

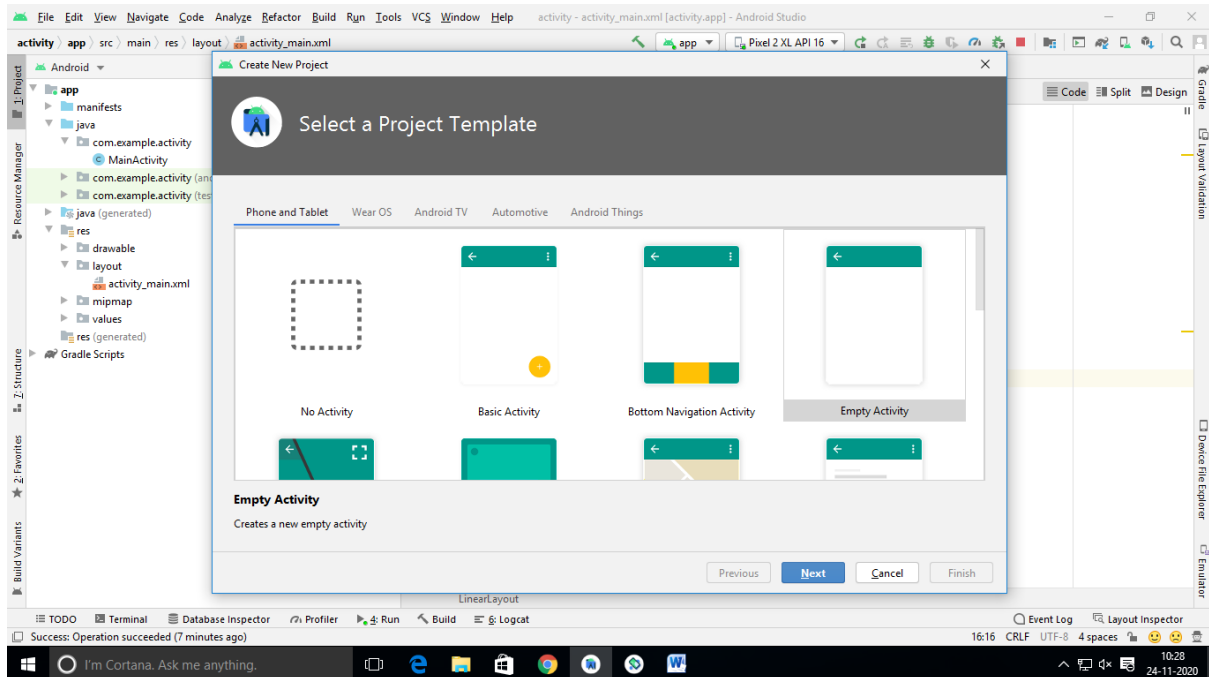
3. Create an application that displays the progress of an operation.

Program:

Step-1: go to android studio

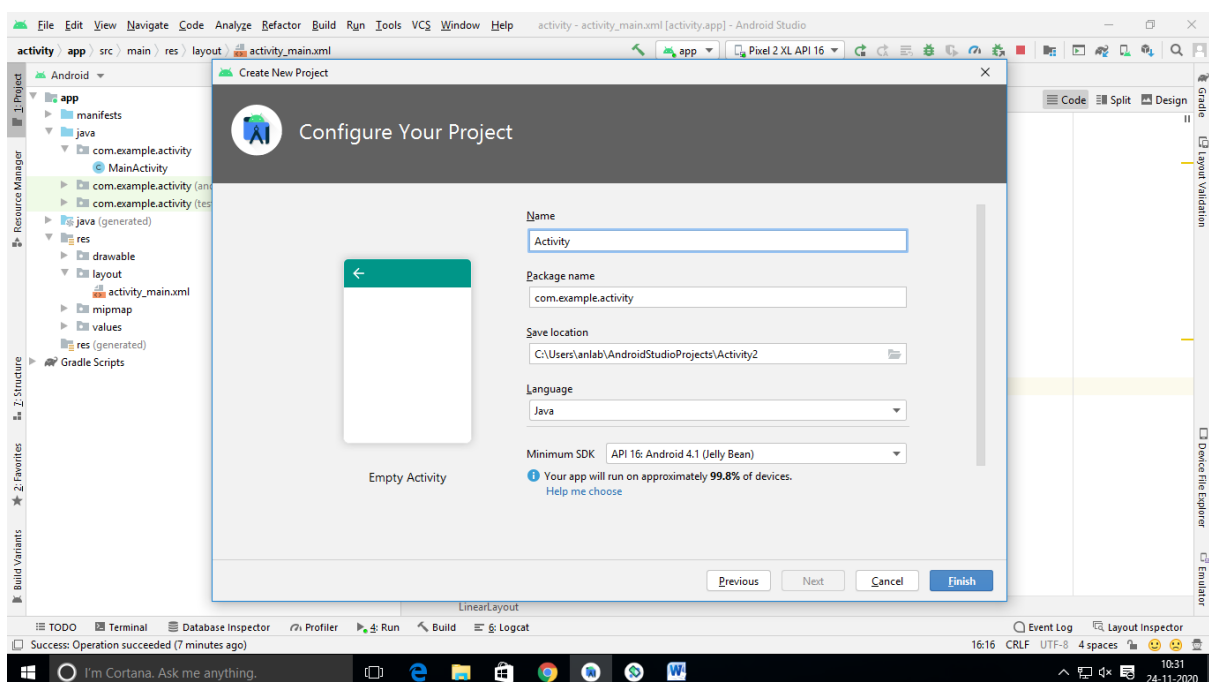
File -> New -> New project

Step-2:



Go to empty activity.

Step-3:



- ➔ Give name progress bar.
- ➔ Click on finish.

Step-4:

- ➔ Write code on activity\_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/button"
        android:layout_width="208dp"
        android:layout_height="59dp"
        android:text="Download File"
        tools:layout_editor_absoluteX="92dp"
        tools:layout_editor_absoluteY="292dp" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

- ➔ Write code on MainActivity.java

```
package com.example.progressbar;

import androidx.appcompat.app.AppCompatActivity;
import android.app.ProgressDialog;
import android.os.Handler;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {
    Button btnStartProgress;
    ProgressDialog progressBar;
    private int progressBarStatus = 0;
    private Handler progressBarHandler = new Handler();
    private long fileSize = 0;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        addListenerOnButtonClick();
    }
    public void addListenerOnButtonClick() {
        btnStartProgress = findViewById(R.id.button);
        btnStartProgress.setOnClickListener(new View.OnClickListener(){

            @Override
            public void onClick(View v) {
                // creating progress bar dialog
                progressBar = new ProgressDialog(v.getContext());
                progressBar.setCancelable(true);
```

```

        progressBar.setMessage("File downloading ...");

        progressBar.setProgressStyle(ProgressDialog.STYLE_HORIZONTAL);
        progressBar.setProgress(0);
        progressBar.setMax(100);
        progressBar.show();
        //reset progress bar and filesize status
        progressBarStatus = 0;
        fileSize = 0;

        new Thread(new Runnable() {
            public void run() {
                while (progressBarStatus < 100) {
                    // performing operation
                    progressBarStatus = doOperation();
                    try {
                        Thread.sleep(1000);
                    } catch (InterruptedException e) {
                        e.printStackTrace();
                    }
                    // Updating the progress bar
                    progressBarHandler.post(new Runnable() {
                        public void run() {

progressBar.setProgress(progressBarStatus);
                        }
                    });
                }
                // performing operation if file is downloaded,
                if (progressBarStatus >= 100) {
                    // sleeping for 1 second after operation
completed
                    try {
                        Thread.sleep(1000);
                    } catch (InterruptedException e) {
                        e.printStackTrace();
                    }
                    // close the progress bar dialog
                    progressBar.dismiss();
                }
            }
        }).start();
    } //end of onClick method
});
}
// checking how much file is downloaded and updating the filesize
public int doOperation() {
    //The range of ProgressDialog starts from 0 to 10000
    while (fileSize <= 10000) {
        fileSize++;
        if (fileSize == 1000) {
            return 10;
        } else if (fileSize == 2000) {
            return 20;
        } else if (fileSize == 3000) {
            return 30;
        } else if (fileSize == 4000) {
            return 40; // you can add more else if
        }
    }
}

```

```
        /* else {  
            return 100;  
        }*/  
    }//end of while  
    return 100;  
}//end of doOperation  
}
```

Step-5:

➔ Click on debug and run the project

Step-6:

Outputs:

