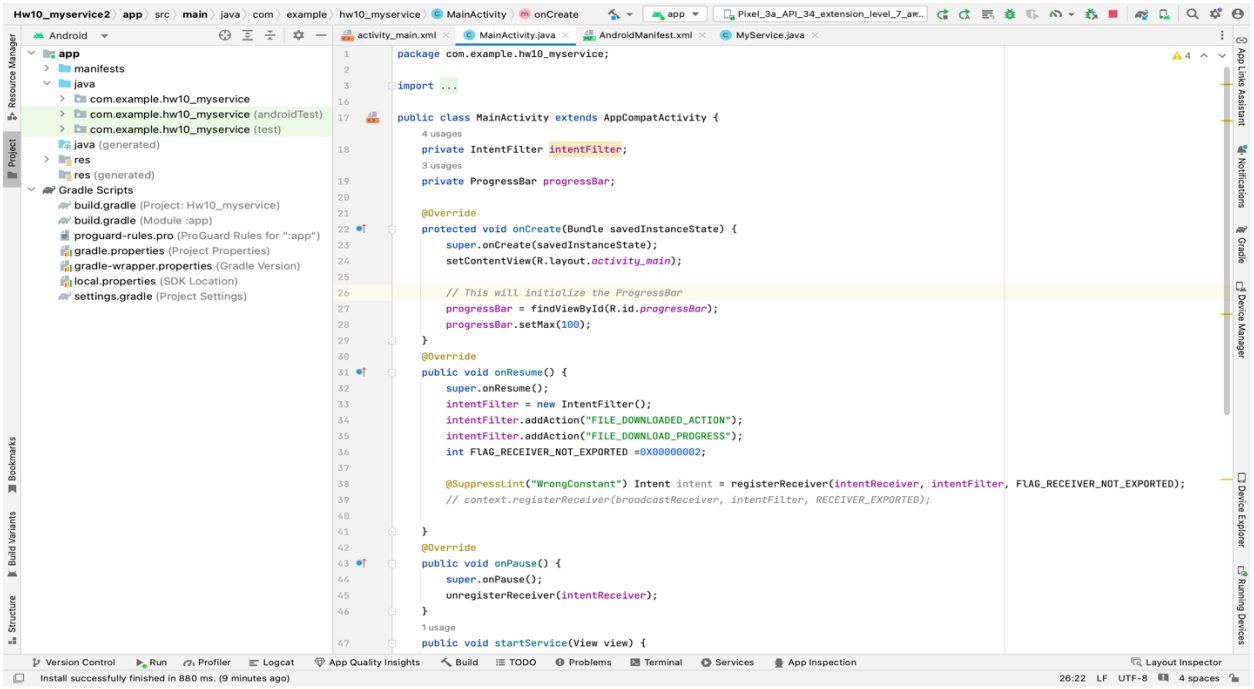
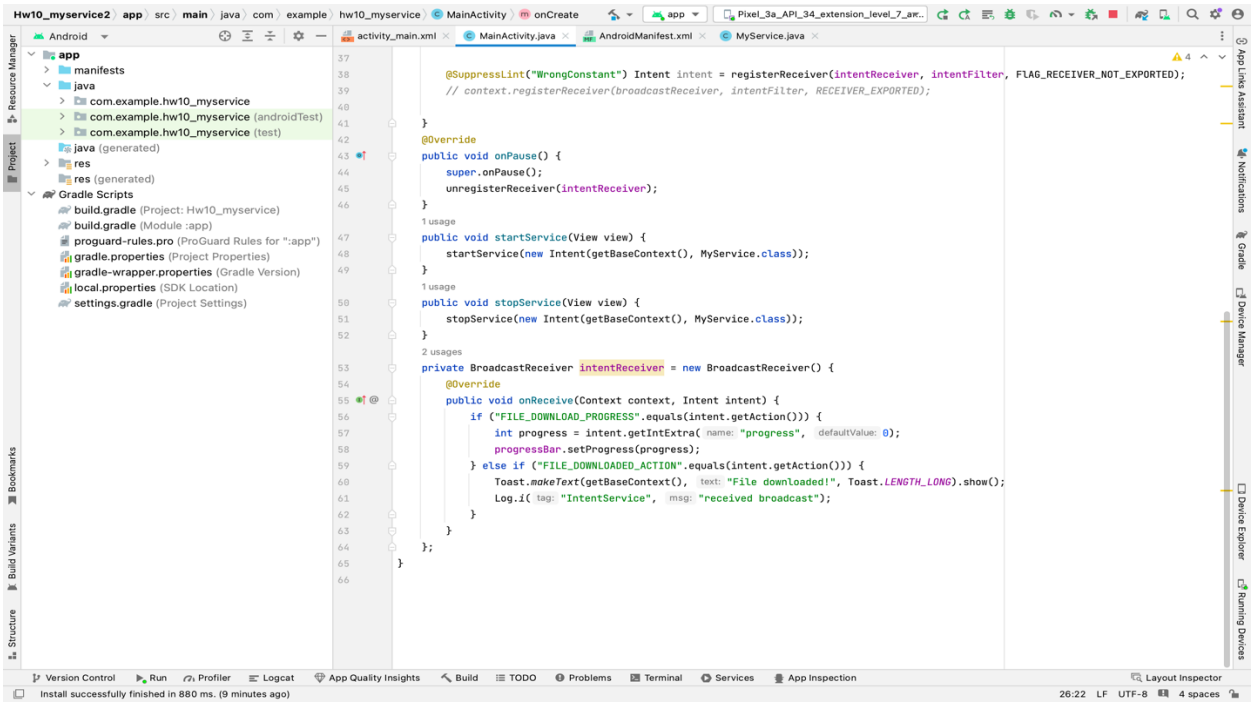
Modify the Services app: Use a Progress Bar view instead/in addition to the Toast to display the progress made by the service.

MainActivity.java : This class extends 'AppCompatActivity' in the Android app's 'MainActivity.java' and manages the UI and service interactions. It creates a 'ProgressBar' to display download progress and handles 'MyService' start/stop via buttons. When a file download is complete, the activity registers a 'BroadcastReceiver' to update the progress bar and display a toast. A manually defined flag for broadcast receiver registration ensures compatibility with API levels 31 and above.

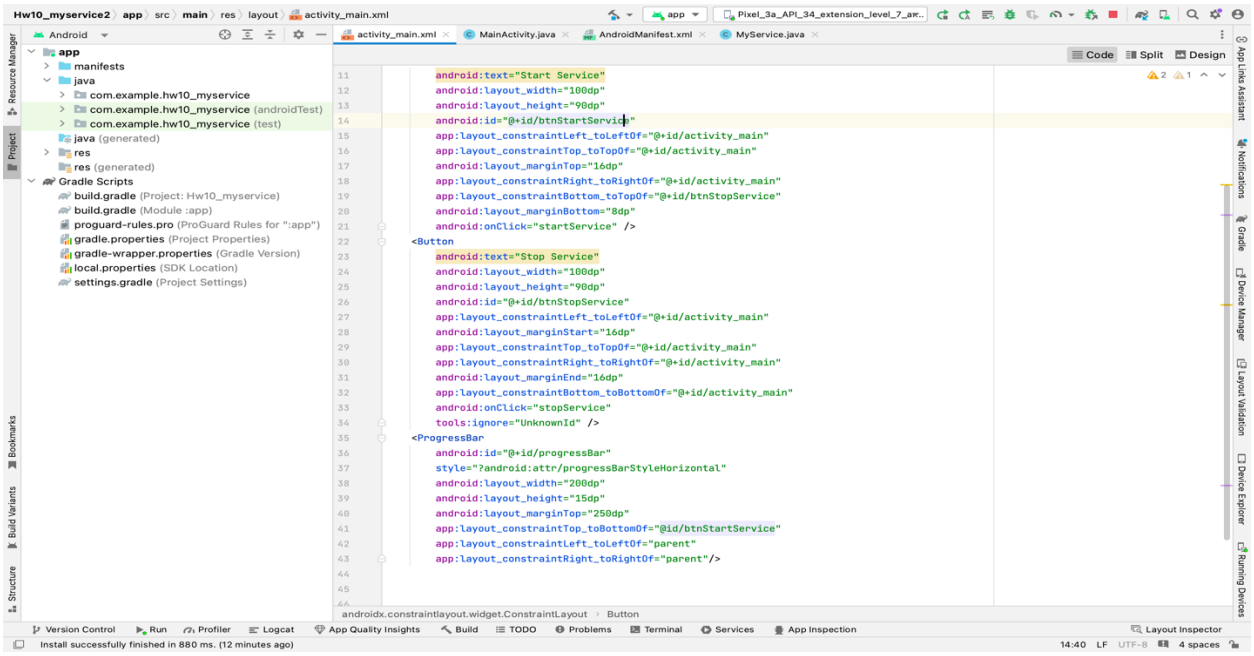


```
package com.example.hw10_myservice;

import ...

public class MainActivity extends AppCompatActivity {
    4 usages
    private IntentFilter intentFilter;
    3 usages
    private ProgressBar progressBar;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        // This will initialize the ProgressBar
        progressBar = findViewById(R.id.progressBar);
        progressBar.setMax(100);
    }
    @Override
    public void onResume() {
        super.onResume();
        intentFilter = new IntentFilter();
        intentFilter.addAction("FILE_DOWNLOADED_ACTION");
        intentFilter.addAction("FILE_DOWNLOAD_PROGRESS");
        int FLAG_RECEIVER_NOT_EXPORTED =0X00000002;

        @SuppressLint("WrongConstant") Intent intent = registerReceiver(intentReceiver, intentFilter, FLAG_RECEIVER_NOT_EXPORTED);
        // context.registerReceiver(broadcastReceiver, intentFilter, RECEIVER_EXPORTED);

    }
    @Override
    public void onPause() {
        super.onPause();
        unregisterReceiver(intentReceiver);
    }
    1 usage
    public void startService(View view) {
```

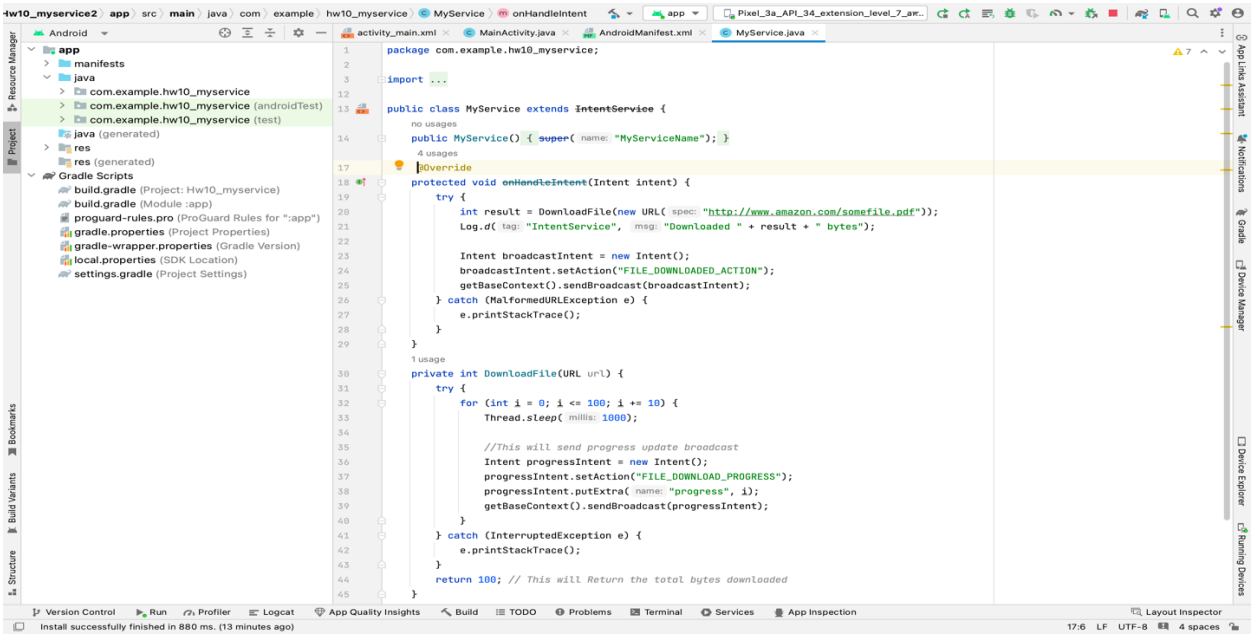


```
        @SuppressLint("WrongConstant") Intent intent = registerReceiver(intentReceiver, intentFilter, FLAG_RECEIVER_NOT_EXPORTED);
        // context.registerReceiver(broadcastReceiver, intentFilter, RECEIVER_EXPORTED);

    }
    @Override
    public void onPause() {
        super.onPause();
        unregisterReceiver(intentReceiver);
    }
    1 usage
    public void startService(View view) {
        startService(new Intent(getBaseContext(), MyService.class));
    }
    1 usage
    public void stopService(View view) {
        stopService(new Intent(getBaseContext(), MyService.class));
    }
    2 usages
    private BroadcastReceiver intentReceiver = new BroadcastReceiver() {
        @Override
        public void onReceive(Context context, Intent intent) {
            if ("FILE_DOWNLOAD_PROGRESS".equals(intent.getAction())) {
                int progress = intent.getIntExtra( name: "progress",  defaultValue: 0);
                progressBar.setProgress(progress);
            } else if ("FILE_DOWNLOADED_ACTION".equals(intent.getAction())) {
                Toast.makeText(getBaseContext(),  text: "File downloaded!", Toast.LENGTH_LONG).show();
                Log.i( tag: "IntentService",  msg: "received broadcast");
            }
        }
    };
}
```

Activity_main.xml : 'ConstraintLayout' is used in this XML layout to establish the user interface for 'MainActivity' in an Android app. It has two buttons for initiating and stopping a service, each with certain dimensions and location requirements. In addition, a horizontal 'ProgressBar' is centrally aligned and positioned below the start service button to graphically display the progress of a task.

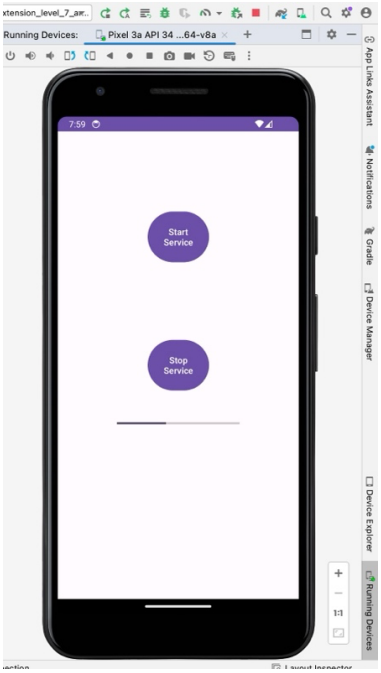


Myservice.java : In the Android app, 'MyService' is a 'IntentService' that handles background tasks. It simulates downloading a file by looping through it and pauses at intervals, during which it broadcasts progress updates using the 'FILE_DOWNLOAD_PROGRESS' action. When the simulated download is complete, it sends a broadcast with the 'FILE_DOWNLOADED_ACTION' action. The main methods are 'onHandleIntent,' which orchestrates the download process and broadcasts completion, and 'DownloadFile,' which mimics the file download and broadcasts the progress on a regular basis.



Below are the screenshots of the app running in the emulator :

SS1 (Main activity).



SS2 (progress bar half way)

SS3 (file downloaded)