|  |  |
| --- | --- |
| **Ex No: 7** | DOWNLOAD IMAGE USING ASYNCHRONOUS |
| **Date:** |  |

**AIM:**

To develop an application for downloading image asynchronously using Android Studio.

**DESCRIPTION:**

The Asynchronous Image Download Application in Android Studio is a user-friendly tool for efficiently downloading images from the internet while maintaining a responsive and seamless user experience. At its core, this application leverages the AsyncTask class, a pivotal component in Android development, to carry out image downloads asynchronously. The AsyncTask class provides several key methods, including doInBackground(), which is used to perform the actual image download from a specified URL in a background thread. This crucial feature ensures that time-consuming network operations do not block the main UI thread, preventing the app from becoming unresponsive.

In addition to doInBackground(), the application also takes advantage of other AsyncTask methods such as onPreExecute(), which allows for any necessary initialization or the display of loading indicators before the background task starts. The onProgressUpdate() method is utilized to update the user interface with progress information during the download process, making it possible to implement features like progress bars. Finally, onPostExecute() plays a crucial role in managing the downloaded image, handling any potential errors, or executing post-processing tasks. Together, these elements work in harmony to create a powerful and efficient image downloading application for Android users.

**PROGRAM:**

MainActivity.java:

package com.example.madex7;

import androidx.appcompat.app.AppCompatActivity;

import android.app.ProgressDialog;

import android.graphics.Bitmap;

import android.graphics.BitmapFactory;

import android.os.AsyncTask;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.ImageView;

import java.io.IOException;

import java.io.InputStream;

import java.net.HttpURLConnection;

import java.net.URL;

public class MainActivity extends AppCompatActivity {

URL ImageUrl = null;

InputStream is = null;

Bitmap bmImg = null;

ImageView image;

ProgressDialog p;

Button download;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

download = findViewById(R.id.button);

image = findViewById(R.id.imageView);

download.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

AsyncTaskExample asyncTask = new AsyncTaskExample();

asyncTask.execute("https://p0.pxfuel.com/preview/875/609/139/butterfly-flower-nature-plant.jpg");

}});}

private class AsyncTaskExample extends AsyncTask<String, String, Bitmap>{

@Override

protected void onPreExecute(){

super.onPreExecute();

p = new ProgressDialog(MainActivity.this);

p.setMessage("Image is being Downloaded");

p.setIndeterminate(false);

p.setCancelable(false);

p.show();

}

@Override

protected Bitmap doInBackground(String... strings){

try{

ImageUrl = new URL(strings[0]);

HttpURLConnection conn = (HttpURLConnection) ImageUrl.openConnection();

conn.setDoInput(true);

conn.connect();

is = conn.getInputStream();

BitmapFactory.Options options = new BitmapFactory.Options();

options.inPreferredConfig = Bitmap.Config.RGB\_565;

bmImg = BitmapFactory.decodeStream(is, null, options);

}catch(IOException e){

e.printStackTrace();

}

return bmImg;

}

@Override

protected void onPostExecute(Bitmap bitmap) {

super.onPostExecute(bitmap);

if(image != null){

p.hide();

image.setImageBitmap(bitmap);

}

else{

p.show();

} } }

}

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="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/download"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.498"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.284" />

<ImageView

android:id="@+id/imageView"

android:layout\_width="410dp"

android:layout\_height="441dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

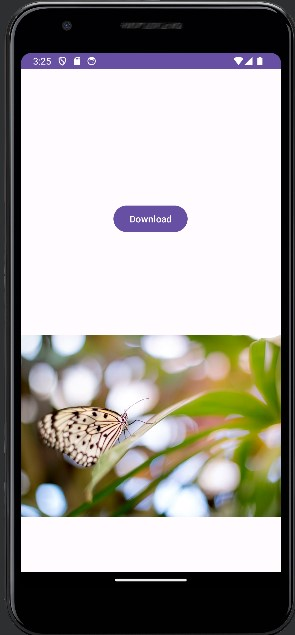
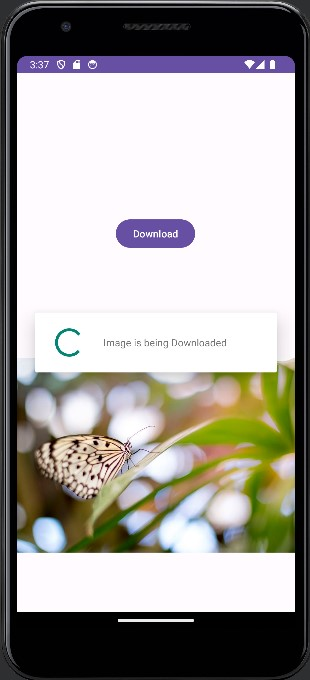
app:layout\_constraintVertical\_bias="1.0"

app:srcCompat="@drawable/butterfly"

android:contentDescription="@string/butterfly" />

</androidx.constraintlayout.widget.ConstraintLayout>

**OUTPUT:**

**RESULT:**

Thus, the application for downloading image asynchronously using Android studio has been executed successfully and the output has been verified.