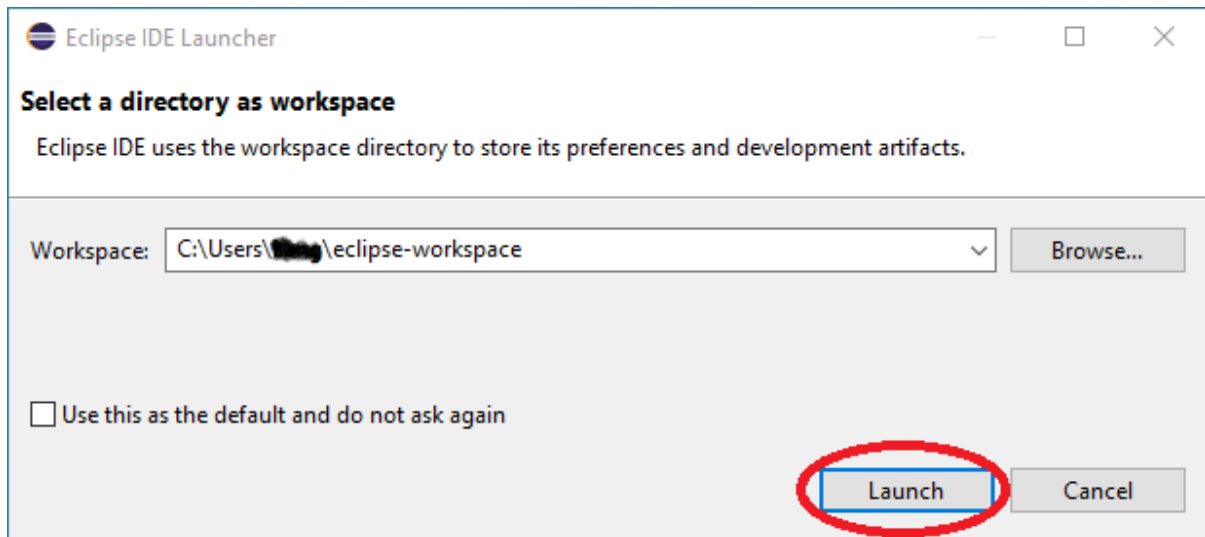
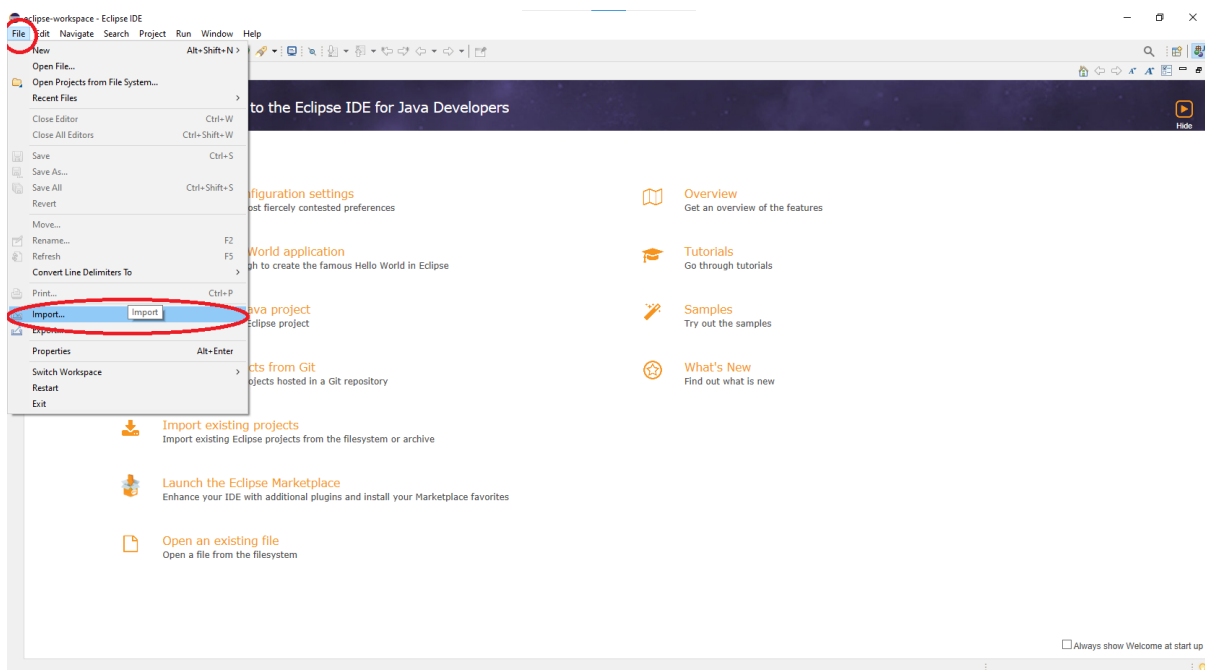


How to import GymBro to Eclipse IDE

1. Open eclipse.

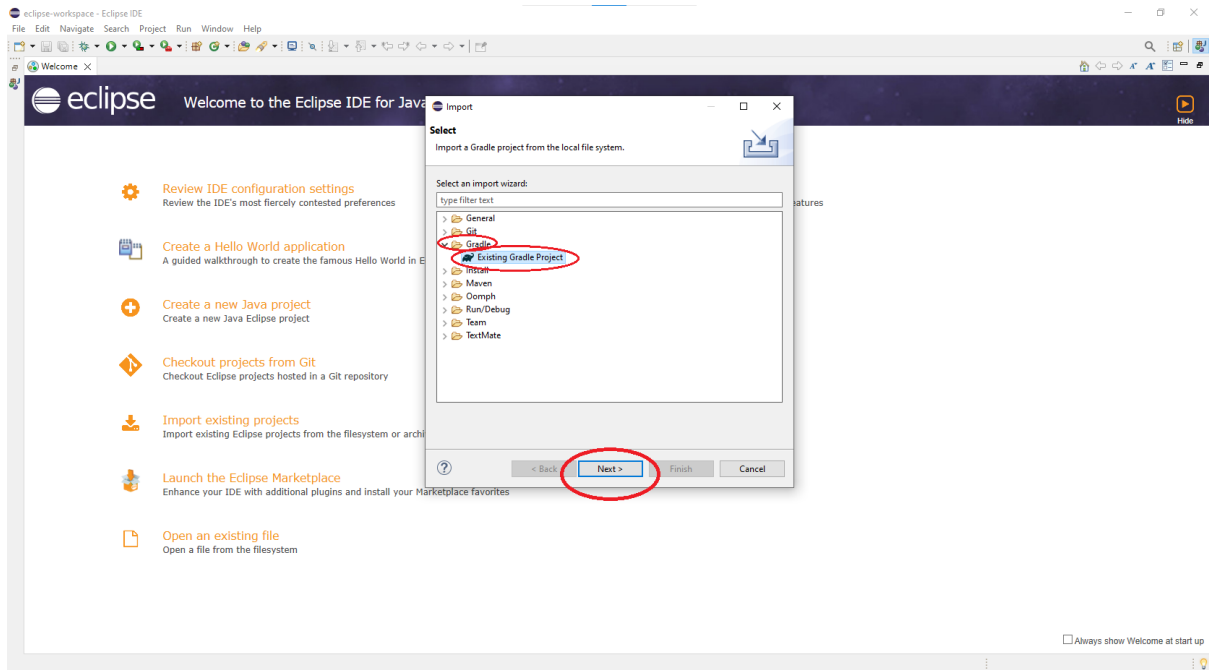


2. Click on File and Import.

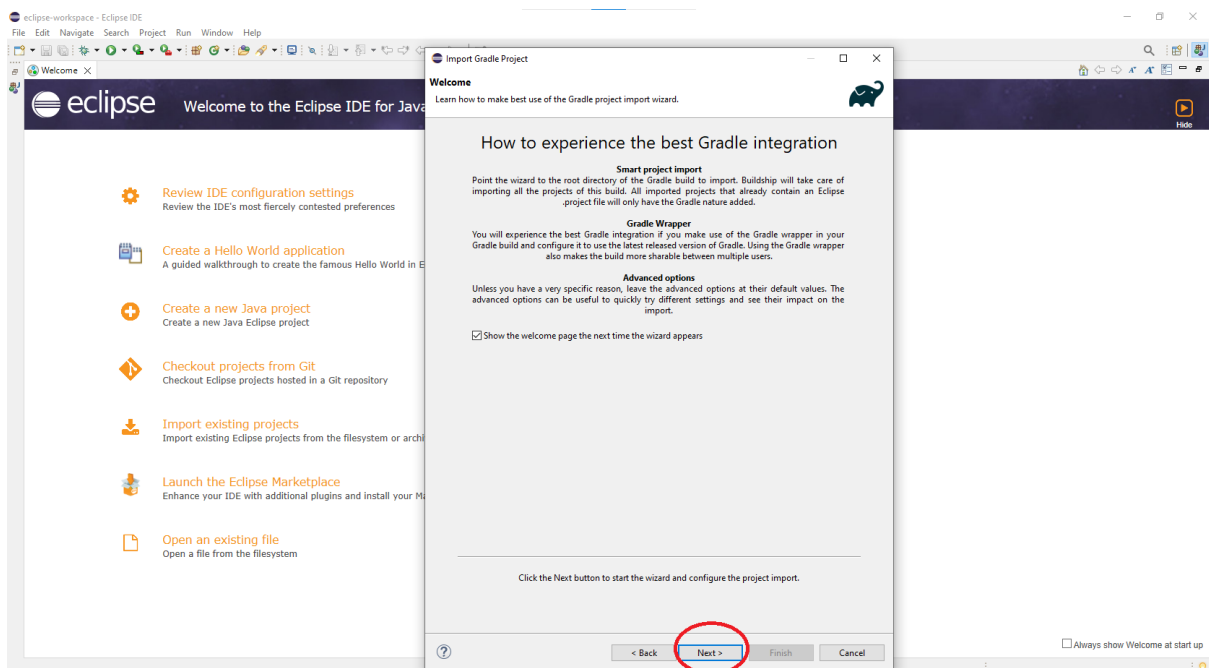


How to import GymBro to Eclipse IDE

3. Select Gradle. Then select Existing Gradle Project. After that click next.

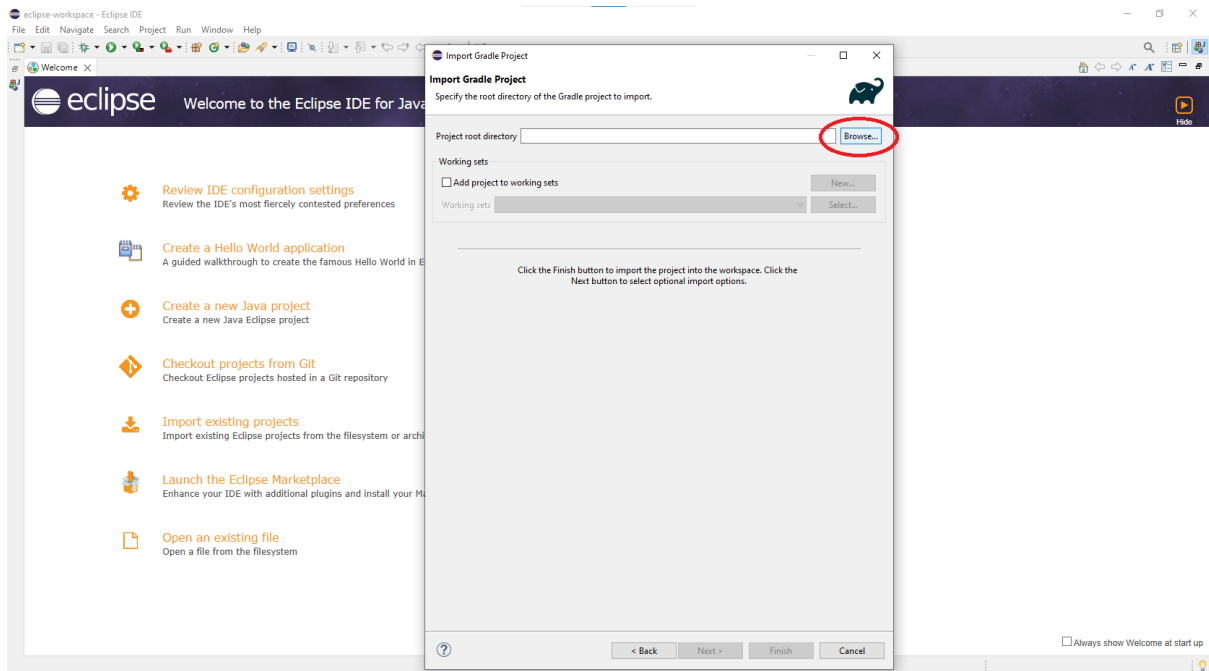


4. Click next.

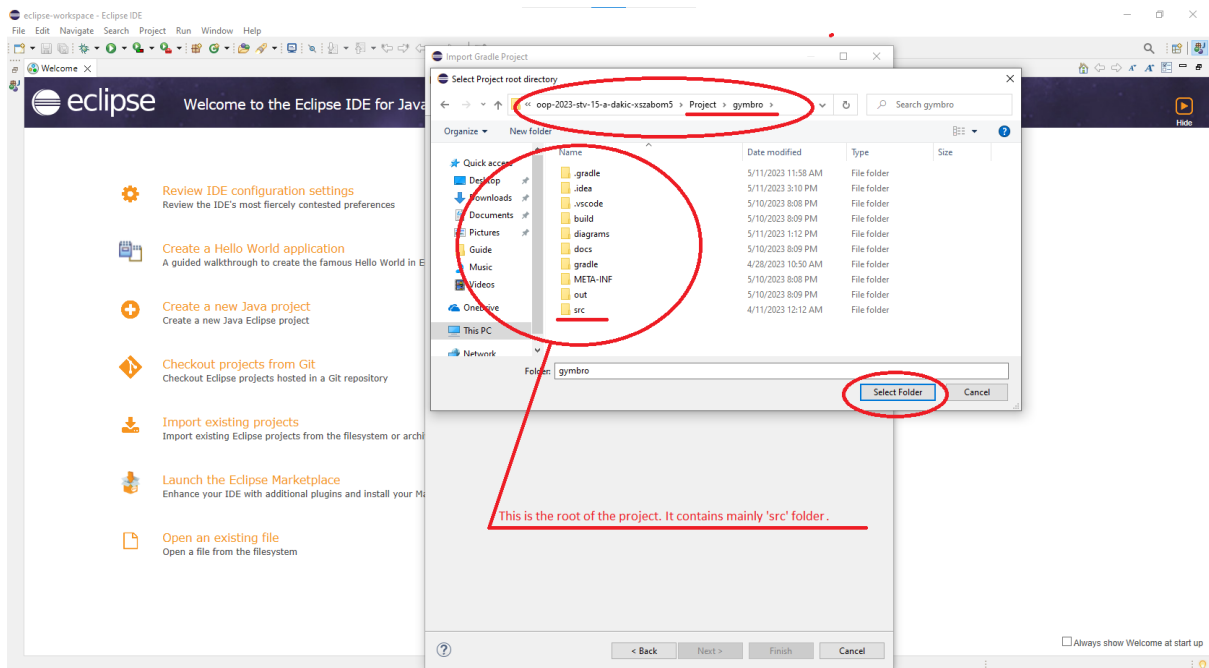


How to import GymBro to Eclipse IDE

5. Select project root directory.

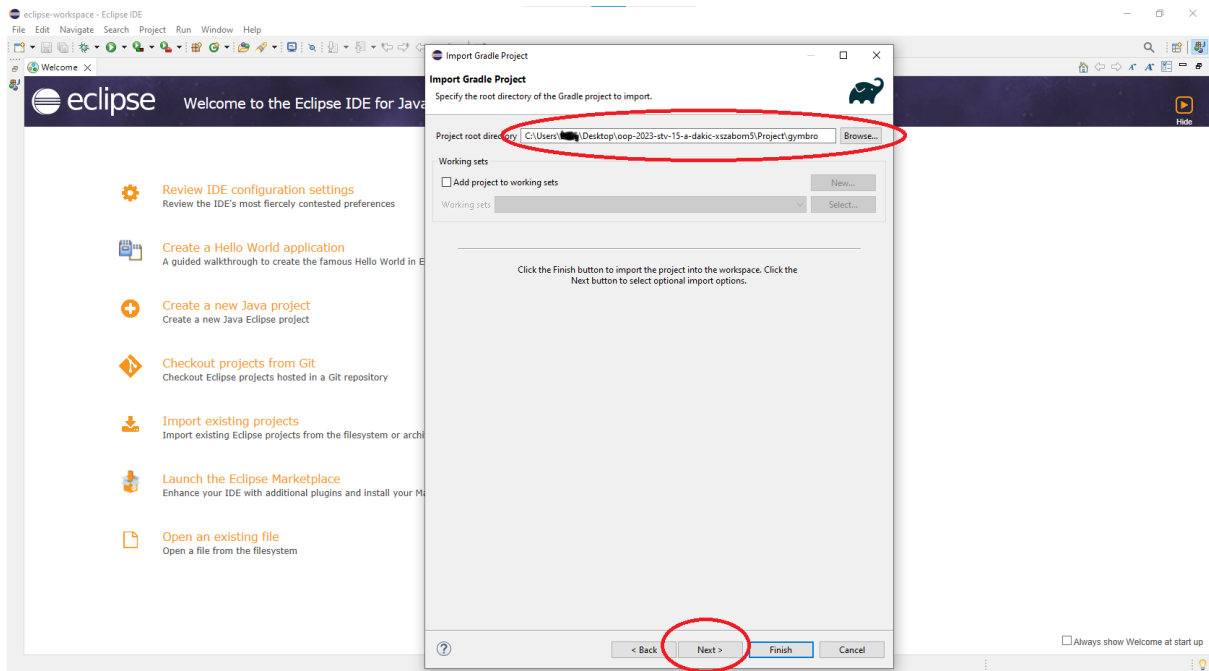


6. Select project root directory.

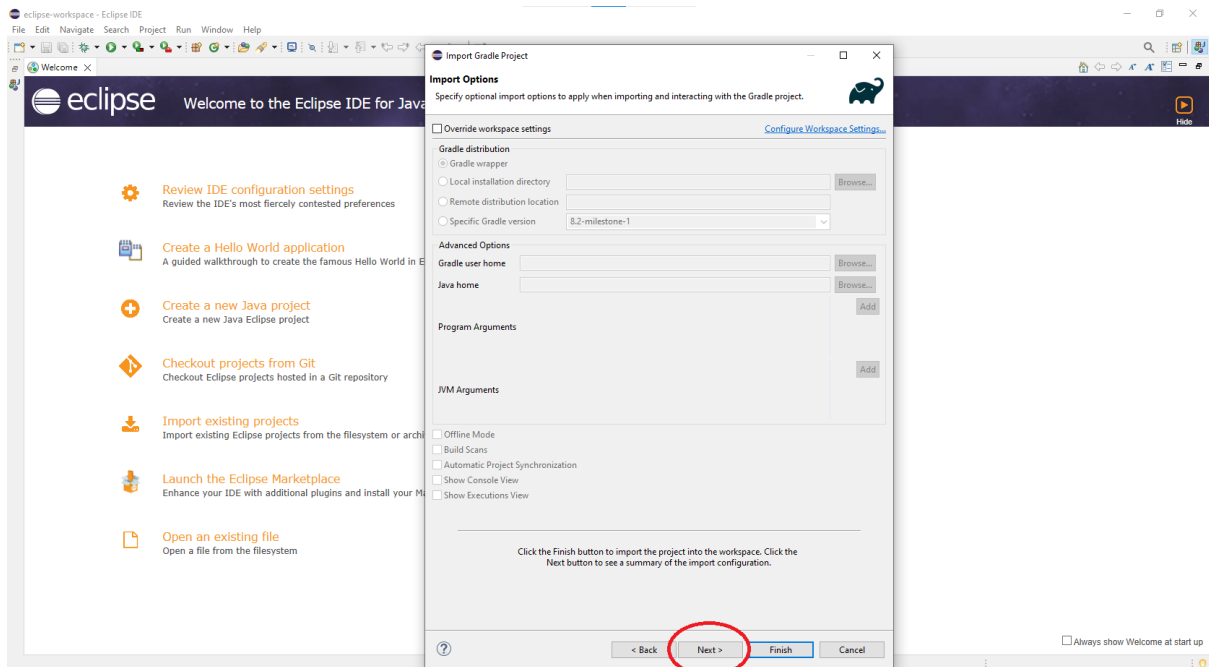


How to import GymBro to Eclipse IDE

7. Select project root directory. Then click next.

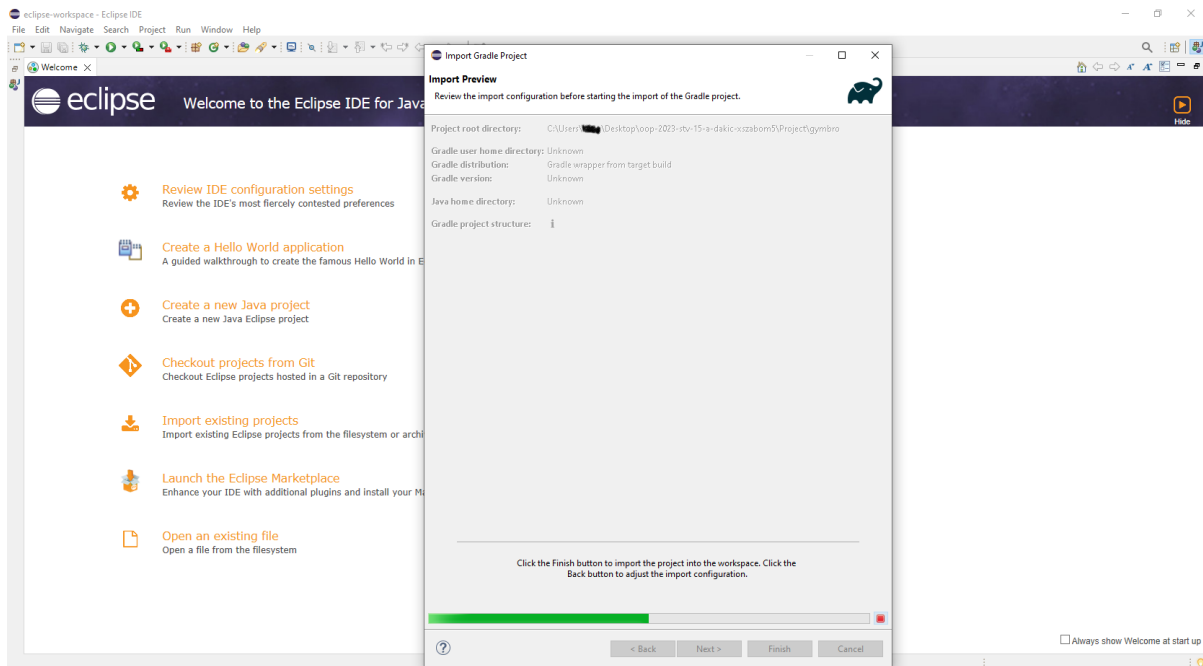


8. Click next.

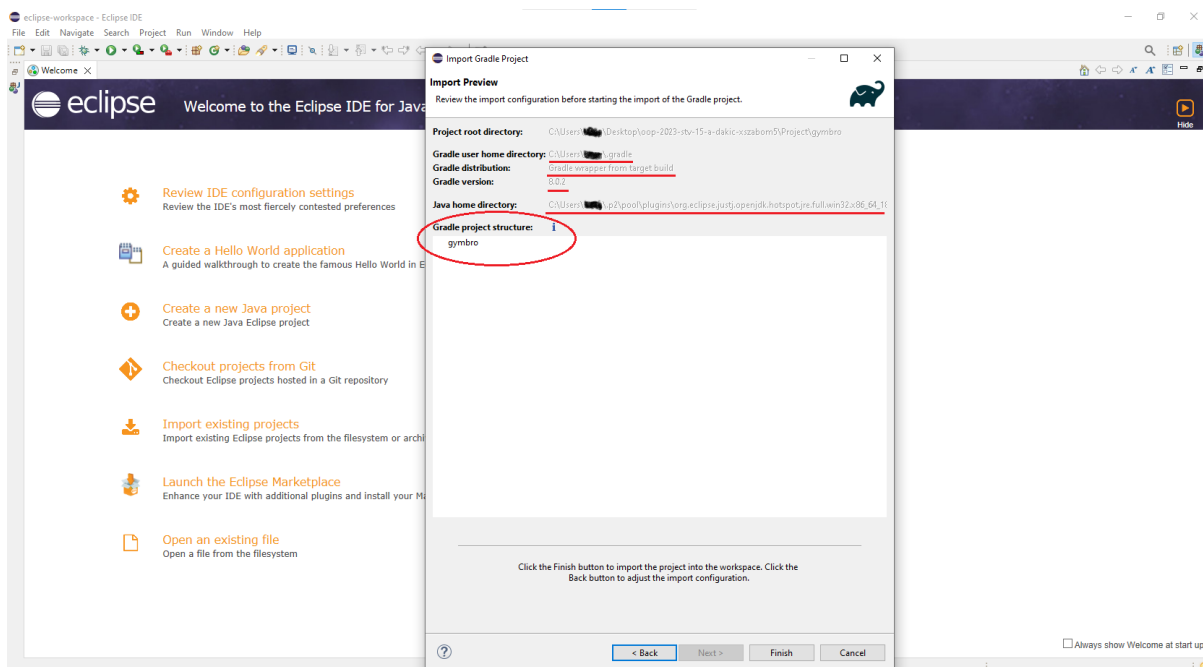


How to import GymBro to Eclipse IDE

9. Wait for eclipse to detect Gradle build system.

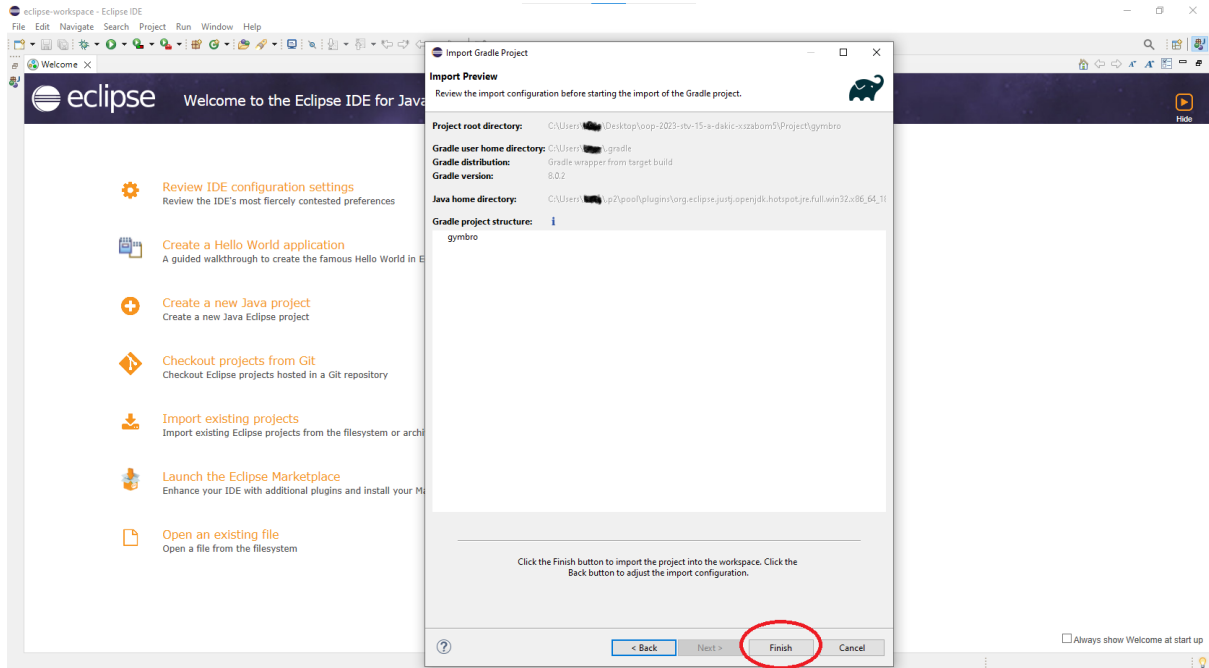


10. Eclipse has successfully detected Gradle.

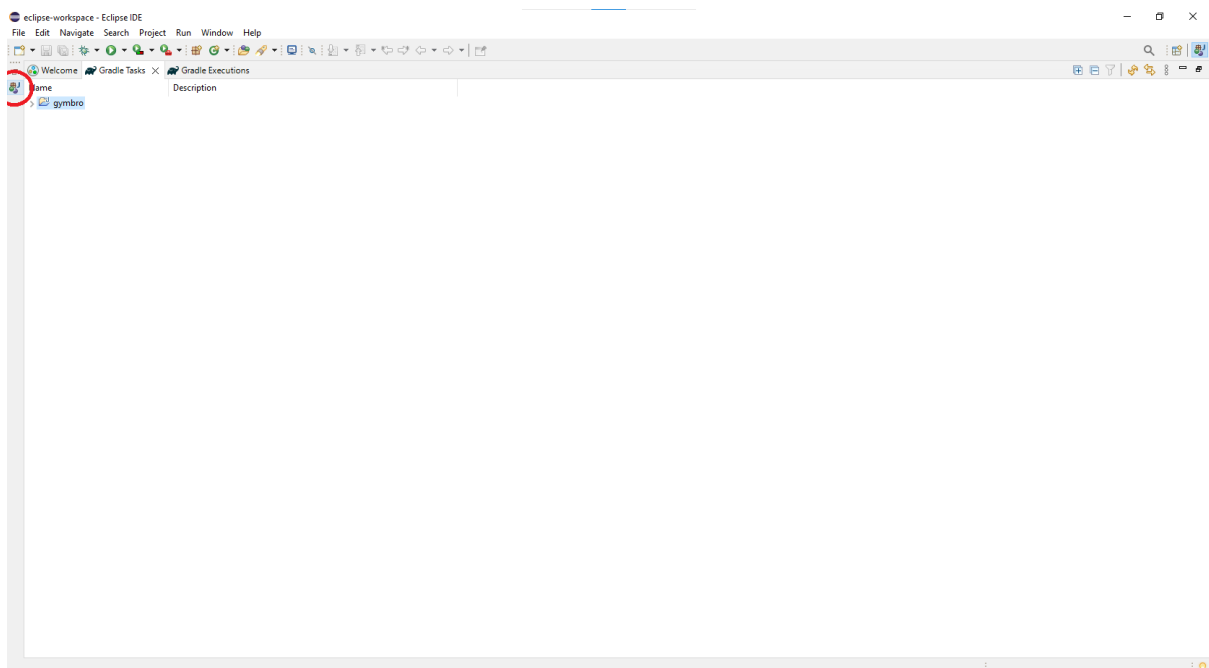


How to import GymBro to Eclipse IDE

11. Click finish. Then wait for eclipse to do the final configuration.

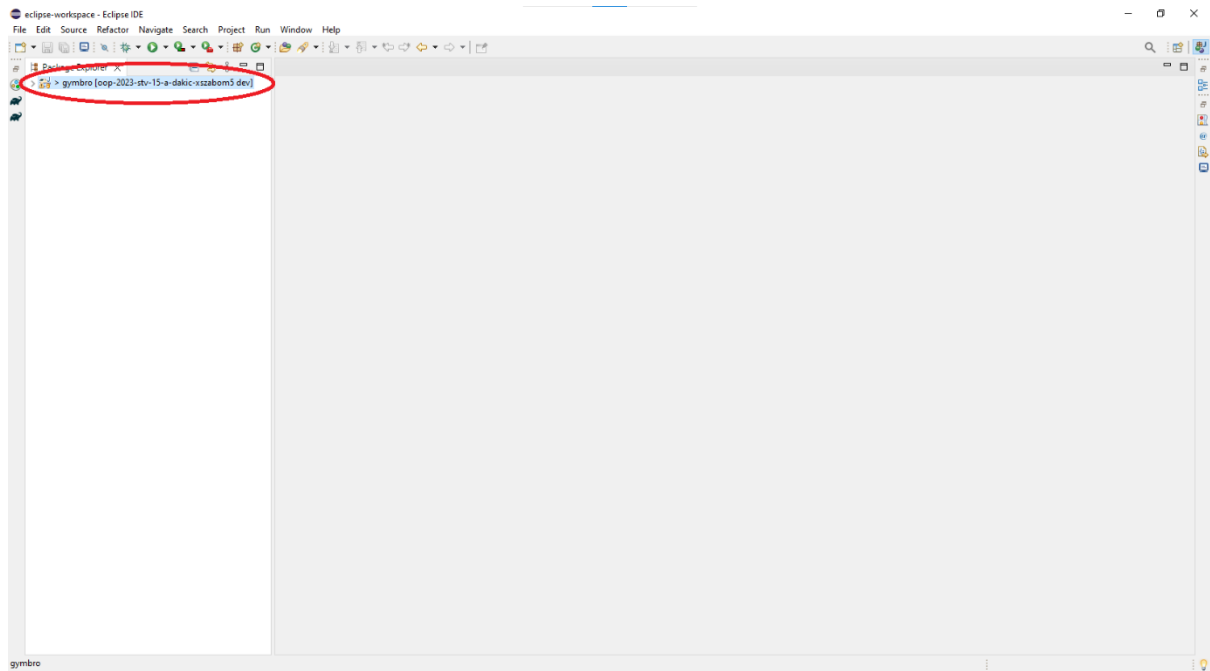


12. Click on package explorer on the left sidebar.

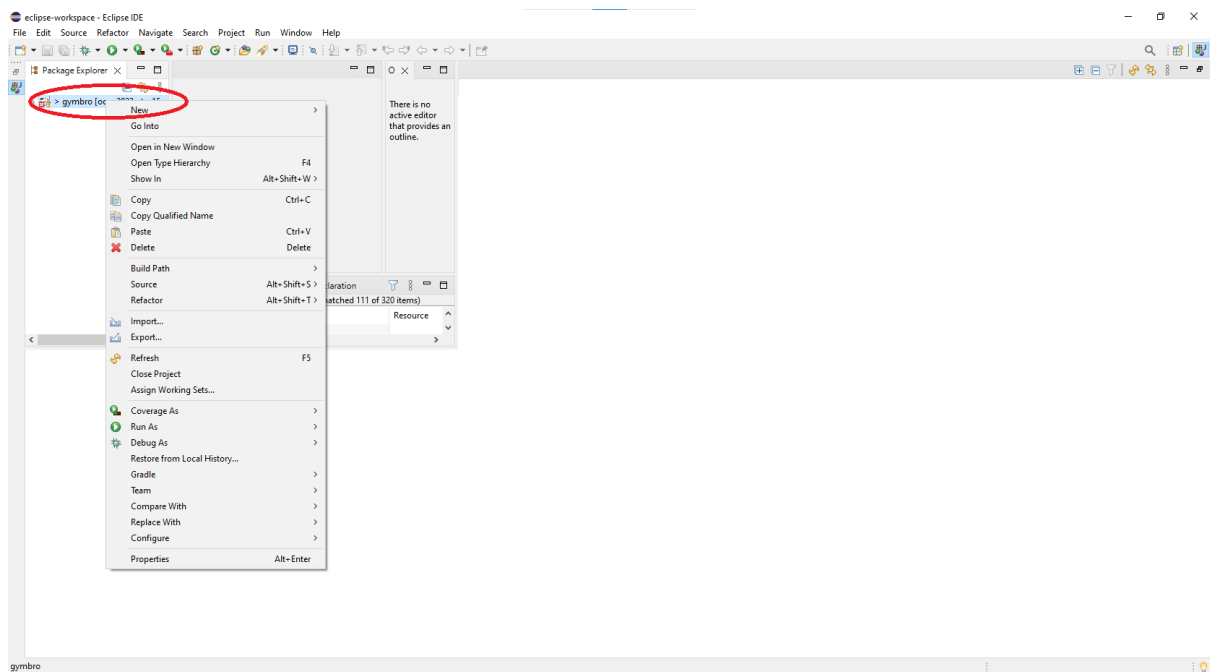


How to import GymBro to Eclipse IDE

13. Right click the project.

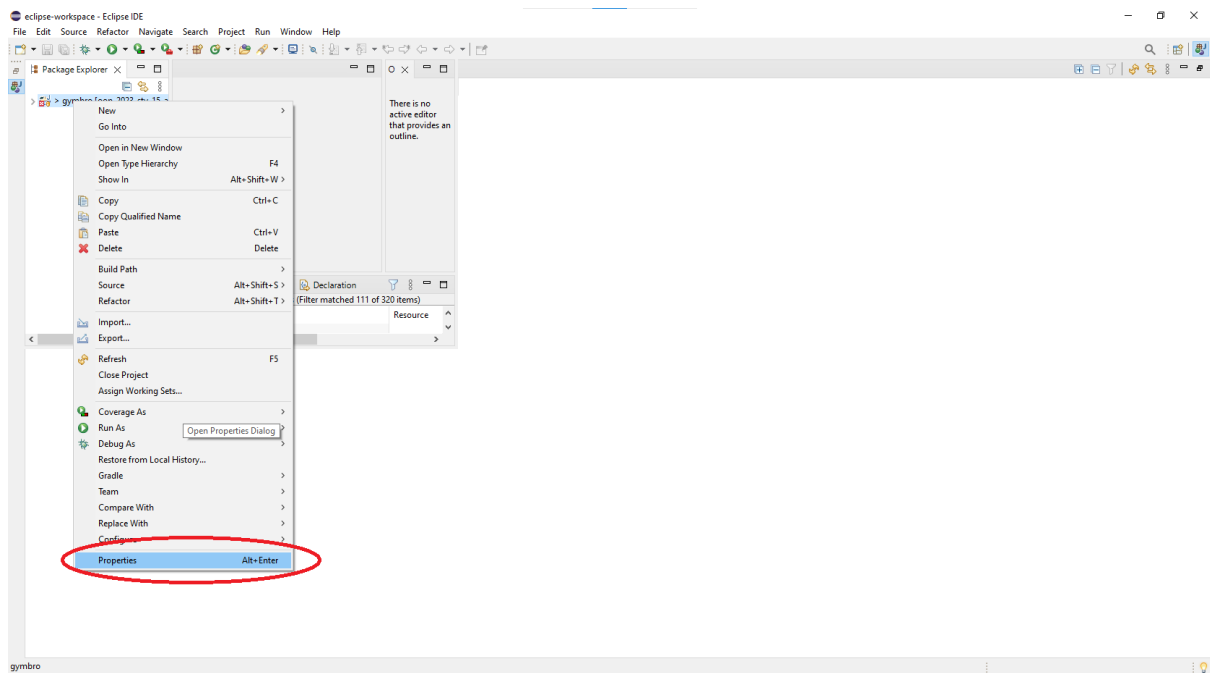


14. Right click project.

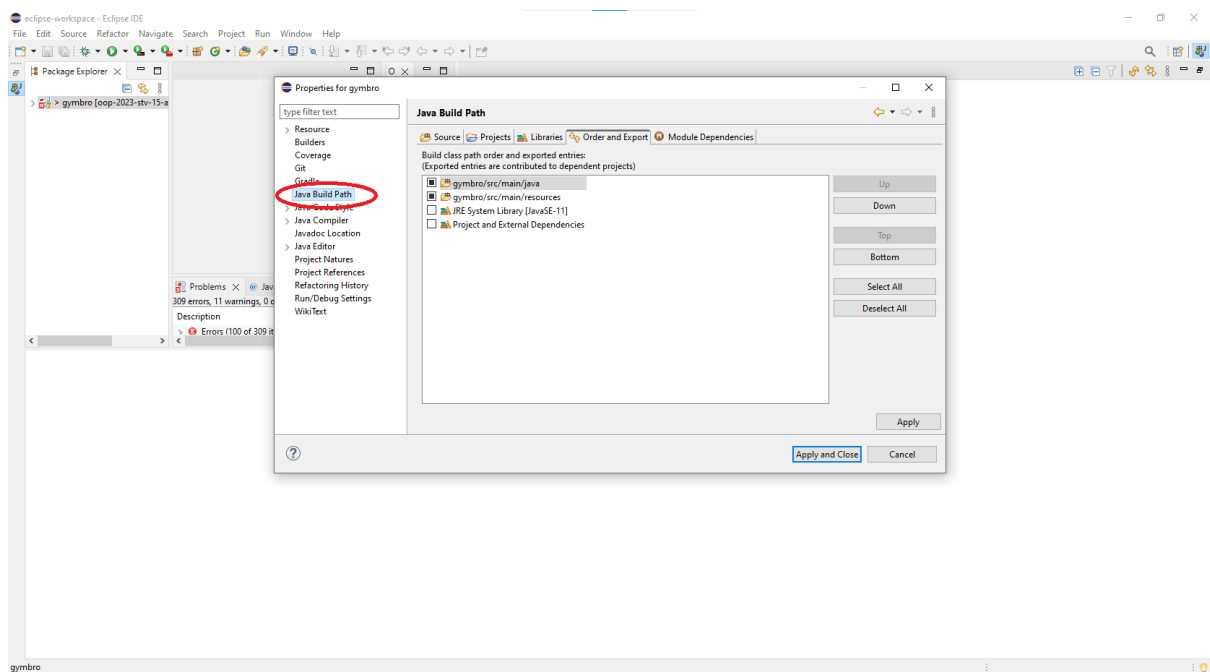


How to import GymBro to Eclipse IDE

15. Select properties.

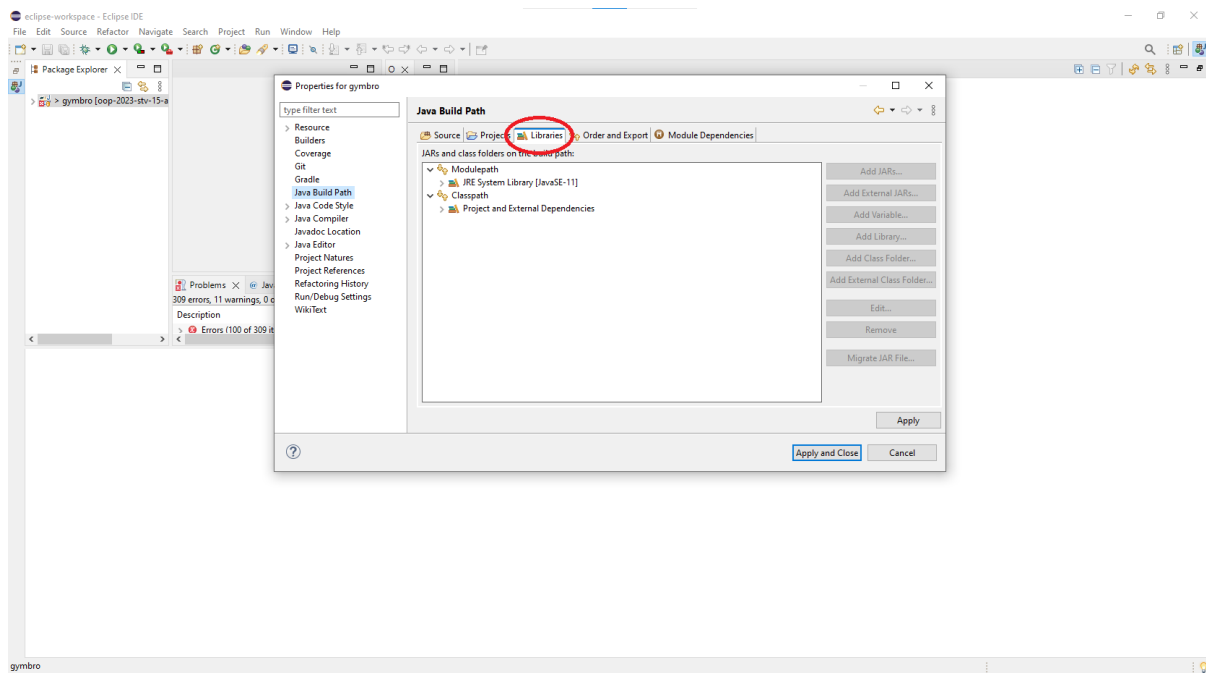


16. Select java build path.

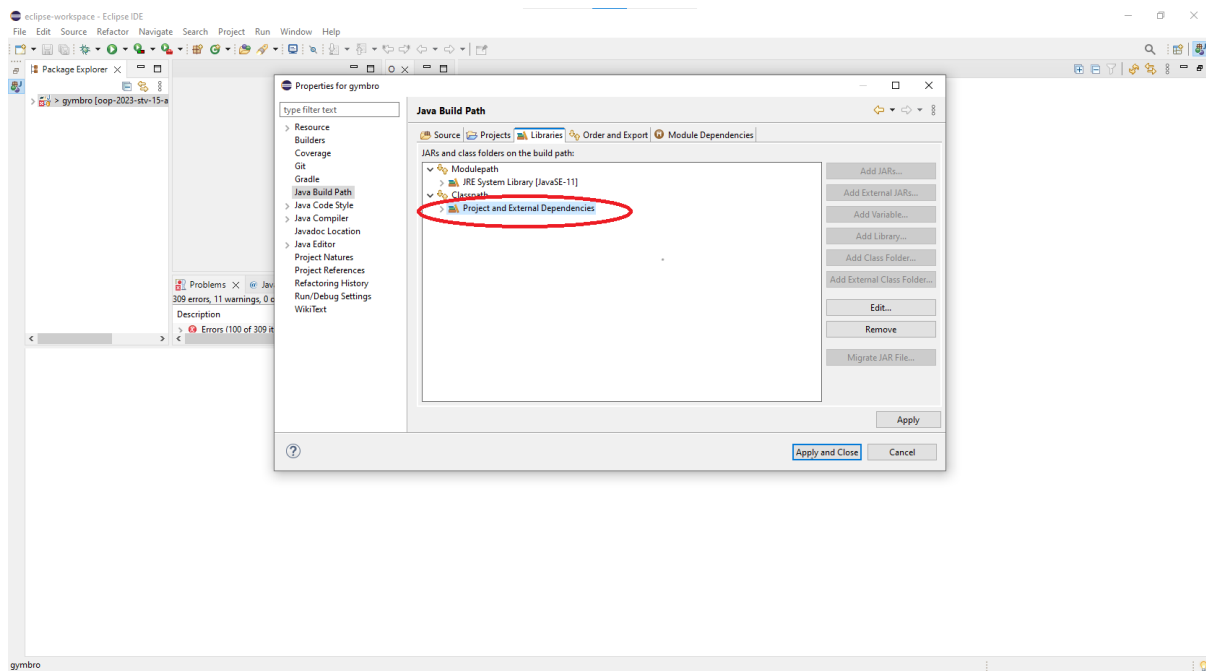


How to import GymBro to Eclipse IDE

17. Select Libraries.

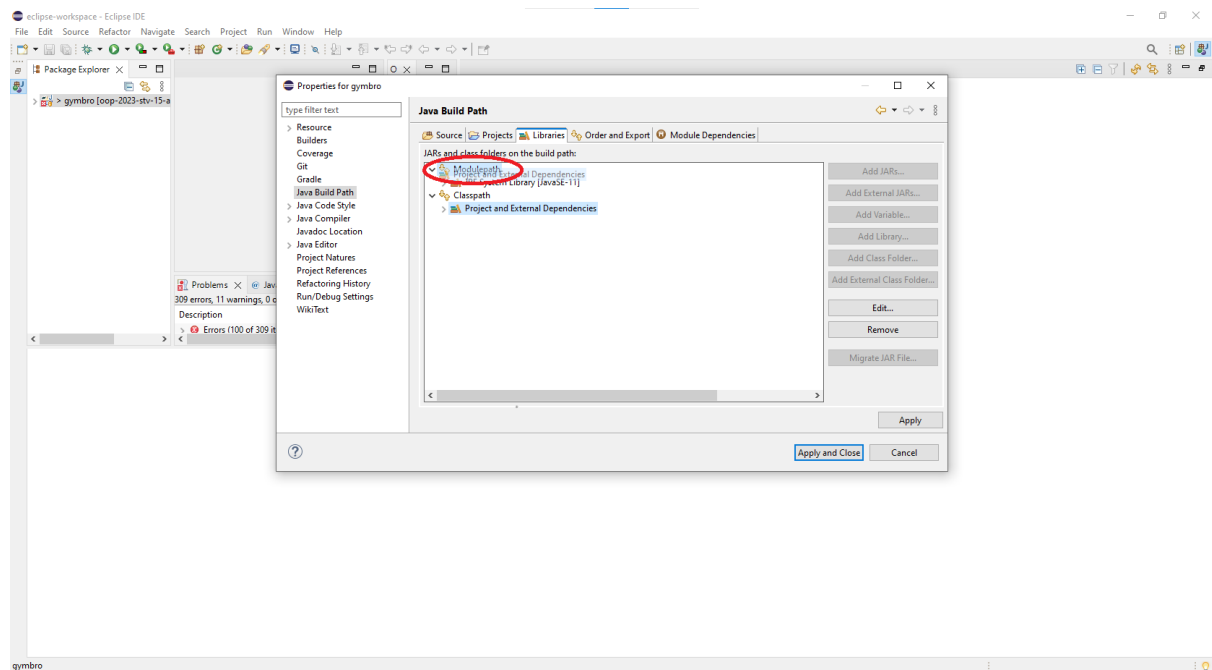


18. Grab “Project and External Dependencies” with your mouse.

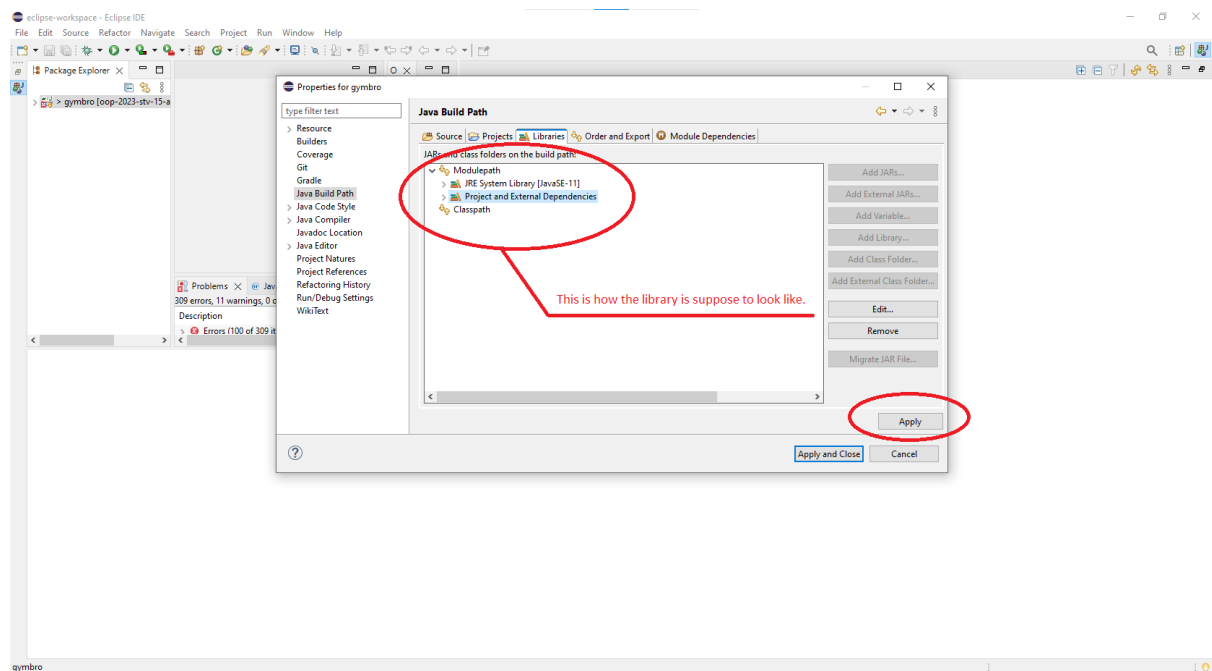


How to import GymBro to Eclipse IDE

19. Move it to “Modulepath”.

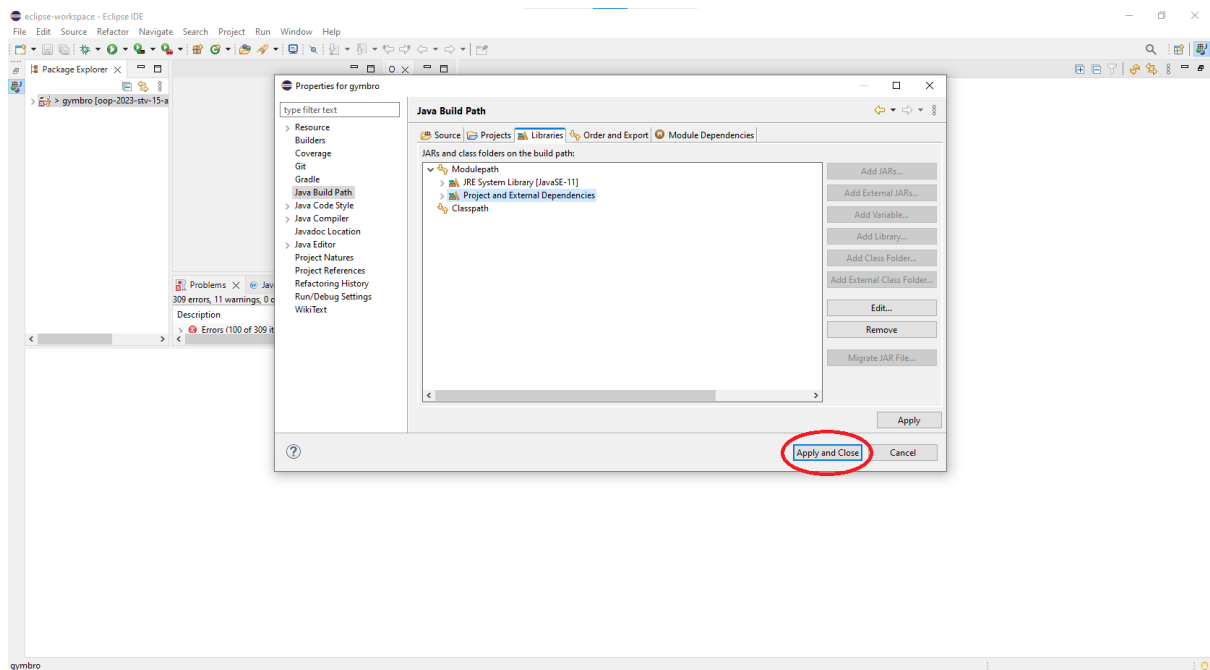


20. Click apply. (IMPORTANT)

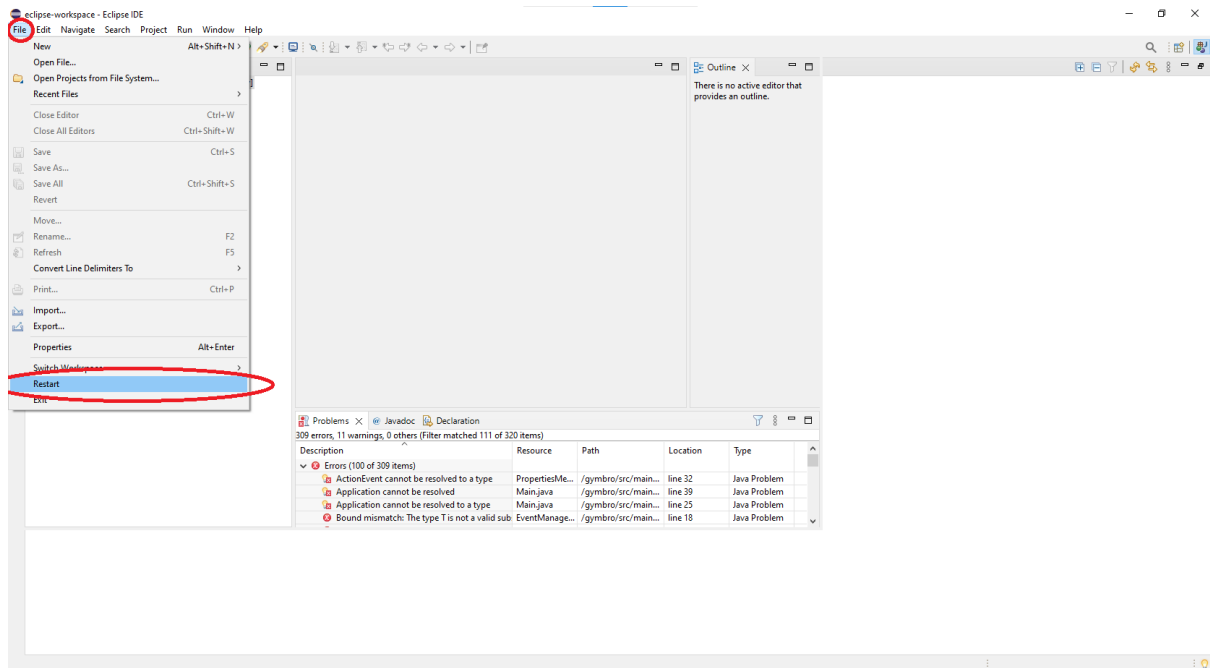


How to import GymBro to Eclipse IDE

21. Click apply and close.

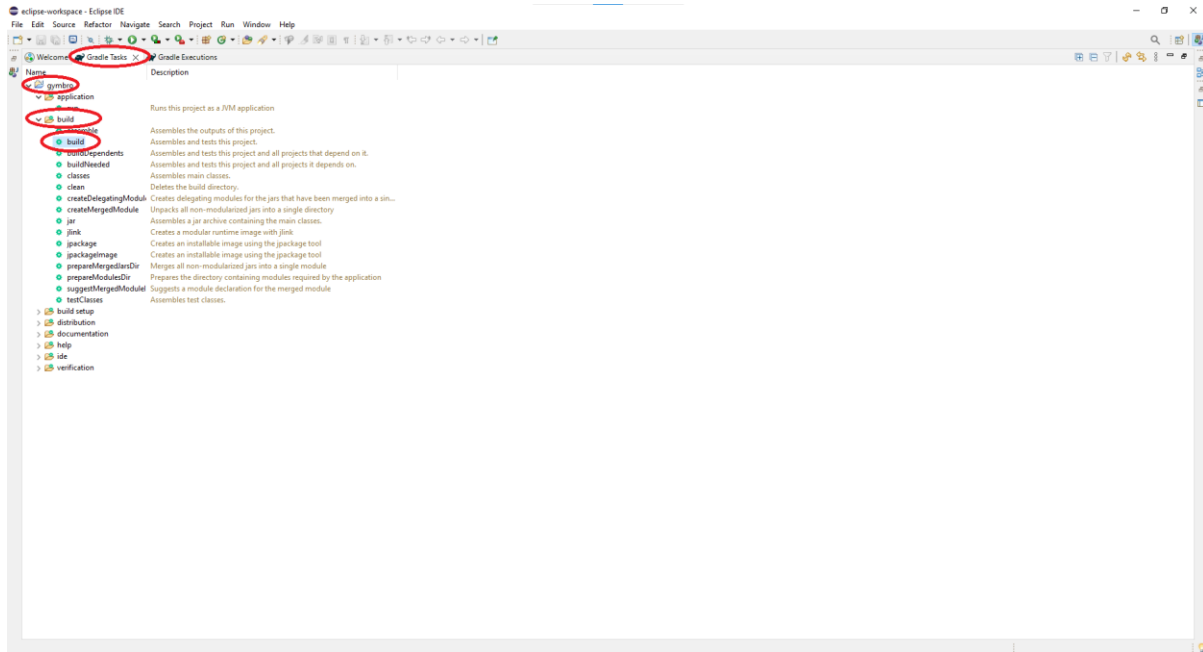


22. Restart Eclipse.

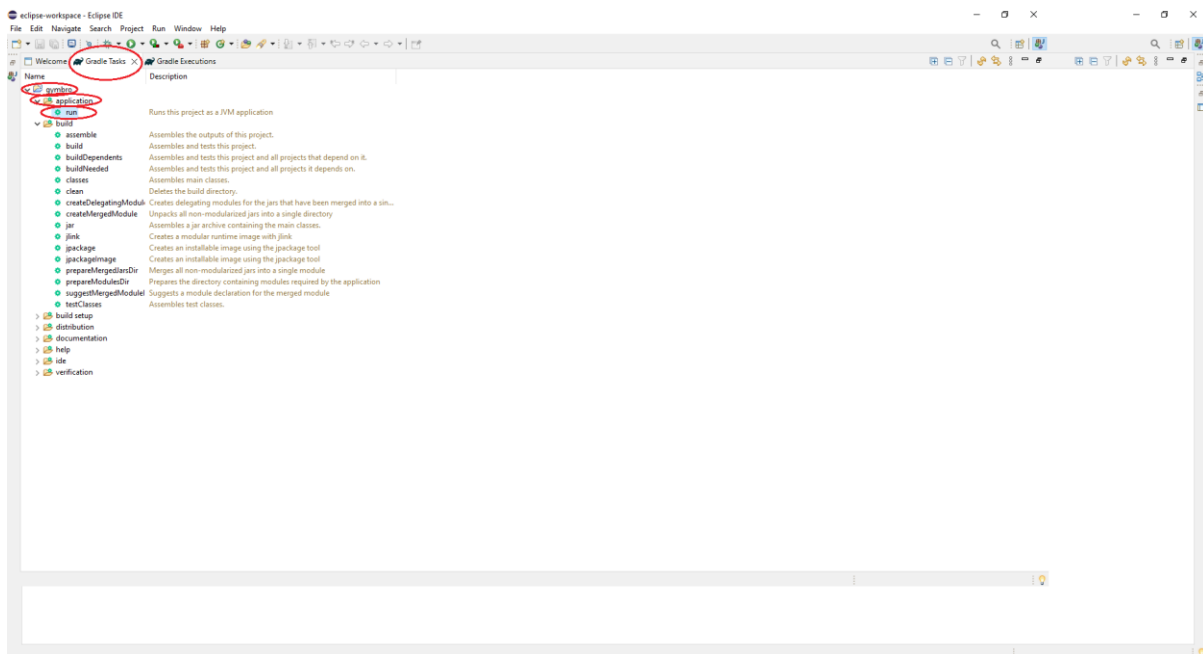


How to import GymBro to Eclipse IDE

23. After restarting run Gradle build task from Gradle Tasks tab. (Gradle Tasks → gymbro → build [folder] → build [the green icon indicates that it is a task, double click it]).

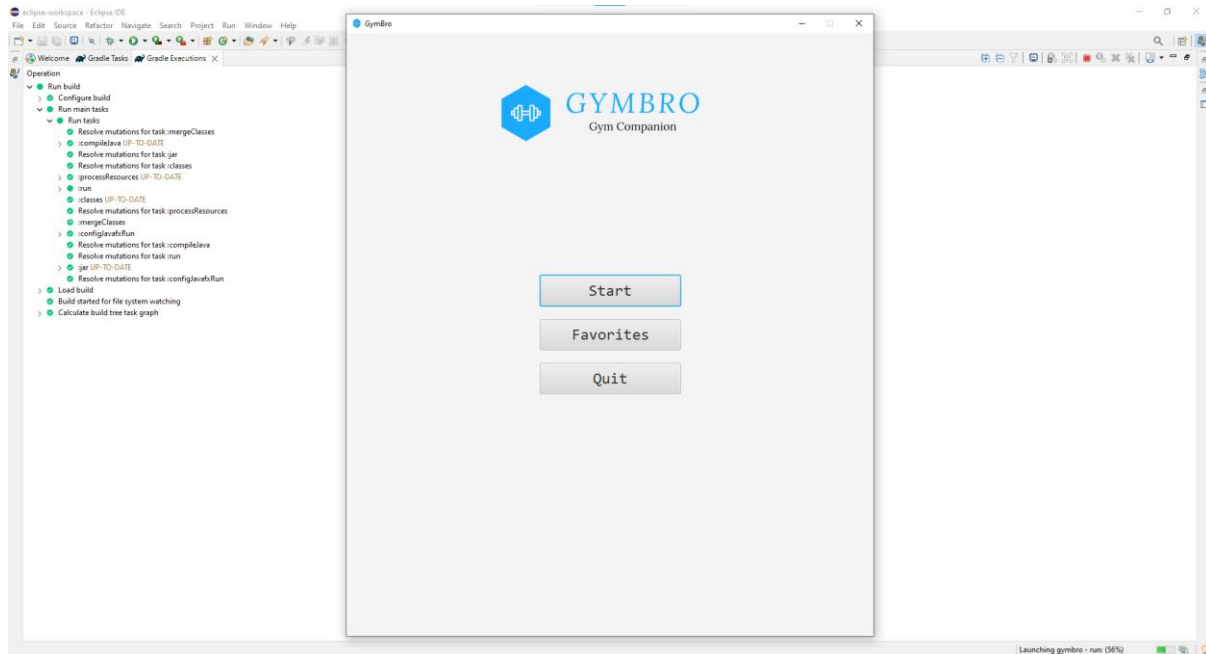


24. Then execute the run task from Gradle Tasks tab the same way. (Gradle Tasks → gymbro → application → run [double click it])

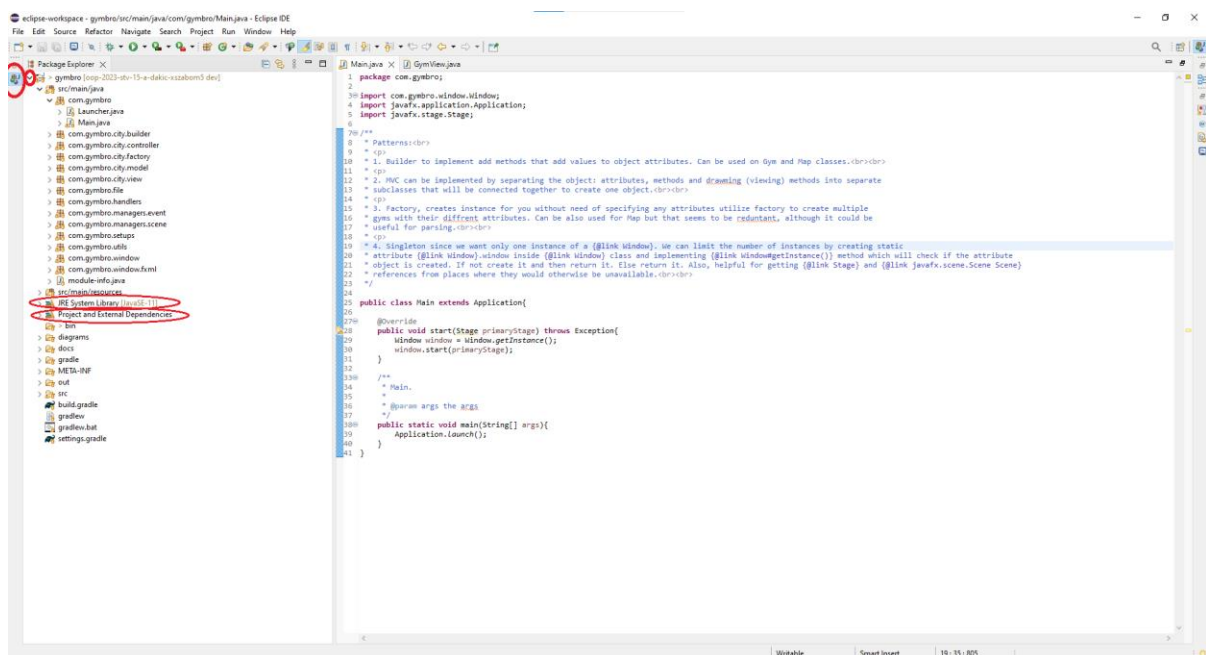


How to import GymBro to Eclipse IDE

25. The app should be running in a couple of seconds. Make sure to check taskbar if it is not running in background.

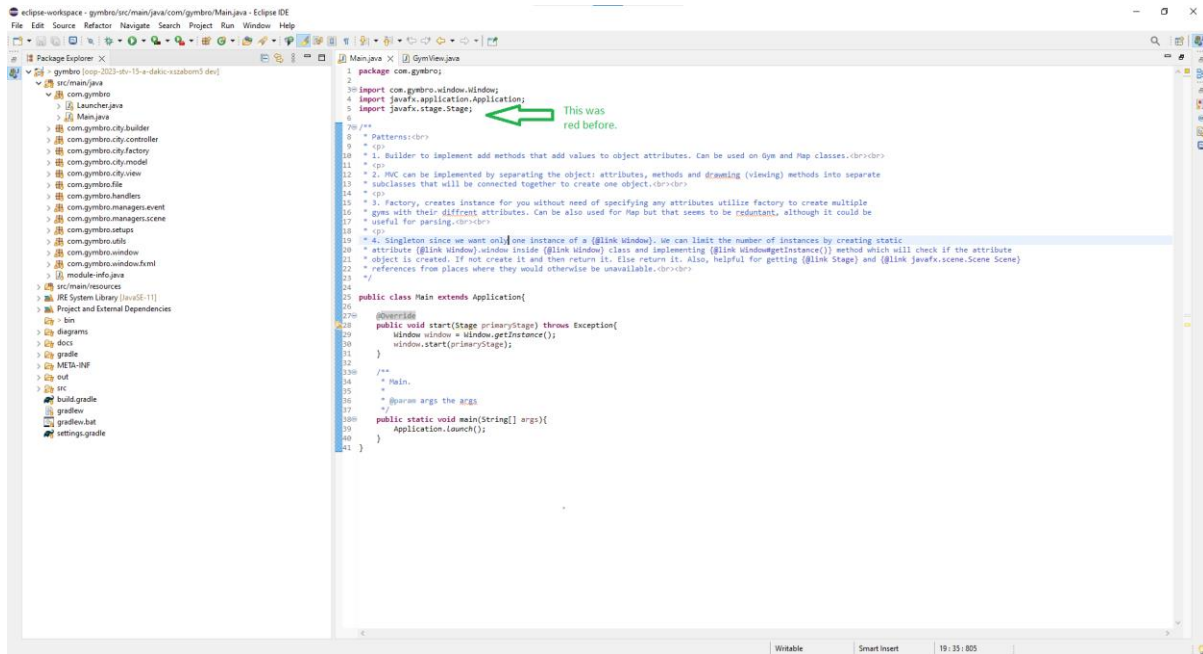


26. Make sure that in package explorer you can see “JRE System Library” and “Project and External Dependencies”. If not, then start over from step 12.

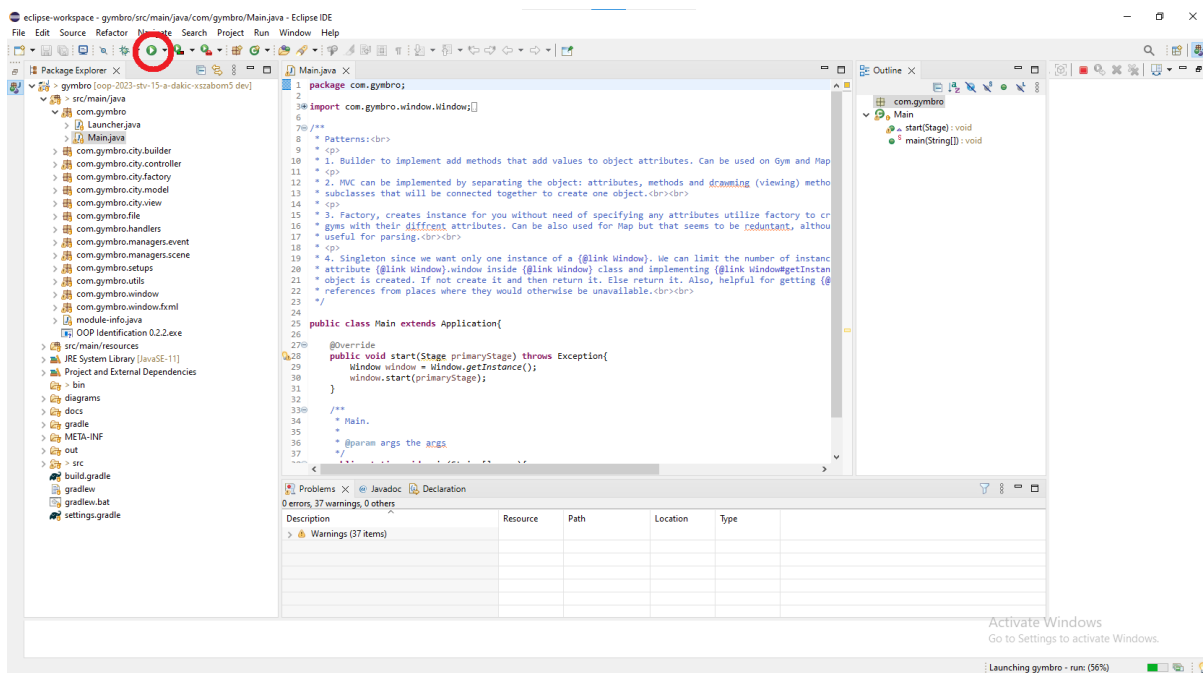


How to import GymBro to Eclipse IDE

27. If you now open the source files (Main.java for example). You won't see any JavaFX dependencies errors.

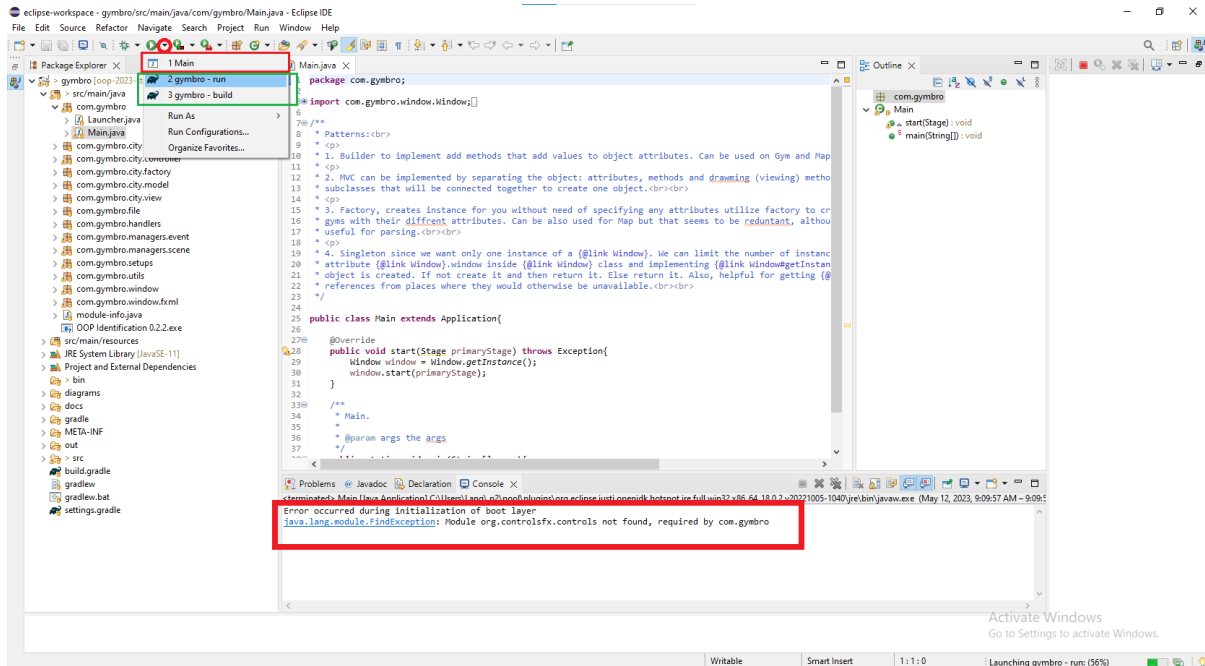


28. Click the run project button.

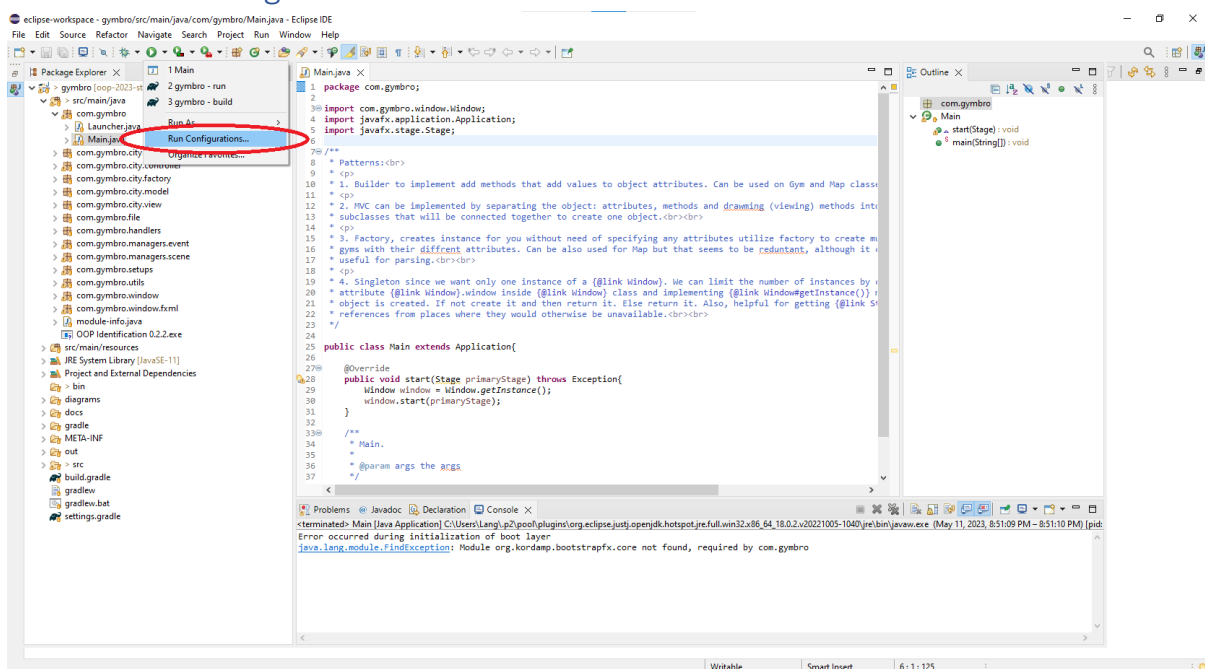


How to import GymBro to Eclipse IDE

29. When you click run project Eclipse will generate “Main” task instead of using Gradle tasks. Make sure to use Gradle tasks from eclipse and not the pre-generated “Main” task. Running the “Main” task instead of Gradle tasks will result in error marked in red square.

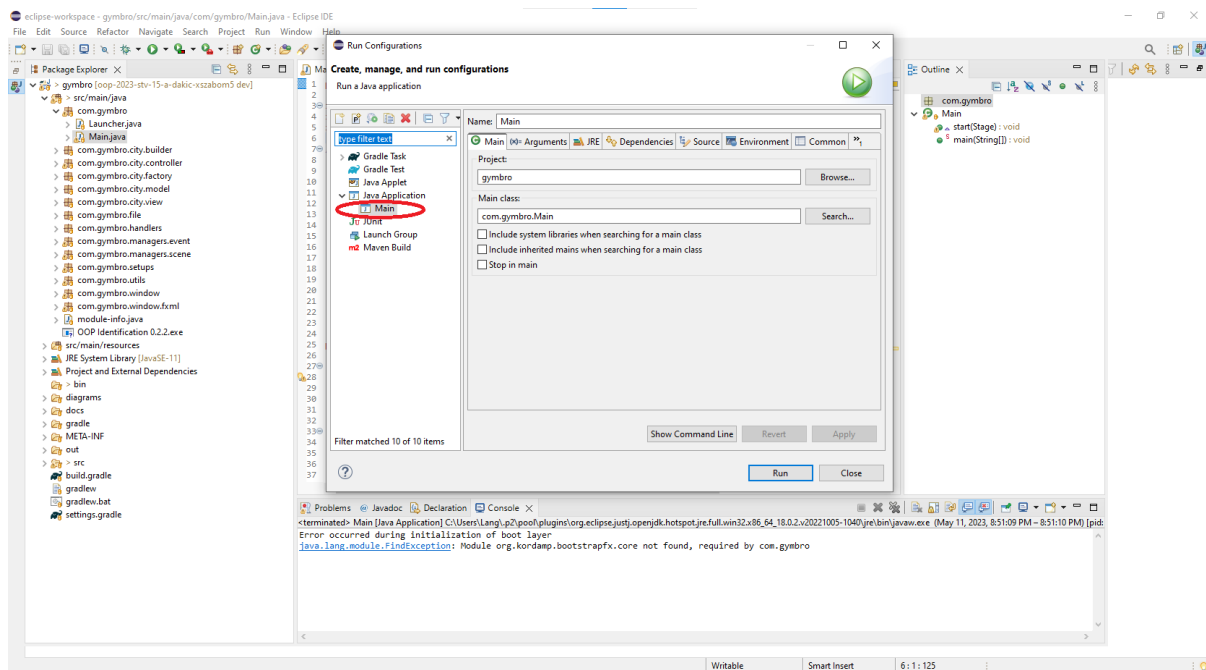


30. To remove the “Main” task. From “Run” right arrow sub menu. Click on “Run Configurations”.

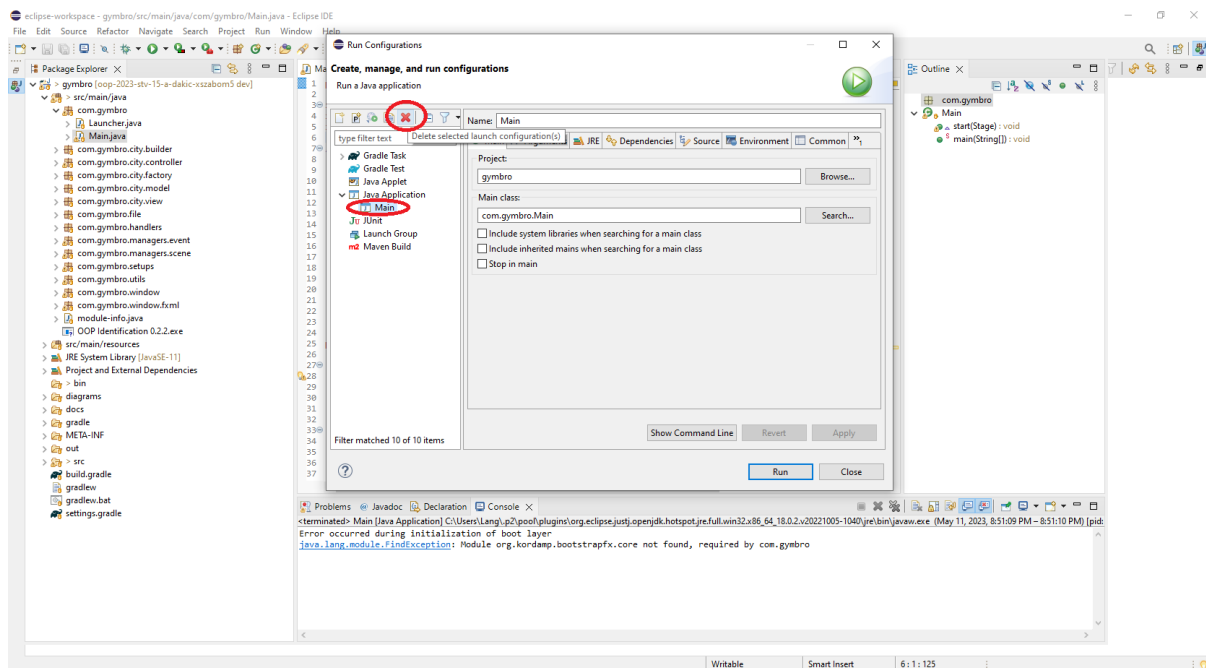


How to import GymBro to Eclipse IDE

31. Select “Main” task.

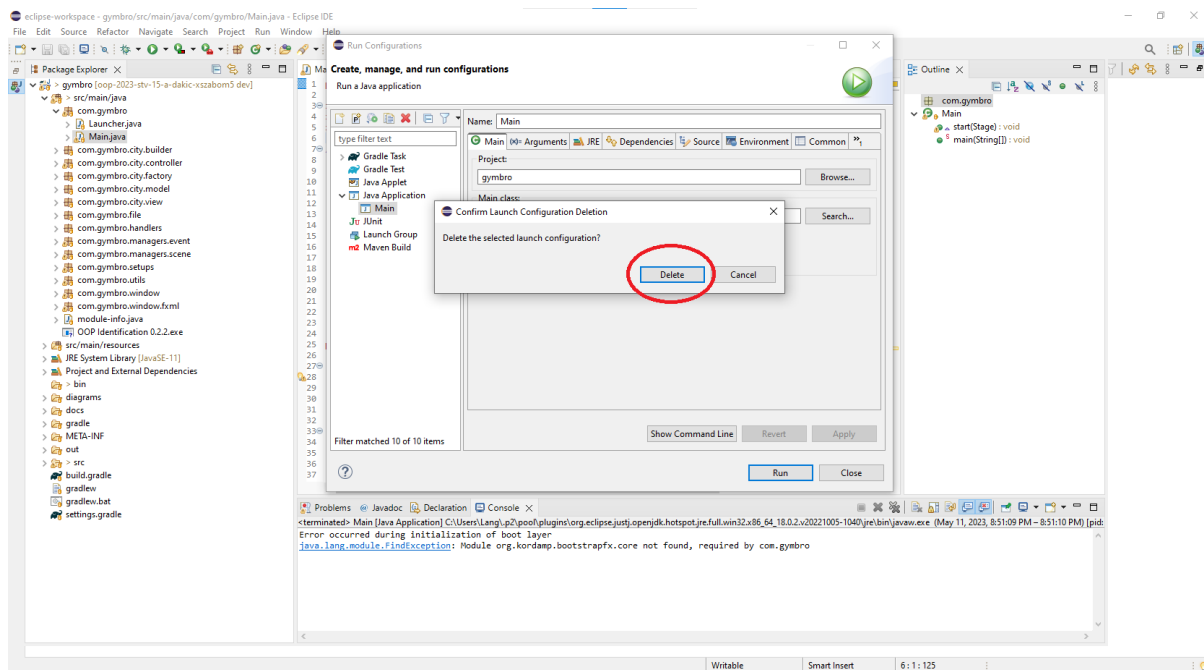


32. Then click on “Delete selected launch configuration(s)”.

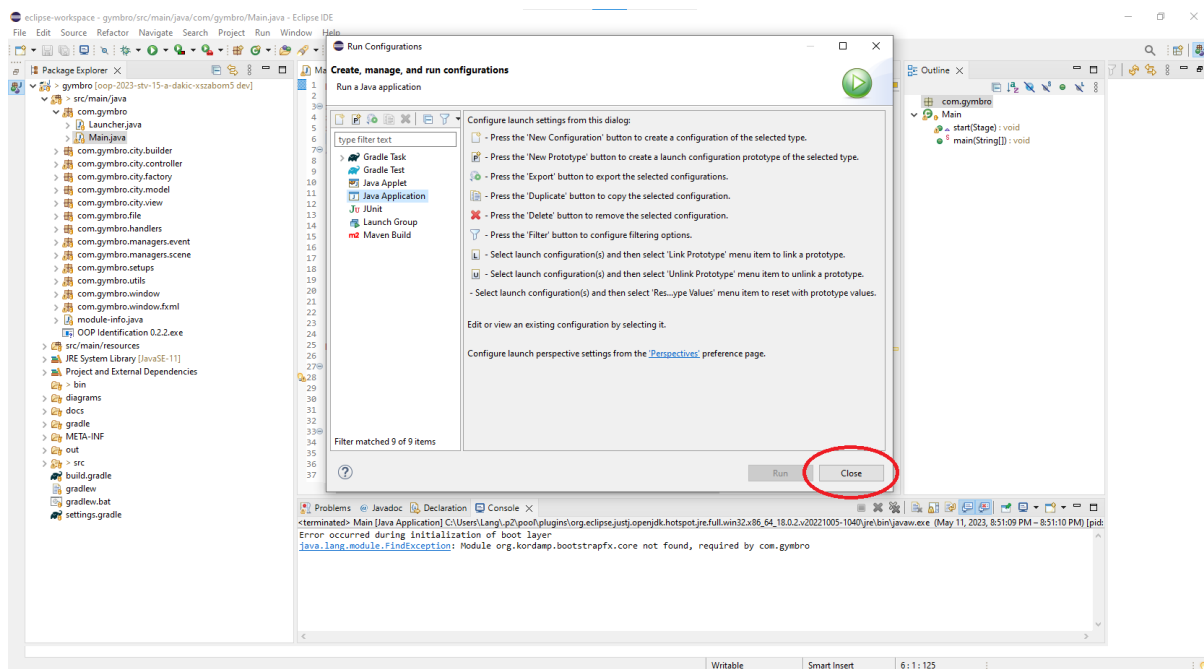


How to import GymBro to Eclipse IDE

33. Click on “Delete”

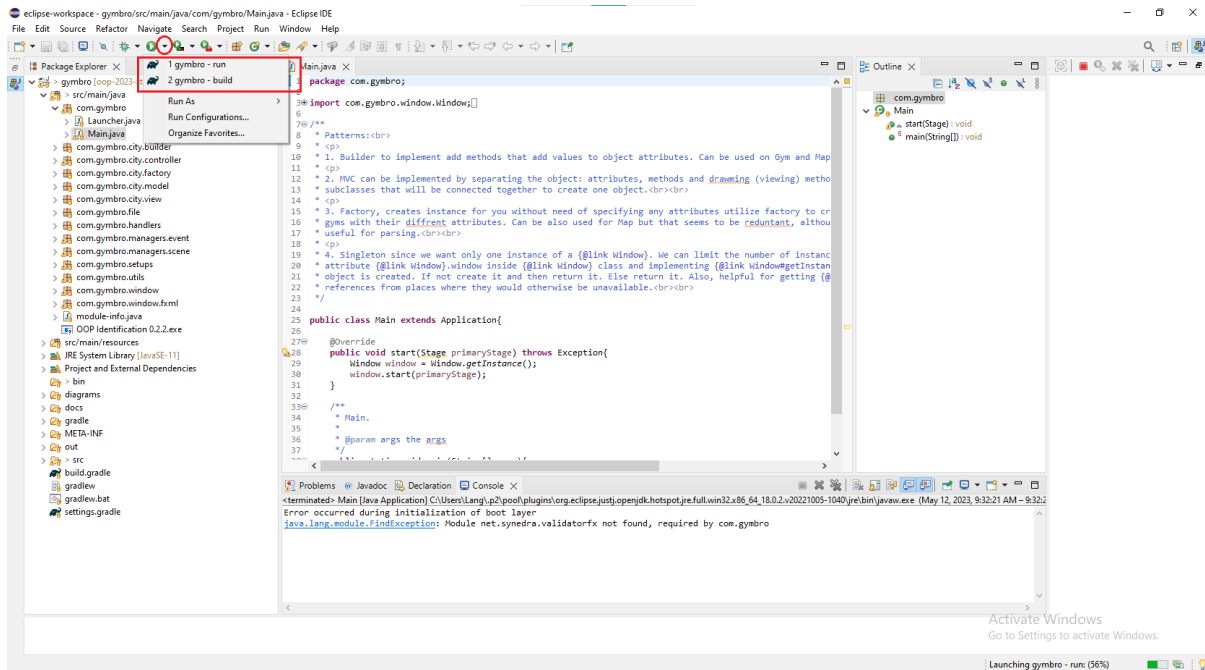


34. Click “Close”

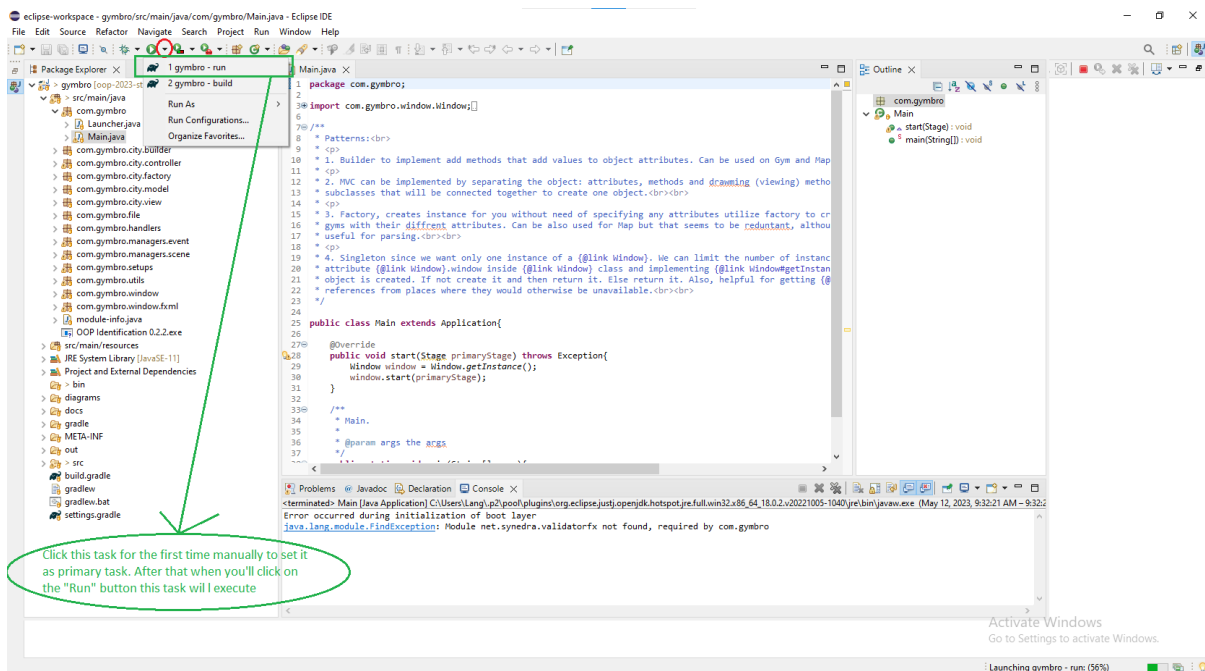


How to import GymBro to Eclipse IDE

35. Now after clicking on “Run” button’s right arrow sub menu we can see only the working Gradle tasks.



36. Now to set Gradle run task as default task. When clicking on “Run” button’s right arrow sub menu. Click on the “1 gymbro – run” Gradle run task. It will run the app. Now next time you’ll click on the eclipse “Run” button. It will use the Gradle run task instead of generating its own “Main” task.



How to import GymBro to Eclipse IDE

37. Now the task is set as default task when clicking on “Run” button.

You have successfully imported GymBro project to Eclipse! 😊

Side note: If you ever receive that initialization of boot layer error from the 29th step. Check if you are using Gradle tasks to run the project as eclipse tends to recreate this “Main” task that is not functional for Gradle projects.