android studio

android APP DEVELOPMENT

LAB MANUAL

Dr. Zeeshan Bhatti

Android Mobile Application Development (Using Android Studio 2021)

LAB Manual (60 Marks)
By: Dr. Zeeshan Bhatti

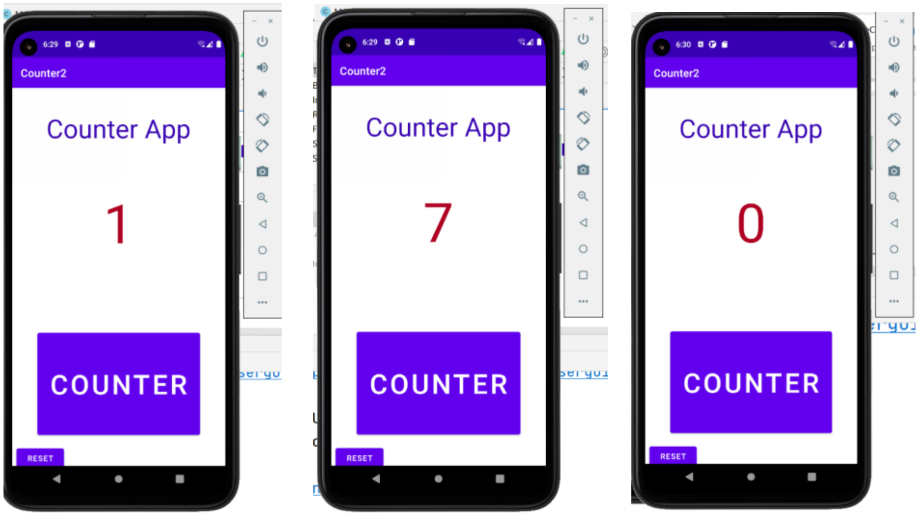
Each student is required to complete the following Lab project works individually. For each project, take Screenshot of the code and results/App output, and create PDF file and Upload it on LMS.

It is required/Compulsory for each student to Watch the given Video Lectures/Tutorials in order to complete the lab projects. Write your Name/Roll Number and status of project (completed) under the comments of each video.

LAB Projects for Week-1
Lab Task 1 to 3 - Last Date Wed-26 Oct

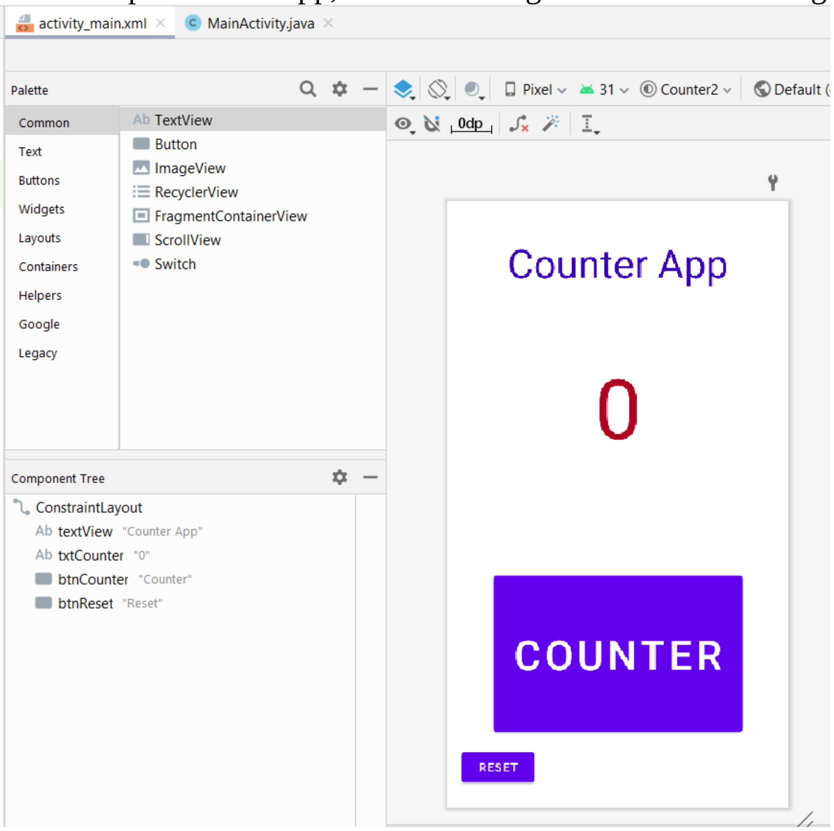
Task 1: Download and Install Android Studio.
Every student must download and install Android Studio in this personal systems.

Task 2: Create a simple Counter App in Android Studio.
The requirements of this app are that there needs to be a Count Button, which increments a number and display it on screen ever time the button is clicked as shown in figure below.



Watch Tutorial on "Code with Zeeshan Academy"
Create a Counting App in Android Studio
https://bit.ly/3TtYjui

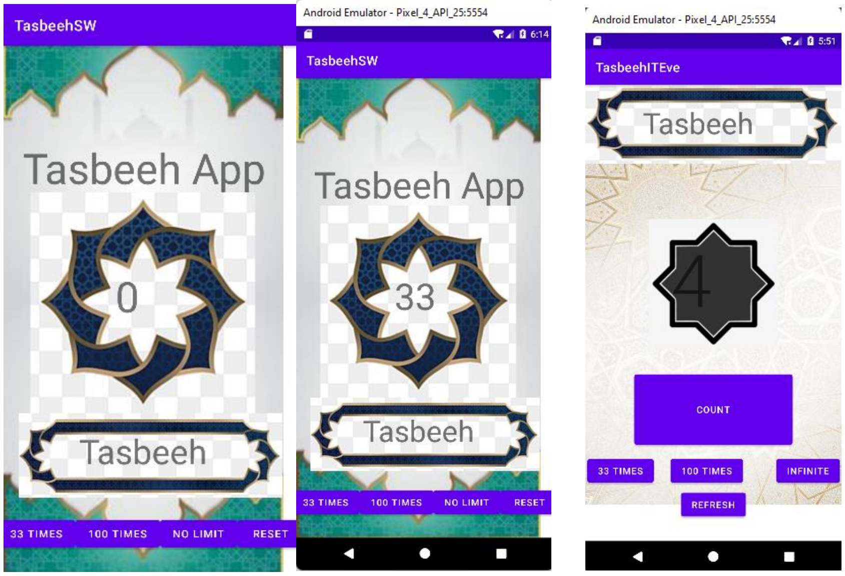
SOLVED:
To develop the above app, create the widgets as shown in the figure below, and write the code in Java file.



```java
public class MainActivity extends AppCompatActivity {

    int counter;

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

        TextView txtCount = findViewById(R.id.txtCounter);
        Button btnCount = findViewById(R.id.btnCounter);
        Button btnReset = findViewById(R.id.btnReset);

        counter=0;
        btnCount.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                counter++;
                txtCount.setText(""+counter);

            }
        });

        btnReset.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                counter=0;
                txtCount.setText(""+counter);
            }
        });
        }
}
```

Task 3: Create a Tasbeeh App in Android Studio.

The requirements of this app are that there needs to be a 5 buttons, button one should count the Tasbeeh, then each separate button for limiting the count to 33 times, 100 times, No limit and reset button, which increments a number and display it on screen ever time the button is clicked as shown in figure below.



हिन्दी / اردو TASBEEH [COUNTING] APP IN Android Studio	Watch Full Lecture on “Code with Zeeshan Academy” Create a full Tasbeeh App in Android Studio https://bit.ly/3F6LZM2

Java Code for TAsbeeh App :

```
package com.usindh.tasbeehsw;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    int counter;
    int limit;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
```

```
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);

TextView txtCount = findViewById(R.id.txtCounter);
TextView btnCount = findViewById(R.id.btnTasbeeh);
Button btn33 = findViewById(R.id.btn33);
Button btn100 = findViewById(R.id.btn100);
Button btnReset = findViewById(R.id.btnReset);
Button btnnoLimit = findViewById(R.id.btnNolimit);

counter=0;
limit=0;

btnCount.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //counter++;
        if(limit==33 && counter<33){
            counter++;
        }
        else if(limit==100 && counter<100){
            counter++;
            }
            else if(limit==0){
                counter++;
            }

        txtCount.setText(""+counter);
    }
});

btnReset.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        counter=0;
        txtCount.setText(""+counter);
    }
});

btn33.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        limit=33;
    }
});

btn100.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        limit=100;
    }
});
btnnoLimit.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        limit=0;
```

```
            }
        });
    }
}
```

Practice TASK: Develop a Roll Dice App .
Create a App that generates a random number

Watch Lecture 4: Develop First Android App For absolute Beginner | Dice App from Scratch in Android 2021 | L-4
https://youtu.be/0sWtsem8BbA

Task 3: Create a Tasbeeh App with sound or Ziker App in Android Studio

The requirements of this app are that there needs to be a buttons with sound played every time the button is clicked. The Sound of Ziker can be of and Islamic Ziker.

Practice Task: Create Login App in Android Studio

Watch Full Lecture on "Code with Zeeshan Academy"

Create a full Tasbeeh App in Android Studio

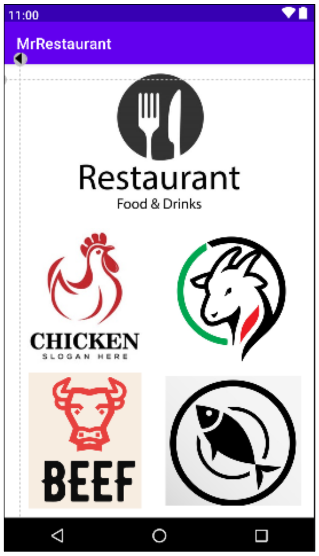
https://bit.ly/3F6LZM2

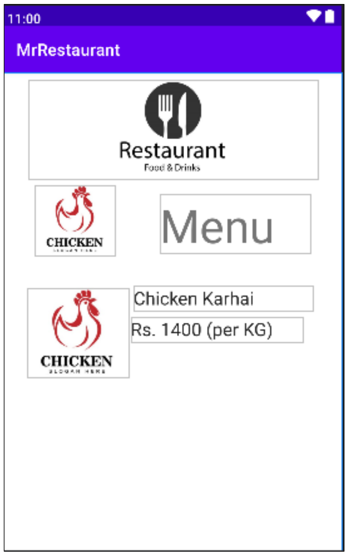
Lab Week 4- 6

- Last Date Thursday-3rd November

Task 4: Create a Food Restaurant Menu Card App

The requirement of this app is that the app contains main menu with at least 4 categories, i.e Chicken Menu, Mutton Menu, Beef Menu, Fish Menu. Then when user selects the menu, a new activity opens and it shows the list of menu items under that category, for the user to choose from.



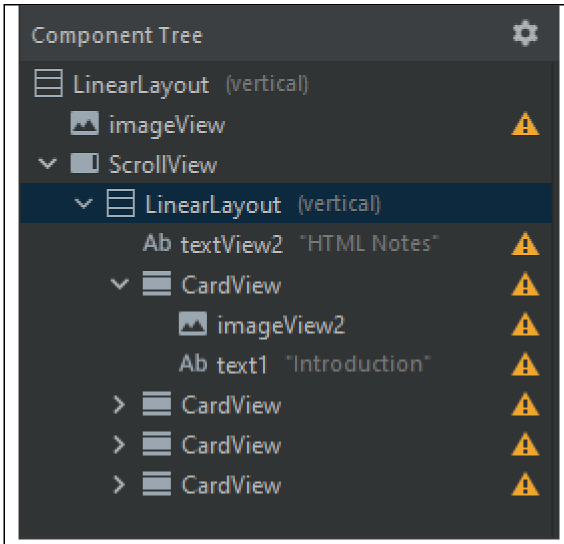


Task 5: Create a Food Recipe App

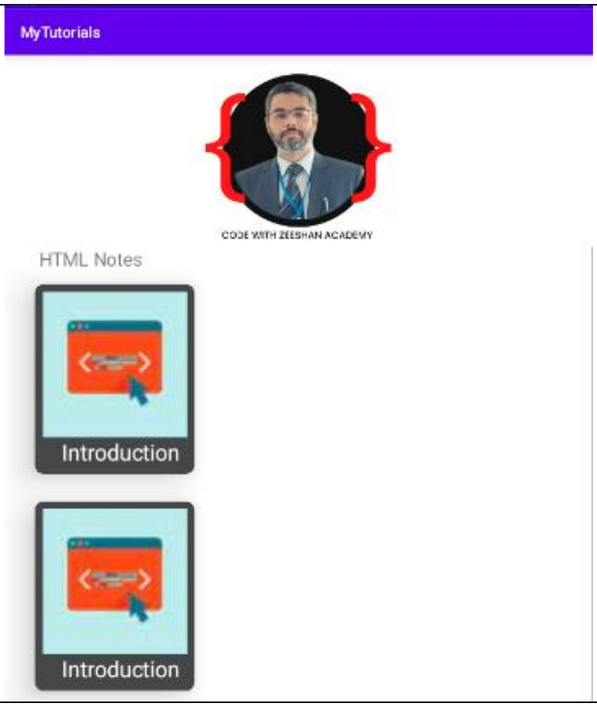
The requirement of this app is that the app contains main menu with at least 6 main categories, i.e Chicken recipes, Mutton recipes, Beef recipes, Fish recipes, etc. Then when user selects the menu, a new activity opens and it shows the list of recipes items under that category, for the user to choose from. Then when each recipe is shown, it gets displayed on another new activity.

Card View Layout Tutorial

Here are some example to create a Card View Layout with example code.

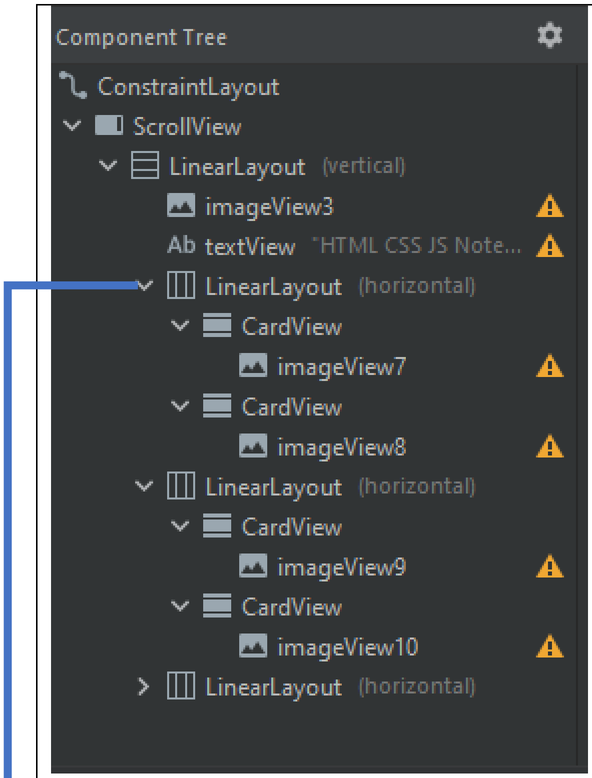


```
Component Tree
LinearLayout (vertical)
  imageView
ScrollView
  LinearLayout (vertical)
    Ab textView2 "HTML Notes"
    CardView
      imageView2
      Ab text1 "Introduction"
    CardView
    CardView
    CardView
```

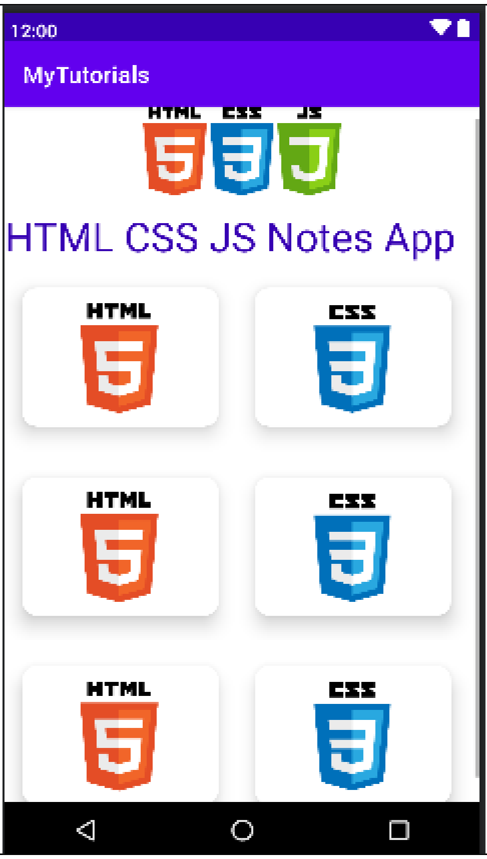


```
<androidx.cardview.widget.CardView
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_marginLeft="30dp"
    app:cardBackgroundColor="@color/material_dynamic_neutral30"
    app:cardCornerRadius="12dp"
    app:cardElevation="10dp"
    app:cardPreventCornerOverlap="true"
    app:cardUseCompatPadding="true"
    app:contentPadding="10dp">

    <ImageView
        android:id="@+id/imageView2"
        android:layout_width="200dp"
        android:layout_height="200dp"
        app:srcCompat="@drawable/web1"                                      />

    <TextView
        android:id="@+id/text1"
        android:layout_width="197dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="200dp"
        android:paddingLeft="20dp"
        android:text="Introduction"
        android:textAlignment="center"
        android:textColor="@color/white"
        android:textSize="30sp"                                             />
</androidx.cardview.widget.CardView>
```

Card View Layout 2



```
Component Tree
ConstraintLayout
  ScrollView
    LinearLayout (vertical)
        imageView3
        Ab textView "HTML CSS JS Note...
      LinearLayout (horizontal)
        CardView
          imageView7
        CardView
          imageView8
      LinearLayout (horizontal)
        CardView
          imageView9
        CardView
          imageView10
      LinearLayout (horizontal)
```



```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal">

    <androidx.cardview.widget.CardView
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        app:cardCornerRadius="12dp"
        app:cardElevation="12dp"
        app:cardUseCompatPadding="true"
        app:contentPadding="10dp">

        <ImageView
            android:id="@+id/imageView7"
            android:layout_width="150dp"
            android:layout_height="100dp"
            app:srcCompat="@drawable/html"                                    />
    </androidx.cardview.widget.CardView>

    <androidx.cardview.widget.CardView
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        app:cardCornerRadius="12dp"
        app:cardElevation="12dp"
        app:cardUseCompatPadding="true"
        app:contentPadding="10dp">

        <ImageView
            android:id="@+id/imageView8"
            android:layout_width="150dp"
            android:layout_height="100dp"
            app:srcCompat="@drawable/css"                                                      />
    </androidx.cardview.widget.CardView>

</LinearLayout>
```

Practice Tasks

Create Facebook Card Layout

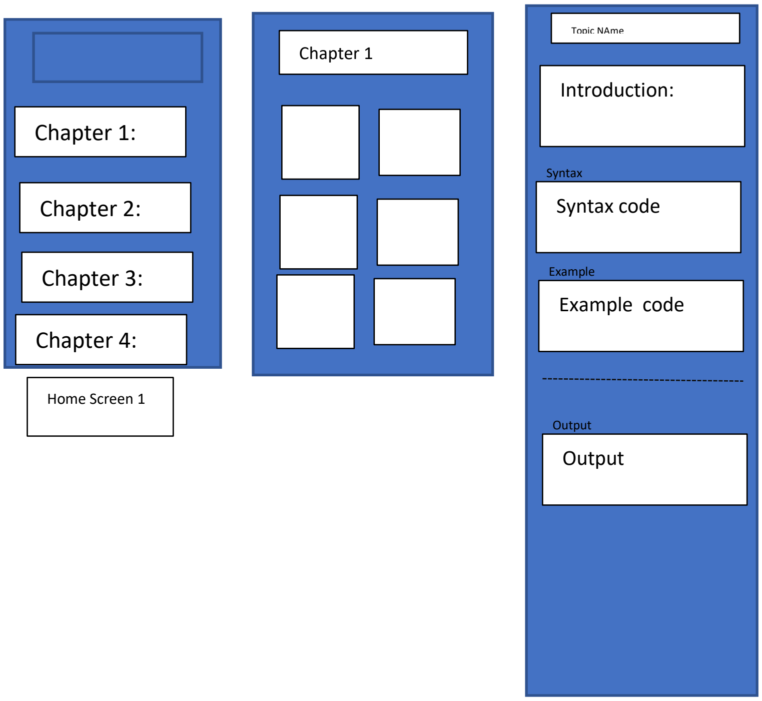
Create Food Panda App Card Layout, Daraz, Instagram, Twitter. Etc etc.

Task 6: University Notes App. Like HTML Notes App, or Android Notes App with Web view

Requirement: App with Notes on HTML only. Complete App. With Main Screen Showing Various Chapters.

Then Second Screen Should show each topic inside the Chapter. Then Finally each topic notes should appear. The final notes screen should contain, Syntax, Example Code, and Final Output.

Design, (Adobe XD or Whimsical.com)



In design stage, you might also be required to design the following diagrams:

- ERD Diagram
- Use case Diagram
- Activity Diagram
- Process Diagram

CODING

To load multiple HTML files, on single activity, we need to pass extra information through INTENTS. This extra information will contain the “file path” to load and in main activity java file, we use Bundle class to read the extra file path from intent.

FIRST: on each Buton, we set onClick listenera and use the following code

```
btnhtml.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent ca = new Intent(getApplicationContext(), Chp1.class);
            ca.putExtra("keyFile", "file:///android_asset/htmlp.html");
            startActivity(ca);
        }
    });
```

In your first activity, pass the right URL in the onClick event (example for button one):

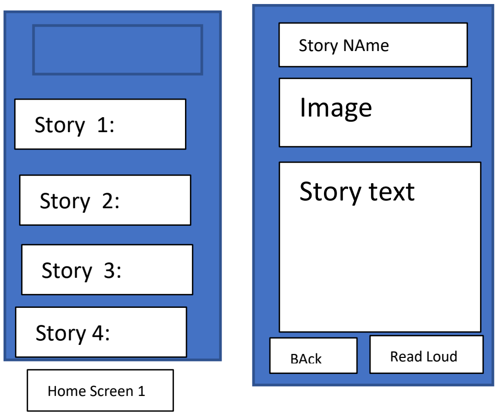
```
Intent i = new Intent(this, WebViewActivity.class);
i.putExtra("keyHTML","file:///android_asset/page_button_1.html");
startActivity(i);
```

Then in your webview activity :

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

// Read the extra path information from the Intent
  Bundle extras = getIntent().getExtras();
   String htmlFile = "file:///android_asset/default_page.html";

//check of variable is NOT empty and contains the path.
  if(extras !=null) {
     htmlFile = extras.getString("keyHTML");          // load the extra html file path in the variable.
   }

   webView = (WebView)findViewById(R.id.webview);

   webView.loadUrl(htmlFile);                //load the html file in the web view.
}
```

Task 7: Create a Stories Reading Book App. Use internet to find free stories, then there needs to be two main activities. Show list of stories on main home screen and once it clicked it shows the stories on second activity.

Design, (Adobe XD or Whimsical.com)



Website for stories.
https://momlovesbest.com/short-moral-stories-kids
https://ofhsoupkitchen.org/short-stories-with-morals

App Idea :

- Poetry App
- Hadith App
- Islamic Quotes App
-

Button READ STORY (Text to Speech)

To add “Read Aloud” feature in your app where we your Text is Read out automatically, using “Text-To-Speech” feature in Android Studio:

```
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.speech.tts.TextToSpeech;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import java.util.Locale;

public class MainActivity extends AppCompatActivity {

    EditText Text;
    Button btnText;
    TextToSpeech textToSpeech;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        Text = findViewById(R.id.Text);
        btnText = findViewById(R.id.btnText);

        // create an object textToSpeech and adding features into it
        textToSpeech = new TextToSpeech(getApplicationContext(), new
TextToSpeech.OnInitListener() {
            @Override
            public void onInit(int i) {

                // if No error is found then only it will run
                if(i!=TextToSpeech.ERROR){
                    // To Choose language of speech
                    textToSpeech.setLanguage(Locale.UK);
                }
            }
        });

        // Adding OnClickListener
        btnText.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                textToSpeech.speak(Text.getText().toString(),TextToSpeech.QUEUE_FLUSH,n
ull);
            }
        });

    }
}
```

//textToSpeech.**stop()** ; ///This method stop the speak.

https://www.tutorialspoint.com/android/android_text_to_speech.htm

https://www.geeksforgeeks.org/

Tutorials Link

https://www.tutorialspoint.com/android