
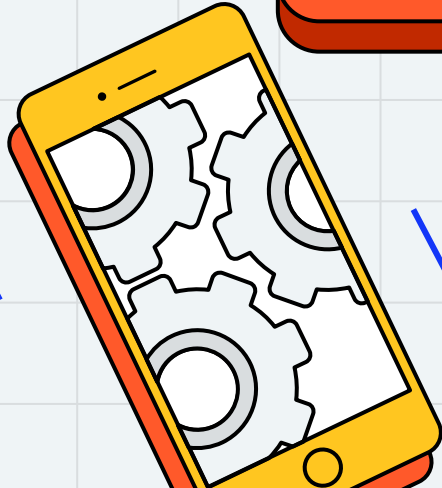


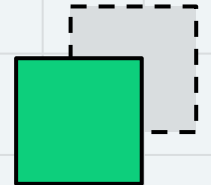


Application

Programming



Hend Alkittawi





Version Control

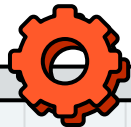
Introduction To Git And Github

INTRODUCTION



We watched a few videos from the following course on Udacity:

[Version Control With Git](#)

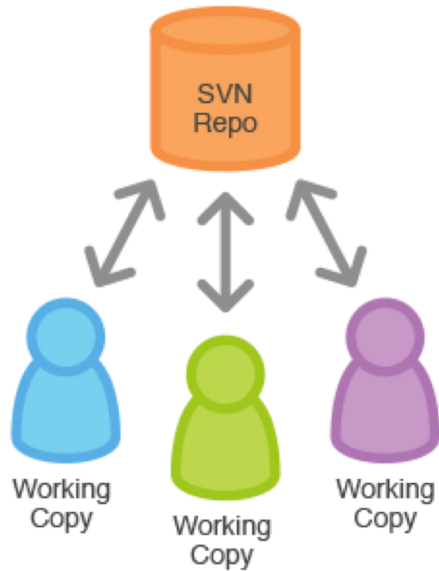


VERSION CONTROL

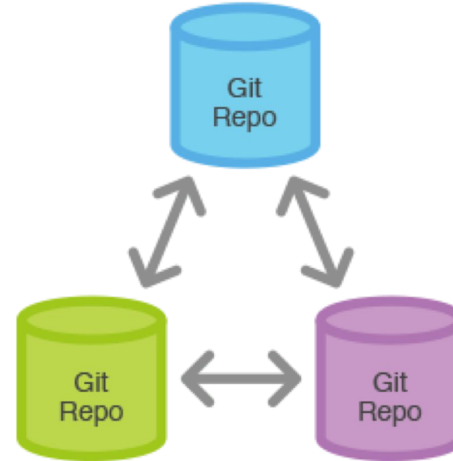
- Collaborative software development necessitates a system for source control and versioning!
- A version control system facilitates simultaneous updates to the same software and manages any conflicts created!
- There are a few version control system options
 - CVS - Concurrent Versions System
 - SVN - Apache Subversion
 - GIT - a distributed VCS (not the only one)

VERSION CONTROL

Central-Repo-to-Working-Copy
Collaboration



Repo-to-Repo
Collaboration



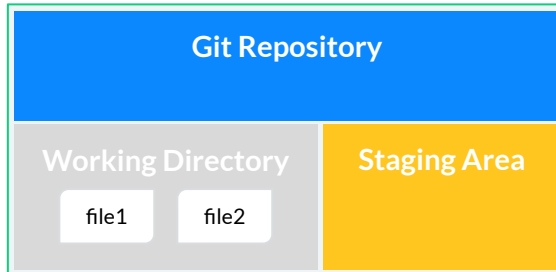
GIT

- Developed by Linus Torvalds around 2005
- Open source, under the GNU GPL (General Public License)
- Distributed, in that every directory on every computer is a full-fledged repository
- It has a complete history, version-tracking capabilities
- Independent of network access, or a central server
- Use as command-line or via GUI

GIT

- To create a local Git Repo

1. Create a new project directory `mkdir WorkingDir`
2. Move into that new directory `cd WorkingDir`
3. Initialize Git repository `git init`
4. Create/edit project files
5. Stage files in the project directory `git add .`
6. Commit tracked files in Git storage `git commit -m "a msg"`



GITHUB

- Web-based Git repository hosting service
- Provides cloud storage, public/private repos, free accounts
- Home to lots of open source software!
- To use Github

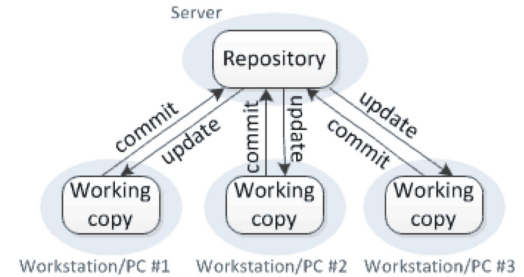
1. Access a remote repository on Github.com

2. Pull down the latest version of the code to local `git pull`

3. Add edited files to the source code `git add File.java`

4. Commit edits to the local version `git commit -m "short msg"`

5. Push the changes out to the server `git push`



VERSION CONTROL

- Summary of Git Terminology
 - Repo - repository
 - Init - initialize a repository
 - Commit
 - Push/Pull
 - Clone - copy a remote repo locally
 - Branch - a separate version of the main repo that allows working on different parts of a project without impacting the main branch.

VERSION CONTROL

- Git and Github
 - You can choose how you want to use Git and Github
 - Command line
 - **Browser** <http://github.com>
 - GitHub Desktop <http://desktop.github.com>
 - **IDE**

VERSION CONTROL

- Some useful resources
 - Hints for using eGit (Eclipse plugin)
 - Resolving merge conflicts
 - Effective Git: <https://git-scm.com/book/en/v2>
 - Tutorial: <https://try.github.io/>

README FILES

- A **code readme file** should include dependencies, description of functionality, etc.
- A **repo readme file** should describe project(s) in the repo, contributors, functionality, license(s), contribution guidelines, known issues, ...
- A README file is created using markdown language!
- Example: README.md template · GitHub

README FILES

- More Readme
 - [A Beginners Guide to writing a Kickass README 📖 | by Akash](#)
 - [Top ten reasons why I won't use your open source project |> Changelog](#)
 - [GitHub - hackergrrrl/art-of-readme: :love_letter: Things I've learned about writing good READMEs.](#)
 - [How To Write A Great README](#)
 - [GitHub - jehna/readme-best-practices: Best practices for writing a README for your open source project](#)
 - [zalando-howto-open-source/READMEtemplate.md at master](#)

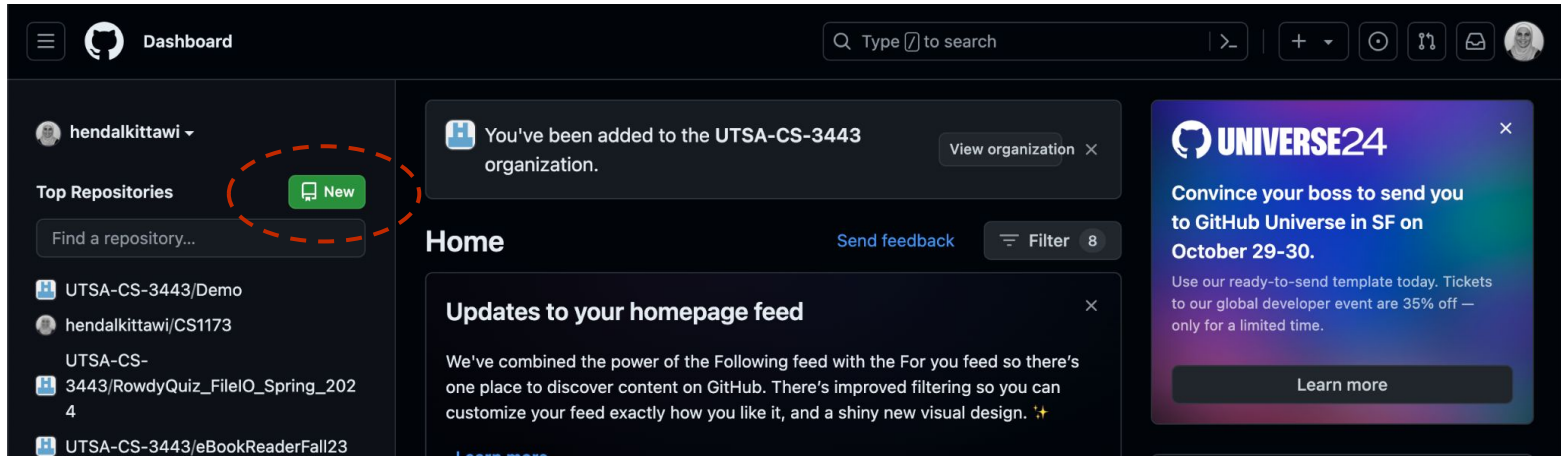


TUTORIAL

Using Git/Github in
Android Studio

Generate a personal access token and save it somewhere! (Under your Github account > Settings > Developer Settings)

Check the repository created under your team in CS3443 Github organization OR create your own repository under your account



The screenshot shows the GitHub Dashboard interface. At the top, there is a search bar with the text "Type to search" and a navigation bar with icons for home, add, refresh, fork, and messages. The user's profile picture is visible in the top right corner. On the left sidebar, the user's name "hendalkittawi" is displayed with a dropdown arrow. Below it, the "Top Repositories" section lists several repositories, including "UTSA-CS-3443/Demo", "hendalkittawi/CS1173", "UTSA-CS-3443/RowdyQuiz_FileO_Spring_2024", and "UTSA-CS-3443/eBookReaderFall23". A green "New" button with a repository icon is circled in red. In the main content area, a notification banner states "You've been added to the UTSA-CS-3443 organization." with a "View organization" link. Below this, the "Home" section features a "Send feedback" link and a "Filter 8" button. A "Updates to your homepage feed" section is also visible. On the right side, there is a promotional banner for "UNIVERSE24" with the text "Convince your boss to send you to GitHub Universe in SF on October 29-30." and a "Learn more" button.



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).


Required fields are marked with an asterisk (*).

Repository template

No template ▾

Start your repository with a template repository's contents.

Owner *

 hendalkittawi ▾

Repository name *

Great repository names are short and memorable. Need inspiration? How about [ubiquitous-dollop](#) ?

Description (optional)

 **Public**

Anyone on the internet can see this repository. You choose who can commit.

 **Private**

You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file

This is where you can write a long description for your project. [Learn more about READMEs](#).

Add .gitignore

.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files](#).

Choose a license

License: None ▾

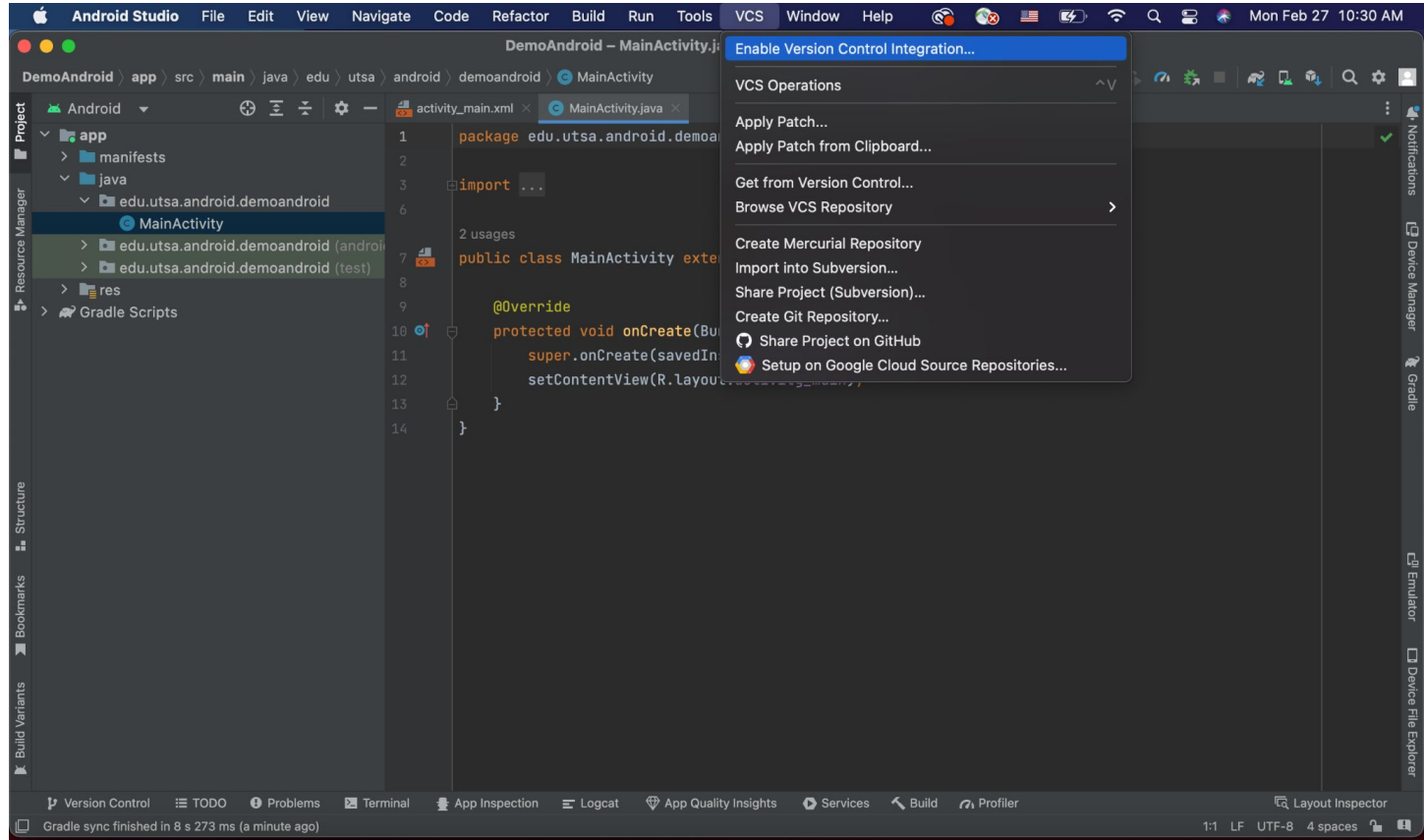
A license tells others what they can and can't do with your code. [Learn more about licenses](#).

This will set `main` as the default branch. Change the default name in your [settings](#).

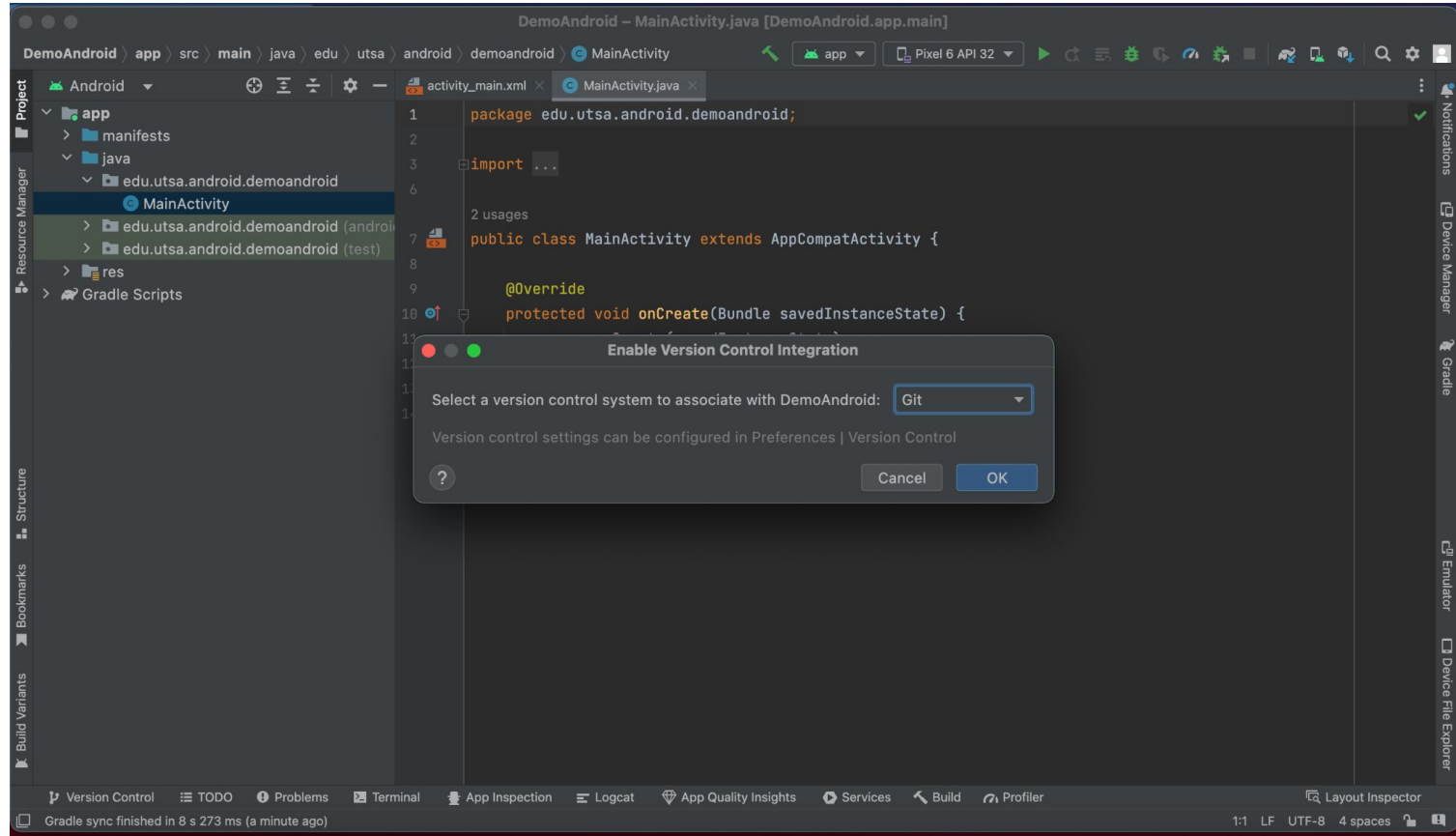
 You are creating a public repository in your personal account.

Create repository

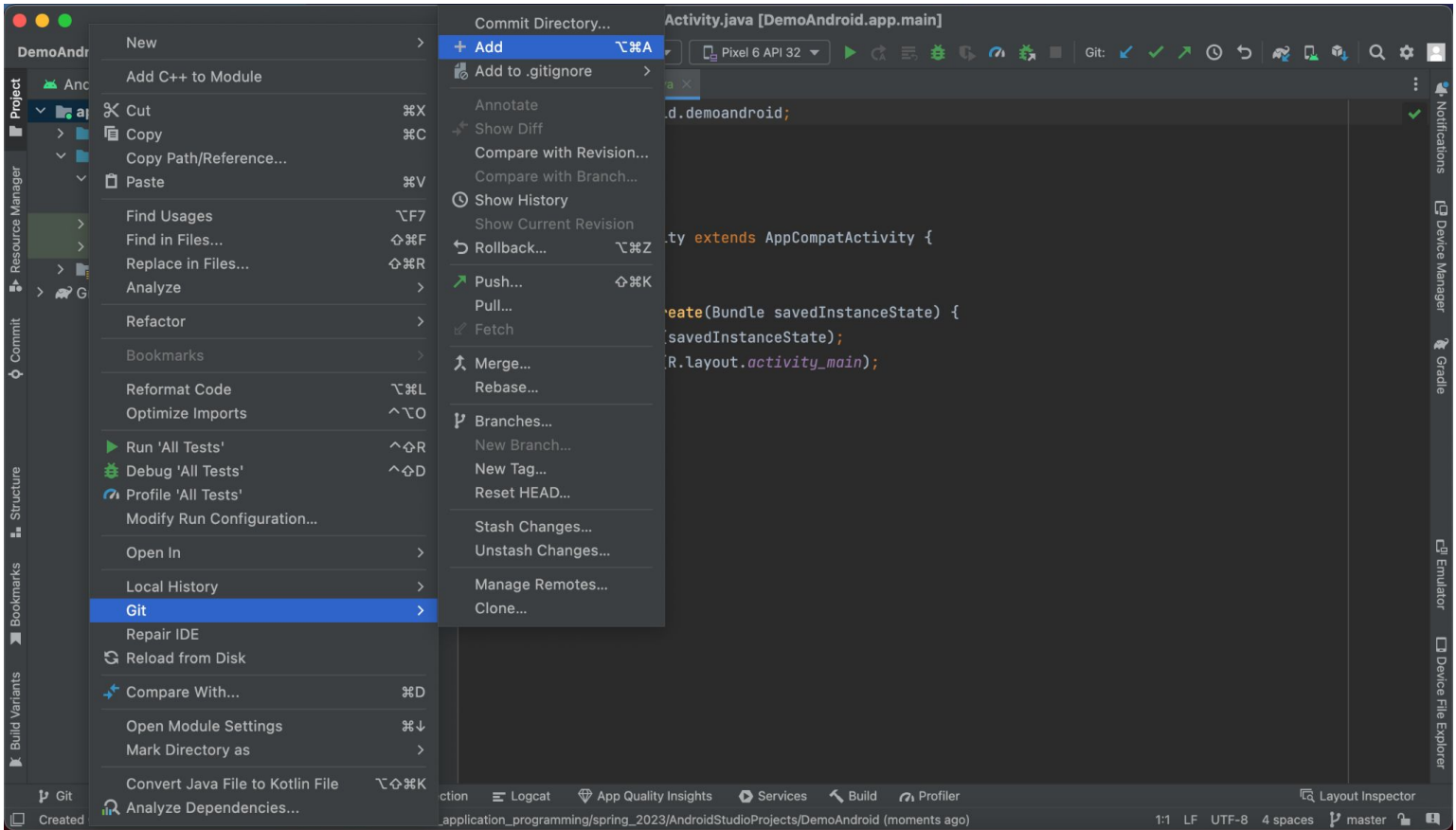
Create your project in Android Studio then create a local Git repo
(Android Studio > VCS > Enable Version Control Integration, select Git)



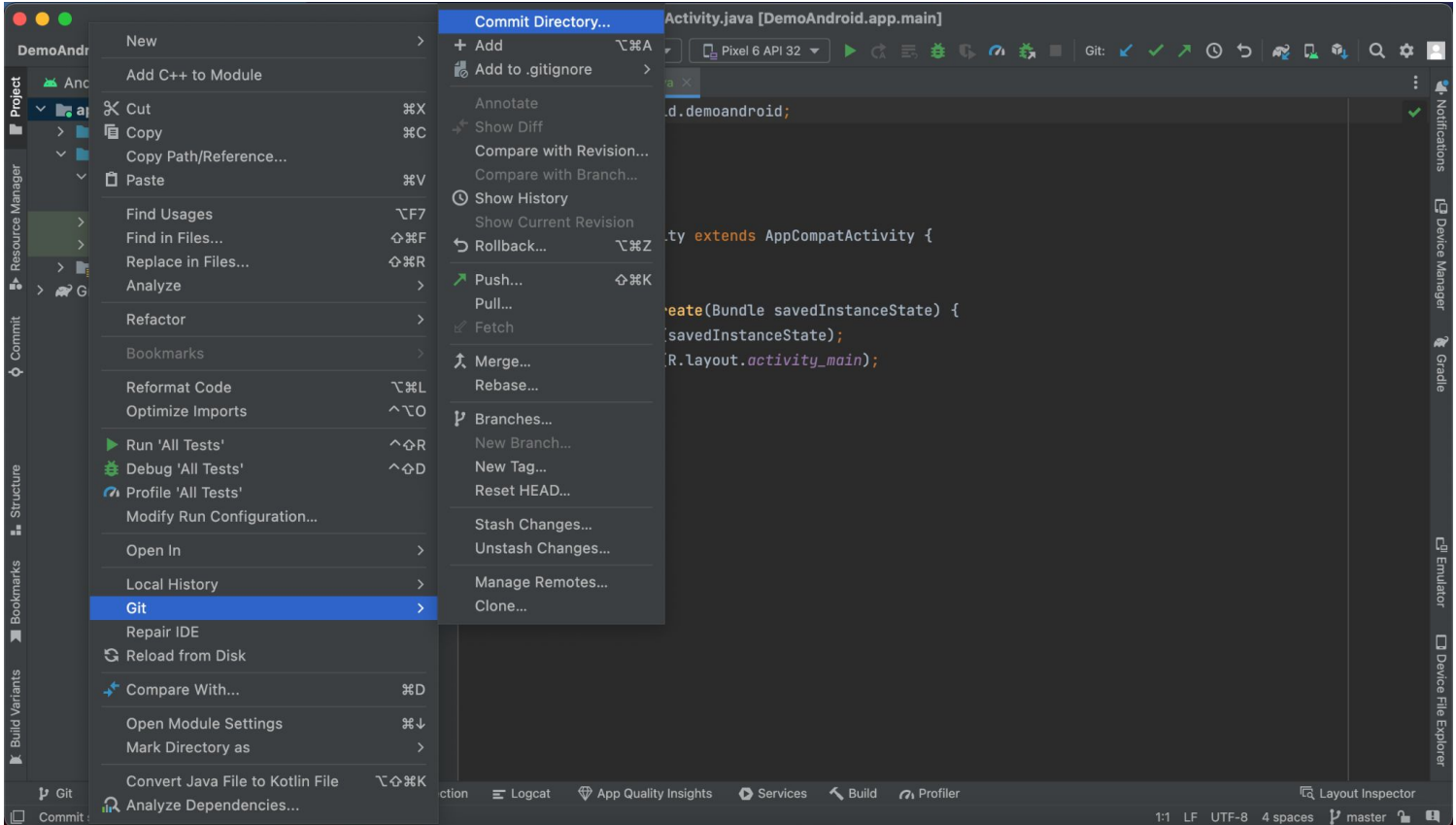
Create your project in Android Studio then create a local Git repo
(Android Studio > VCS > Enable Version Control Integration, select Git)



Stage files (Right click on app > Git > Add)



Commit to local repo (Right click on app > Git > Commit Directory...) and add the unversioned files



Rename local branch to main (to match the remote branch name)

-- IMPORTANT --

The screenshot shows an IDE interface with the following components:

- Code Editor:** Displays the `MainActivity.java` file with the following code:

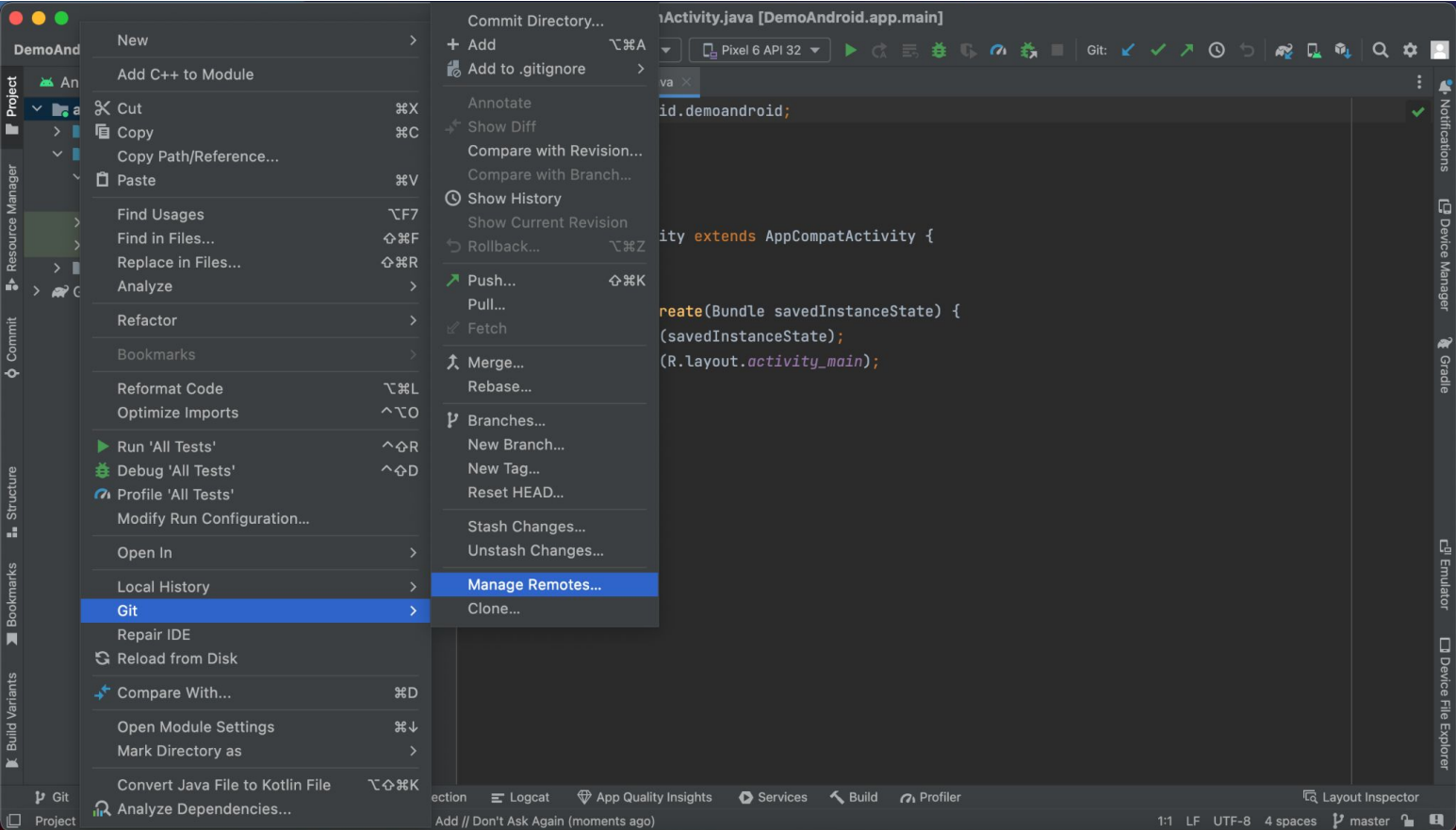
```
1 package edu.utsa.android.demos.android;  
2  
3 import ...  
4  
5 2 usages  
6  
7 public class MainActivity extends AppCompatActivity {  
8  
9     @Override  
10    protected void onCreate(Bundle savedInstanceState) {  
11        super.onCreate(savedInstanceState);  
12        setContentView(R.layout.activity_main);  
13    }  
14 }
```
- Project Explorer:** Shows the project structure for `DemoAndroid`, including `app`, `manifests`, `java`, `res`, and `Gradle Scripts`.
- Git Log:** Shows the commit history with a context menu open over the local branch `main`. The menu options are:
 - New Branch from 'master'...
 - Show Diff with Working Tree
 - Update
 - Push...
 - Rename...** (highlighted)
- Git Log Table:**

Commit	Author	Date
initial commit	hendalkittawi	2 minutes ago
Create README.md	hendalkittawi*	4 minutes ago
- Bottom Bar:** Shows various IDE tools like `Git`, `TODO`, `Problems`, `Terminal`, `App Inspection`, `Logcat`, `App Quality Insights`, `Services`, `Build`, and `Profiler`.

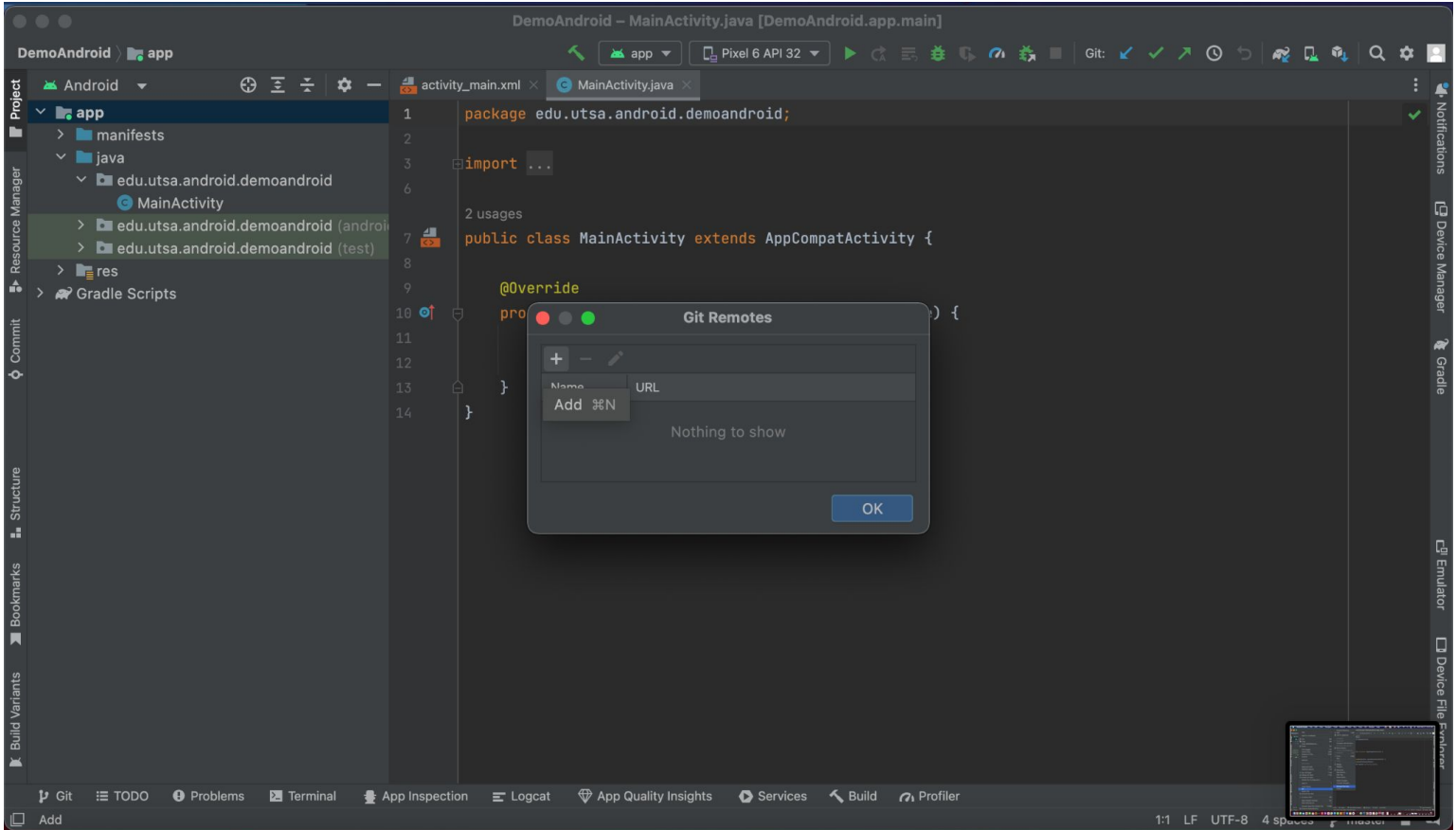
Copy the https URL for the remote repo

The screenshot shows a web browser window displaying a GitHub repository page for 'UTSA-CS-3443/DemoAndroid'. The browser's address bar shows the URL 'github.com/UTSA-CS-3443/DemoAndroid'. The page header includes navigation links for 'Pull requests', 'Issues', 'Codespaces', 'Marketplace', and 'Explore'. The repository name 'UTSA-CS-3443 / DemoAndroid' is displayed with a 'Private' label. Below the repository name, there are buttons for 'Edit Pins', 'Watch 3', 'Fork 0', and 'Star 0'. The main navigation bar includes 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. The 'Code' button is highlighted with a red border. A dropdown menu is open under 'Code', showing options for 'Local' and 'Codespaces (New)'. Under 'Local', there are three options: 'Clone', 'Open with GitHub Desktop', and 'Download ZIP'. The 'Clone' option is selected, and a sub-menu is open showing 'HTTPS', 'SSH', and 'GitHub CLI'. The 'HTTPS' option is selected, and the URL 'https://github.com/UTSA-CS-3443/DemoAndro' is displayed with a copy icon. The 'About' section on the right shows 'No description, website, or topics provided.' and '0 stars', '3 watching', and '0 forks'. The 'Releases' section shows 'No releases published' and a link to 'Create a new release'. The 'Packages' section is partially visible at the bottom.

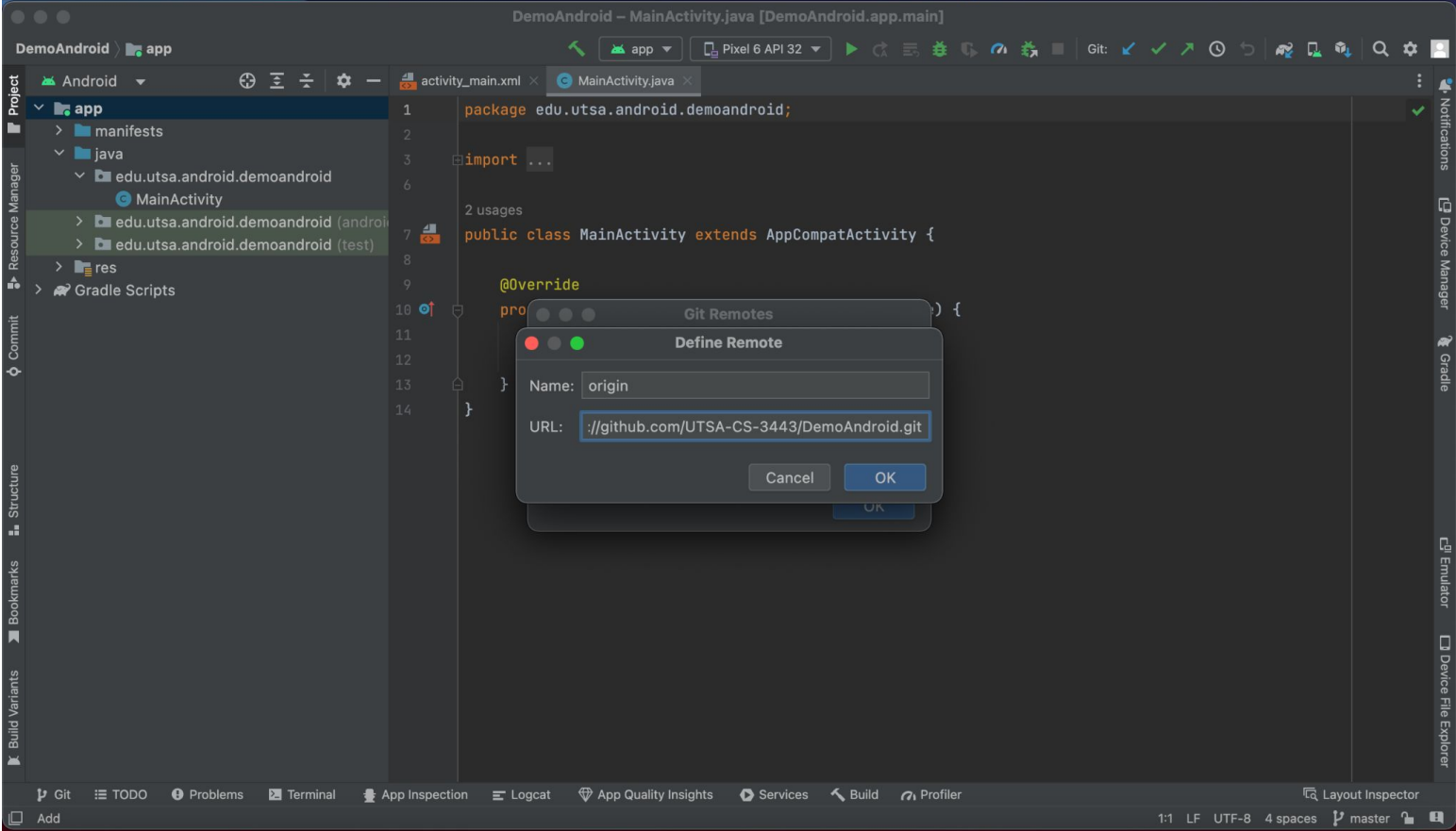
Add the remote repo



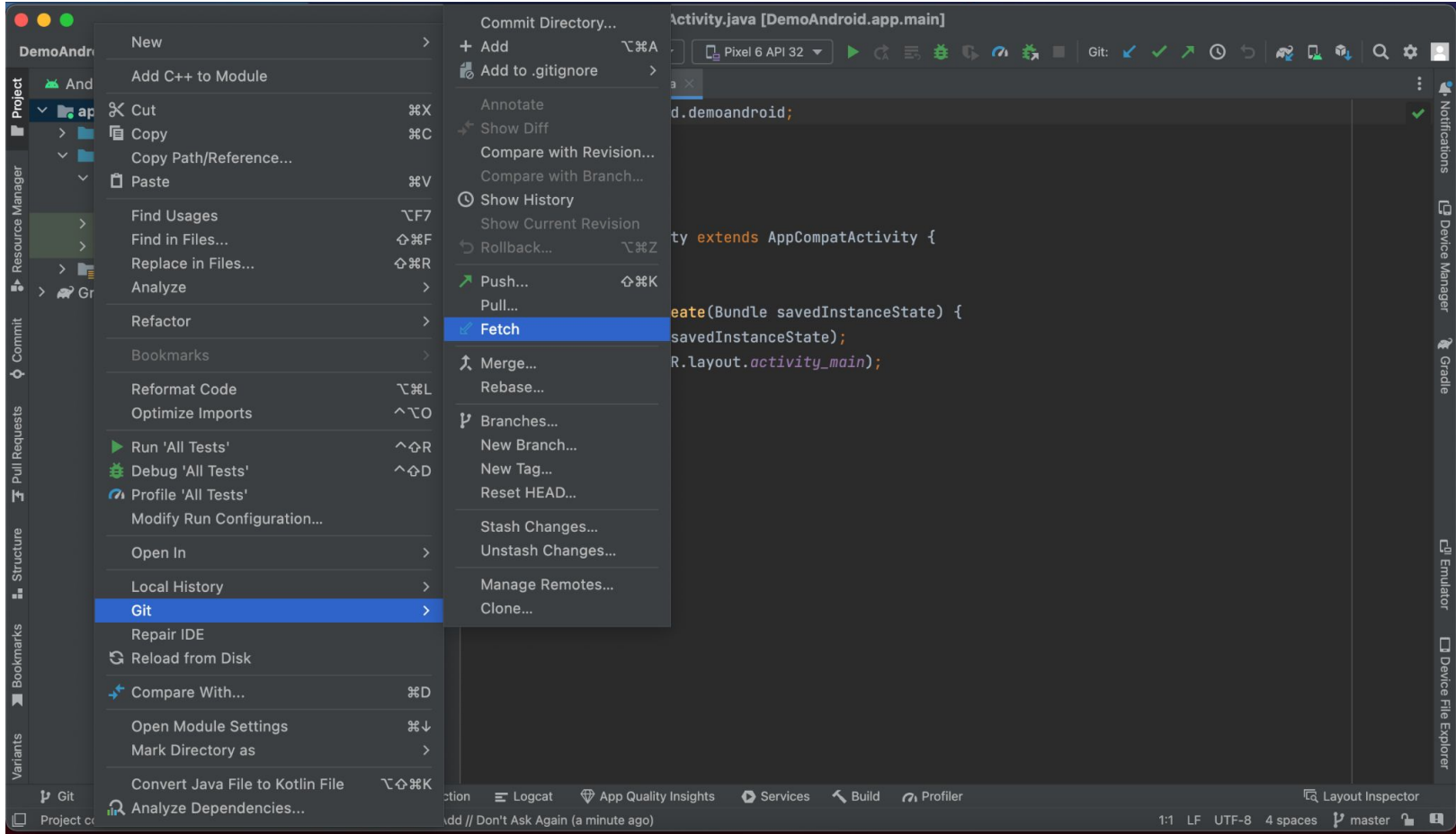
Add the remote repo



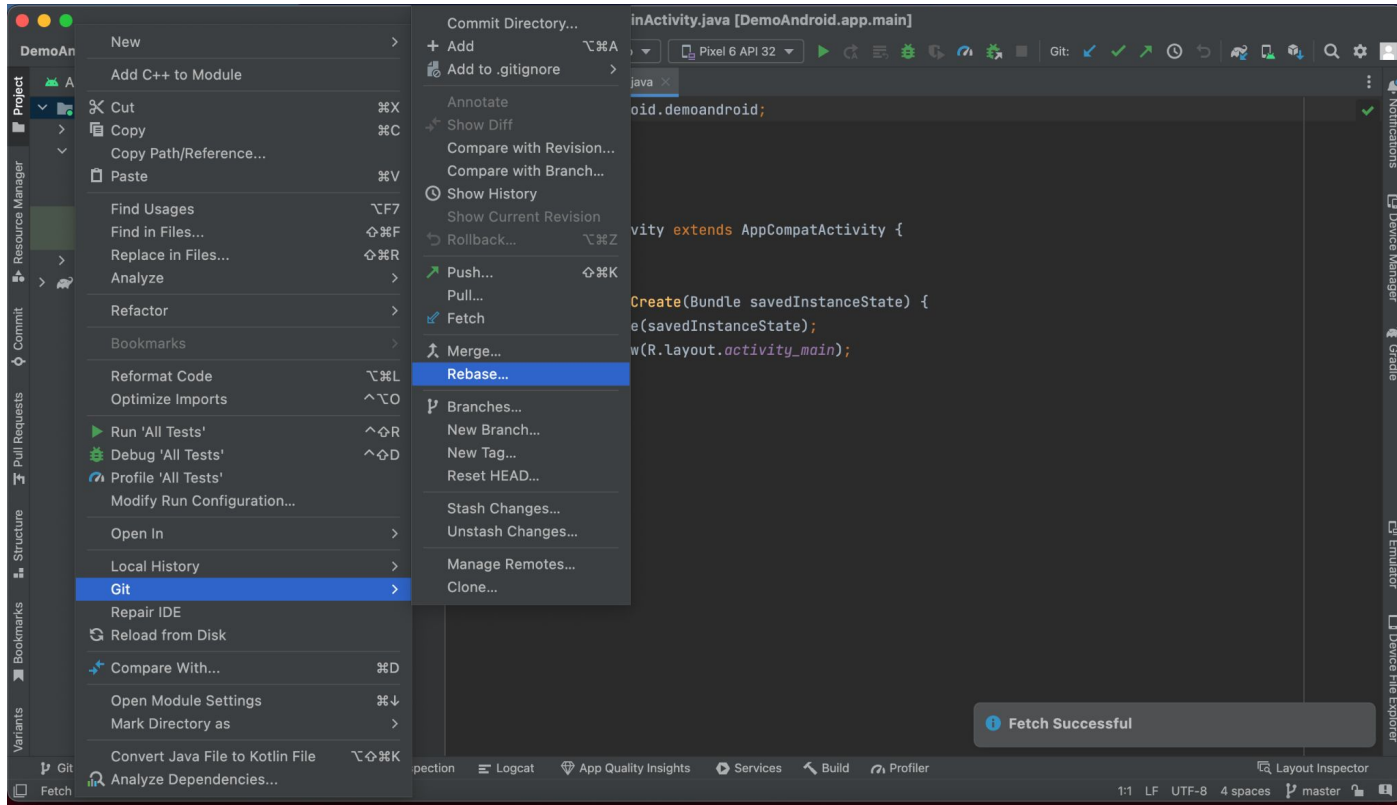
Add the remote repo



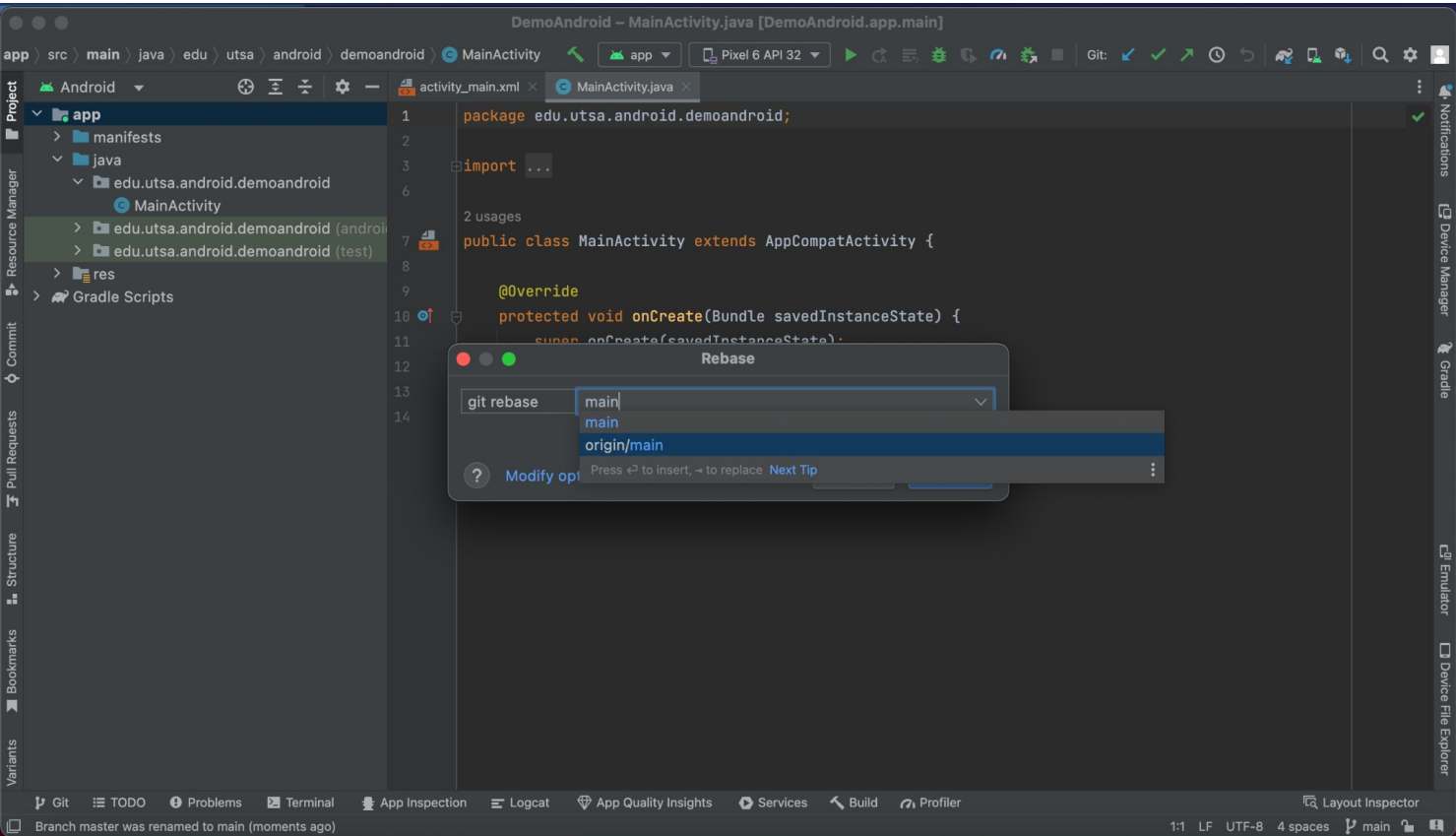
Fetch



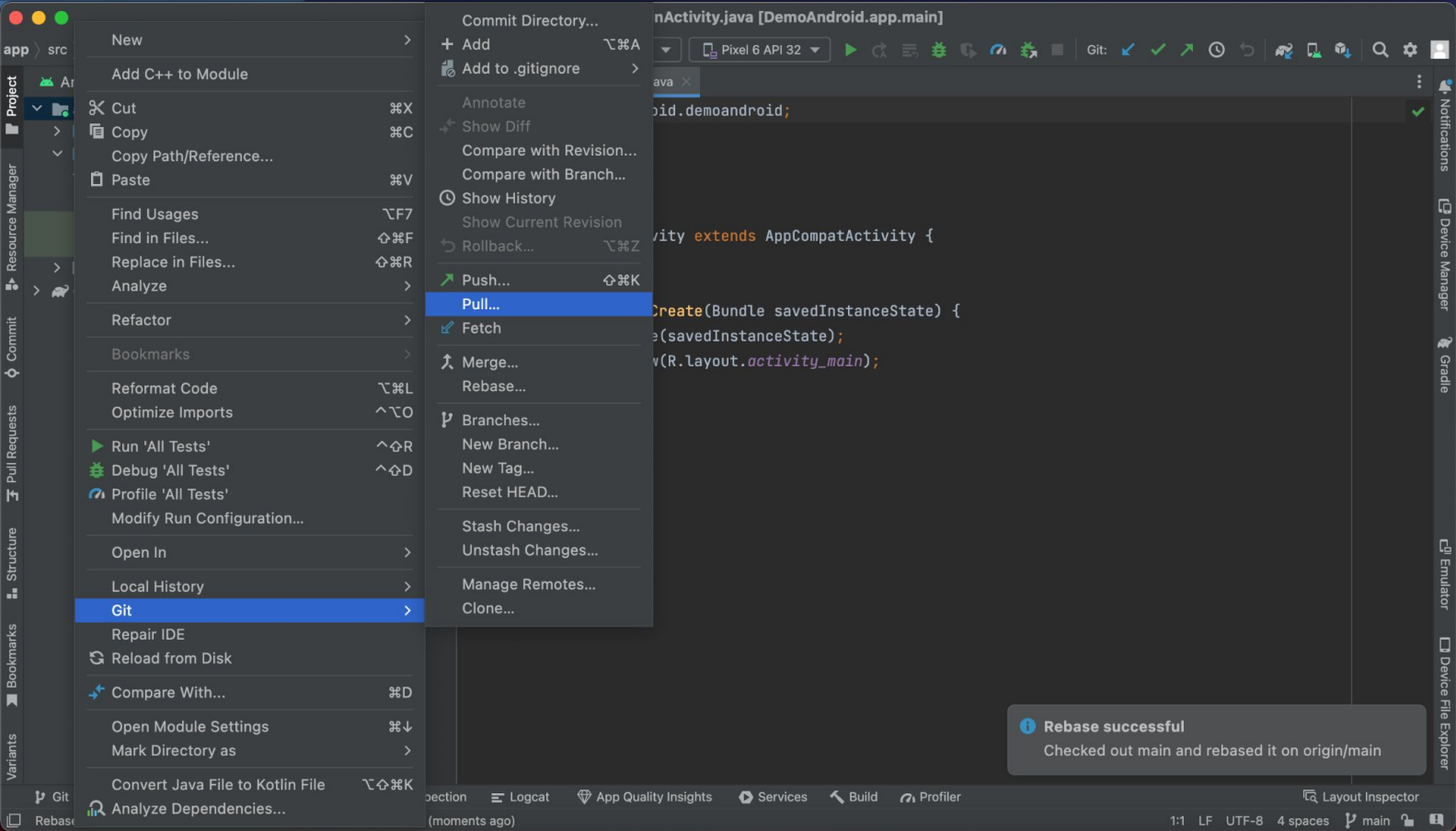
Note: Since we added a README to the remote repo we need Rebase before we Pull. You do not need to Rebase before you Pull if you did not create a README in the remote repo - you can skip slides 27 and 28.



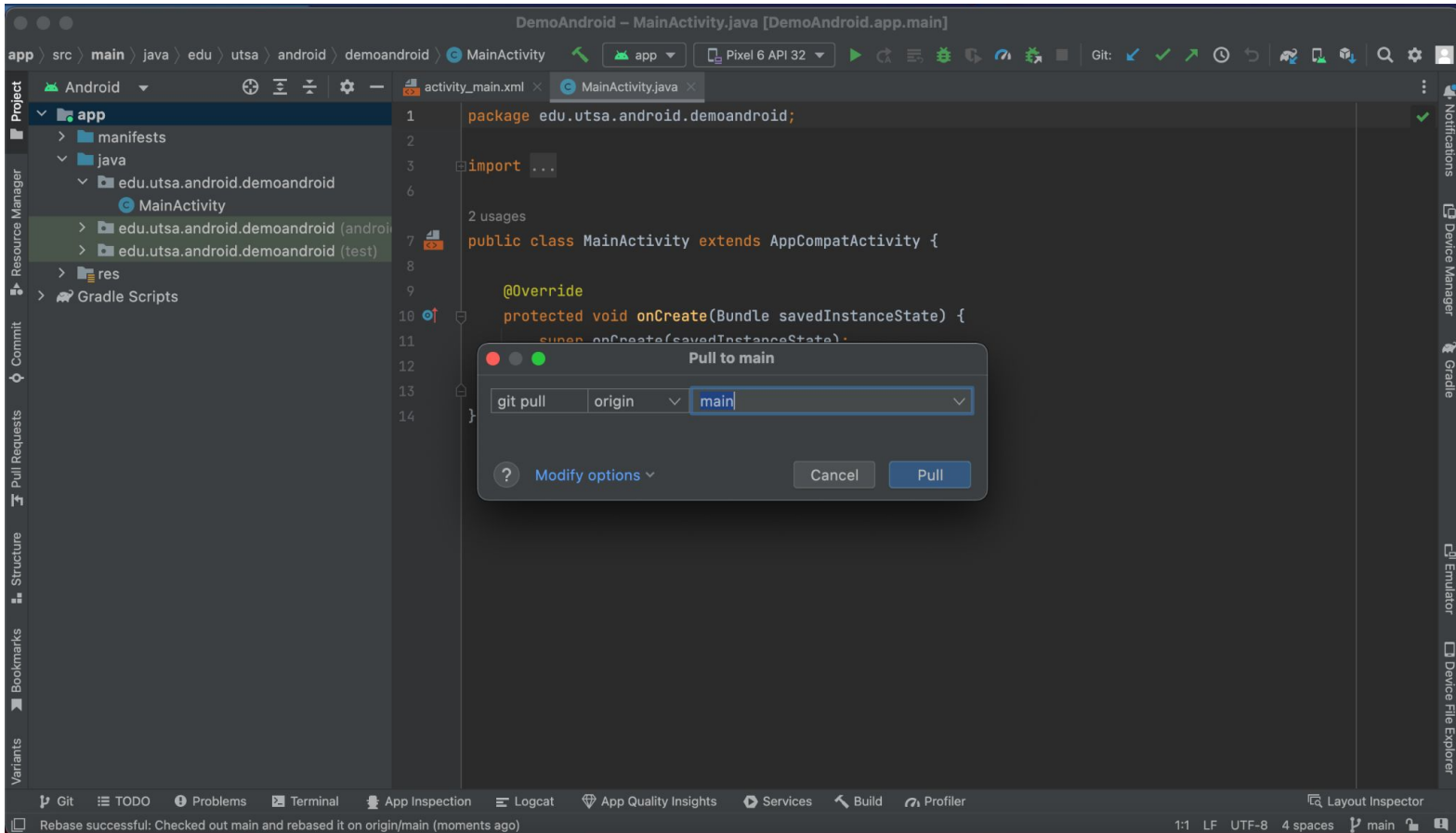
Rebase (we are just using it here because our local and remote repos initially do not have a “common history”)



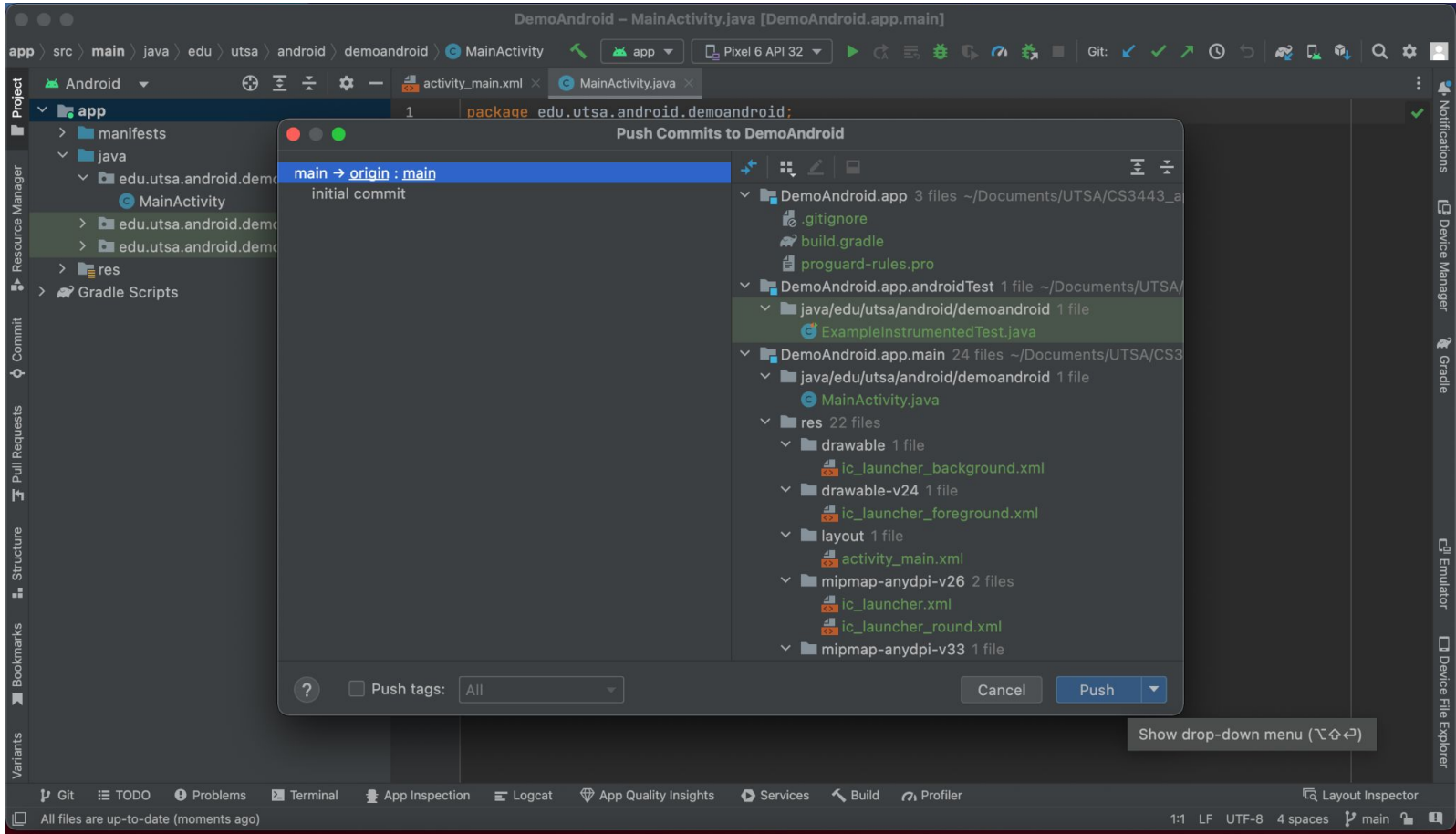
Pull from remote repo



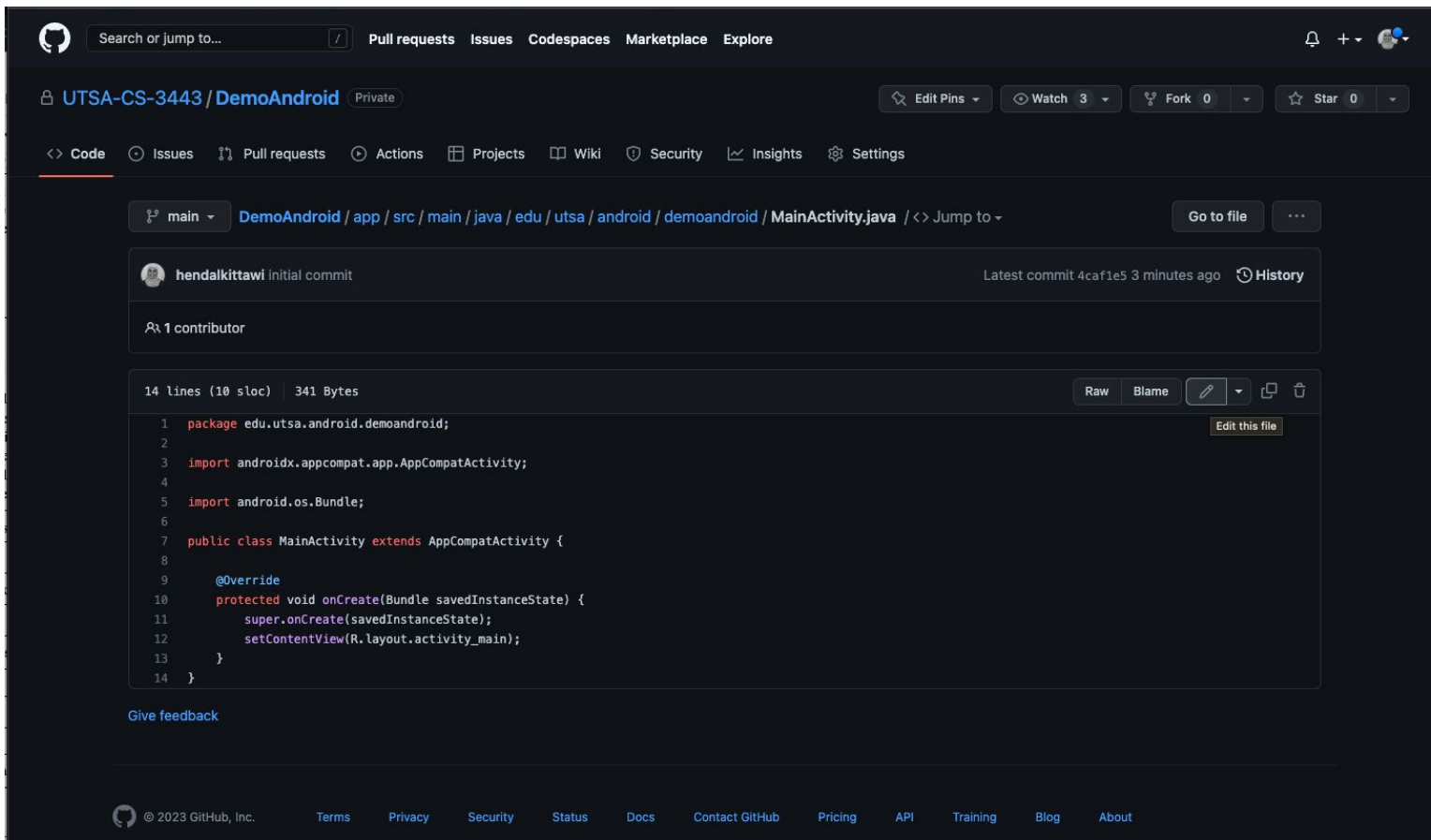
Pull from remote repo



Push to remote repo



Modify file(s) on Github (similar to if someone else modifies the files)

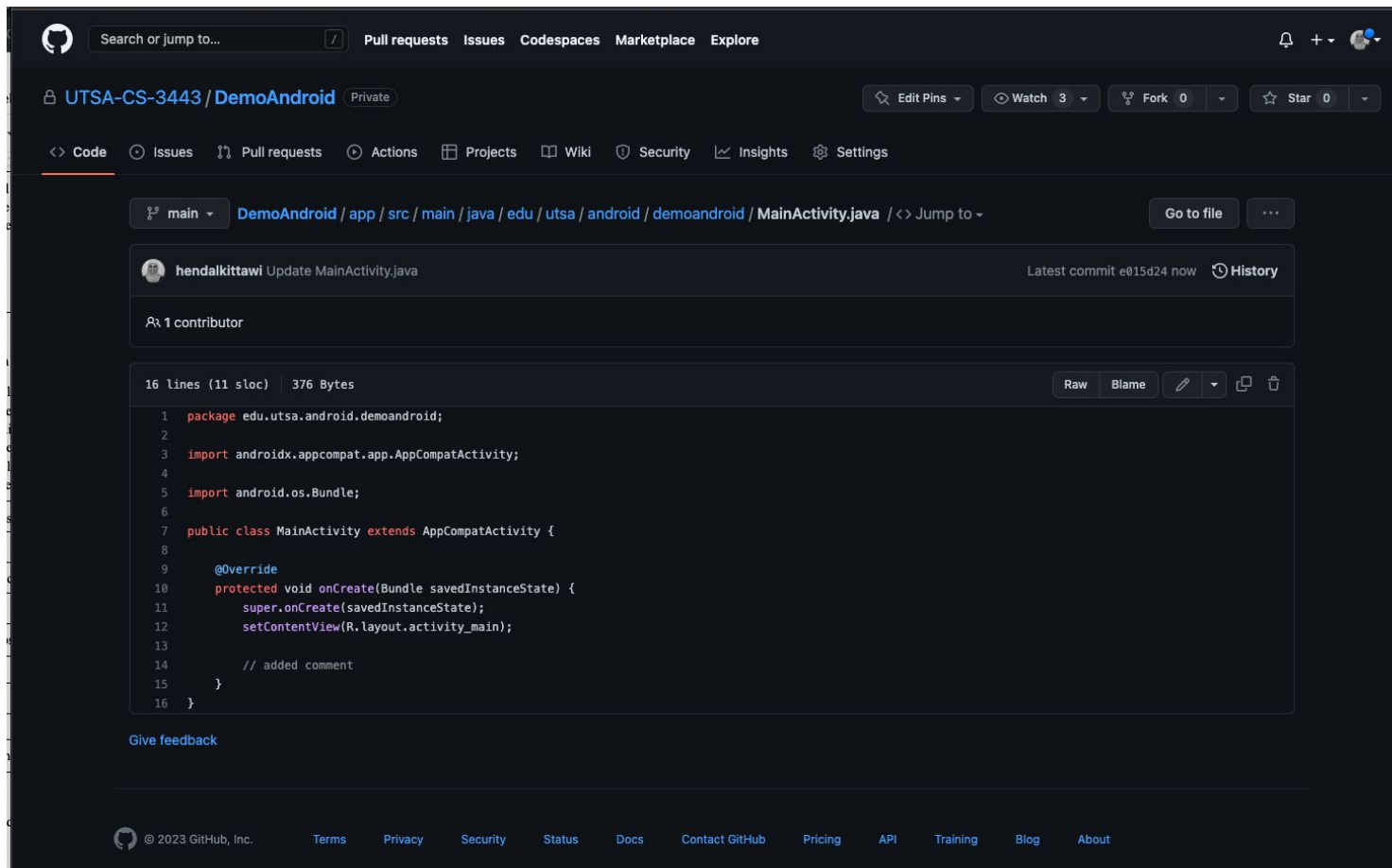


The screenshot shows the GitHub interface for a repository named "UTSA-CS-3443 / DemoAndroid". The page displays a commit by user "hendalkittawi" with the message "initial commit". Below the commit information, the code for the file "MainActivity.java" is shown. The code is a Java class that extends AppCompatActivity and includes an onCreate method. The code is as follows:

```
1 package edu.utsa.android.demoandroid;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.os.Bundle;
6
7 public class MainActivity extends AppCompatActivity {
8
9     @Override
10    protected void onCreate(Bundle savedInstanceState) {
11        super.onCreate(savedInstanceState);
12        setContentView(R.layout.activity_main);
13    }
14 }
```

The interface also includes navigation links for "Code", "Issues", "Pull requests", "Actions", "Projects", "Wiki", "Security", "Insights", and "Settings". At the bottom, there is a footer with copyright information and various links like "Terms", "Privacy", "Security", "Status", "Docs", "Contact GitHub", "Pricing", "API", "Training", "Blog", and "About".

Modify file(s) on Github (similar to if someone else modifies the files)



The screenshot displays the GitHub interface for a repository named "UTSA-CS-3443 / DemoAndroid". The page shows a commit by user "hendalkittawi" titled "Update MainActivity.java". The commit message is "Update MainActivity.java" and the latest commit hash is "e015d24". The commit is attributed to 1 contributor. The code file "MainActivity.java" is shown with 16 lines of Java code. The code includes package declarations, imports for AppCompatActivity and Bundle, and an override of the onCreate method. A comment "// added comment" is present on line 14. The interface includes navigation tabs for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The footer contains the GitHub logo, copyright information for 2023, and various links like Terms, Privacy, Security, Status, Docs, Contact GitHub, Pricing, API, Training, Blog, and About.

Search or jump to... Pull requests Issues Codespaces Marketplace Explore

UTSA-CS-3443 / DemoAndroid Private Edit Pins Watch 3 Fork 0 Star 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main DemoAndroid / app / src / main / java / edu / utsa / android / demoandroid / MainActivity.java /<> Jump to Go to file

hendalkittawi Update MainActivity.java Latest commit e015d24 now History

1 contributor

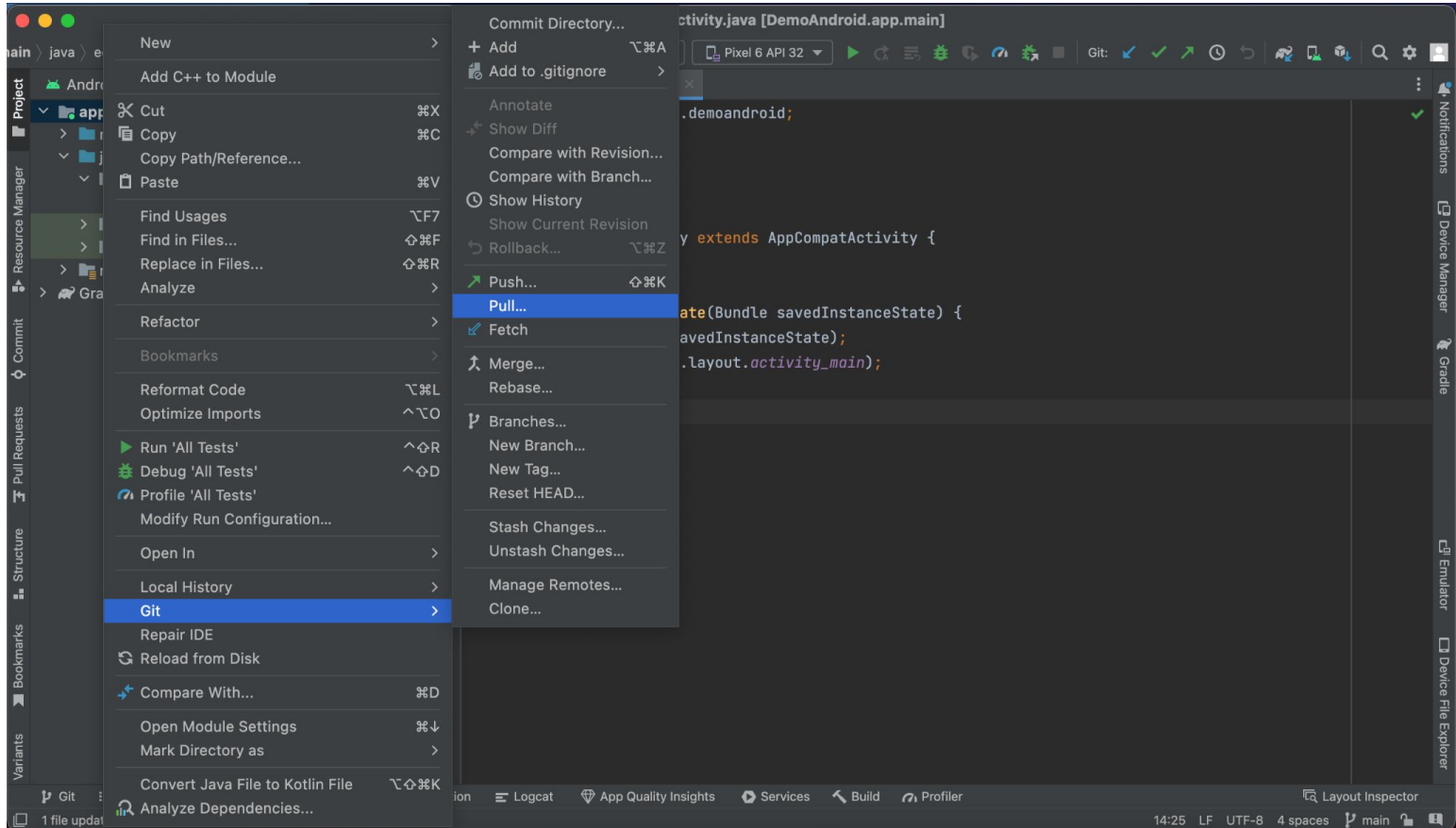
16 lines (11 sloc) 376 Bytes Raw Blame

```
1 package edu.utsa.android.demoandroid;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.os.Bundle;
6
7 public class MainActivity extends AppCompatActivity {
8
9     @Override
10    protected void onCreate(Bundle savedInstanceState) {
11        super.onCreate(savedInstanceState);
12        setContentView(R.layout.activity_main);
13
14        // added comment
15    }
16 }
```

Give feedback

© 2023 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About

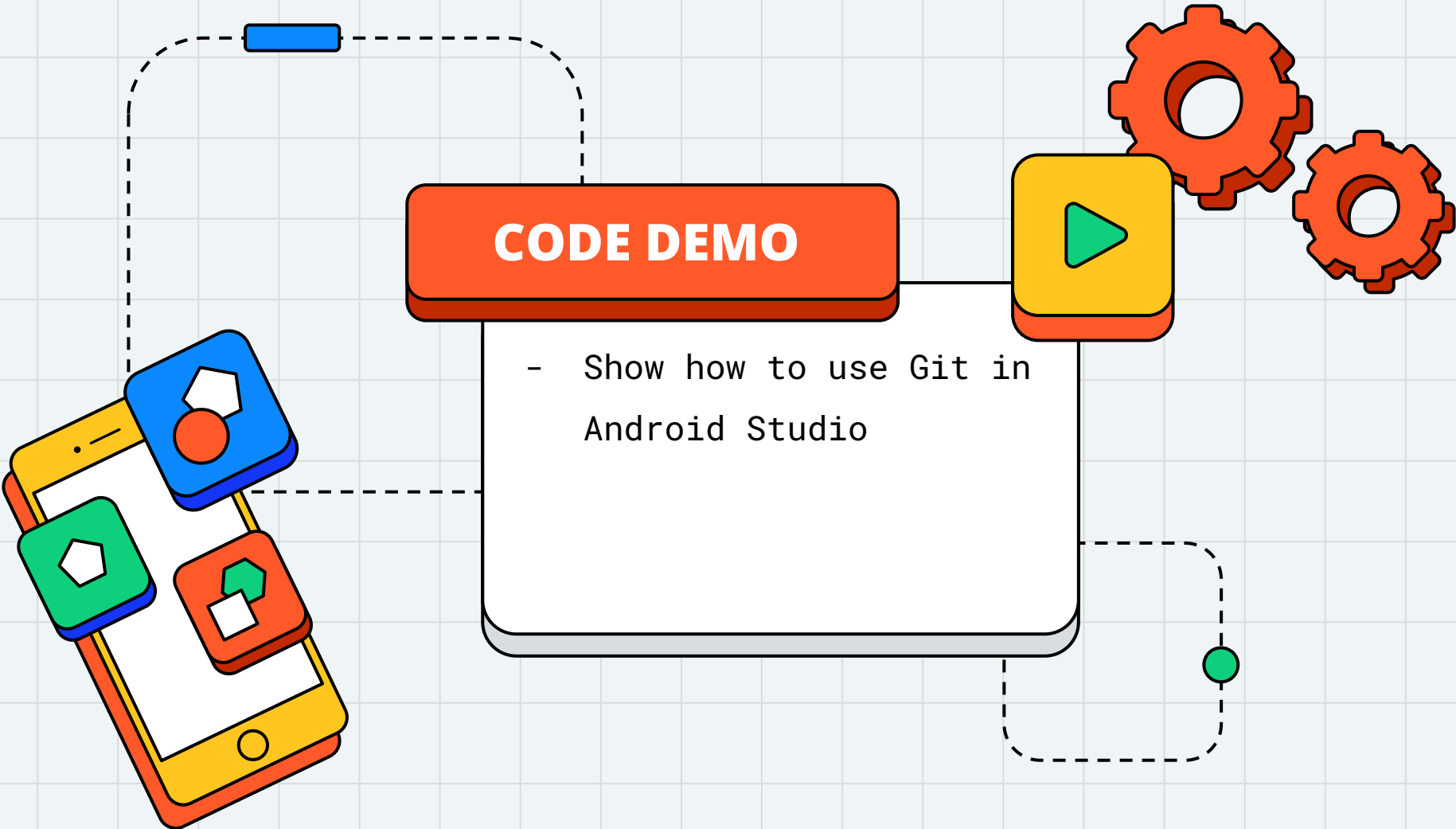
Pull changes to local repo



Modify file(s) locally

Stage, commit and push to remote repo (like above)

Check the remote repo



CODE DEMO

- Show how to use Git in Android Studio



THANK

YOU!



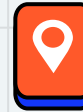
DO YOU HAVE ANY QUESTIONS?



hend.alkittawi@utsa.edu



By Appointment



Online