HW10:

Services

Martin Mudenda Bbela

2582912

Overview

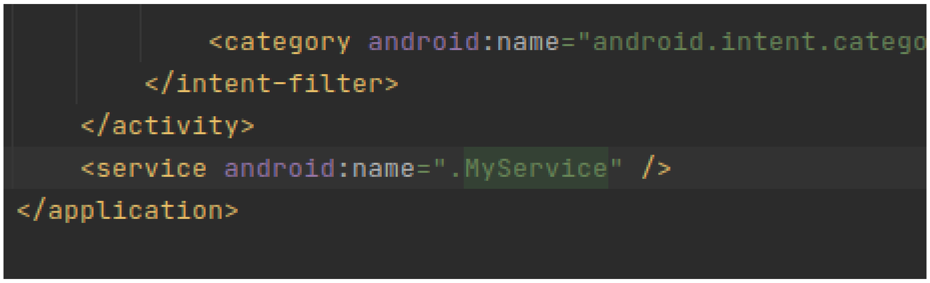
Modify the Services app:

- Use a ProgressBar view instead/in addition to the Toast to display the progress made by the service

Method

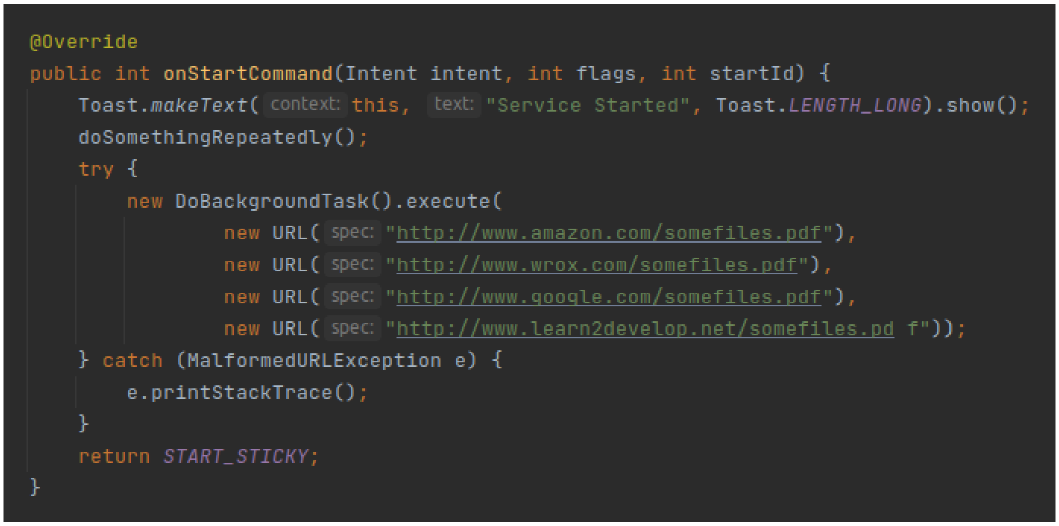
MyService

The service is refrenced in the Manifest so we can listen to calls from it



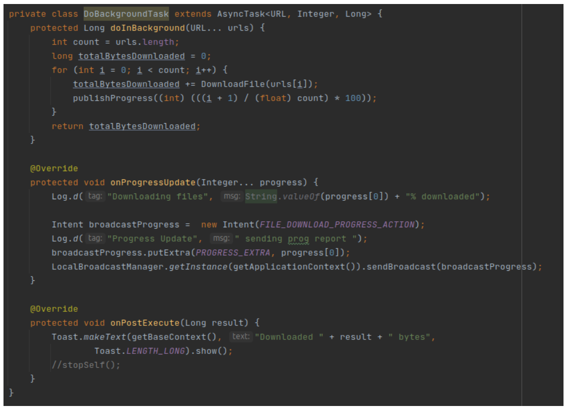
```
            <category android:name="android.intent.catego
        </intent-filter>
    </activity>
    <service android:name=".MyService" />
</application>
```

We create a service which will be started by the Main activity that will will wait and send progress to the Main activity.



```
@Override
public int onStartCommand(Intent intent, int flags, int startId) {
    Toast.makeText( context: this,  text: "Service Started", Toast.LENGTH_LONG).show();
    doSomethingRepeatedly();
    try {
        new DoBackgroundTask().execute(
                new URL( spec: "http://www.amazon.com/somefiles.pdf"),
                new URL( spec: "http://www.wrox.com/somefiles.pdf"),
                new URL( spec: "http://www.google.com/somefiles.pdf"),
                new URL( spec: "http://www.learn2develop.net/somefiles.pd f"));
    } catch (MalformedURLException e) {
        e.printStackTrace();
    }
    return START_STICKY;
}
```

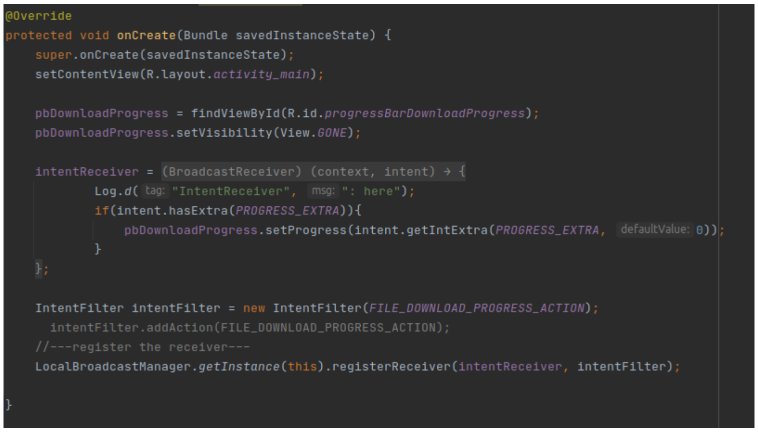
The DoBackGroundTask() is a custom Async Task and internally will send progress to the main activity.



```
private class DoBackgroundTask extends AsyncTask<URL, Integer, Long> {
    protected Long doInBackground(URL... urls) {
        int count = urls.length;
        long totalBytesDownloaded = 0;
        for (int i = 0; i < count; i++) {
            totalBytesDownloaded += DownloadFile(urls[i]);
            publishProgress((int) (((i + 1) / (float) count) * 100));
        }
        return totalBytesDownloaded;
    }

    @Override
    protected void onProgressUpdate(Integer... progress) {
        Log.d( tag: "Downloading files",  msg: String.valueOf(progress[0]) + "% downloaded");

        Intent broadcastProgress =  new Intent(FILE_DOWNLOAD_PROGRESS_ACTION);
        Log.d( tag: "Progress Update",  msg: " sending prog report ");
        broadcastProgress.putExtra(PROGRESS_EXTRA, progress[0]);
        LocalBroadcastManager.getInstance(getApplicationContext()).sendBroadcast(broadcastProgress);
    }

    @Override
    protected void onPostExecute(Long result) {
        Toast.makeText(getBaseContext(),  text: "Downloaded " + result + " bytes",
                Toast.LENGTH_LONG).show();
        //stopSelf();
    }
}
```

Main Activity

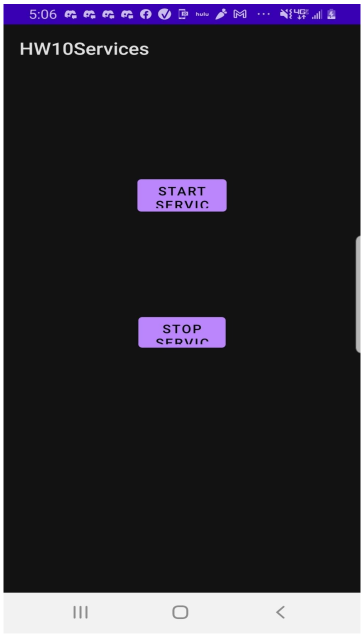
The Main Activity is simply 2 buttons and a progress Bar. We hide the Progress Bar on Create and only make it visible when the the service has started.

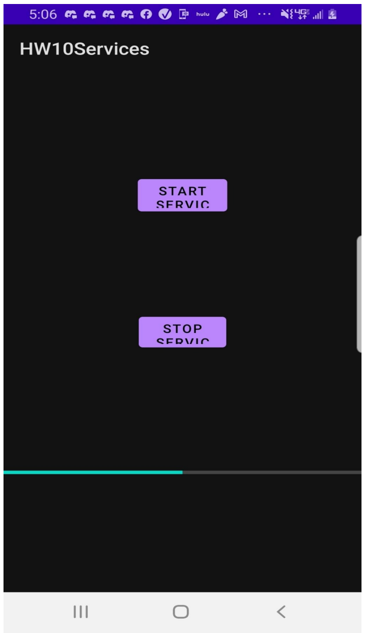


```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    pbDownloadProgress = findViewById(R.id.progressBarDownloadProgress);
    pbDownloadProgress.setVisibility(View.GONE);

    intentReceiver = (BroadcastReceiver) (context, intent) → {
            Log.d( tag: "IntentReceiver",  msg: ": here");
            if(intent.hasExtra(PROGRESS_EXTRA)){
                pbDownloadProgress.setProgress(intent.getIntExtra(PROGRESS_EXTRA,  defaultValue: 0));
            }
    };

    IntentFilter intentFilter = new IntentFilter(FILE_DOWNLOAD_PROGRESS_ACTION);
      intentFilter.addAction(FILE_DOWNLOAD_PROGRESS_ACTION);
    //---register the receiver---
    LocalBroadcastManager.getInstance(this).registerReceiver(intentReceiver, intentFilter);

}
```

Result

The user is able to start the service and see the download progress in the Main activty.


5:06
HW10Services
START SERVIC
STOP SERVIC


5:06
HW10Services
START SERVIC
STOP SERVIC