How to Create a JAR File in Eclipse

There are several ways to create a JAR (Java Archive) file.
This guide walks you through one of the methods using Eclipse IDE.

Requirements

Before starting, make sure you have the following ready:

- **🧩 Eclipse IDE (latest version recommended)**

https://www.eclipse.org/downloads

- **☕ Java Development Kit (JDK 21 or newer)**

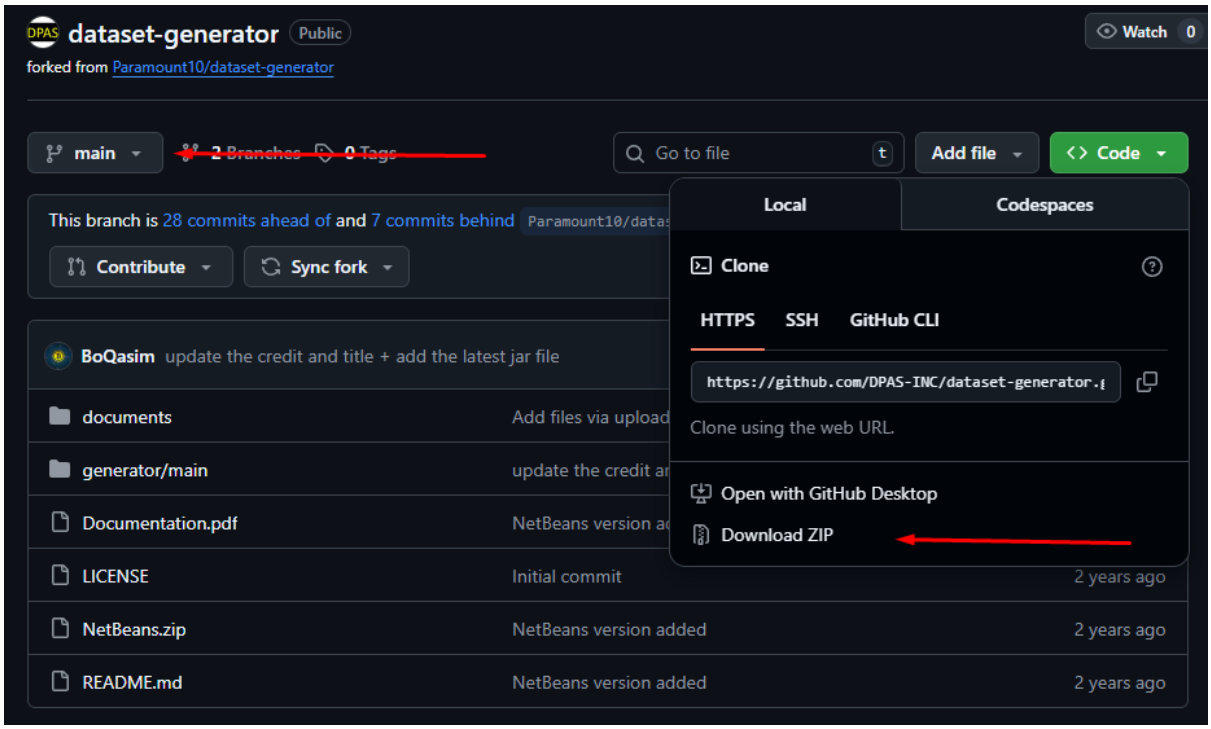
https://www.oracle.com/java/technologies/javase/jdk21-archive-downloads.html

---

Step 1: Download the Project from GitHub

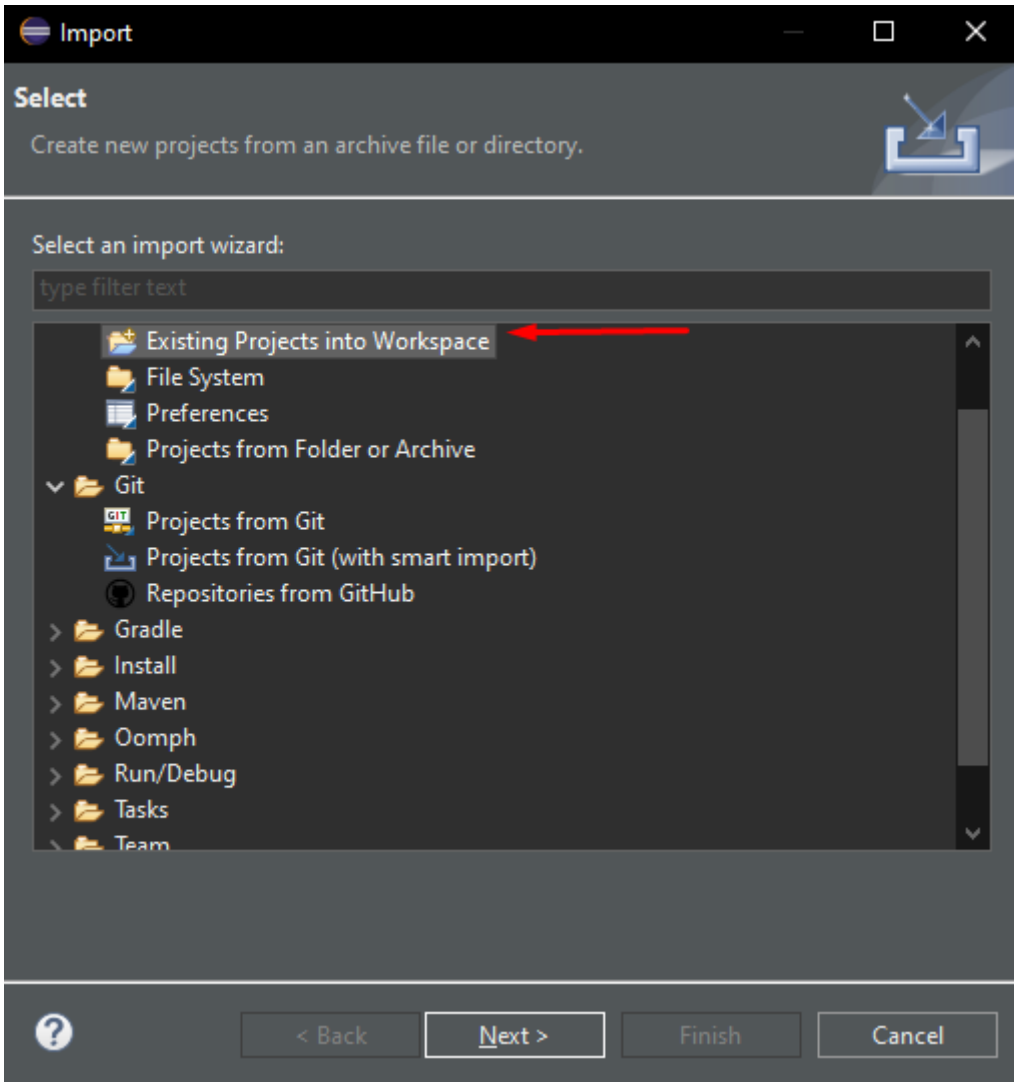
1. **Visit the GitHub repository: https://github.com/DPAS-INC/dataset-generator**
   Make sure you're on the main branch.
2. **Click Code → Download ZIP.**

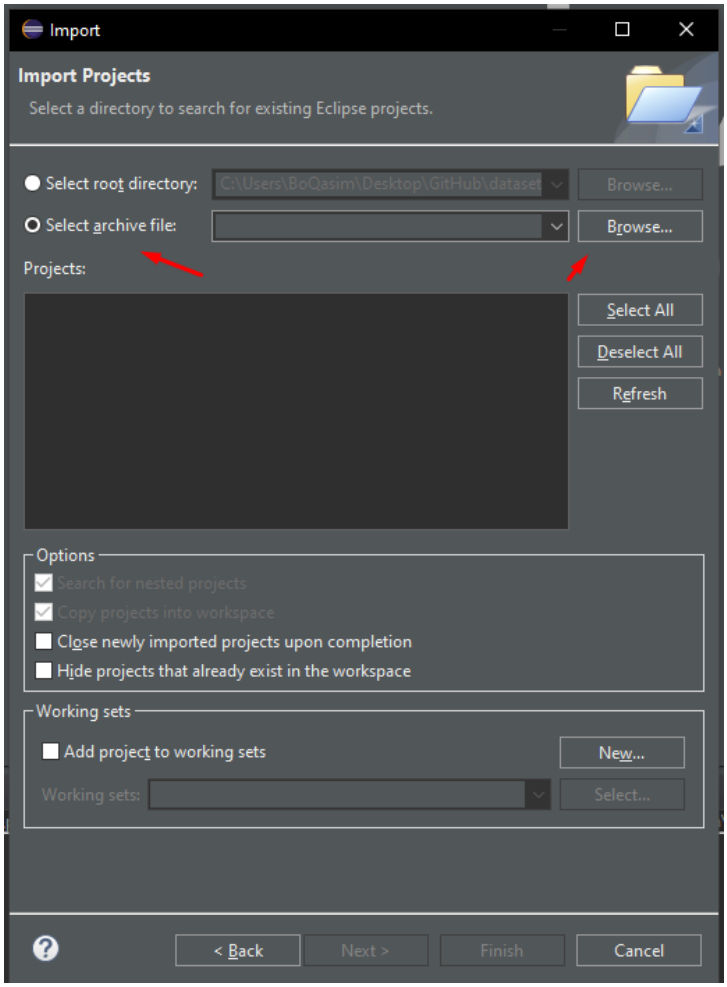
📸 ***Example screenshot:***

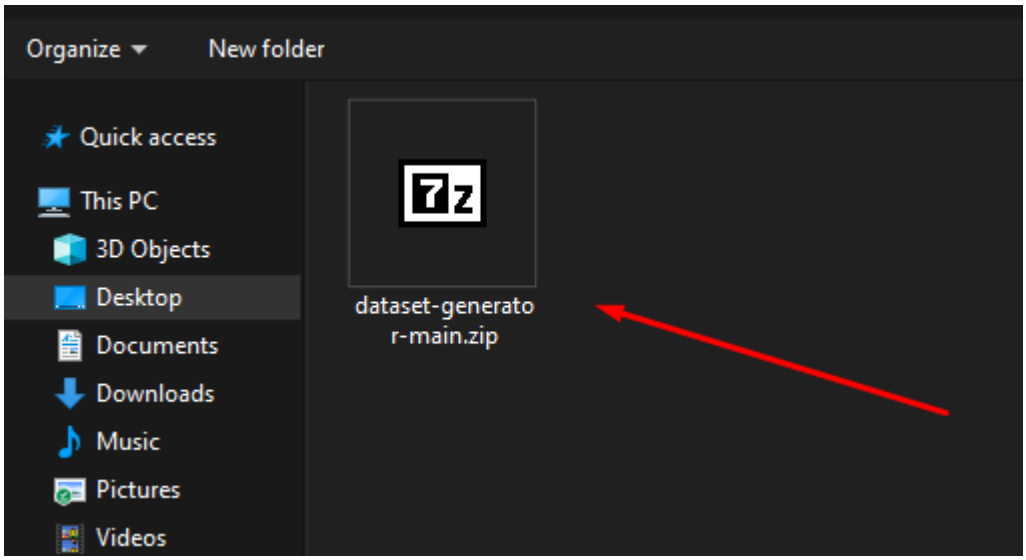


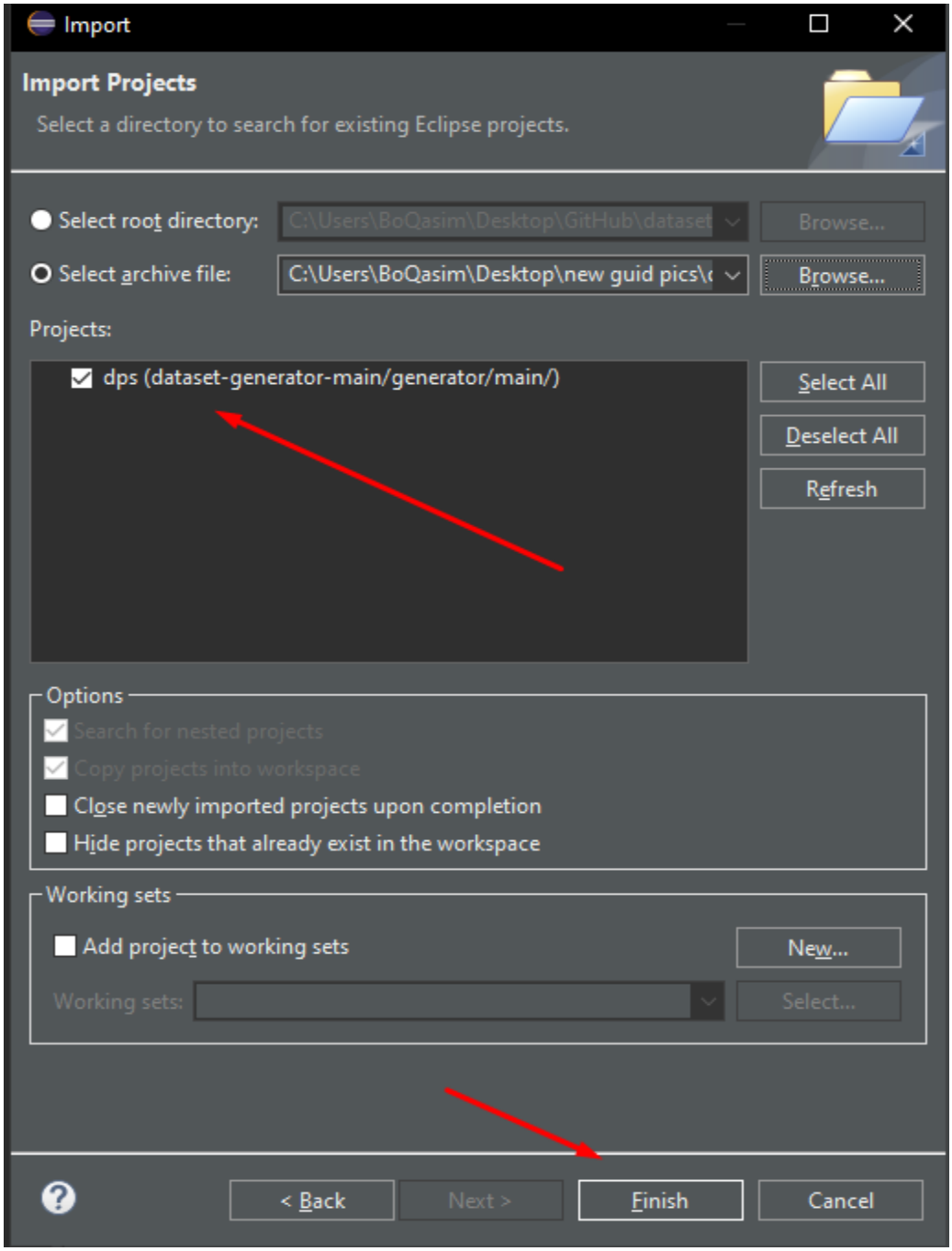
Step 2: Import the Project

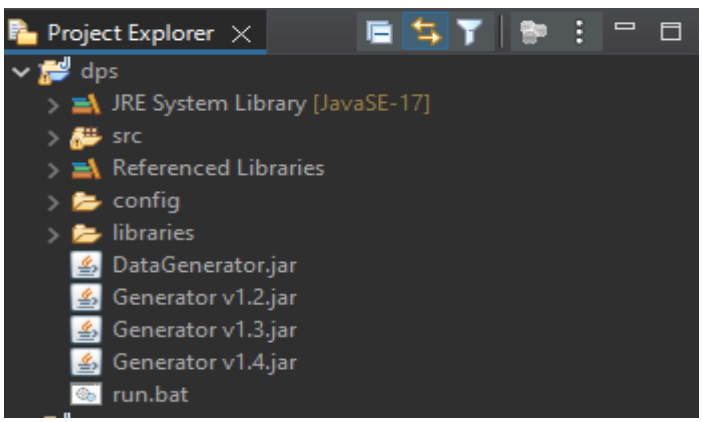
3. **Open Eclipse IDE.**
4. **Navigate to File → Import → Existing Projects into Workspace**
5. **Select "Select archive file", then browse to your downloaded ZIP file (dataset-generator-main.zip).**
6. **Eclipse will detect the project automatically (for example, dps). Check the box beside it and click Finish.**
7. 📸 ***Example screenshot:***




Import
Import Projects
Select a directory to search for existing Eclipse projects.
Select root directory:
C:\Users\BoQasim\Desktop\GitHub\dataset
Browse...
Select archive file:
Browse...
Projects:
Select All
Deselect All
Refresh
Options
Search for nested projects
Copy projects into workspace
Close newly imported projects upon completion
Hide projects that already exist in the workspace
Working sets
Add project to working sets
New...
Working sets:
Select...
< Back
Next >
Finish
Cancel


Organize
New folder
Quick access
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
dataset-generato
r-main.zip

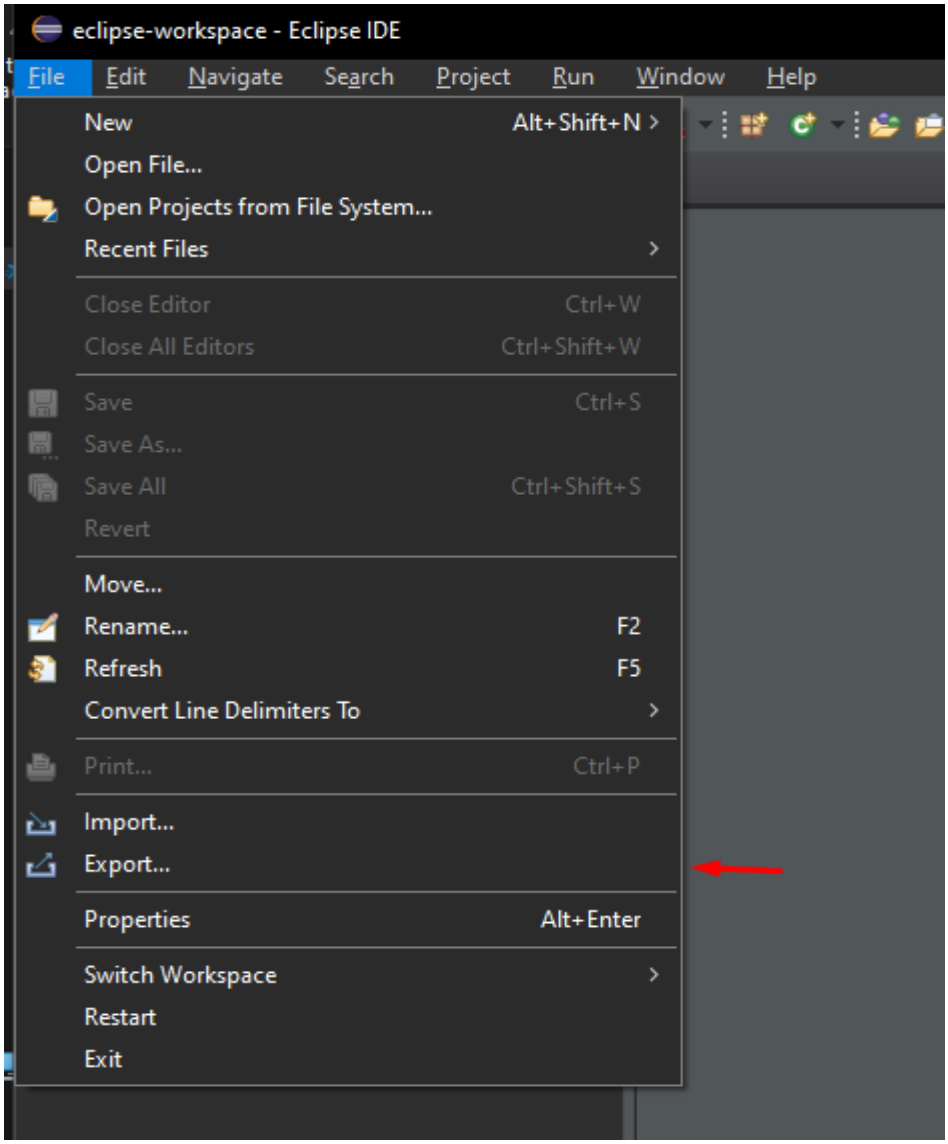

Import
Import Projects
Select a directory to search for existing Eclipse projects.
Select root directory:
C:\Users\BoQasim\Desktop\GitHub\dataset
Browse...
Select archive file:
C:\Users\BoQasim\Desktop\new guid pics\
Browse...
Projects:
dps (dataset-generator-main/generator/main/)
Select All
Deselect All
Refresh
Options
Search for nested projects
Copy projects into workspace
Close newly imported projects upon completion
Hide projects that already exist in the workspace
Working sets
Add project to working sets
New...
Working sets:
Select...
< Back
Next >
Finish
Cancel

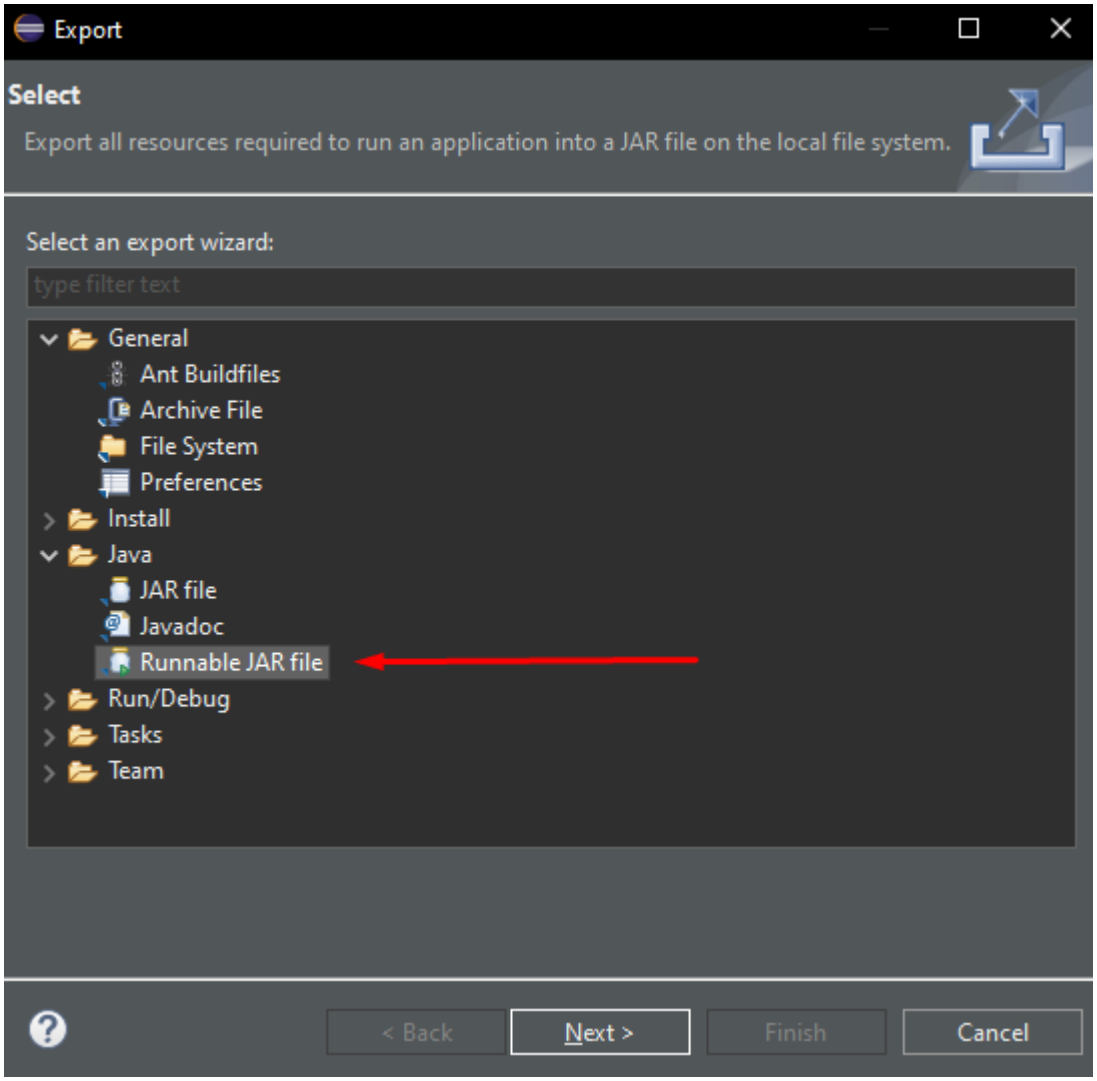

Project Explorer
dps
JRE System Library [JavaSE-17]
src
Referenced Libraries
config
libraries
DataGenerator.jar
Generator v1.2.jar
Generator v1.3.jar
Generator v1.4.jar
run.bat

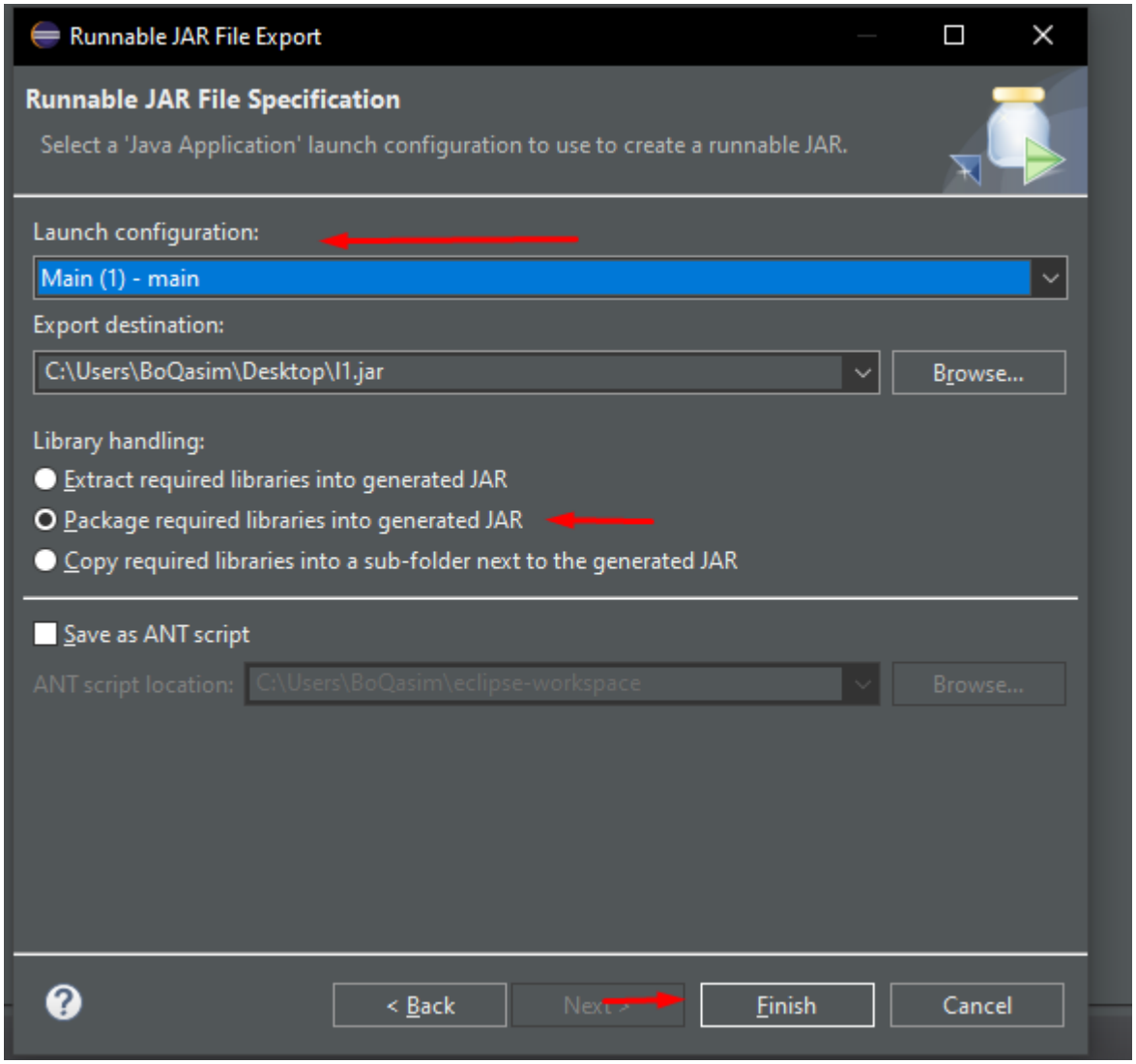
Step 4: Export the Project as a Runnable (Fat) JAR

1. **Go to File → Export.**
2. **Under Java, select Runnable JAR file and click Next.**
3. **Configure the export options as follows:**
    - **Launch configuration: select your project's main class.**
    - **Export destination: choose the location where the JAR file will be saved.**
    - **Library handling: select**
      👉 ***"Package required libraries into generated JAR."***
4. **Click Finish to create the JAR file.**

📸 ***Example screenshot:***



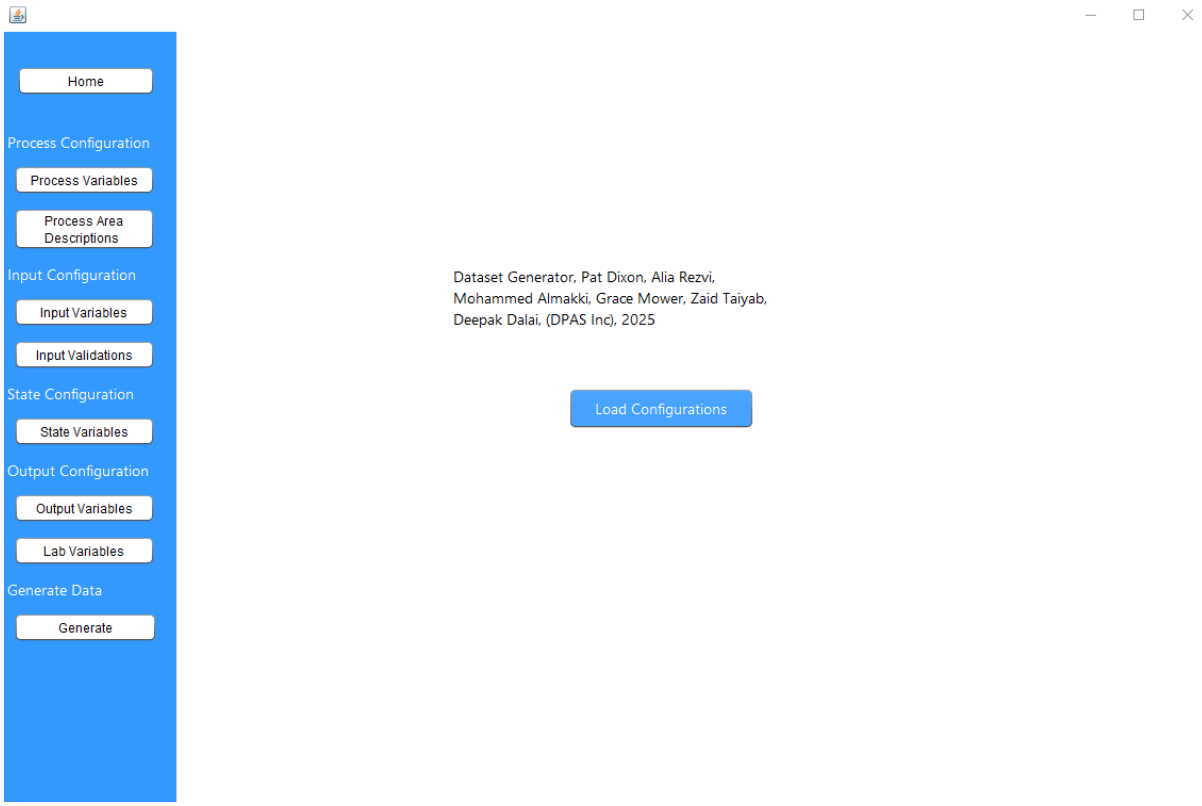

Export
Select
Export all resources required to run an application into a JAR file on the local file system.
Select an export wizard:
type filter text
General
Ant Buildfiles
Archive File
File System
Preferences
Install
Java
JAR file
Javadoc
Runnable JAR file
Run/Debug
Tasks
Team
< Back
Next >
Finish
Cancel



Step 4: Verify the Generated JAR

1. **Open the folder where the exported .jar file was saved.**
2. **Double-click the file to run it, or use a terminal:** `java -jar filename.jar`
3. **Confirm that your program launches successfully and works as expected.**

Example screenshot:



Troubleshooting:

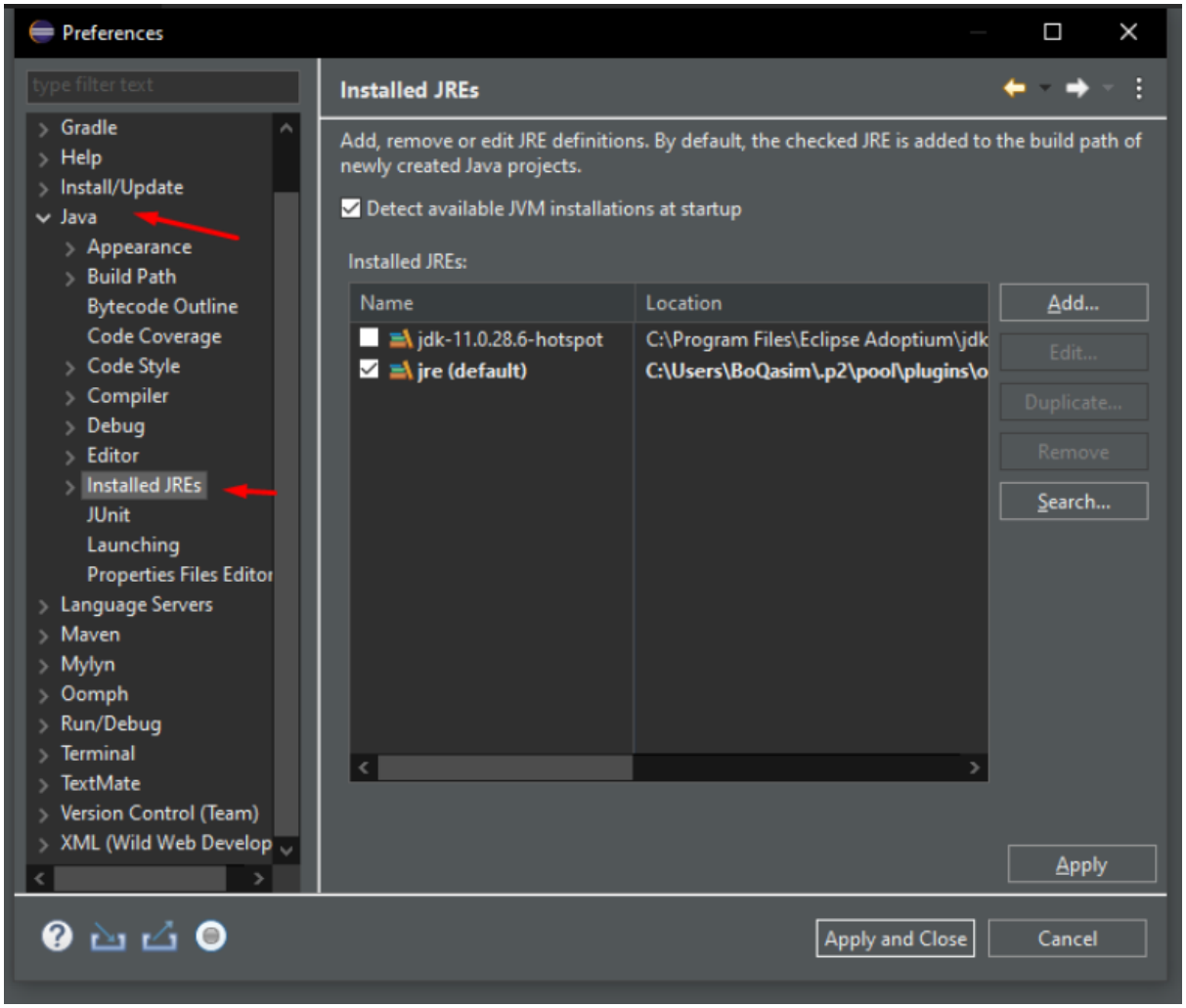
Unsupported major.minor version 65.0

```
Error occurred during initialization of boot layer
java.lang.module.FindException: Error reading module: /Users/patrick/eclipse-workspace/dps/bin
Caused by: java.lang.module.InvalidModuleDescriptorException: Unsupported major.minor version 65.0
```

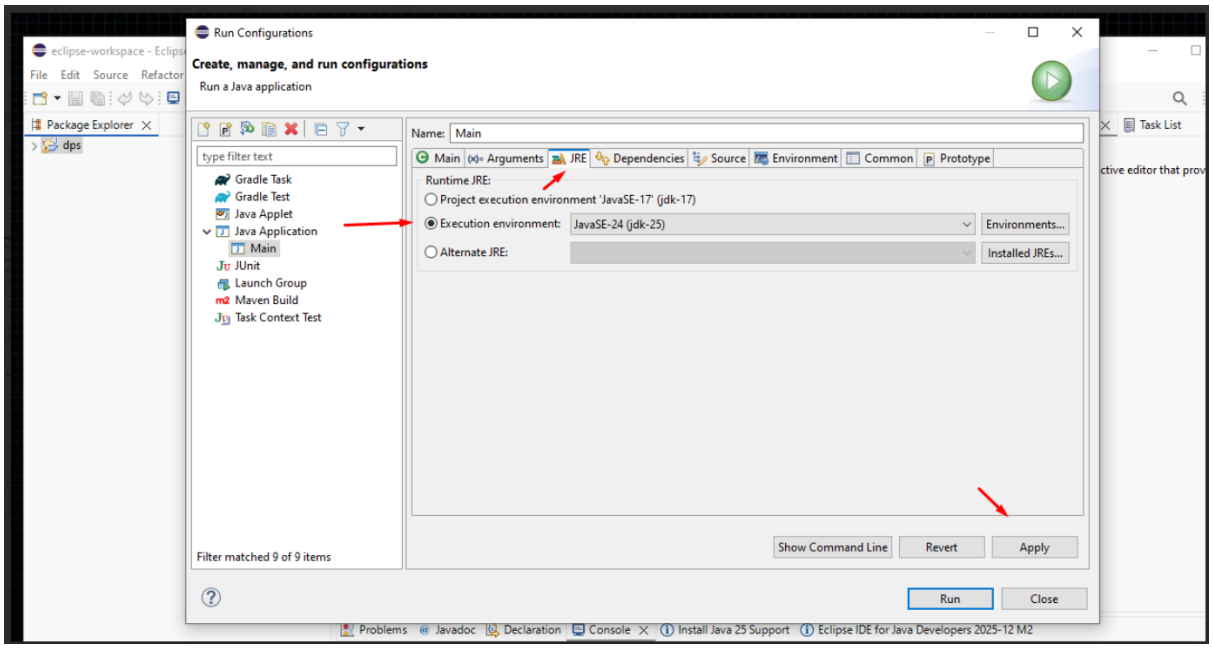
This error happens when the project is compiled with a newer Java version (e.g., Java 21 or 25) but Eclipse tries to run it using an older version (e.g., Java 17).

How to Fix

1. **In Eclipse, go to:**
   Window → Preferences → Java → Installed JREs
2. **Select the newest JDK (Java 21 or Java 25).**
   If it's not listed, click Add... → Standard VM and add it manually.
3. **Set it as Default.**



4. **Open your project → Run As → Run Configuration → Java Application**
   Make sure the compiler version matches the installed JDK.



5. **Re-run the project.**
6. **Export again if needed.**

Why This Happens

- **Newer Java → creates class version 65 (Java 21)**
- **Older Java → cannot run newer class versions**
- **Result → Unsupported major.minor version 65.0**