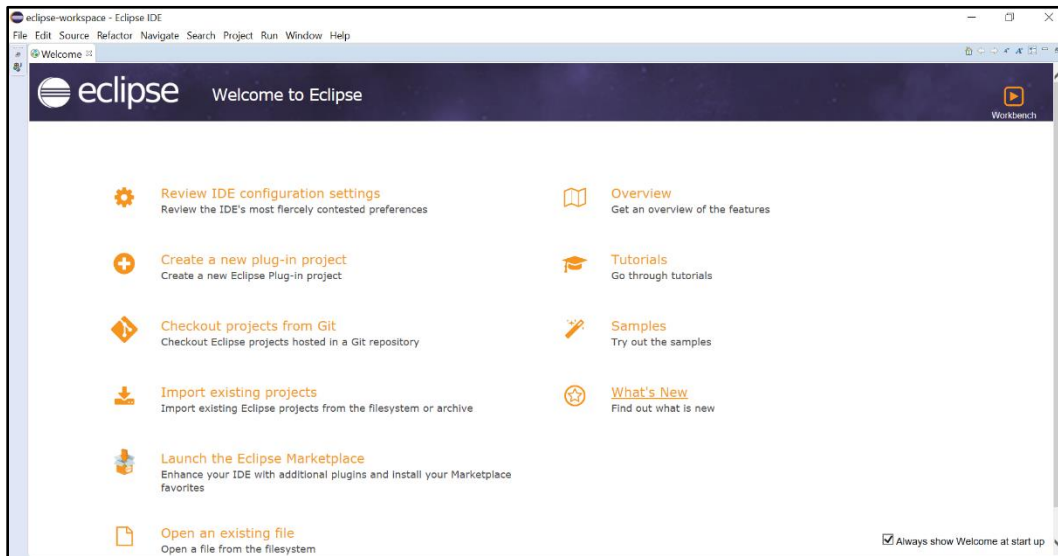
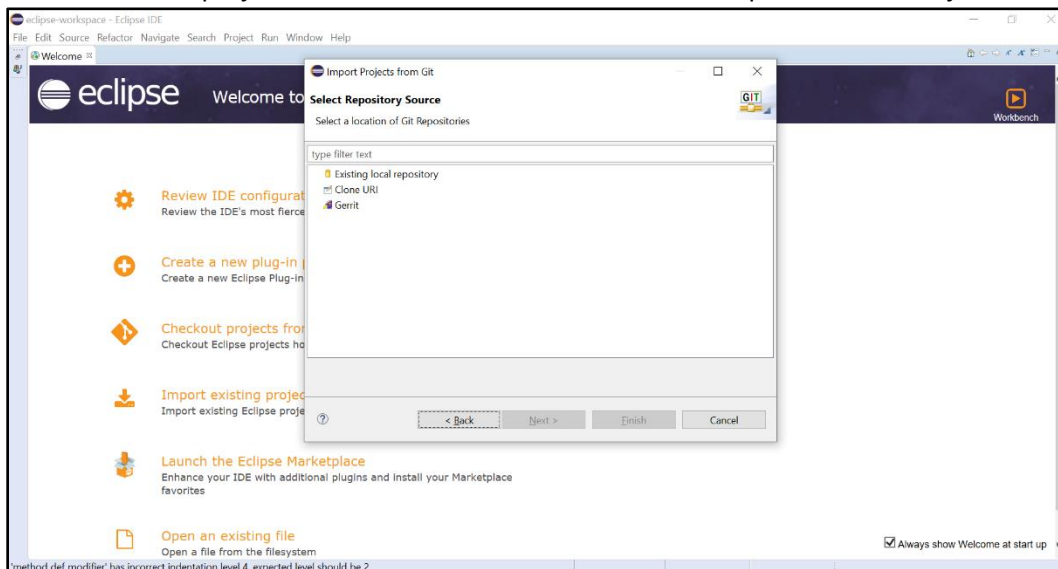


To open the projects and setup the environment on your device please follow the steps:

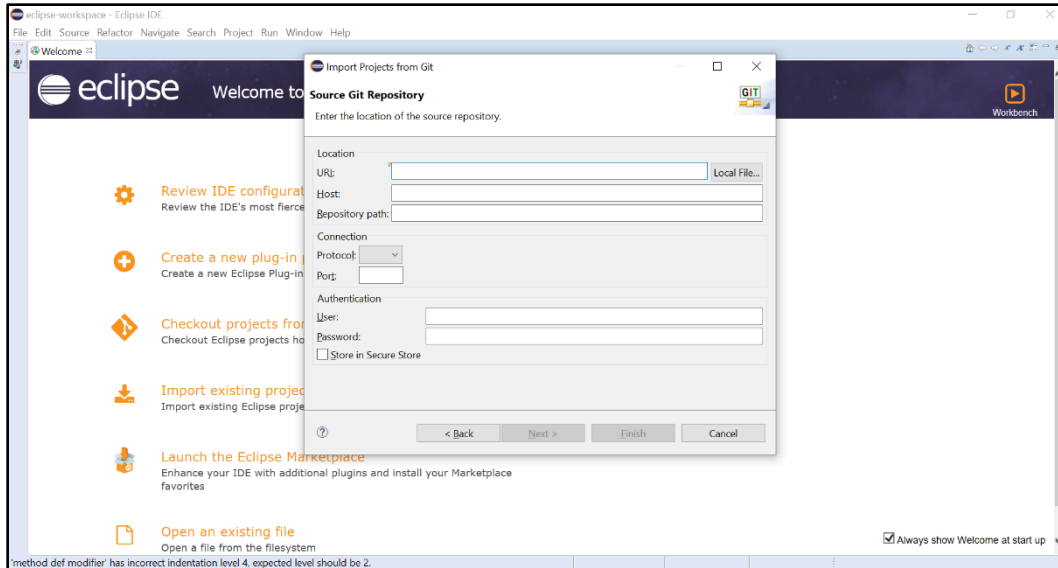
1. Open the Eclipse on your device.



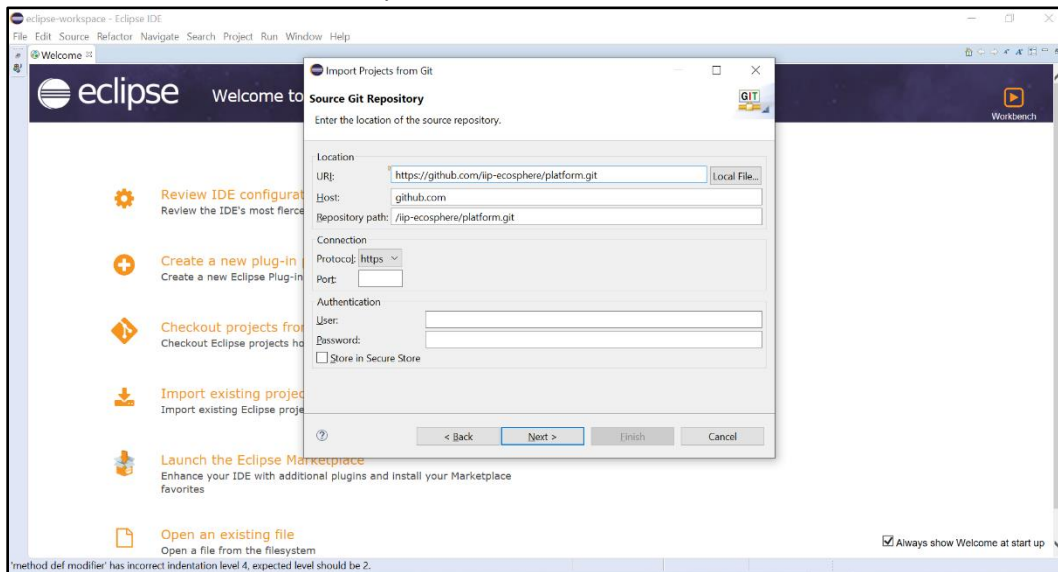
2. Select Checkout projects from Git or from tool menu File -> Import -> Git -> Projects from Git.



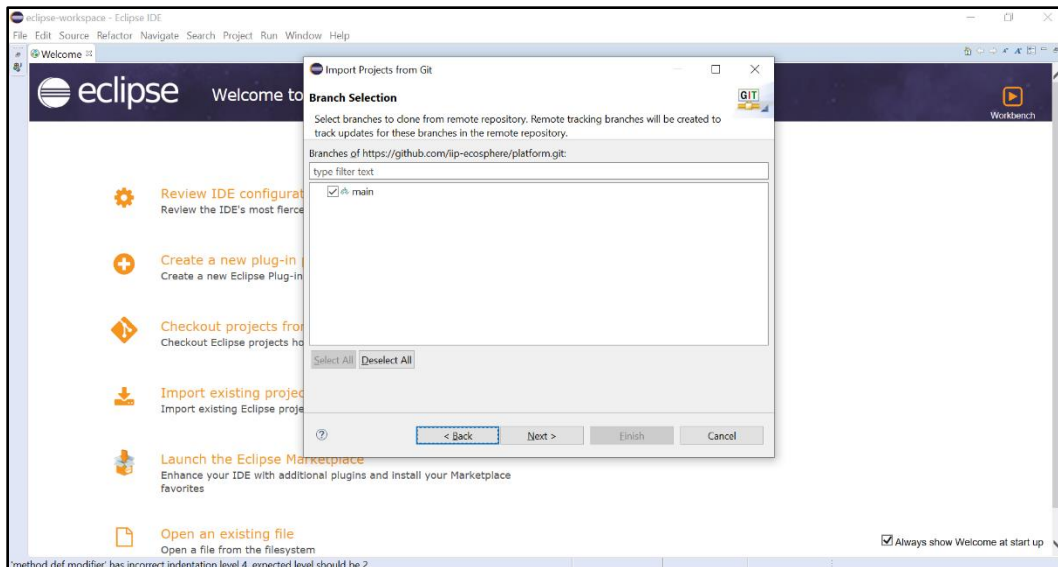
3. Select Clone URI and Next.



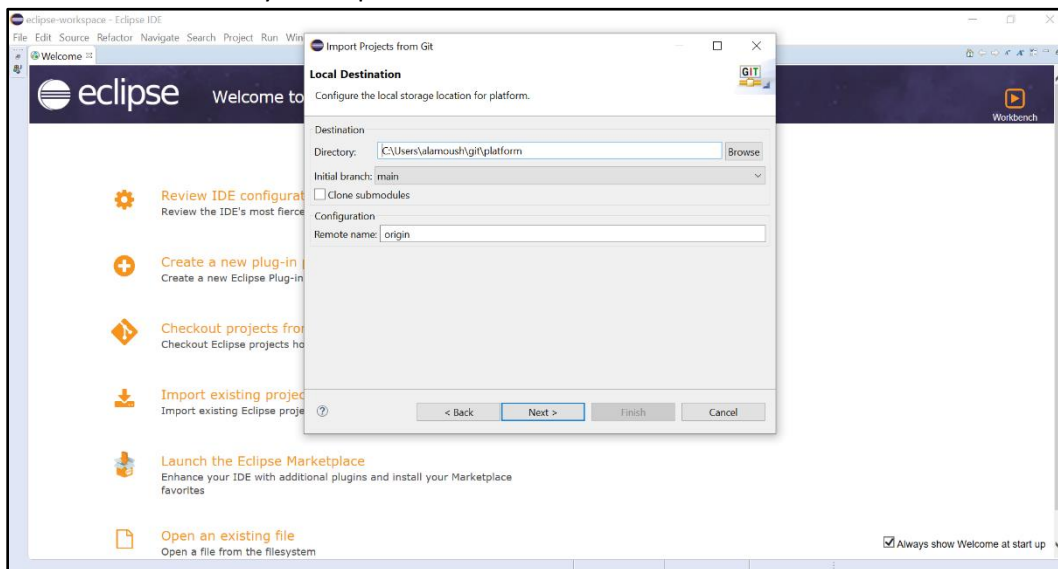
4. Insert the URI "<https://github.com/iip-ecosphere/platform.git>". Host, Repository path, and Protocol will be fill automatically.



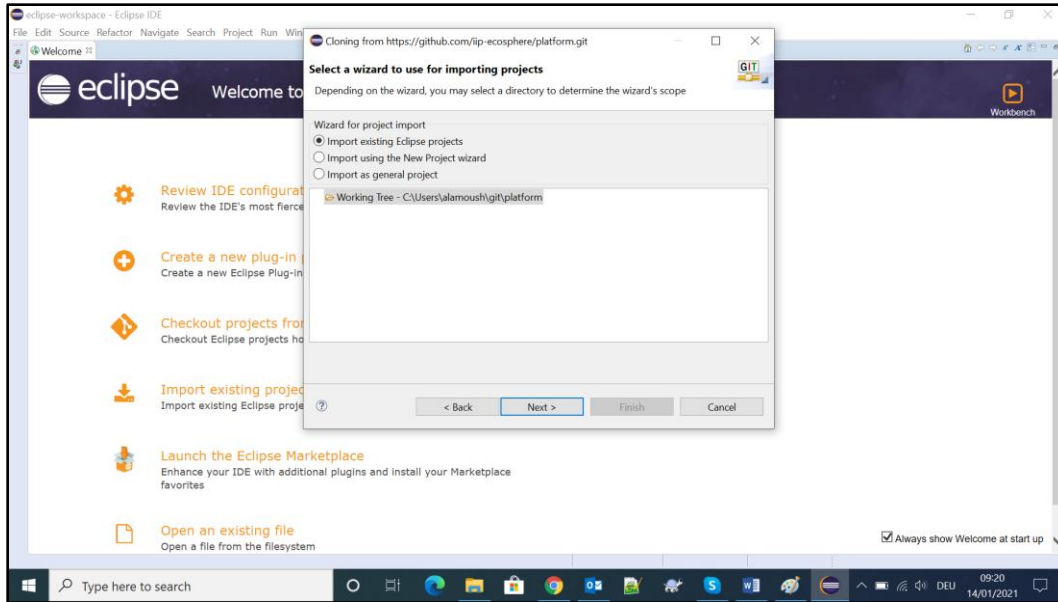
5. Next and keep main selected



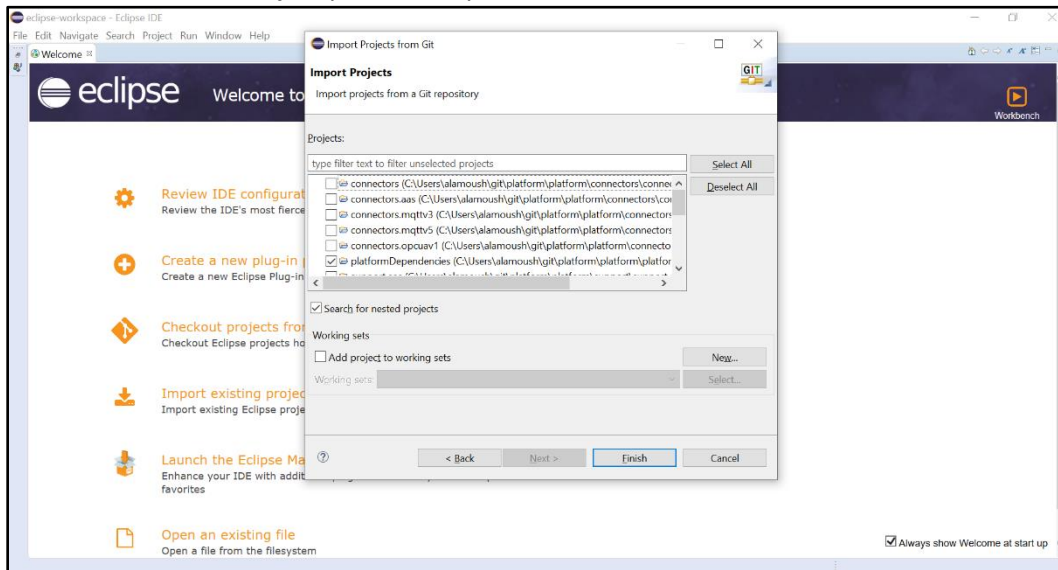
6. Next add the Directory or keep it as default.



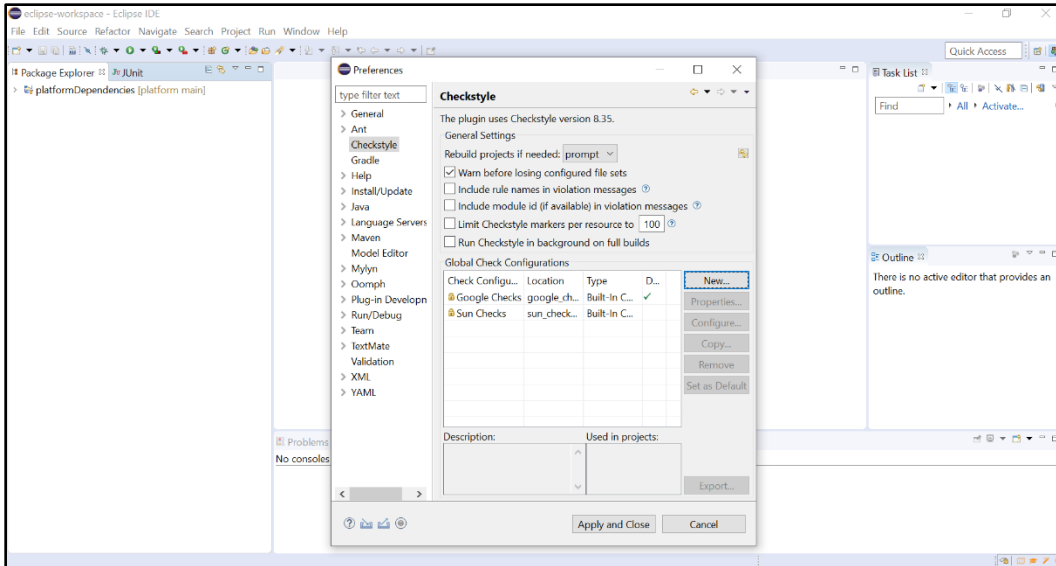
7. Select Import existing Eclipse projects and Next.



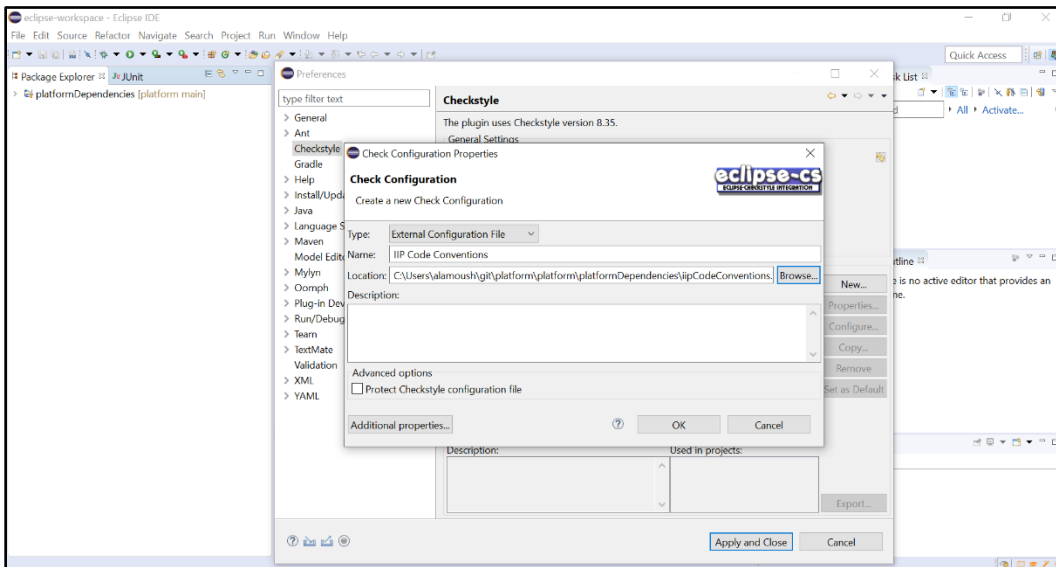
8. Deselect All and select just platformDependencies and Finish.



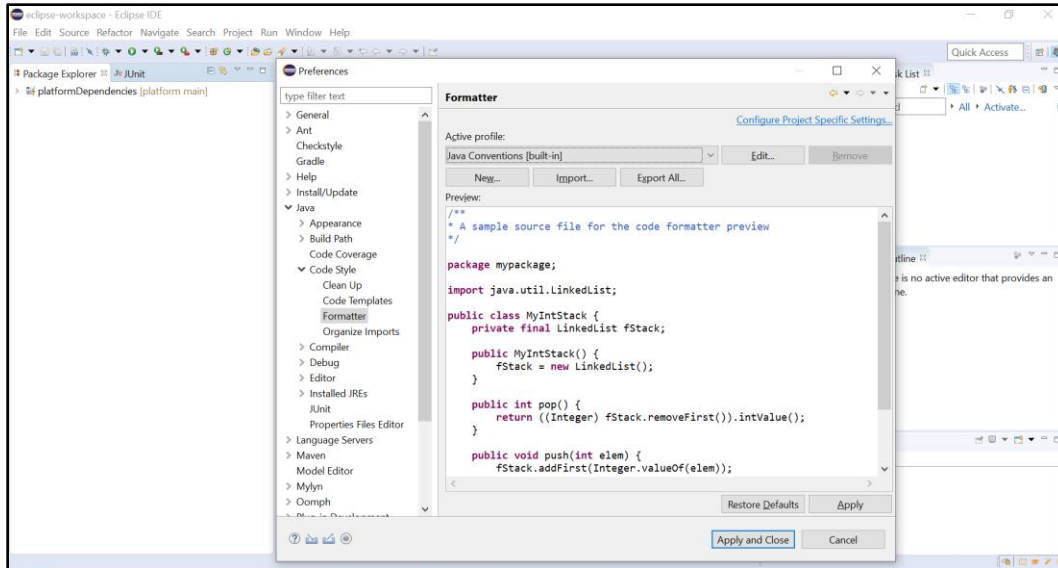
9. From the tool menu Window -> Preferences -> Checkstyle – in case you don't have Checkstyle go to Help -> Eclipse Marketplace and install the Checkstyle Plug-in.



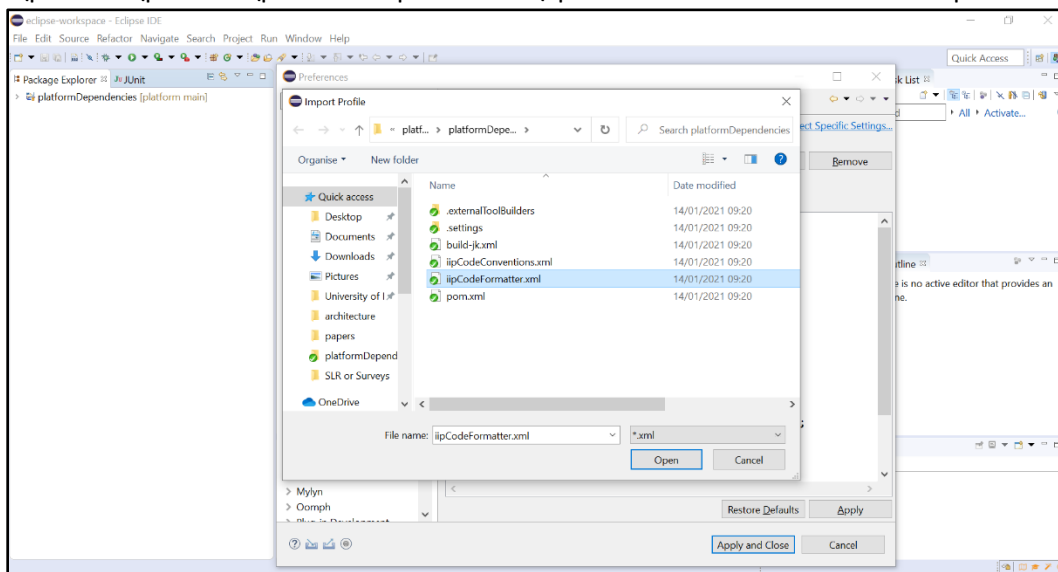
10. Select New and select External Configuration file, insert name “IIP Code Conventions”, and browse to the location of the following file in your device – same directory for the selected for Git “\platform\platform\platformDependencies\iipCodeConventions.xml” then select OK.



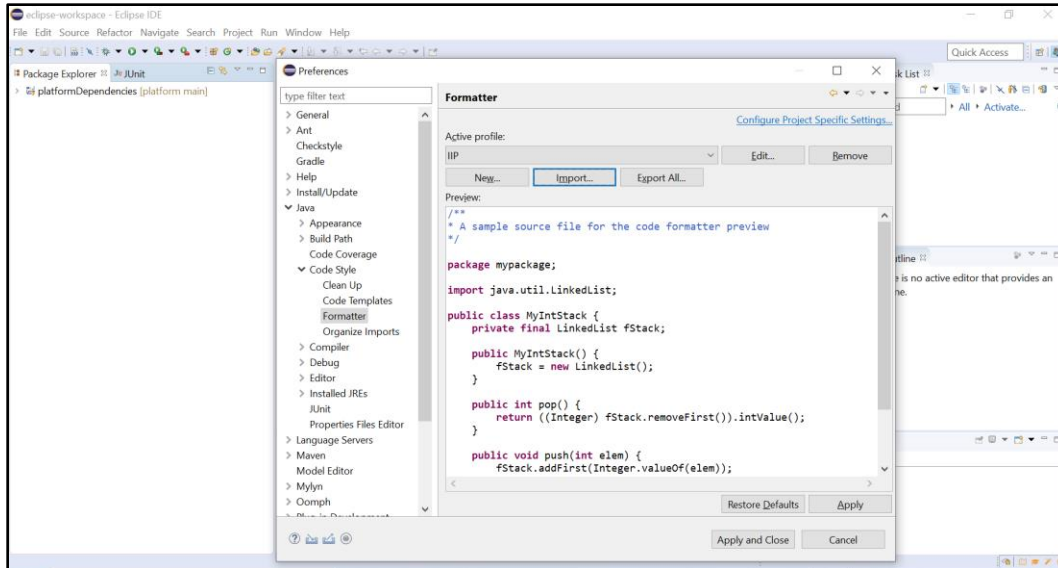
11. Select Java -> Code Style -> Formatter.



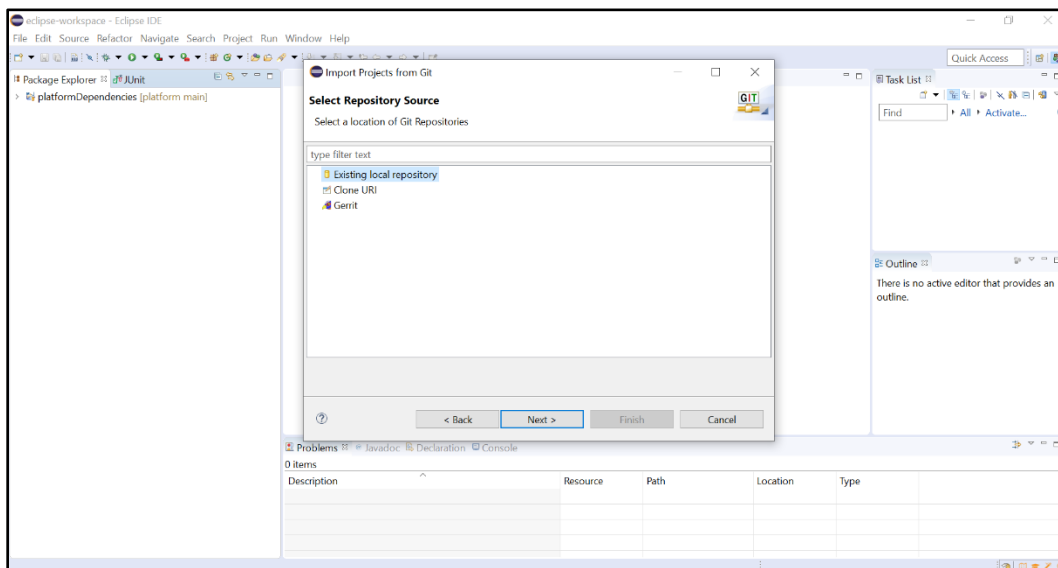
12. Select Import and browse to the location of the Git and select “\platform\platform\platformDependencies\iipCodeFormatter.xml” and click Open.



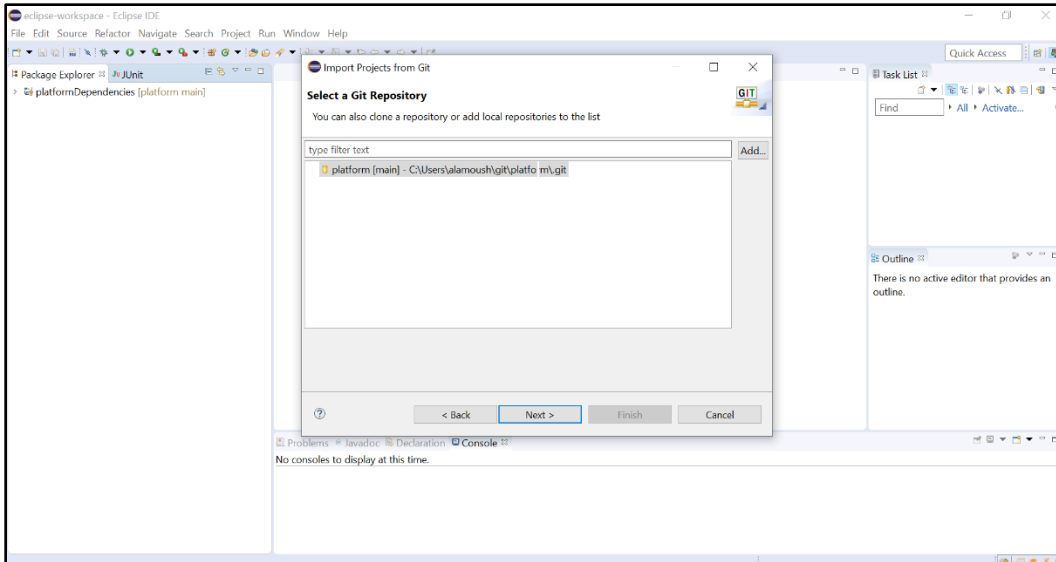
13. Click on Apply and Close.



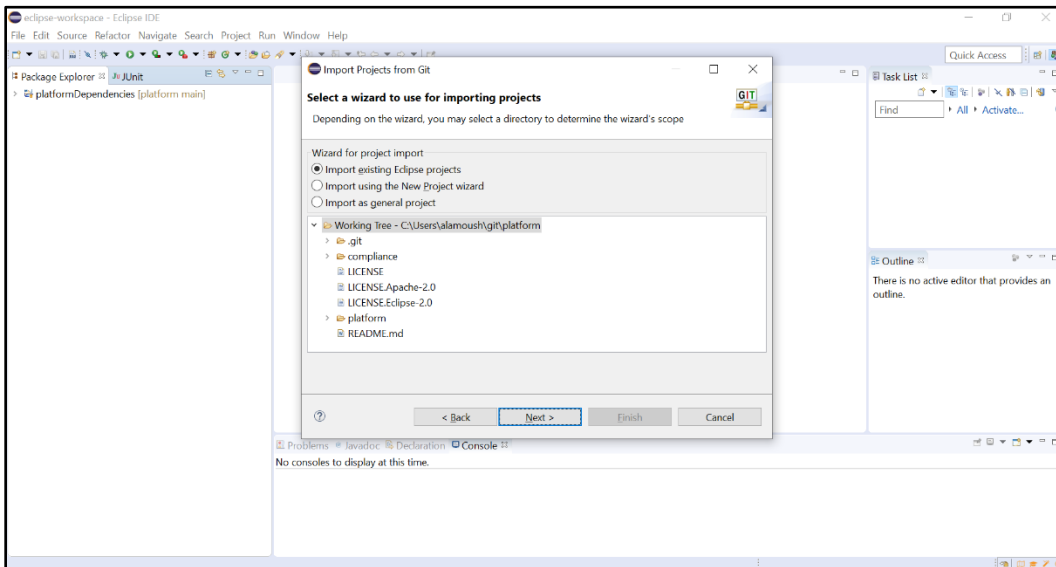
14. From tool menu File -> Import -> Git -> Projects from Git. Select Existing local repository and Next.



15. Select the repository that You have already cloned “in our case Platform” and Next.

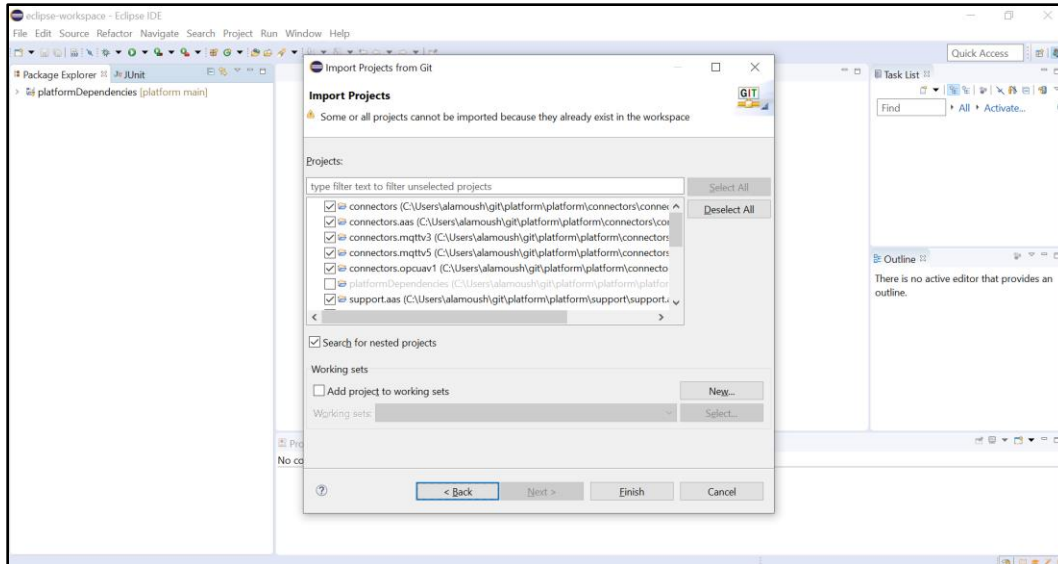


16. Select Import existing Eclipse projects and Next.

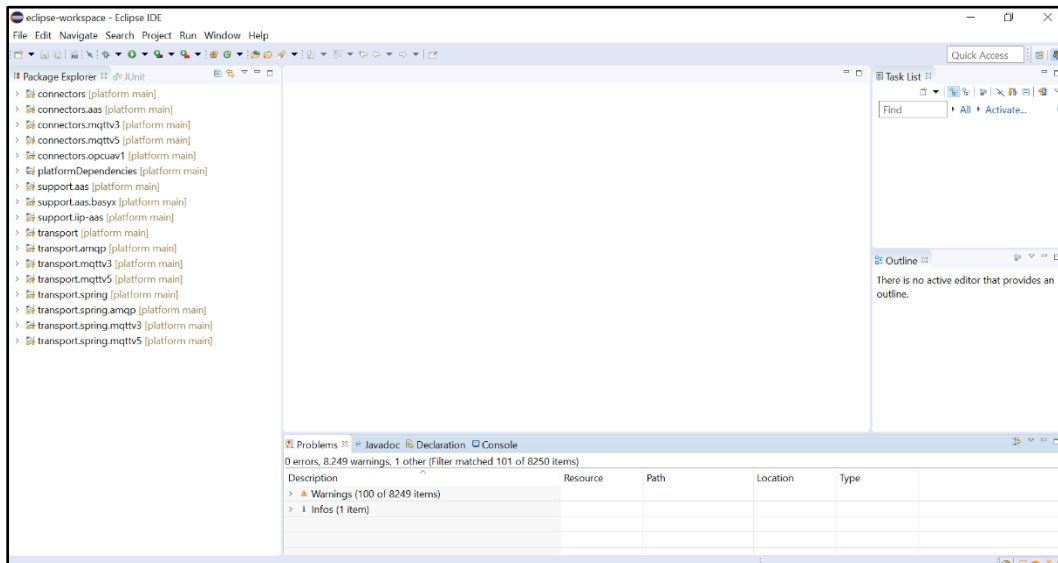


17. Selected the rest of the projects and Finish.



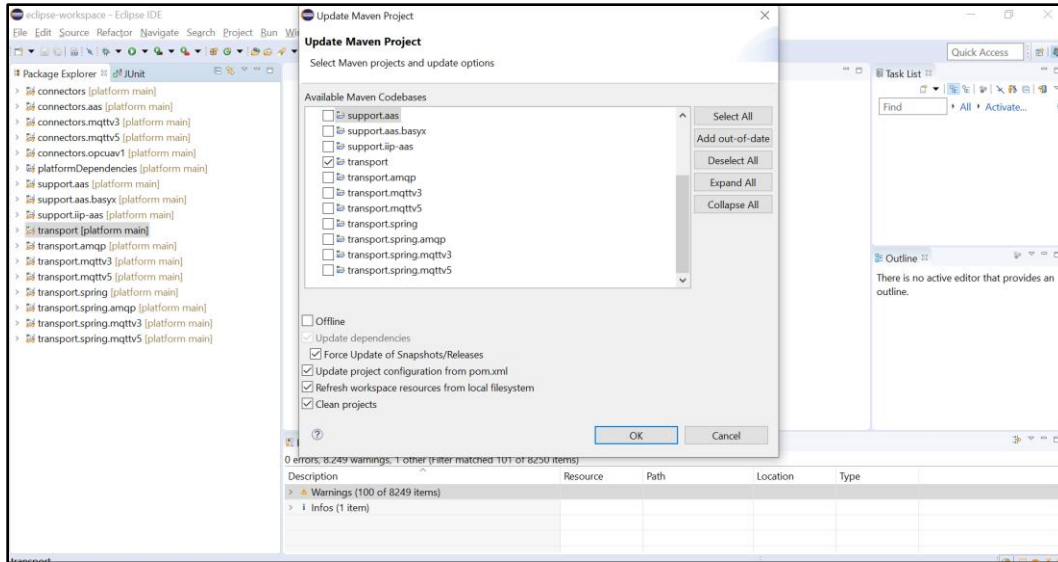


18. You should have all the projects imported without any errors or problems.

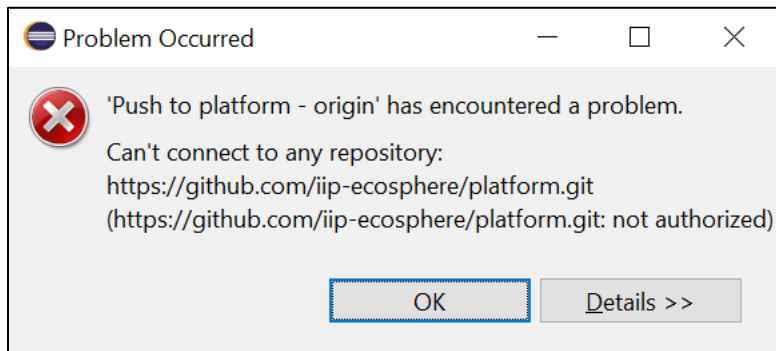


In case there are errors with some projects try:

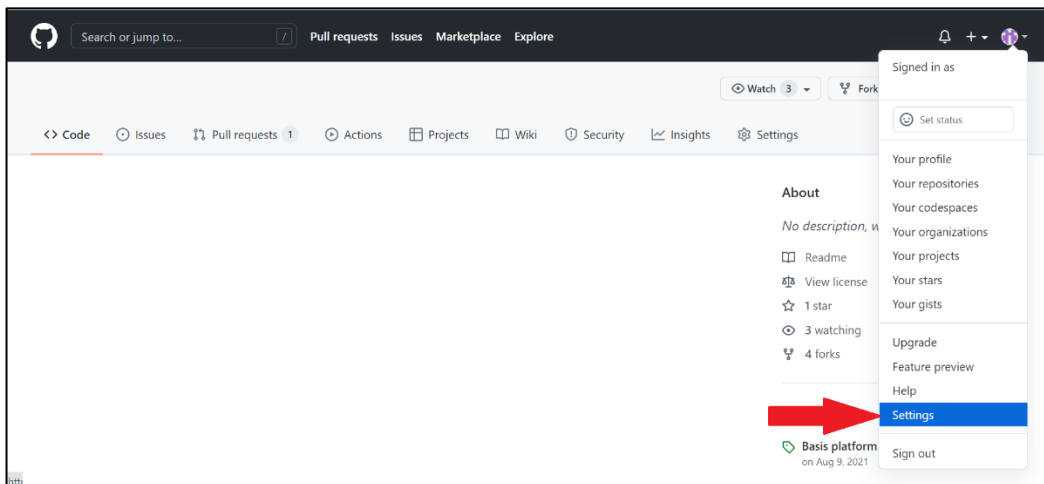
1. Right click on the project -> Maven -> Update Project
2. Select the project with errors.
3. Force Update of Snapshots/Releases
4. Ok.



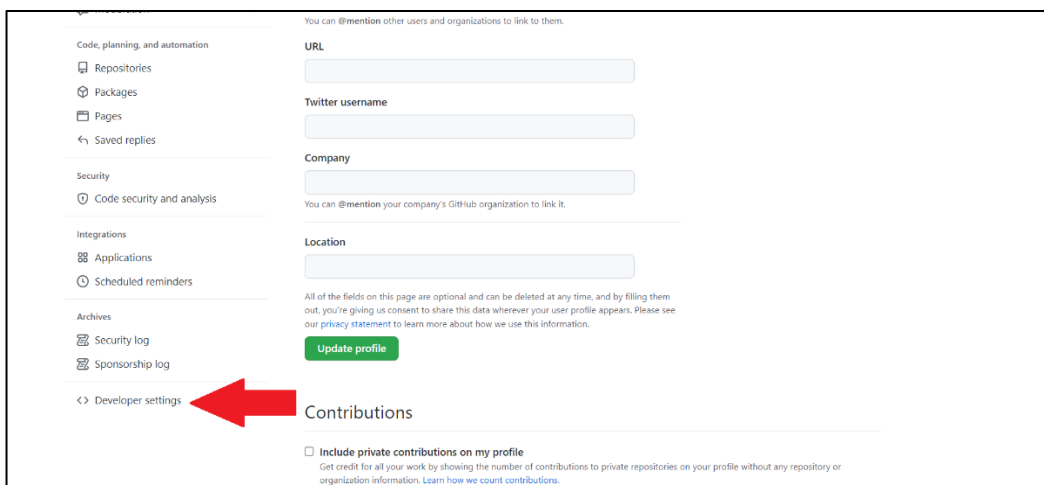
## Solving not authorized error in pushing update from Eclipse to GitHub using Personal access tokens from GitHub



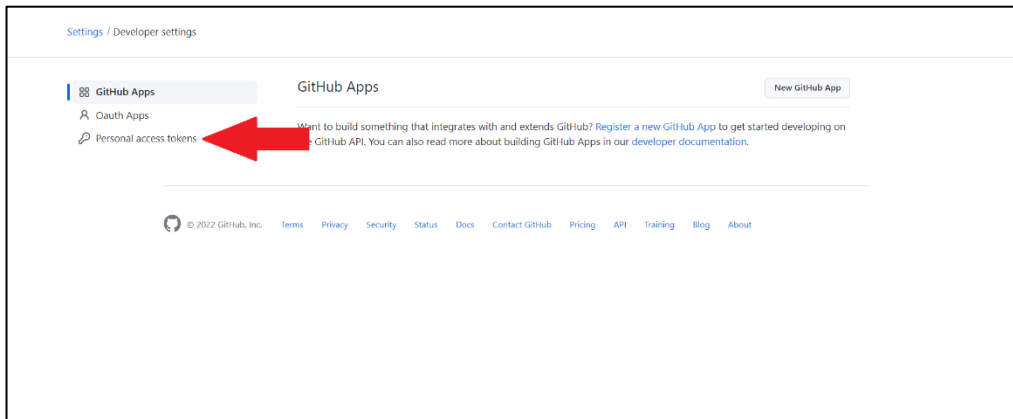
1. Select Settings from the account in the GitHub



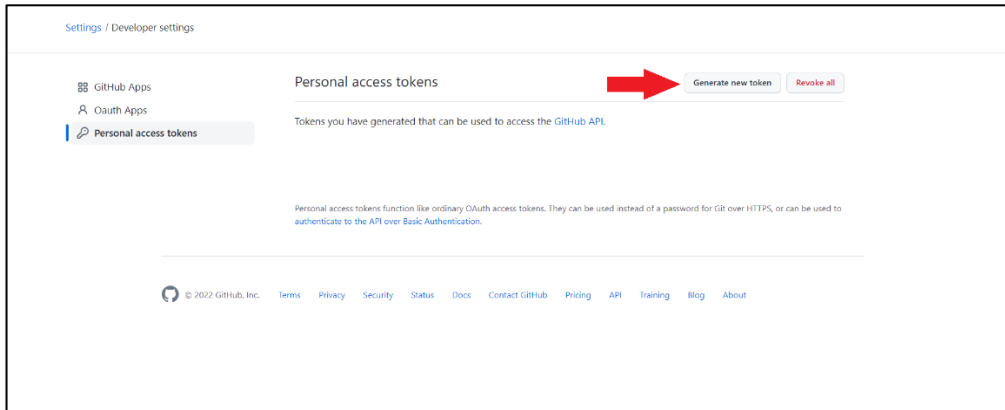
2. From the left Menu select Developer settings



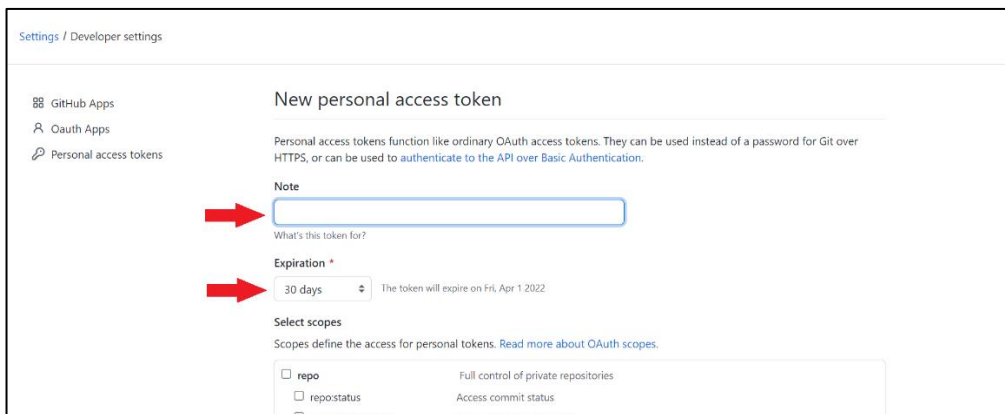
### 3. Select Personal access tokens



### 4. Select Generate new token (confirm your password in next page)



### 5. Add note (optional) and select expiration date for the token



6. You might keep the other option in default setting, click on Generate token

The screenshot shows the 'Generate new token' dialog in GitHub's settings. It contains a table of permissions with checkboxes. A red arrow points to the 'Generate token' button at the bottom.

Access user email	Access user email addresses (read-only)
<input type="checkbox"/> user:follow	Follow and unfollow users
<input type="checkbox"/> delete_repo	Delete repositories
<input type="checkbox"/> write:discussion	Read and write team discussions
<input type="checkbox"/> read:discussion	Read team discussions
<input type="checkbox"/> admin:enterprise	Full control of enterprises
<input type="checkbox"/> manage_runners:enterprise	Manage enterprise runners and runner-groups
<input type="checkbox"/> manage_billing:enterprise	Read and write enterprise billing data
<input type="checkbox"/> read:enterprise	Read enterprise profile data
<input type="checkbox"/> admin:gpg_key	Full control of public user GPG keys (Developer Preview)
<input type="checkbox"/> write:gpg_key	Write public user GPG keys
<input type="checkbox"/> read:gpg_key	Read public user GPG keys

**Generate token** Cancel

7. Now the token is created, you should use it instead of the password in Eclipse while pushing update.

The screenshot shows the 'Personal access tokens' page in GitHub's settings. It displays a list of generated tokens. A red arrow points to the first token.

Settings / Developer settings

Personal access tokens **Generate new token** **Revoke all**

Tokens you have generated that can be used to access the GitHub API.

Make sure to copy your personal access token now. You won't be able to see it again!

✓ ghp\_eI7bXX6zs2RLIwZlmmRUXH4FtjvrVG2NIINVe **Delete**

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

The screenshot shows the 'Login' dialog in Eclipse. It contains fields for Repository, User, Password, and a checkbox for 'Store in Secure Store'. A red arrow points to the Password field.

**Login**

Repository `https://github.com/iip-ecosphere/platform.git`

User

Password

Store in Secure Store ☐

**Log in** **Cancel**