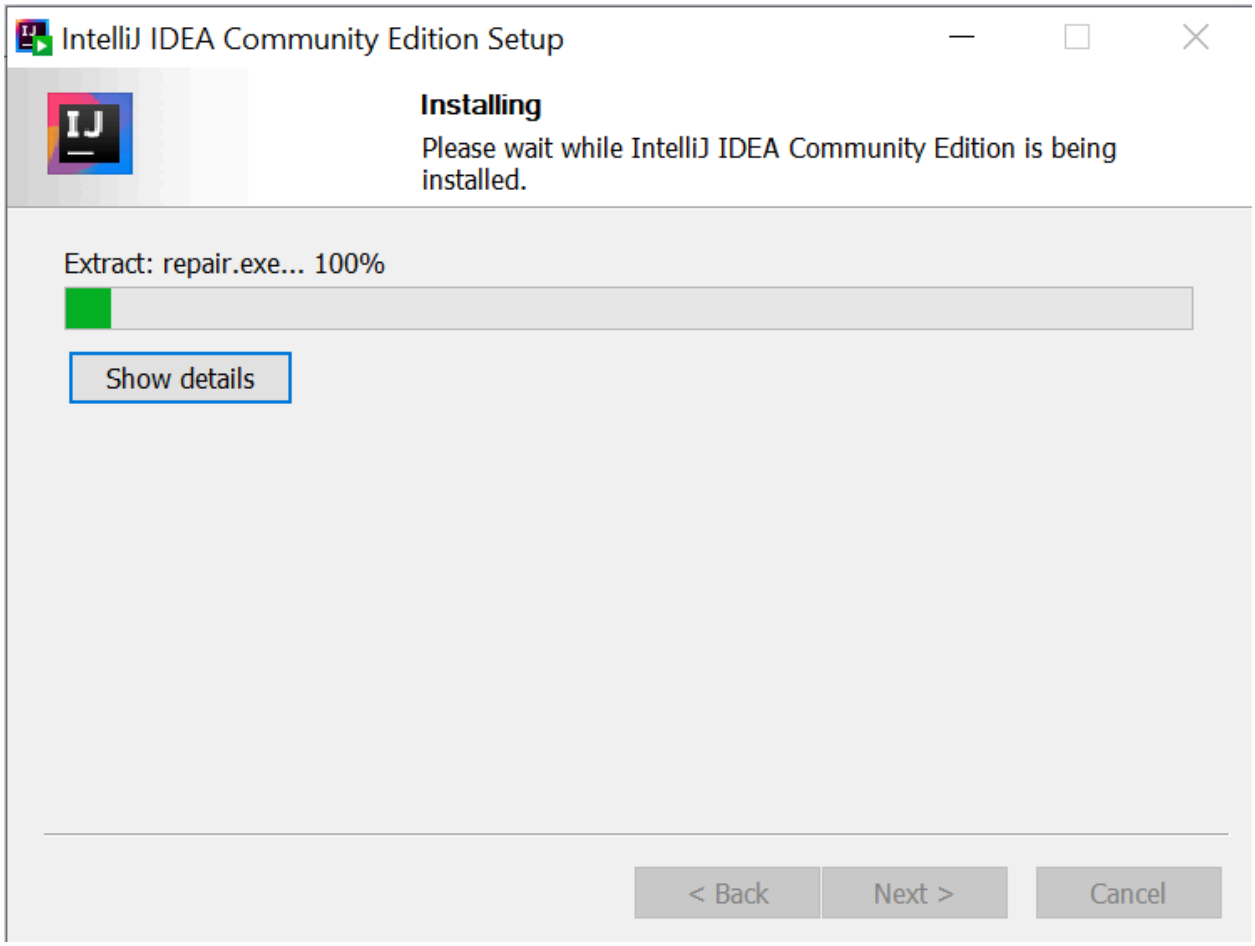


3. Check all the boxes and click next.

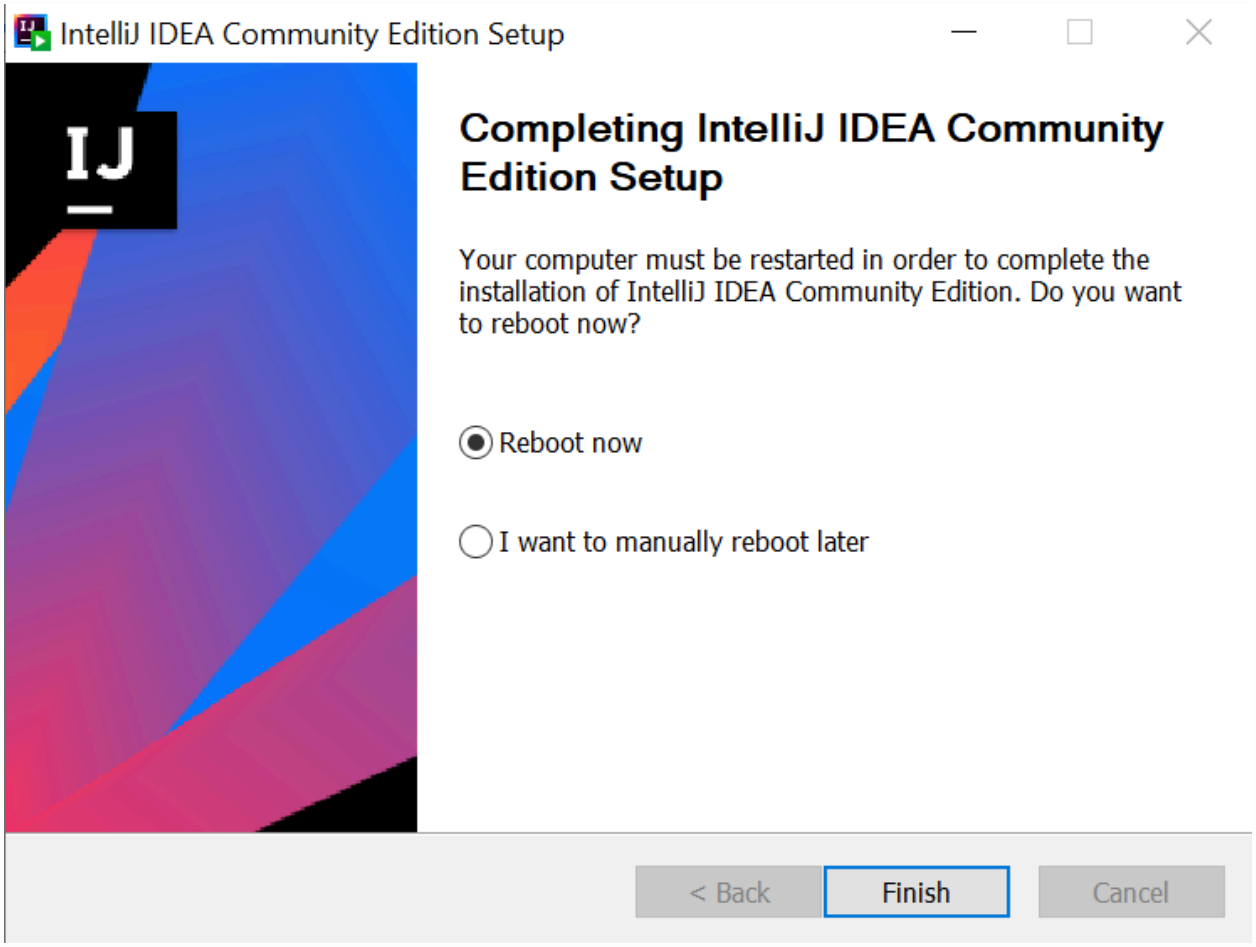

IntelliJ IDEA Community Edition Setup
Installation Options
Configure your IntelliJ IDEA Community Edition installation
Create Desktop Shortcut
IntelliJ IDEA Community Edition
Update PATH Variable (restart needed)
Add "bin" folder to the PATH
Update Context Menu
Add "Open Folder as Project"
Create Associations
.java
.gradle
.groovy
.kt
.kts
.pom
< Back
Next >
Cancel

4. Select JetBrains and click **install**.

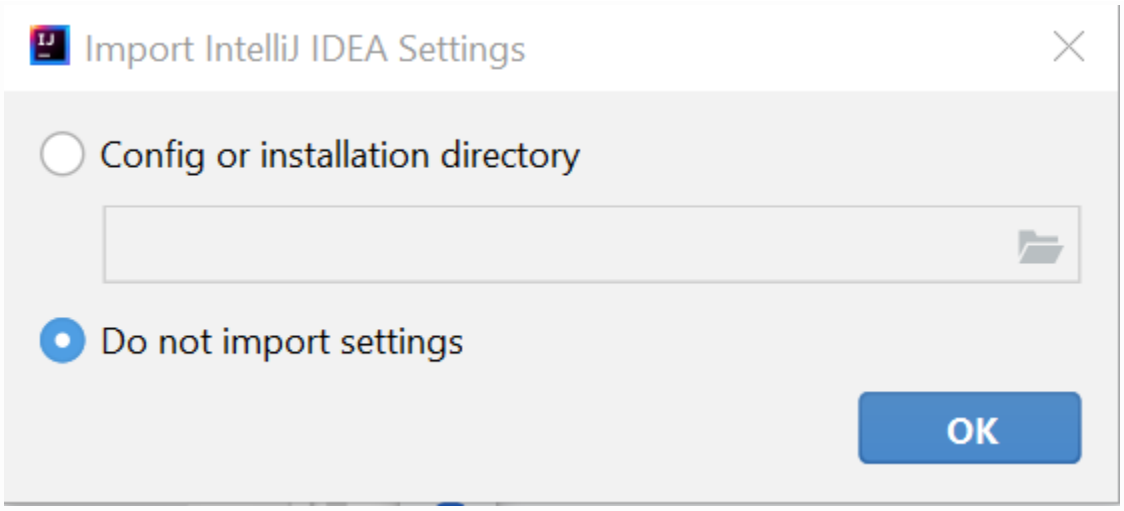




5. Choose your option according to your preference and click **Finish**.
   (Recommended to **Reboot Now**)



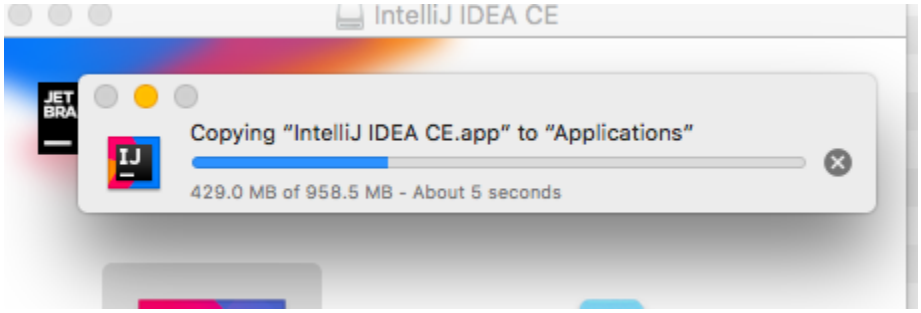
6. Leave the default chosen for importing (if you do not have configuration files or a folder).



Mac:

(source: GeekforGeeks)

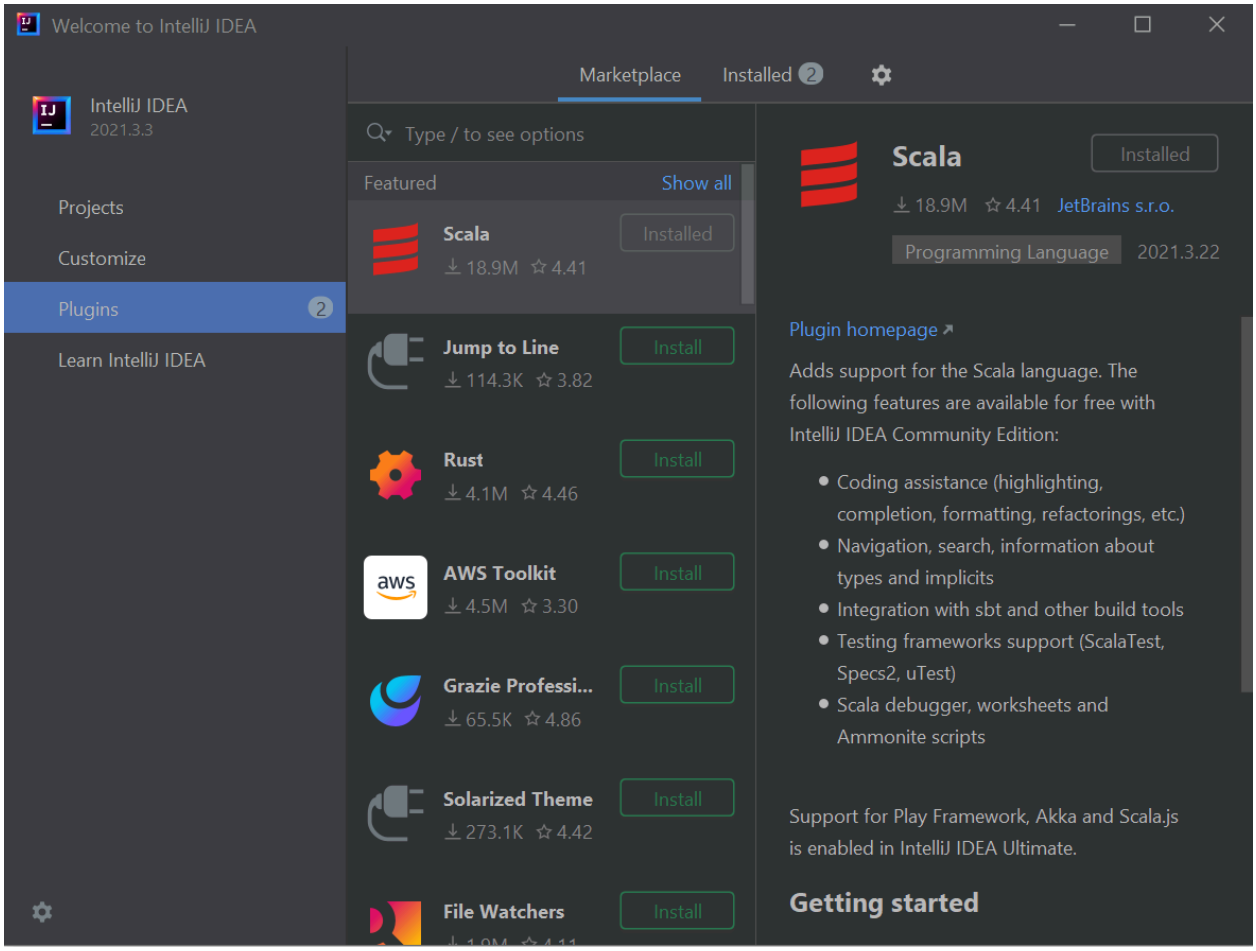
1. Select the platform as Mac and download the Community edition.
2. Open the downloaded file. A message pops up on the screen which says *verifying and opening*. After this screen pops up, drag and drop the IntelliJ to the *Applications* folder.
3. After dragging and dropping it into the applications folder, the following screen is obtained:



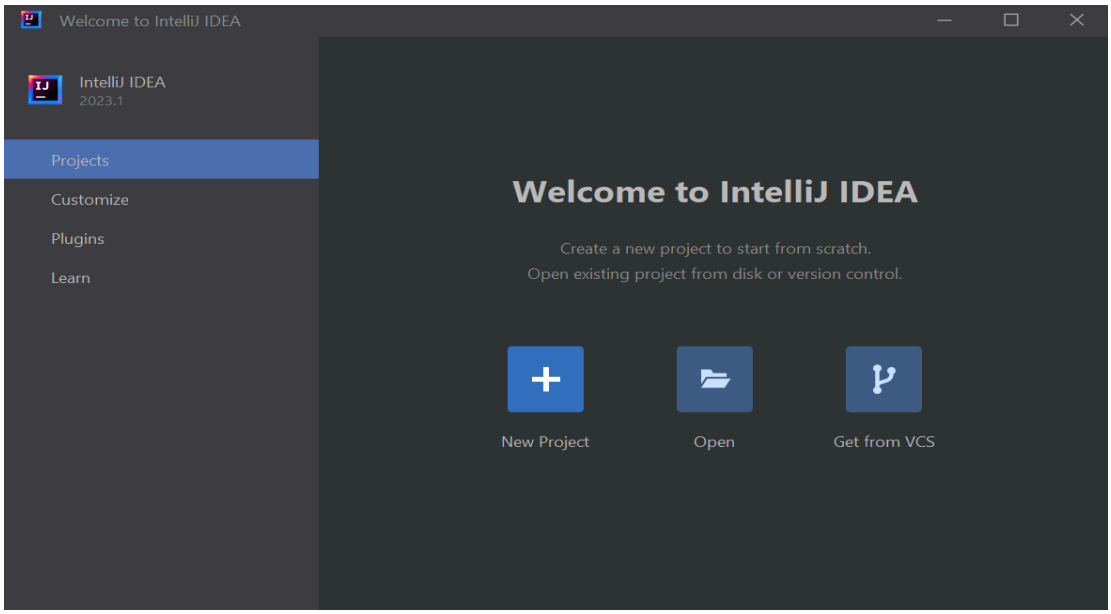
4. After the copying process is done, click on the applications folder in the side panel and open IntelliJ.
5. A welcome screen now pops up:

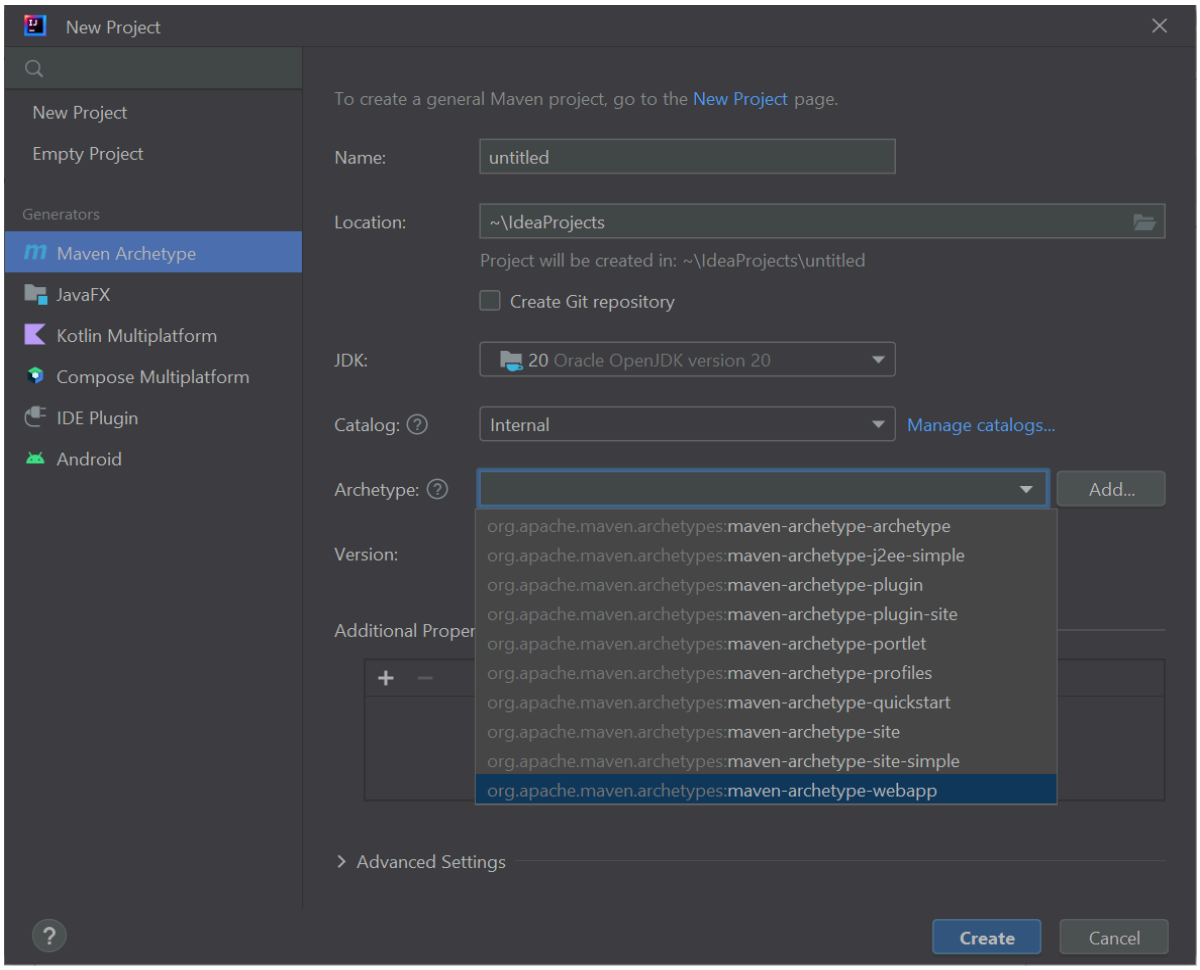
IntelliJ Setup:

1. Open IntelliJ IDE(In case you already have IntelliJ, go to marketplace(ctrl + alt + s))
2. Install the Scala plugin.

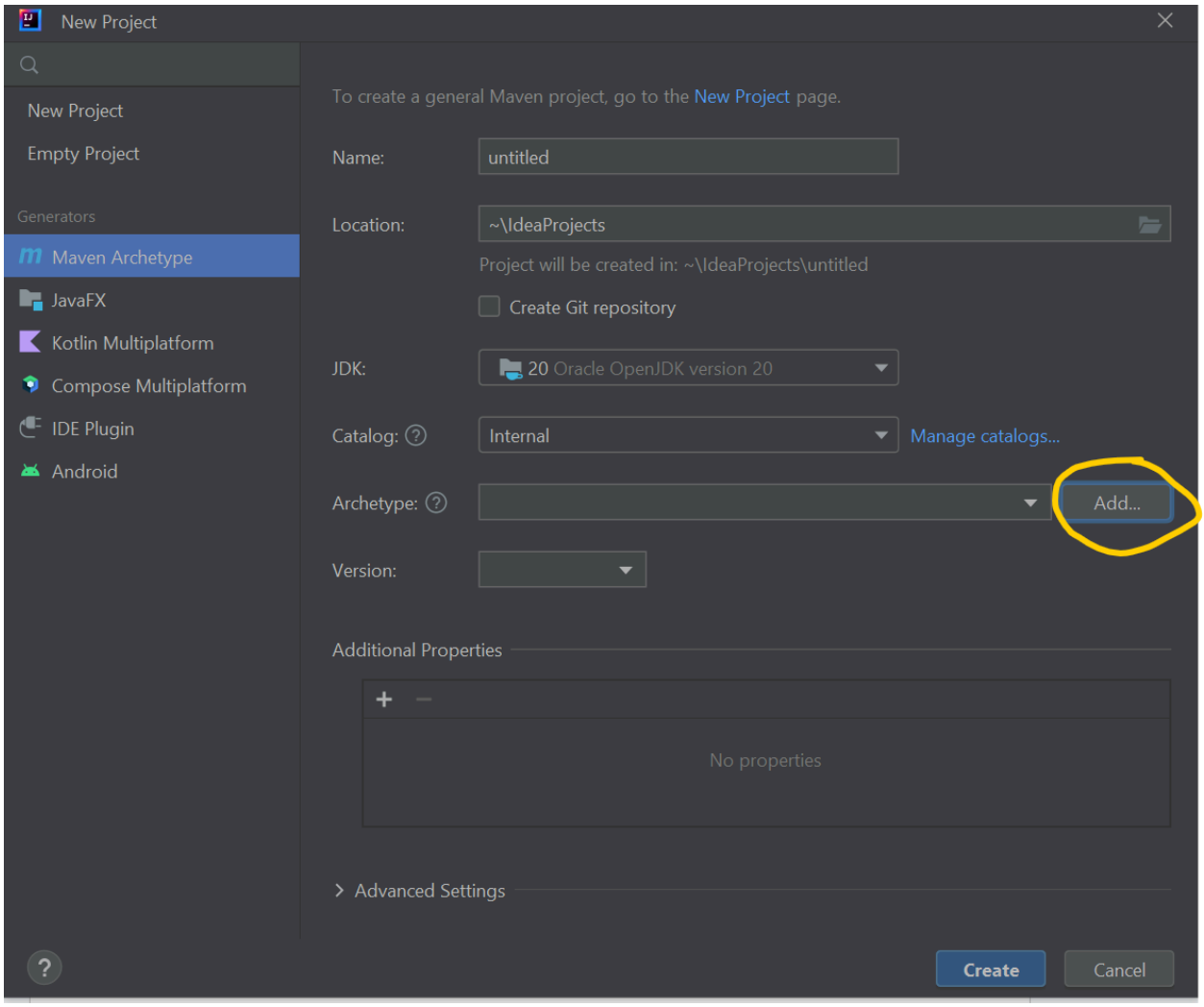


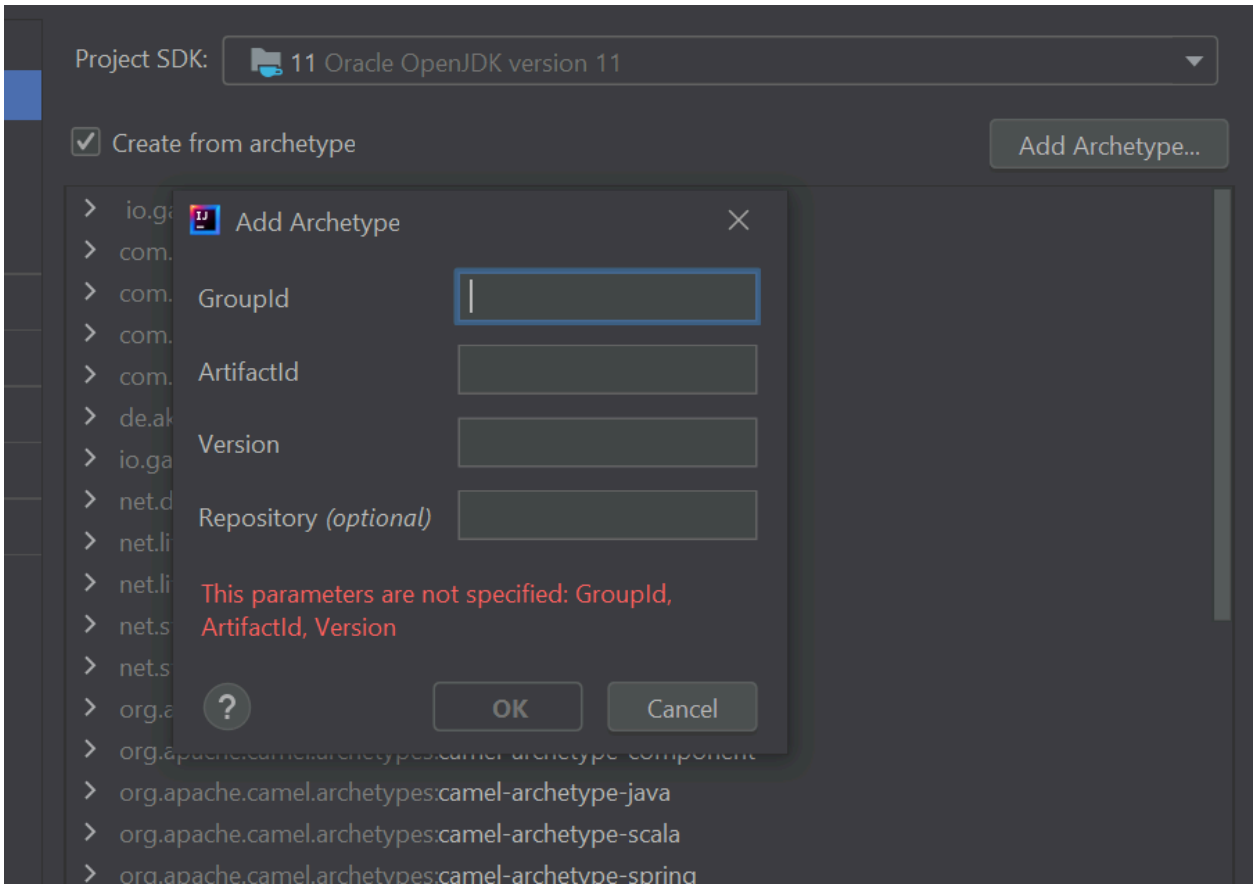
3. Then, create a new maven project, and select **New Project**.





4. Select “**maven archetype**” from the generators and click add archetype.





5. Go to gatling highcharts maven repository and click the previous stable version.

Home » io.gatling.highcharts » gatling-highcharts-maven-archetype

Gatling Highcharts Maven Archetype

gatling-highcharts-maven-archetype

Tags	build archetype build-system maven chart
Ranking	**#458671 in MvnRepository (See Top Artifacts)**

Central (42) Spring Plugins (7)

	Version	Vulnerabilities	Repository	Usages	Date
3.6.x	3.6.1		Central	0	Jul 06, 2021
	3.6.0		Central	0	May 22, 2021
3.5.x	3.5.1		Central	0	Jan 31, 2021
	3.5.0		Central	0	Dec 15, 2020

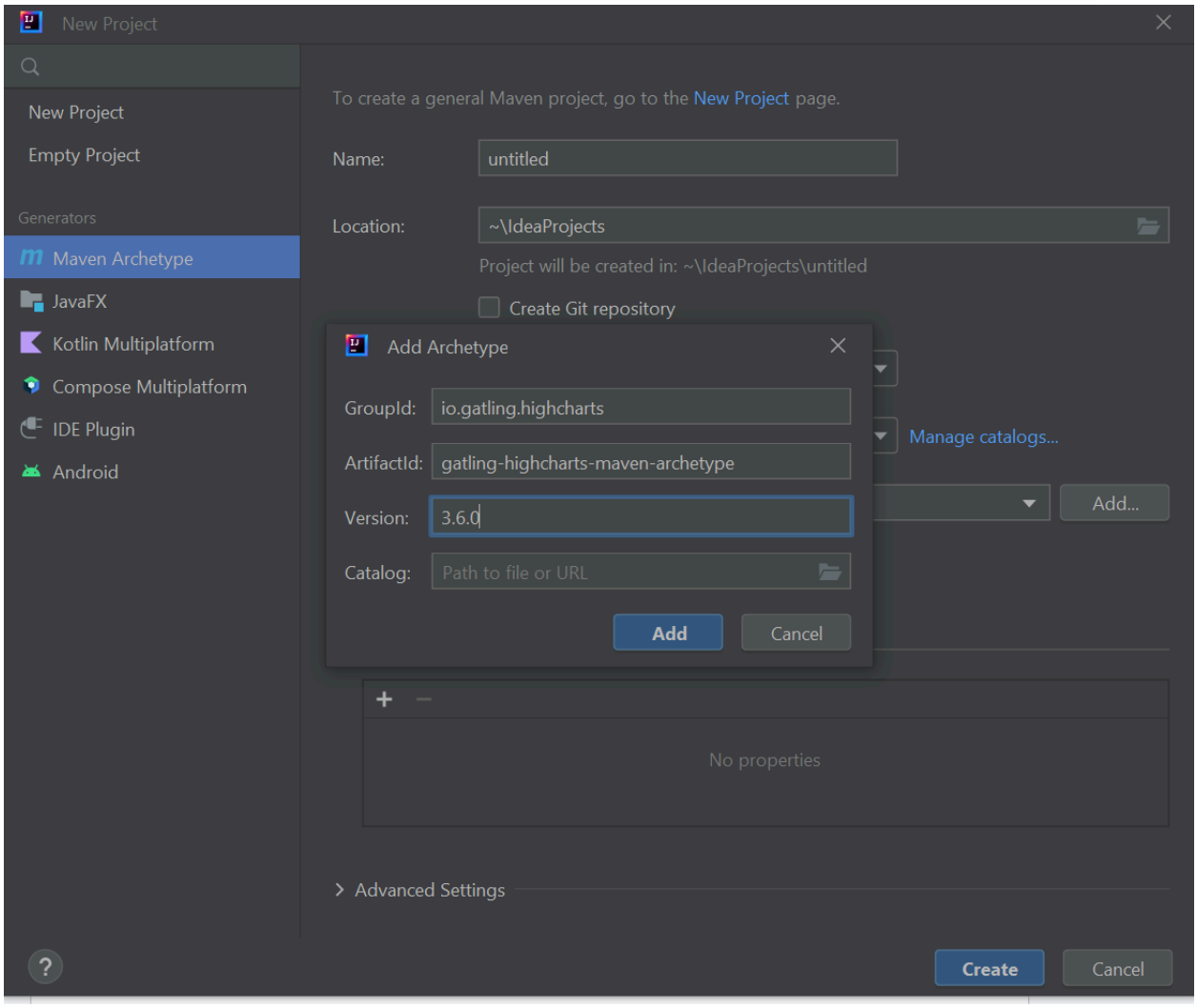
6. Then copy and paste the group id, artifact id, and version into archetype input fields.

Maven Gradle Gradle (Short) Gradle (Kotlin) SBT Ivy Grape Leiningen Buildr

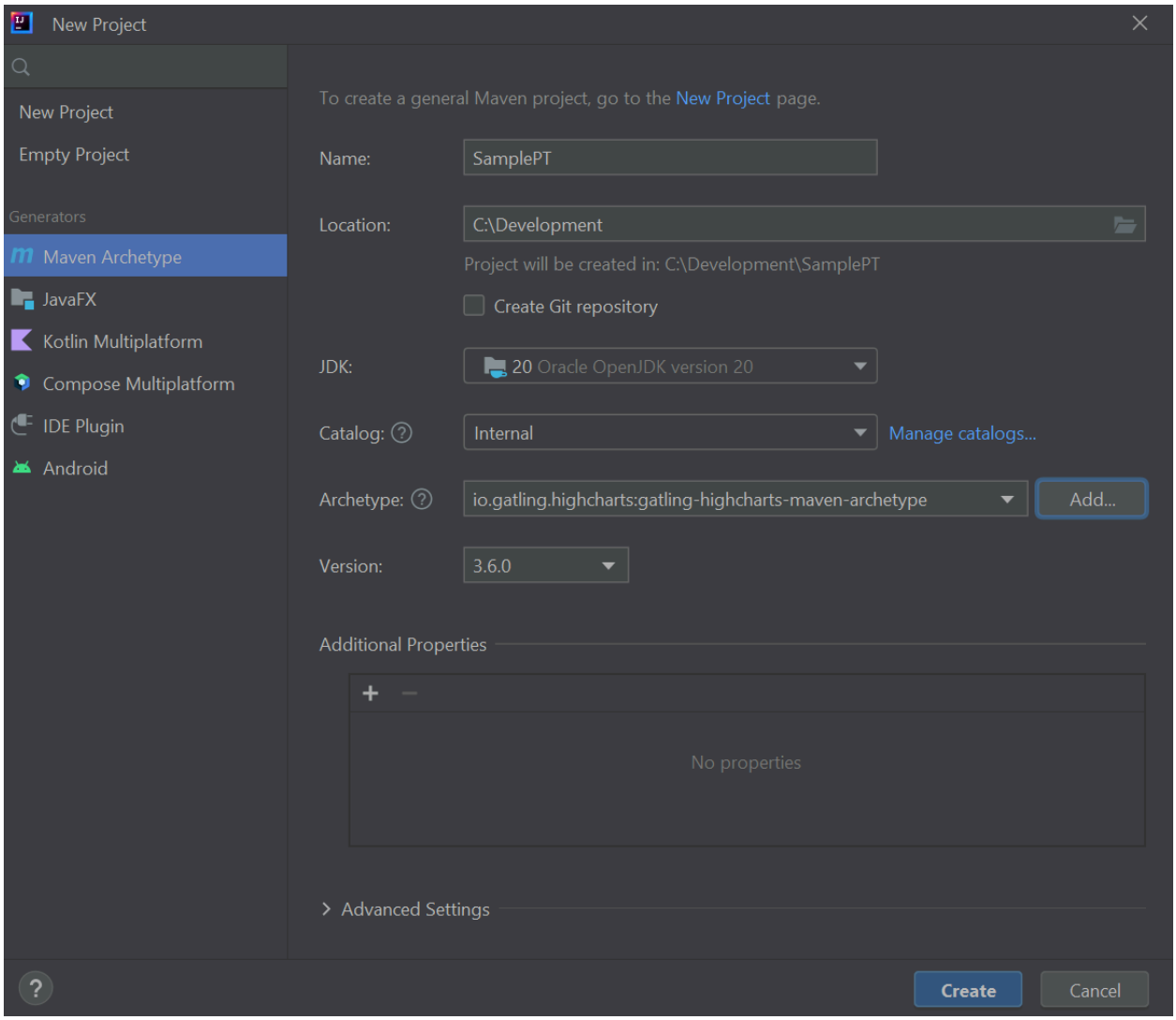
```
<!-- https://mvnrepository.com/artifact/io.gatling.highcharts/gatling-highcharts-maven-archetype -->
<dependency>
    <groupId>io.gatling.highcharts</groupId>
    <artifactId>gatling-highcharts-maven-archetype</artifactId>
    <version>3.6.0</version>
</dependency>
```

☑ Include comment with link to declaration

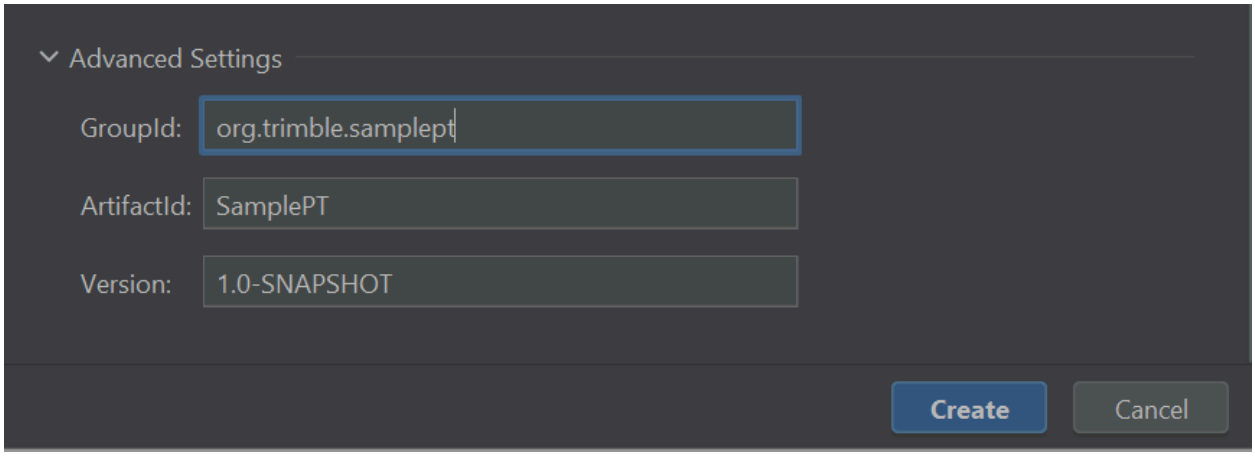
7. Copy and paste the GroupId, ArtifactId & version and put them accordingly, and then click **Add**.



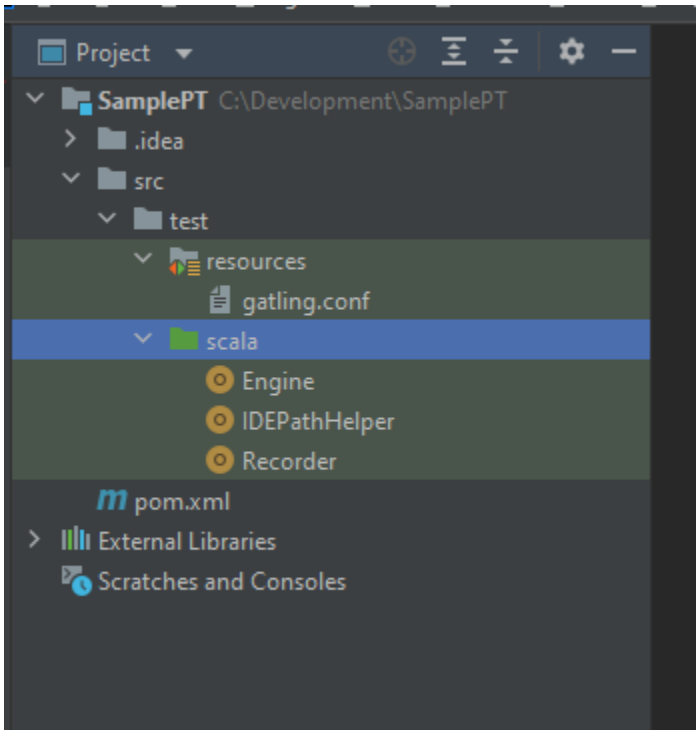
8. Fill in the necessary fields and also choose the desired destination path.



9. Go to advanced settings and change GroupId, ArtifactId, and Version based on your project.

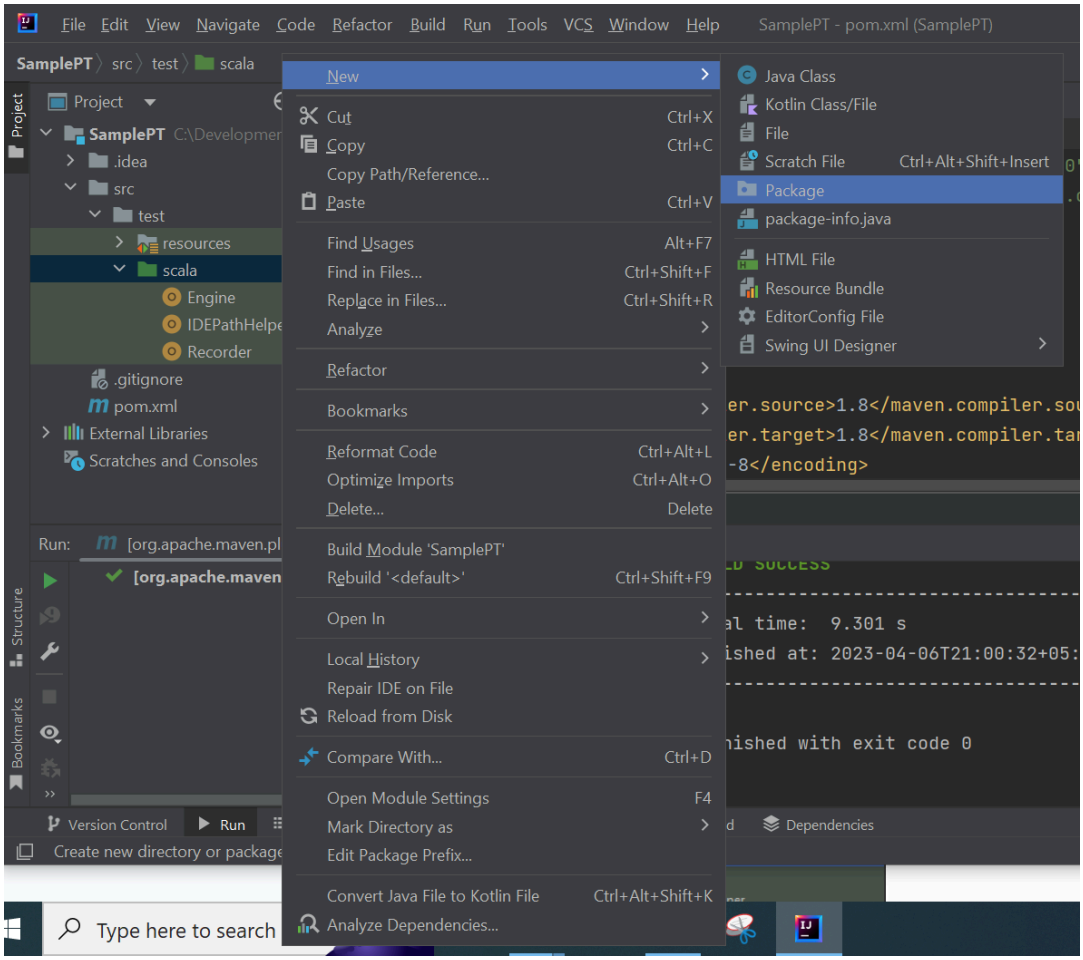


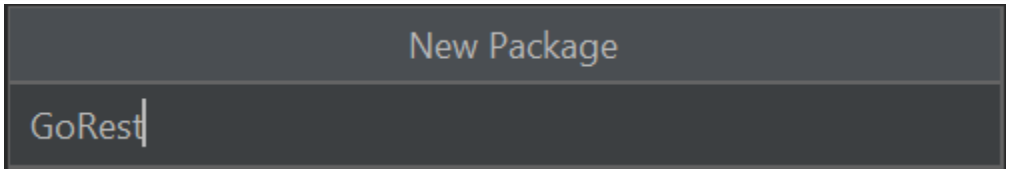
10. Project has been successfully set up.

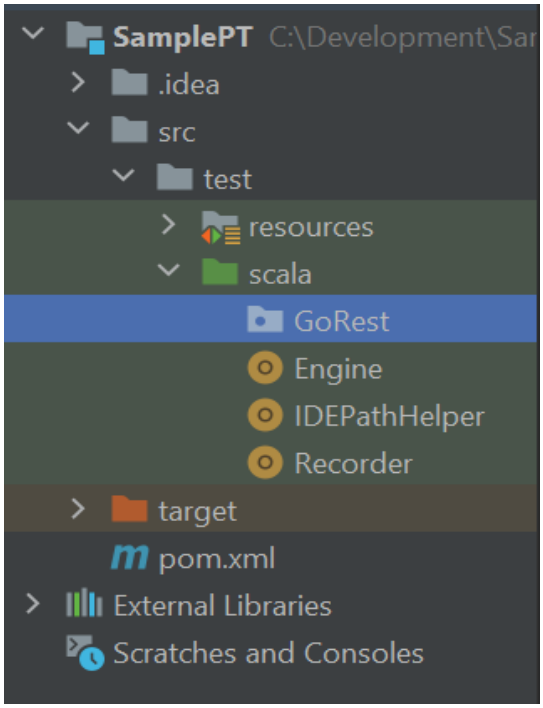


Script Development:

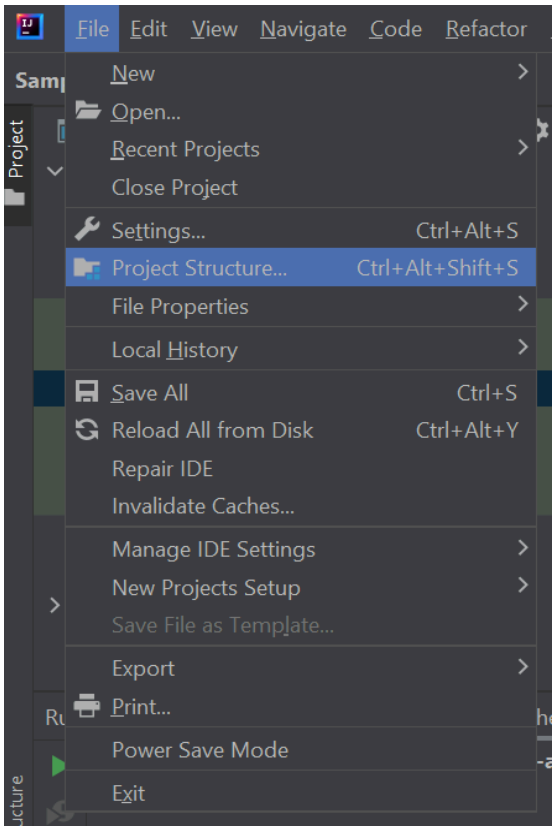
1. Let's create a PT script for GoRest. First, create a package to maintain all objects and controller related to GoRest.


File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
SamplePT - pom.xml (SamplePT)
SamplePT 〉 src 〉 test 〉 scala
Project
SamplePT C:\Developmer
.idea
src
test
resources
scala
Engine
IDEPathHelpe
Recorder
.gitignore
pom.xml
External Libraries
Scratches and Consoles
New
Cut Ctrl+X
Copy Ctrl+C
Copy Path/Reference...
Paste Ctrl+V
Find Usages Alt+F7
Find in Files... Ctrl+Shift+F
Replace in Files... Ctrl+Shift+R
Analyze
Refactor
Bookmarks
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Build Module 'SamplePT'
Rebuild '<default>' Ctrl+Shift+F9
Open In
Local History
Repair IDE on File
Reload from Disk
Compare With... Ctrl+D
Open Module Settings F4
Mark Directory as
Edit Package Prefix...
Convert Java File to Kotlin File Ctrl+Alt+Shift+K
Analyze Dependencies...
Java Class
Kotlin Class/File
File
Scratch File Ctrl+Alt+Shift+Insert
Package
package-info.java
HTML File
Resource Bundle
EditorConfig File
Swing UI Designer
er.source>1.8</maven.compiler.sou
er.target>1.8</maven.compiler.tar
-8</encoding>
Run:
[org.apache.maven.pl
[org.apache.maven
al time: 9.301 s
ished at: 2023-04-06T21:00:32+05:
ished with exit code 0
Version Control
Run
Dependencies
Create new directory or package
Type here to search

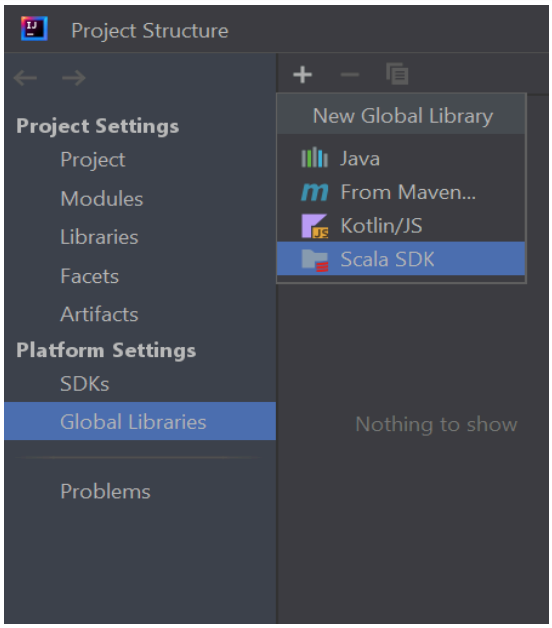

New Package
GoRest



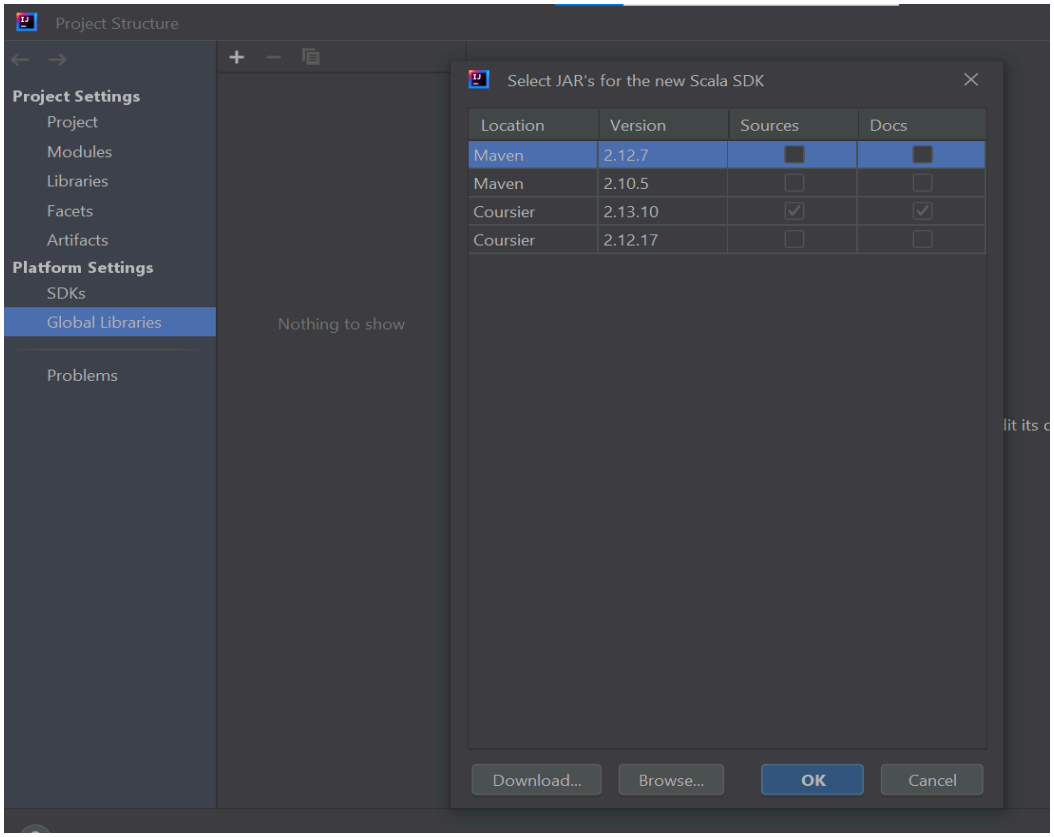
2. Before development, please make sure the Scala library is downloaded and attached globally. Go to **File- >Project Structure**.

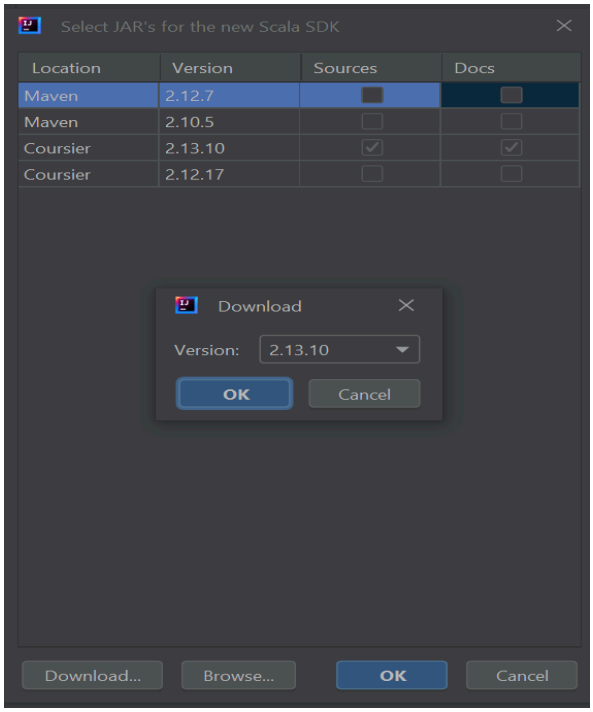


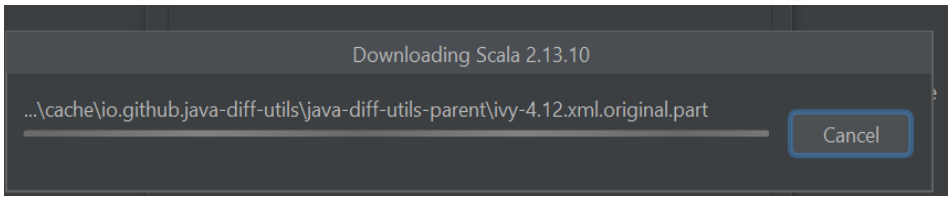
3. Go to **Global Libraries** and click the "**+**" icon and select **Scala SDK**.



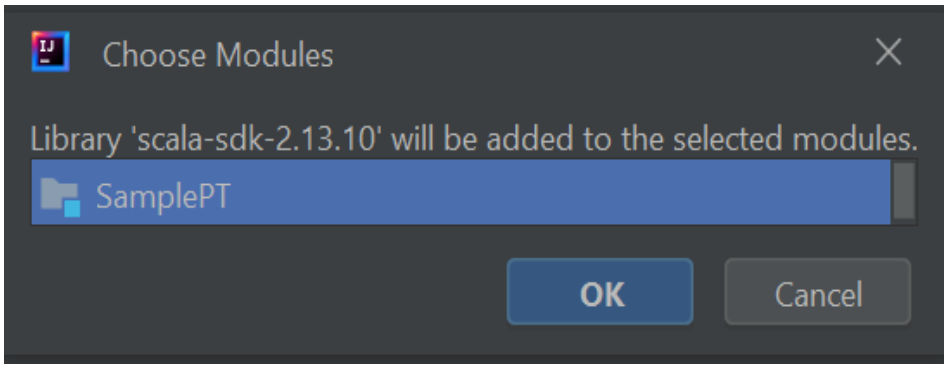
4. Select Download and choose Scala version

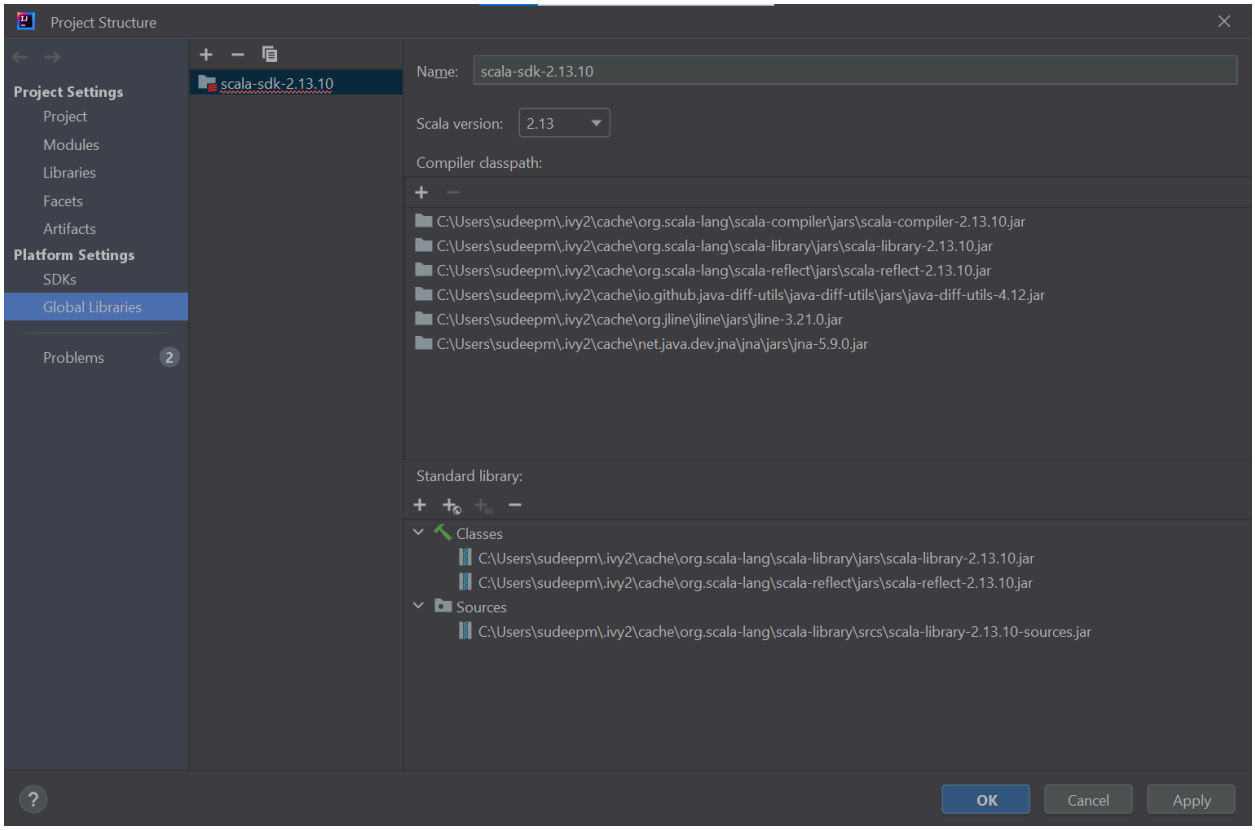


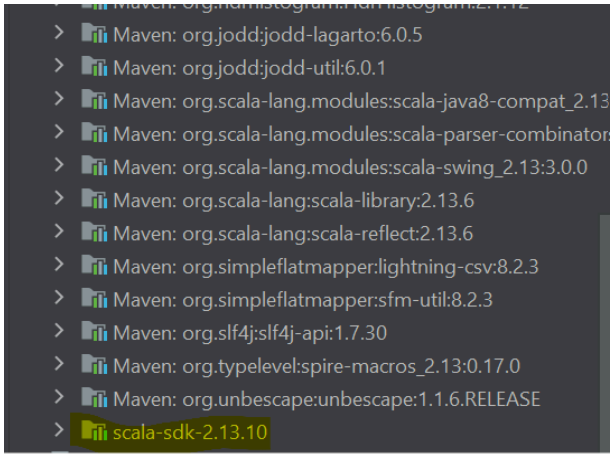




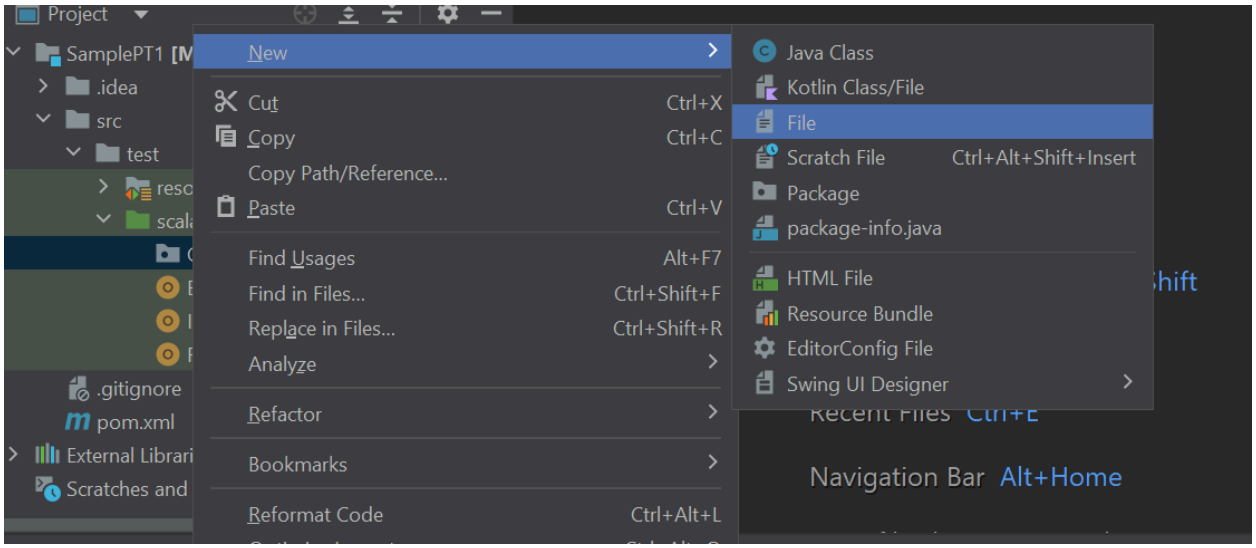
5. After scala is downloaded, it will prompt you to add in the maven project we created. Click **OK**. It will be added under the **External Libraries** of the SamplePT project.

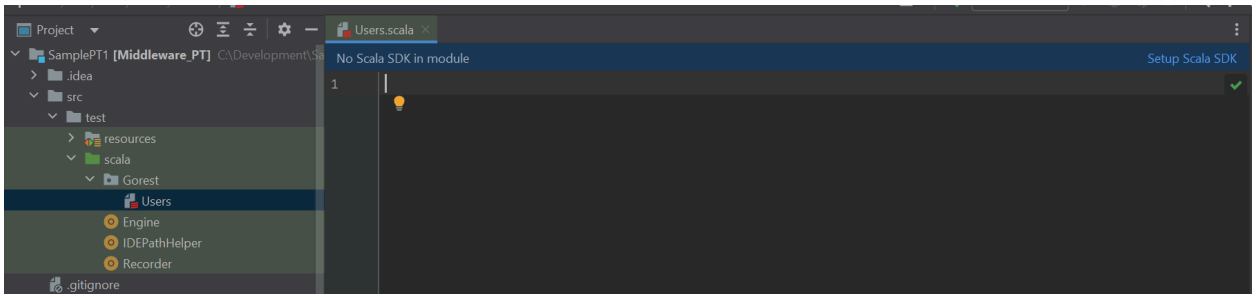




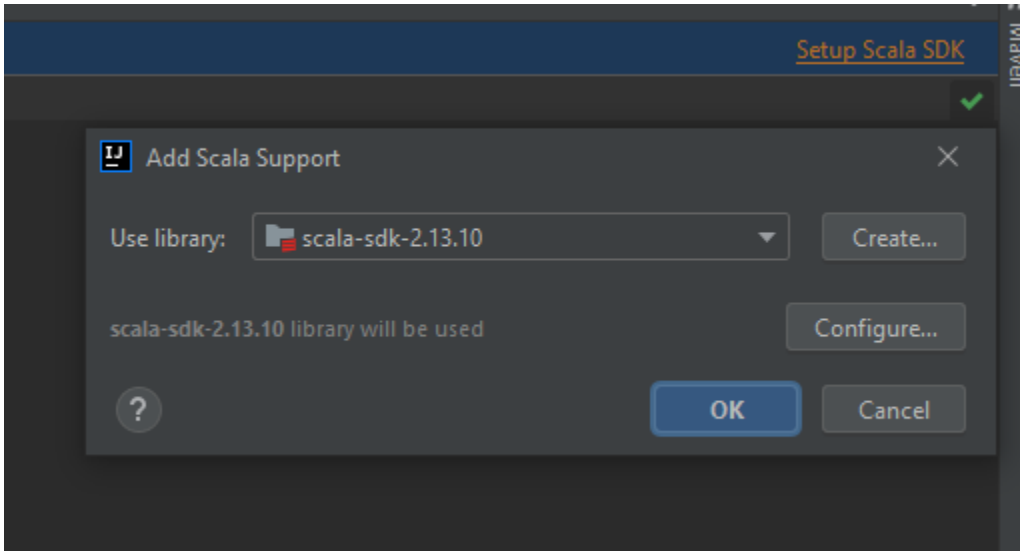


Note: We have to set up Scala SDK for each new project. Initially, the Scala SDK library will not be available under External Libraries. To include that library, first, create a new file with the "**.scala**" extension. Then, in the top right corner, you can see the pop-up "**Setup Scala SDK**". Click it.

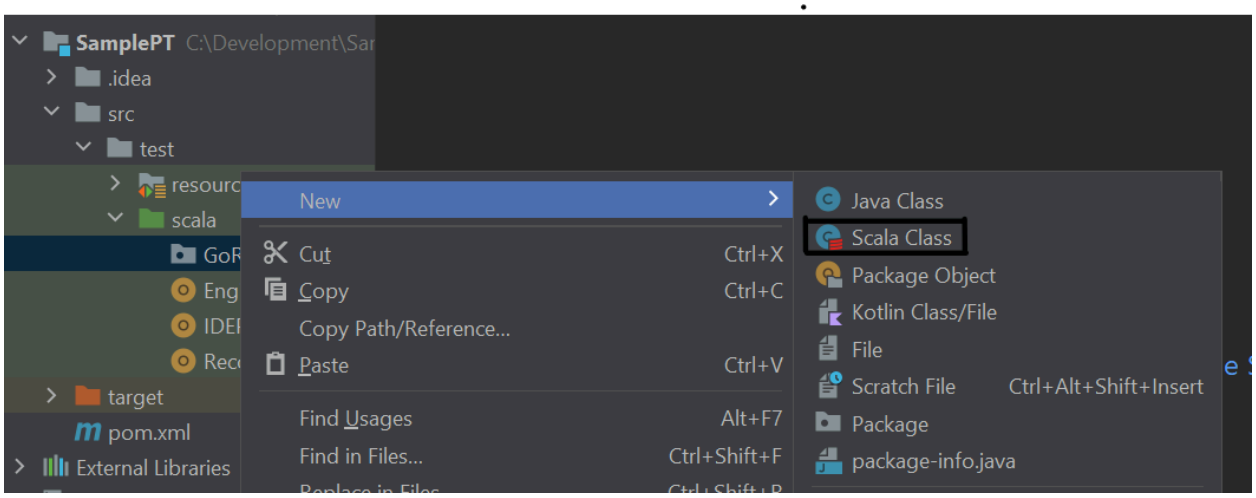


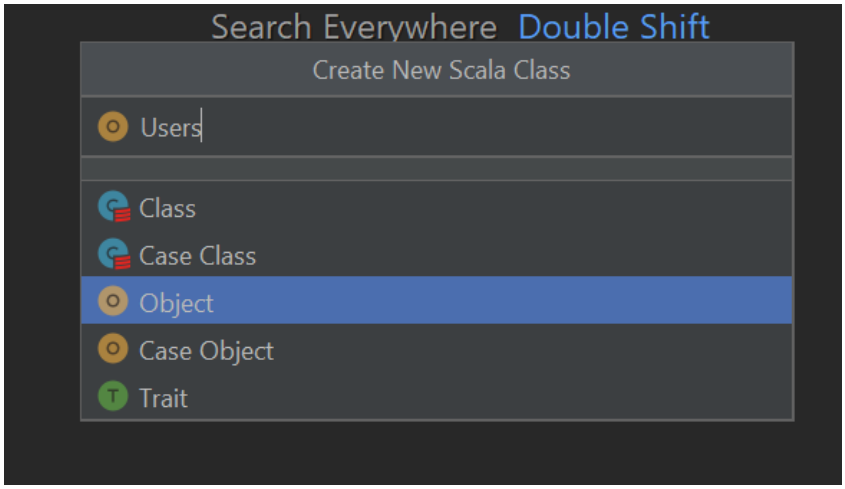


Choose the Scala SDK version you want to add and click ok.



6. Then, for demo purposes, I have written a scala program for two APIs from GoRest.
7. Create a scala object as “**Users.scala**”.



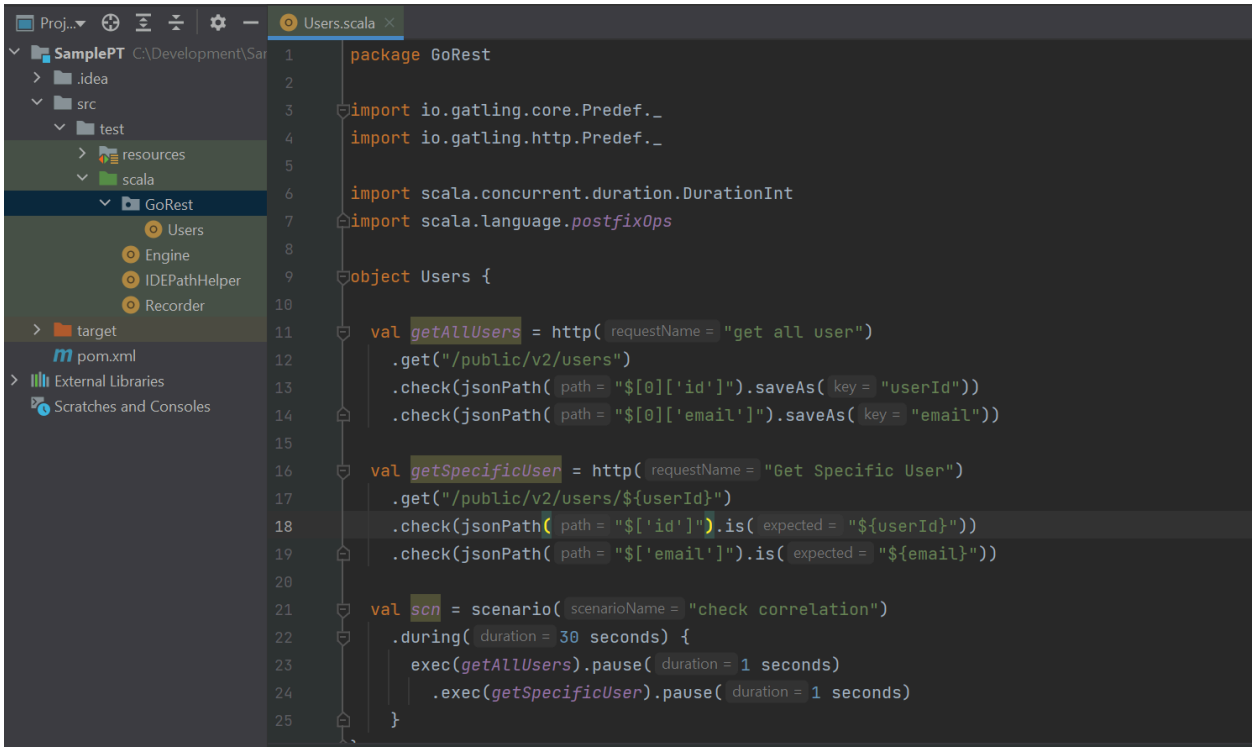


1. Copy and paste the below code into **Users.scala** file.

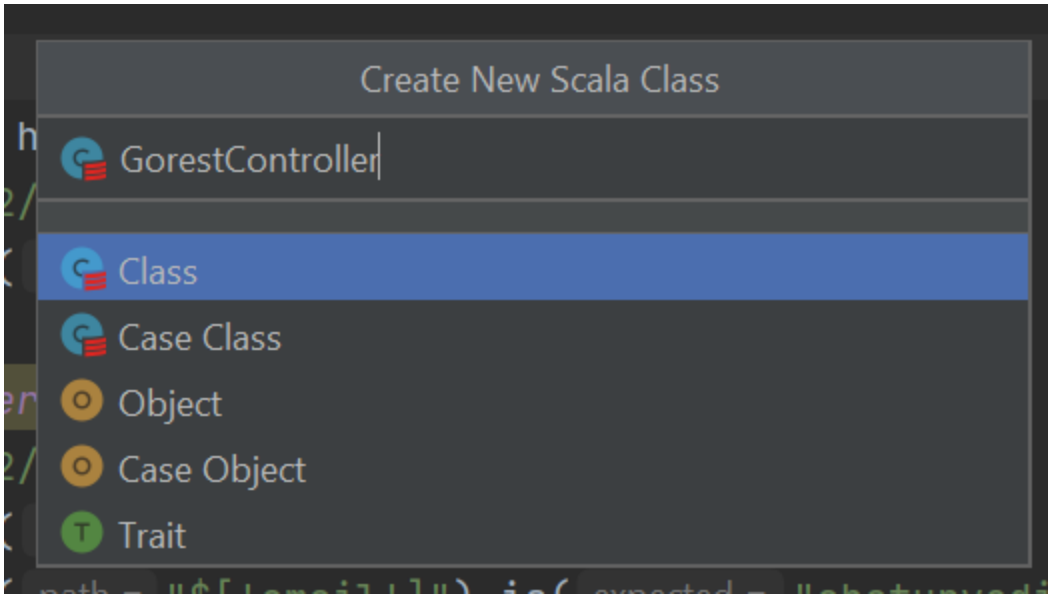
Users.scala

```
package GoRest

import io.gatling.core.Predef._
import io.gatling.http.Predef._

import scala.concurrent.duration.DurationInt
import scala.language.postfixOps

object Users {

 val getAllUsers = http("get all user")
   .get("/public/v2/users")
   .check(jsonPath("$[0]['id']").saveAs("userId"))
   .check(jsonPath("$[0]['email']").saveAs("email"))

 val getSpecificUser = http("Get Specific User")
   .get("/public/v2/users/${userId}")
```

```
   .check(jsonPath("$['id']").is("${userId}"))
   .check(jsonPath("$['email']").is("${email}"))

 val scn = scenario("check correlation")
   .during(30 seconds) {
     exec(getAllUsers).pause(1 seconds)
       .exec(getSpecificUser).pause(1 seconds)
   }
}
```



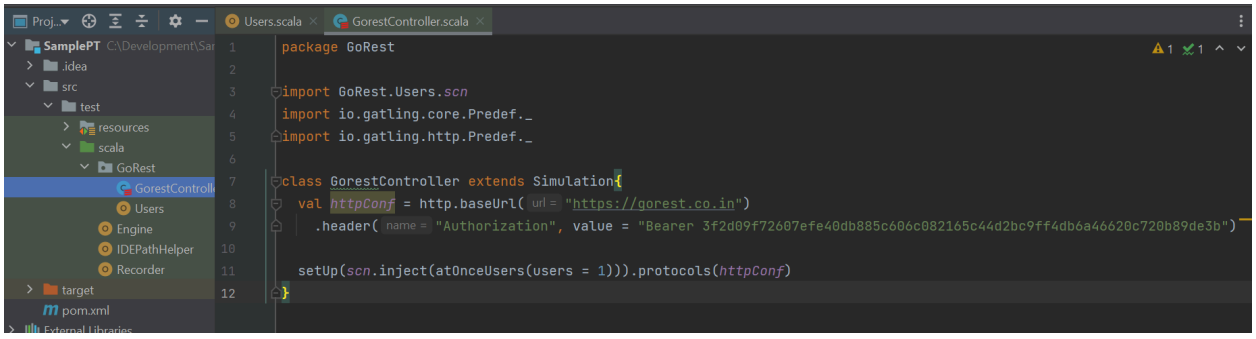
1. Create a new controller file in the same GoRest folder.



1. Copy and paste the below code into **GorestController.scala** file.

GorestController.scala

```
package GoRest

import GoRest.Users.scn
import io.gatling.core.Predef._
import io.gatling.http.Predef._

class GorestController extends Simulation{
 val httpConf = http.baseUrl("https://gorest.co.in")
   .header("Authorization", value = "Bearer
3f2d09f72607efe40db885c606c082165c44d2bc9ff4db6a46620c720b89de3b")

 setUp(scn.inject(atOnceUsers(users = 1))).protocols(httpConf)
}
```



8. After completed with the development, open the terminal and run the following command:

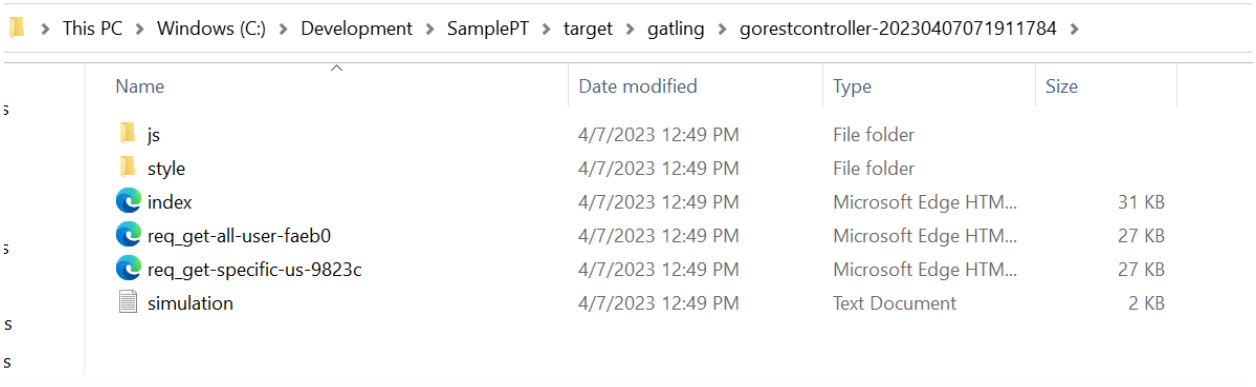
mvn gatling:test -D gatling.simulationClass=GoRest.GorestController

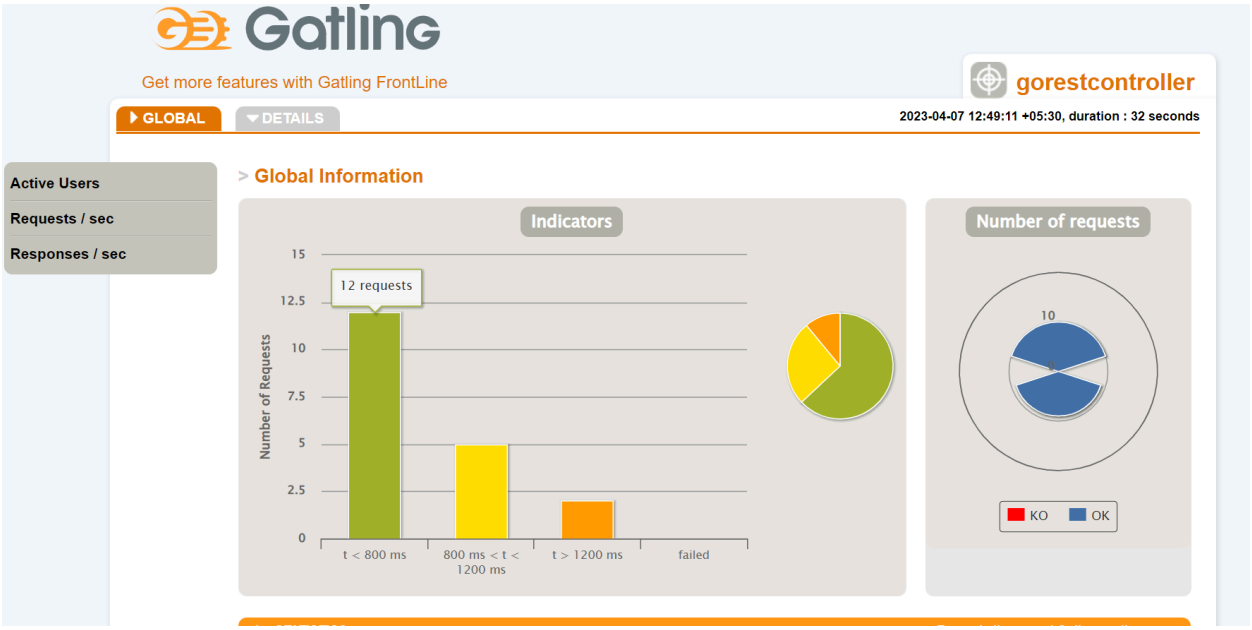
```
PS C:\Development\SamplePT> mvn gatling:test -D gatling.simulationClass=GoRest.GorestController
[INFO] Scanning for projects...
[INFO]
[INFO] ------------------< org.trimble.samplept:SamplePT >-------------------
[INFO] Building SamplePT 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- gatling-maven-plugin:3.1.2:test (default-cli) @ SamplePT ---

Simulation GoRest.GorestController started...

================================================================================
2023-04-07 12:41:25                                           5s elapsed
---- Requests ------------------------------------------------------------------
> Global                                                   (OK=3      KO=1     )
> get all user                                             (OK=2      KO=0     )
> Get Specific User                                        (OK=1      KO=1     )
---- Errors --------------------------------------------------------------------
> jsonPath($['id']).find.is(3704), but actually found 803462          1 (100.0%)

---- check correlation ----------------------------------------------------------
[--------------------------------------------------------------------------]  0%
```

```
================================================================================
2023-04-07 12:49:44                                           30s elapsed
---- Requests ------------------------------------------------------------------
> Global                                                   (OK=19     KO=0     )
> get all user                                             (OK=10     KO=0     )
> Get Specific User                                        (OK=9      KO=0     )

---- check correlation ----------------------------------------------------------
[##########################################################################]100%
          waiting: 0      / active: 0      / done: 1
================================================================================

Simulation GoRest.GorestController completed in 30 seconds
Parsing log file(s)...
Parsing log file(s) done
Generating reports...
```

```
Terminal:  Local
================================================================================
---- Global Information --------------------------------------------------------
> request count                                         19 (OK=19     KO=0     )
> min response time                                    169 (OK=169    KO=-     )
> max response time                                   2390 (OK=2390   KO=-     )
> mean response time                                   603 (OK=603    KO=-     )
> std deviation                                        614 (OK=614    KO=-     )
> response time 50th percentile                        244 (OK=244    KO=-     )
> response time 75th percentile                        868 (OK=868    KO=-     )
> response time 95th percentile                       1981 (OK=1981   KO=-     )
> response time 99th percentile                       2308 (OK=2308   KO=-     )
> mean requests/sec                                0.613 (OK=0.613  KO=-     )
---- Response Time Distribution ------------------------------------------------
> t < 800 ms                                           12 ( 63%)
> 800 ms < t < 1200 ms                                  5 ( 26%)
> t > 1200 ms                                           2 ( 11%)
> failed                                                0 (  0%)
================================================================================

Reports generated in 0s.
Please open the following file: C:\Development\SamplePT\target\gatling\gorestcontroller-20230407071911784\index.html
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  42.857 s
```

9. After the successful execution of the PT script, the html report will be generated. We can find that in the target folder.




Gatling
Get more features with Gatling FrontLine
gorestcontroller
GLOBAL
DETAILS
2023-04-07 12:49:11 +05:30, duration : 32 seconds
Active Users
Requests / sec
Responses / sec
Global Information
Indicators
Number of Requests
12 requests
15
12.5
10
7.5
5
2.5
0
t < 800 ms
800 ms < t < 1200 ms
t > 1200 ms
failed
Number of requests
10
KO
OK