**Practical – 1**

**Aim: Installation of android SDK, android studio and creating android AVD.**

Step 1: Head over to the Android Studio install page https://developer.android.com/studio

Step 2: Click on the Download Android Studio Button.

Graphical user interface, text, application, chat or text message

Description automatically generated

* Agree the terms and conditions and download

Graphical user interface, text, application, email

Description automatically generated

Step 3: After the downloading has finished, open the file from downloads and run it. It will prompt the following dialog box.



Graphical user interface, text

Description automatically generated

Graphical user interface, text

Description automatically generated

Step 4: It will start the installation, and once it is completed, it will be like the image shown below.

Graphical user interface, text, application

Description automatically generated

Step 5: Once “Finish” is clicked, it will ask whether the previous settings need to be imported [if the android studio had been installed earlier, or not. It is better to choose the ‘Don’t import Settings option’.

Step 6: This will start the Android Studio.



Meanwhile, it will be finding the available SDK components.

Step 7: After it has found the SDK components, it will redirect to the Welcome dialog box.

Graphical user interface, text, application

Description automatically generated

Step 8: Now it is time to download the SDK components.

Table

Description automatically generated

Step 9: Click on Start a new Android Studio project to build a new app.

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

Text

Description automatically generated

Step10: Open the Device Manager.

Graphical user interface, application

Description automatically generated

Step11: Click Create Device.

Graphical user interface, application

Description automatically generated

The Select Hardware window appears.

A screenshot of a computer

Description automatically generated with medium confidence

Step12: Select a hardware profile, and then click Next.

If you don't see the hardware profile you want, you can create or import a hardware profile. The System Image page appears.

Select the system image for a particular API level, and then click Next.

Step13: Change AVD properties as needed, and then click Finish.

**Practical – 2**

**Aim: Create an android application that will demonstrate the use of user interface elements and layouts.**

To get started, set up your workspace as follows:

1. In the Project window, open app > res > layout > activity\_main.xml.
2. To make room for the Layout Editor, hide the Project window. To do so, select View > Tool Windows > Project, or just click Project on the left side of the Android Studio screen.
3. If your editor shows the XML source, click the Design tab at the top right of the window.
4. Click (Select Design Surface) and select Blueprint.
5. Click (View Options) in the Layout Editor toolbar and make sure that Show All Constraints is checked.

**Layout Editor**

**Graphical user interface

Description automatically generated**

Diagram

Description automatically generated

**Illustration of two views positioned inside ConstraintLayout**

Graphical user interface, application

Description automatically generated

**The text box is constrained to the top and left of the parent layout**

Graphical user interface, application

Description automatically generated

**The button is constrained to the right side of the text box and its baseline**

**Practical Code:**

**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=".Registration">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textAlignment="center"

android:padding="64dp"

android:textStyle="bold"

android:textAppearance="?android:textAppearanceLarge"

android:text="Registration"/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_marginHorizontal="64dp"

android:weightSum="10">

<ImageView

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:scaleType="centerCrop"

android:layout\_weight="1"

android:src="@drawable/ic\_baseline\_account\_box\_24"/>

<EditText

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="9"

android:inputType="text"

android:hint="Username"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_marginHorizontal="64dp"

android:weightSum="10">

<ImageView

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:scaleType="centerCrop"

android:layout\_weight="1"

android:src="@drawable/ic\_baseline\_alternate\_email\_24"/>

<EditText

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="9"

android:inputType="text"

android:hint="Email"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_marginHorizontal="64dp"

android:weightSum="10">

<ImageView

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:scaleType="centerCrop"

android:layout\_weight="1"

android:src="@drawable/ic\_baseline\_vpn\_key\_24"/>

<EditText

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="9"

android:inputType="textPassword"

android:hint="Password"/>

</LinearLayout>

<CheckBox

android:text="I accept the Terms of Service"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"/>

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:layout\_margin="8dp"

android:text="Create Account"/>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_above="@id/loginbtn"

android:textAlignment="center"

android:text="OR"/>

<Button

android:id="@+id/loginbtn"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentBottom="true"

android:layout\_centerHorizontal="true"

android:text="Login" />

</RelativeLayout>

</LinearLayout>

</androidx.constraintlayout.widget.ConstraintLayout>

**Output:**

Graphical user interface, application

Description automatically generated

**Practical – 3**

**Aim: Create an application that should be able to find entered query (search view) from RecyclerView.**

**Code:**

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"

android:background="@drawable/bgg"

tools:context=".MainActivity">

<TextView

android:id="@+id/heading"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textAppearance="?android:textAppearanceLarge"

android:text="Person Index"

android:textAlignment="center"

app:layout\_constraintTop\_toTopOf="parent"/>

<androidx.appcompat.widget.SearchView

android:id="@+id/search"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

app:layout\_constraintTop\_toBottomOf="@+id/heading"

app:actionViewClass="androidx.appcompat.widget.SearchView"

/>

<androidx.recyclerview.widget.RecyclerView

android:id="@+id/recycler"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

app:layout\_constraintTop\_toBottomOf="@+id/search"/>

</androidx.constraintlayout.widget.ConstraintLayout>

row.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:padding="16dp">

<TextView

android:id="@+id/tview"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textAppearance="?android:textAppearanceMedium"/>

</LinearLayout>

Adapter.java

public class Adapter extends RecyclerView.Adapter<Adapter.ViewHolder> {

String[] data;

@NonNull

@Override

public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {

View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.row, parent, false);

return new ViewHolder(view);

}

@Override

public void onBindViewHolder(@NonNull ViewHolder holder, int position) {

holder.textView.setText(data[position]);

}

@Override

public int getItemCount() {

return data.length;

}

public void setData(String[] data) {

this.data = data;

notifyDataSetChanged();

}

public class ViewHolder extends RecyclerView.ViewHolder {

TextView textView;

public ViewHolder(@NonNull View itemView) {

super(itemView);

textView = itemView.findViewById(R.id.tview);

}

}

}

MainActivity.java

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

RecyclerView recyclerView = findViewById(R.id.recycler);

recyclerView.setLayoutManager(new LinearLayoutManager(this));

String person[] = {"ramesh","suresh","ashish","bs","singh","cash","dash"};

Adapter adapter = new Adapter();

recyclerView.setAdapter(adapter);

adapter.setData(animals);

SearchView searchView = findViewById(R.id.search);

searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {

@RequiresApi(api = Build.VERSION\_CODES.N)

@Override

public boolean onQueryTextSubmit(String s) {

adapter.setData(Arrays.stream(animals).filter(x -> x.contains(s)).toArray(String[]::new));

return false;

}

@Override

public boolean onQueryTextChange(String s) {

return false;

}

});

}

}

Shape

Description automatically generatedA picture containing chart

Description automatically generated

**Practical - 4**

**Aim: Create an application that have two activities:**

**- Registration of student**

**- Login**

**Login activity check for user id and password. On successful login**

**go to home screen and display user data on home page.**

**Registration of student activity have “Registration” button. If user**

**clicks on “Registration” button alert dialog will be display. If user**

**click on yes, registration details will be stored in database by using**

**SQLite. If user click on no registration details will be display only**

**in toast notification. Registrations of students contain student**

**name, address, and contact number, emailed, date of birth.**

**Code:**

**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">

<com.google.android.material.floatingactionbutton.FloatingActionButton android:id="@+id/floatingActionButton" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_marginEnd="16dp" android:layout\_marginBottom="16dp"

android:clickable="true" app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:srcCompat="@android:drawable/ic\_input\_add" android:focusable="true" android:contentDescription="@string/add\_user" />

<androidx.recyclerview.widget.RecyclerView android:id="@+id/recyclerView"

android:layout\_width="match\_parent"

android:layout\_height="0dp"

android:background="#DFBEBE"

android:orientation="vertical"

android:scrollbars="vertical"

app:layoutManager="androidx.recyclerview.widget.LinearLayoutManager"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" tools:listitem="@layout/view\_user" />

</androidx.constraintlayout.widget.ConstraintLayout>

**activity\_login.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=".LoginActivity">

<TextView

android:id="@+id/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="16dp"

android:text="@string/login\_page"

android:textSize="32sp"

android:textStyle="bold"

app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" />

<com.google.android.material.textfield.TextInputLayout android:id="@+id/textField1" style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox" android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginStart="16dp"

android:layout\_marginTop="32dp"

android:layout\_marginEnd="16dp"

android:hint="@string/enter\_the\_email"

app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toBottomOf="@+id/textView">

<com.google.android.material.textfield.TextInputEditText android:id="@+id/edEmailId"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:inputType="textEmailAddress" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout android:id="@+id/textFieawld1" style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox" android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginStart="16dp"

android:layout\_marginTop="8dp"

android:layout\_marginEnd="16dp"

android:hint="@string/enter\_the\_password"

app:endIconMode="password\_toggle"

app:errorEnabled="true"

app:errorIconDrawable="@null"

app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/textField1">

<com.google.android.material.textfield.TextInputEditText android:id="@+id/edPassword"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:inputType="textPassword" />

</com.google.android.material.textfield.TextInputLayout>

<Button

android:id="@+id/signInBtn"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginStart="16dp"

android:layout\_marginTop="16dp"

android:layout\_marginEnd="16dp"

android:text="@string/sign\_in"

app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toBottomOf="@+id/textFieawld1" />

<Button

android:id="@+id/displayAllStudentBtn"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginStart="16dp"

android:layout\_marginTop="8dp"

android:layout\_marginEnd="16dp"

android:text="@string/display\_all\_student" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toBottomOf="@+id/signInBtn" />

<androidx.cardview.widget.CardView

android:id="@+id/view\_user\_card"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:layout\_margin="8dp"

android:layout\_marginTop="8dp" android:background="?attr/selectableItemBackground" android:elevation="8dp" android:foreground="?attr/selectableItemBackground" app:cardCornerRadius="8dp"

android:visibility="gone" app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/displayAllStudentBtn">

<LinearLayout android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:layout\_margin="4dp" android:orientation="vertical">

<TextView

android:id="@+id/studentId"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" android:textColor="@color/black"

android:textSize="16sp"

android:textStyle="bold" />

<TextView

android:id="@+id/studentName"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:textColor="@color/black"

android:textSize="16sp"

android:textStyle="bold" />

<TextView

android:id="@+id/studentAddress" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:textColor="@color/black"

android:textSize="16sp"

android:textStyle="bold" />

<TextView

android:id="@+id/studentPhoneNo" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:textColor="@color/black"

android:textSize="16sp"

android:textStyle="bold" />

<TextView android:id="@+id/studentEmail" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:textColor="@color/black"

android:textSize="16sp"

android:textStyle="bold" />

<TextView android:id="@+id/studentDOB" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:textColor="@color/black"

android:textSize="16sp"

android:textStyle="bold" />

</LinearLayout>

</androidx.cardview.widget.CardView>

</androidx.constraintlayout.widget.ConstraintLayout>

add\_student\_layout

<?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" android:layout\_width="match\_parent" android:layout\_height="match\_parent" xmlns:tools="http://schemas.android.com/tools">

<include android:id="@+id/toolbarLayout" layout="@layout/toolbar\_layout"

app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" />

<ScrollView android:id="@+id/scrollView2" android:layout\_width="match\_parent" android:layout\_height="0dp"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toBottomOf="@+id/toolbarLayout">

<LinearLayout

android:id="@+id/linearLayout"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:layout\_margin="8dp"

android:orientation="vertical">

<com.google.android.material.textfield.TextInputLayout style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:hint="@string/enter\_the\_name">

<com.google.android.material.textfield.TextInputEditText android:id="@+id/edStudentName" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:inputType="textCapWords" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp" android:hint="@string/enter\_the\_address">

<com.google.android.material.textfield.TextInputEditText android:id="@+id/edStudentAddress" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:gravity="top|start"

android:inputType="textMultiLine"

android:lines="4"

android:maxLines="4"

android:minLines="4" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp" android:hint="@string/enter\_the\_mobile\_no">

<com.google.android.material.textfield.TextInputEditText android:id="@+id/edStudentPhoneNo" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:inputType="phone" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp" android:hint="@string/enter\_the\_email">

<com.google.android.material.textfield.TextInputEditText android:id="@+id/edStudentEmail" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:inputType="textEmailAddress" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout android:id="@+id/edDateBirthLayout" style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp" android:background="?attr/selectableItemBackground" android:hint="@string/enter\_the\_date\_of\_birth" android:textAllCaps="false">

<com.google.android.material.textfield.TextInputEditText android:id="@+id/edDateBirth"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:cursorVisible="false" android:drawableEnd="@drawable/ic\_calendar\_today" android:focusable="false"

android:inputType="none"

android:longClickable="false" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout android:id="@+id/txtPasswordL" style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp" android:hint="@string/enter\_the\_password" app:endIconMode="password\_toggle"

app:errorEnabled="true"

app:errorIconDrawable="@null">

<com.google.android.material.textfield.TextInputEditText android:id="@+id/edPassword"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:inputType="textPassword"

android:textColorHint="#757575"

tools:ignore="TextContrastCheck" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout android:id="@+id/txtConPasswordL" style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp" android:hint="@string/enter\_the\_confirm\_password" app:endIconMode="password\_toggle"

app:errorEnabled="true"

app:errorIconDrawable="@null">

<com.google.android.material.textfield.TextInputEditText android:id="@+id/edConPassword" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:inputType="textPassword"

android:textColorHint="#757575"

tools:ignore="TextContrastCheck" />

</com.google.android.material.textfield.TextInputLayout>

</LinearLayout>

</ScrollView>

</androidx.constraintlayout.widget.ConstraintLayout>

**toolbar\_layout.xml**

<?xml version="1.0" encoding="utf-8"?>

<com.google.android.material.appbar.AppBarLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto" android:layout\_width="match\_parent" android:layout\_height="?attr/actionBarSize" android:elevation="4dp" android:theme="@style/AppTheme.AppBarOverlay" app:popupTheme="@style/AppTheme.PopupOverlay">

<androidx.appcompat.widget.Toolbar android:id="@+id/toolbar" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" app:contentInsetStartWithNavigation="0dp" app:navigationIcon="@drawable/ic\_baseline\_close\_32"

/>

</com.google.android.material.appbar.AppBarLayout>

**select\_layout.xml**

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:app="http://schemas.android.com/apk/res-auto" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:orientation="vertical">

<Button

style="@style/Widget.AppCompat.Button.Borderless.Colored" android:id="@+id/edit\_student"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:text="@string/edit"

android:textAllCaps="false"

android:textColor="@color/black"

android:textSize="28sp"

android:textStyle="bold"

app:drawableEndCompat="@drawable/ic\_baseline\_edit\_24" />

<Button

style="@style/Widget.AppCompat.Button.Borderless.Colored" android:id="@+id/delete\_student"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:text="@string/delete"

android:textAllCaps="false"

android:textColor="@color/black"

android:textSize="28sp"

android:textStyle="bold"

app:drawableEndCompat="@drawable/ic\_baseline\_delete\_24" />

</LinearLayout>

**View\_user.xml**

<?xml version="1.0" encoding="utf-8"?>

<androidx.cardview.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:layout\_margin="8dp" android:background="?attr/selectableItemBackground" android:elevation="8dp" android:foreground="?attr/selectableItemBackground" app:cardCornerRadius="8dp">

<LinearLayout android:layout\_width="match\_parent" android:layout\_height="match\_parent"

android:layout\_margin="4dp"

android:orientation="vertical">

<TextView android:id="@+id/studentId" android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" android:textColor="@color/black"

android:textSize="16sp"

android:textStyle="bold" />

<TextView android:id="@+id/studentName" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:textColor="@color/black"

android:textSize="16sp"

android:textStyle="bold" />

<TextView android:id="@+id/studentAddress" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:textColor="@color/black"

android:textSize="16sp"

android:textStyle="bold" />

<TextView android:id="@+id/studentPhoneNo" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:textColor="@color/black"

android:textSize="16sp"

android:textStyle="bold" />

<TextView android:id="@+id/studentEmail" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:textColor="@color/black"

android:textSize="16sp"

android:textStyle="bold" />

<TextView android:id="@+id/studentDOB" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:textColor="@color/black"

android:textSize="16sp"

android:textStyle="bold" />

</LinearLayout>

</androidx.cardview.widget.CardView>

MainActivity.java

package com.android.practical4;

import androidx.appcompat.app.AppCompatActivity;

import androidx.recyclerview.widget.RecyclerView;

import android.app.Dialog;

import android.database.Cursor;

import android.os.Bundle;

import android.widget.Button;

import android.widget.LinearLayout;

import com.google.android.material.dialog.MaterialAlertDialogBuilder;

import com.google.android.material.floatingactionbutton.FloatingActionButton;

import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {

DatabaseHelper myDB;

ArrayList < Student > studentList;

StudentAdapter studentAdapter;

AddStudentDialog addStudentDialog;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

studentList = new ArrayList < > ();

myDB = new DatabaseHelper(this);

FloatingActionButton addUserFloatBtn = findViewById(R.id.floatingActionButton);

Dialog selectDialog = new Dialog(this);

selectDialog.setContentView(R.layout.select\_layout);

selectDialog.getWindow().setLayout(LinearLayout.LayoutParams.MATCH\_PARENT,

LinearLayout.LayoutParams.WRAP\_CONTENT);

selectDialog.setCancelable(true);

Button editStudentBtn = selectDialog.findViewById(R.id.edit\_student);

Button deleteStudentBtn = selectDialog.findViewById(R.id.delete\_student);

RecyclerView recyclerView = findViewById(R.id.recyclerView);

Cursor cursor = myDB.readAllStudentData();

while (cursor.moveToNext()) {

studentList.add(new Student(

cursor.getString((int) cursor.getColumnIndex("studentId")), cursor.getString((int) cursor.getColumnIndex("studentName")), cursor.getString((int) cursor.getColumnIndex("studentAddress")), cursor.getString((int) cursor.getColumnIndex("studentPhoneNo")), cursor.getString((int) cursor.getColumnIndex("studentEmail")), cursor.getString((int) cursor.getColumnIndex("studentDOB")), cursor.getString((int) cursor.getColumnIndex("password"))

));

}

studentAdapter = new StudentAdapter(studentList, (position, key) - > {

selectDialog.show();

editStudentBtn.setOnClickListener(v - > {

selectDialog.dismiss();

addStudentDialog = new AddStudentDialog(true, position, studentList, studentAdapter, myDB);

addStudentDialog.show(getSupportFragmentManager(), "AddStudentDialog");

});

deleteStudentBtn.setOnClickListener(v - > {

selectDialog.dismiss();

new MaterialAlertDialogBuilder(this)

.setTitle("Delete Student")

.setMessage("Are you sure, you want to delete this Student?")

.setPositiveButton("Delete", (dialog, which) - > {

studentList.remove(position);myDB.deleteStudent(key);studentAdapter.notifyItemRemoved(position);

})

.setNegativeButton("Cancel", null)

.setIcon(android.R.drawable.ic\_dialog\_alert)

.show();

});

});

recyclerView.setAdapter(studentAdapter);

addUserFloatBtn.setOnClickListener(v - > {

addStudentDialog = new AddStudentDialog(false, -1, studentList, studentAdapter, myDB);

addStudentDialog.show(getSupportFragmentManager(), "AddStudentDialog");

});

}

}

**StudentAdapter.java**

package com.android.practical4;

import android.annotation.SuppressLint;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.TextView;

import androidx.annotation.NonNull;

import androidx.recyclerview.widget.RecyclerView;

import java.util.List;

class StudentAdapter extends RecyclerView.Adapter < StudentAdapter.ViewHolder > {

private final List < Student > studentList;

private final SelectListener selectListener;

public StudentAdapter(List < Student > studentList, SelectListener selectListener) {

this.studentList = studentList;

this.selectListener = selectListener;

}

@NonNull @Override

public StudentAdapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {

View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.view\_user, parent, false);

return new ViewHolder(view);

}

@SuppressLint("SetTextI18n") @Override

public void onBindViewHolder(@NonNull StudentAdapter.ViewHolder holder, int position) {

Student student = studentList.get(position);

holder.studentId.setText("Id: " + student.getId());

holder.studentName.setText("Name: " + student.getStudentName());

holder.studentAddress.setText("Address: " + student.getStudentAddress());

holder.studentPhoneNo.setText("Mobile No.: " + student.getStudentPhoneNo());

holder.studentEmail.setText("Email id: " + student.getStudentEmail());

holder.studentDOB.setText("DOB: " + student.getStudentDOB());

holder.itemView.setOnLongClickListener(v - > {

selectListener.onSelect(position, student.getId());

return false;

});

}

@Override

public int getItemCount() {

return studentList.size();

}

static class ViewHolder extends RecyclerView.ViewHolder {

private final TextView studentId;

private final TextView studentName;

private final TextView studentAddress;

private final TextView studentPhoneNo;

private final TextView studentEmail;

private final TextView studentDOB;

public ViewHolder(@NonNull View itemView) {

super(itemView);

studentId = itemView.findViewById(R.id.studentId);

studentName = itemView.findViewById(R.id.studentName);

studentAddress = itemView.findViewById(R.id.studentAddress);

studentPhoneNo = itemView.findViewById(R.id.studentPhoneNo);

studentEmail = itemView.findViewById(R.id.studentEmail);

studentDOB = itemView.findViewById(R.id.studentDOB);

}

}

public interface SelectListener {

void onSelect(int position, String id);

}

}

**Login.java**

package com.android.practical4;

import androidx.appcompat.app.AppCompatActivity;

import androidx.appcompat.app.AppCompatDelegate;

import android.content.Intent;

import android.database.Cursor;

import android.os.Bundle;

import android.text.Editable;

import android.text.TextWatcher;

import android.util.Patterns;

import android.view.View;

import android.view.inputmethod.InputMethodManager;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;

import com.google.android.material.textfield.TextInputLayout;

public class LoginActivity extends AppCompatActivity {

private EditText edEmailId, edPassword;

Button signInBtn;

private TextInputLayout txtPasswordL;

private DatabaseHelper myDB;

TextView studentId;

TextView studentName;

TextView studentAddress;

TextView studentPhoneNo;

TextView studentEmail;

TextView studentDOB;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

AppCompatDelegate.setDefaultNightMode(AppCompatDelegate.MODE\_NIGHT\_NO);

setContentView(R.layout.activity\_login);

studentId = findViewById(R.id.studentId);

studentName = findViewById(R.id.studentName);

studentAddress = findViewById(R.id.studentAddress);

studentPhoneNo = findViewById(R.id.studentPhoneNo);

studentEmail = findViewById(R.id.studentEmail);

studentDOB = findViewById(R.id.studentDOB);

myDB = new DatabaseHelper(this);

findViewById(R.id.displayAllStudentBtn).setOnClickListener(v - > startActivity(new Intent(LoginActivity.this, MainActivity.class)));

edEmailId = findViewById(R.id.edEmailId);

edPassword = findViewById(R.id.edPassword);

signInBtn = findViewById(R.id.signInBtn);

txtPasswordL = findViewById(R.id.textFieawld1);

signInBtn.setOnClickListener(v - > {

Validation();hideKeyboard(v);

});

edPassword.addTextChangedListener(new TextWatcher() {

@Override

public void beforeTextChanged(CharSequence s, int start, int count, int after) {

}

@Override

public void onTextChanged(CharSequence s, int start, int before, int count) {

}

@Override

public void afterTextChanged(Editable s) {

if (edPassword.getText().toString().trim().isEmpty()) {

txtPasswordL.setError("Required");

} else if (edPassword.getText().toString().trim().length() < 8 || edPassword.getText().toString().trim().length() > 10) {

txtPasswordL.setError("Password must be 8 to 10 character!");

} else {

txtPasswordL.setError(null);

}

}

});

}

private void Validation() {

if (edEmailId.getText().toString().trim().isEmpty()) {

edEmailId.setError("Required");

} else if (!validEmail(edEmailId.getText().toString().trim())) {

edEmailId.setError("Enter valid e-mail!");

} else if (edPassword.getText().toString().trim().isEmpty()) {

txtPasswordL.setError("Required");

} else if (edPassword.getText().toString().trim().length() < 8 || edPassword.getText().toString().trim().length() > 10) {

txtPasswordL.setError("Password must be 8 to 10 character!");

} else {

txtPasswordL.setError(null);

Cursor cursor = myDB.checkEmailPassword(edEmailId.getText().toString().trim(), edPassword.getText().toString().trim());

if (cursor.getCount() <= 0) {

findViewById(R.id.view\_user\_card).setVisibility(View.GONE);

Toast.makeText(this, "Not Found!", Toast.LENGTH\_SHORT).show();

}

while (cursor.moveToNext()) {

findViewById(R.id.view\_user\_card).setVisibility(View.VISIBLE);

studentId.setText(cursor.getString((int) cursor.getColumnIndex("studentId")));

studentName.setText(cursor.getString((int) cursor.getColumnIndex("studentName")));

studentAddress.setText(cursor.getString((int) cursor.getColumnIndex("studentAddress")));

studentPhoneNo.setText(cursor.getString((int) cursor.getColumnIndex("studentPhoneNo")));

studentEmail.setText(cursor.getString((int) cursor.getColumnIndex("studentEmail")));

studentDOB.setText(cursor.getString((int) cursor.getColumnIndex("studentDOB")));

}

}

}

private boolean validEmail(String email) {

return Patterns.EMAIL\_ADDRESS.matcher(email).matches();

}

public void hideKeyboard(View view) { try {

InputMethodManager imm = (InputMethodManager) getSystemService(INPUT\_METHOD\_SERVICE);

imm.hideSoftInputFromWindow(view.getWindowToken(), 0);

} catch (Exception ignored) {

}

}

}

DatabaseHelper.java

package com.android.practical4;

import android.content.ContentValues;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteOpenHelper;

import android.widget.Toast

public class DatabaseHelper extends SQLiteOpenHelper {

private final Context context;

private static final String DATABASE\_NAME = "student.db";

private static final int DATABASE\_VERSION = 1;

private static final String TABLE\_NAME\_NEW\_STUDENT = "Student";

private static final String COLUMN\_studentId = "studentId";

private static final String COLUMN\_studentName = "studentName";

private static final String COLUMN\_studentAddress = "studentAddress";

private static final String COLUMN\_studentPhoneNo = "studentPhoneNo";

private static final String COLUMN\_studentEmail = "studentEmail";

private static final String COLUMN\_studentDOB = "studentDOB";

private static final String COLUMN\_password = "password";

public DatabaseHelper(Context context) {

super(context, DATABASE\_NAME, null, DATABASE\_VERSION);

this.context = context;

}

@Override

public void onCreate(SQLiteDatabase db) {

String new\_user = "CREATE TABLE " + TABLE\_NAME\_NEW\_STUDENT + " (" + COLUMN\_studentId + " TEXT, " +

COLUMN\_studentName + " TEXT, " + COLUMN\_studentAddress + " TEXT, " + COLUMN\_studentPhoneNo + " TEXT, " + COLUMN\_studentEmail + " TEXT, " + COLUMN\_studentDOB + " TEXT, " + COLUMN\_password + " TEXT);";

db.execSQL(new\_user);

}

@Override

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

db.execSQL("DROP TABLE IF EXISTS " + TABLE\_NAME\_NEW\_STUDENT);

onCreate(db);

}

public void addStudent(String studentName, String password,

String studentAddress, String studentPhoneNo, String studentEmail, String studentDOB, String id) {

SQLiteDatabase db = getWritableDatabase();

ContentValues cv = new ContentValues();

cv.put(COLUMN\_studentName, studentName);

cv.put(COLUMN\_studentAddress, studentAddress);

cv.put(COLUMN\_studentPhoneNo, studentPhoneNo);

cv.put(COLUMN\_studentEmail, studentEmail);

cv.put(COLUMN\_studentDOB, studentDOB);

cv.put(COLUMN\_password, password);

cv.put(COLUMN\_studentId, id);

long result = db.insert(TABLE\_NAME\_NEW\_STUDENT, null, cv);

if (result == -1) {

Toast.makeText(context, "Failed", Toast.LENGTH\_SHORT).show();

} else {

Toast.makeText(context, "Added Successfully!", Toast.LENGTH\_SHORT).show();

}

}

public Cursor readAllStudentData() {

String query = "SELECT \* FROM " + TABLE\_NAME\_NEW\_STUDENT + " ORDER

BY " + COLUMN\_studentName + "

ASC "; SQLiteDatabase db = this.getReadableDatabase();

Cursor cursor = null;

if (db != null) {

cursor = db.rawQuery(query, null);

}

return cursor;

}

public void updateStudent(String studentName, String studentAddress, String studentPhoneNo,

String studentEmail, String studentDOB, String id) {

SQLiteDatabase db = this.getWritableDatabase();

ContentValues cv = new ContentValues();

cv.put(COLUMN\_studentName, studentName);

cv.put(COLUMN\_studentAddress, studentAddress);

cv.put(COLUMN\_studentPhoneNo, studentPhoneNo);

cv.put(COLUMN\_studentEmail, studentEmail);

cv.put(COLUMN\_studentDOB, studentDOB);

long result = db.update(TABLE\_NAME\_NEW\_STUDENT, cv, "studentId=?", new String[] {

id

});

if (result == -1) {

Toast.makeText(context, "Failed", Toast.LENGTH\_SHORT).show();

} else {

Toast.makeText(context, "Updated Successfully!", Toast.LENGTH\_SHORT).show();

}

}

public void deleteStudent(String id) {

SQLiteDatabase db = this.getWritableDatabase();

long result = db.delete(TABLE\_NAME\_NEW\_STUDENT, "studentId=?", new String[] {

id

});

if (result == -1) {

Toast.makeText(context, "Failed to Delete.", Toast.LENGTH\_SHORT).show();

}

}

public Cursor checkEmailPassword(String email, String password) {

String[] columns = {

COLUMN\_studentId,

COLUMN\_studentEmail,

COLUMN\_studentName,

COLUMN\_studentPhoneNo,

COLUMN\_studentAddress,

COLUMN\_studentDOB

};

String selection = COLUMN\_studentEmail + " = ?" + " AND " + COLUMN\_password +

" = ?";

String[] selectionArgs = {

email,

password

};

SQLiteDatabase db = this.getReadableDatabase();

Cursor cursor = null;

if (db != null) {

cursor = db.query(TABLE\_NAME\_NEW\_STUDENT, //Table to query columns, selection, selectionArgs, null, null,null);

}

return cursor;

}

}

**AddStudentdialog.java**

package com.android.practical4;

import android.app.Dialog;

import android.os.Bundle;

import android.text.Editable;

import android.text.TextWatcher;

import android.util.Patterns;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.EditText;

import androidx.annotation.NonNull;

import androidx.appcompat.widget.Toolbar;

import androidx.fragment.app.DialogFragment;

import com.google.android.material.datepicker.MaterialDatePicker;

import com.google.android.material.textfield.TextInputLayout;

import java.text.ParseException;

import java.text.SimpleDateFormat;

import java.util.ArrayList;

import java.util.UUID;

public class AddStudentDialog extends DialogFragment {

private Toolbar toolbar;

ArrayList < Student > studentList;

StudentAdapter studentAdapter;

DatabaseHelper myDB;

boolean updateSate;

int updatePos;

public AddStudentDialog(boolean updateSate, int updatePos, ArrayList < Student > studentList, StudentAdapter studentAdapter, DatabaseHelper myDB) {

this.updateSate = updateSate;

this.updatePos = updatePos;

this.studentList = studentList;

this.studentAdapter = studentAdapter;

this.myDB = myDB;

}

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setStyle(DialogFragment.STYLE\_NORMAL, R.style.FullScreenDialog);

}

@Override

public void onStart() {

super.onStart();

Dialog dialog = getDialog();

if (dialog != null) {

int width = ViewGroup.LayoutParams.MATCH\_PARENT;

int height = ViewGroup.LayoutParams.MATCH\_PARENT;

dialog.getWindow().setLayout(width, height);

dialog.getWindow().setWindowAnimations(R.style.AppTheme\_Slide);

}

}

private EditText edStudentName, edStudentAddress, edStudentPhoneNo, edStudentEmail, edDateBirth, edPassword, edConPassword;

private TextInputLayout txtPasswordL, txtConPasswordL;

private boolean isValidPassword = false;

private boolean isValidConPassword = false;

@Override

public View onCreateView(@NonNull LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {

super.onCreateView(inflater, container, savedInstanceState);

View view = inflater.inflate(R.layout.add\_student\_layout, container, false);

toolbar = view.findViewById(R.id.toolbar);

edStudentName = view.findViewById(R.id.edStudentName);

edStudentAddress = view.findViewById(R.id.edStudentAddress);

edStudentPhoneNo = view.findViewById(R.id.edStudentPhoneNo);

edStudentEmail = view.findViewById(R.id.edStudentEmail);

edPassword = view.findViewById(R.id.edPassword);

edDateBirth = view.findViewById(R.id.edDateBirth);

edConPassword = view.findViewById(R.id.edConPassword);

txtPasswordL = view.findViewById(R.id.txtPasswordL);

txtConPasswordL = view.findViewById(R.id.txtConPasswordL);

if (!updateSate) {

edPassword.addTextChangedListener(new TextWatcher() {

@Override

public void beforeTextChanged(CharSequence s, int start, int count, int after) {

}

@Override

public void onTextChanged(CharSequence s, int start, int before, int count) {

}

@Override

public void afterTextChanged(Editable s) {

validateEditTextPassword(edPassword, txtPasswordL, "password");

}

});

edConPassword.addTextChangedListener(new TextWatcher() {

@Override

public void beforeTextChanged(CharSequence s, int start, int count, int after) {

}

@Override

public void onTextChanged(CharSequence s, int start, int before, int count) {}

@Override

public void afterTextChanged(Editable s) {

validateEditTextPassword(edConPassword, txtConPasswordL, "conPassword");

}

});

}

MaterialDatePicker materialDatePicker = MaterialDatePicker.Builder.datePicker()

.setTitleText("Select Birthday")

.setSelection(getDate())

.build();

materialDatePicker.addOnPositiveButtonClickListener(selection - >

edDateBirth.setText(materialDatePicker.getHeaderText()));

edDateBirth.setOnClickListener(v - >

materialDatePicker.show(getParentFragmentManager(), "MATERIAL\_DATE\_PICKER"));

if (updateSate) {

txtPasswordL.setVisibility(View.GONE);

txtConPasswordL.setVisibility(View.GONE);

edStudentName.setText(studentList.get(updatePos).getStudentName());

edStudentAddress.setText(studentList.get(updatePos).getStudentAddress());

edStudentPhoneNo.setText(studentList.get(updatePos).getStudentPhoneNo());

edStudentEmail.setText(studentList.get(updatePos).getStudentEmail());

edDateBirth.setText(studentList.get(updatePos).getStudentDOB());

}

return view;

}

private Long getDate() {

if (updateSate) {

try {

return new SimpleDateFormat("dd MMM yyyy").parse(studentList.get(updatePos).getStudentDOB()).getTime();

} catch (ParseException e) {

e.printStackTrace();

}

}

return null;

}

@Override

public void onViewCreated(@NonNull View view, Bundle savedInstanceState) {

super.onViewCreated(view, savedInstanceState);

toolbar.setNavigationOnClickListener(v - > dismiss());

if (updateSate) {

toolbar.setTitle("Update Student");

} else {

toolbar.setTitle("Add Student");

}

toolbar.inflateMenu(R.menu.save\_menu);

toolbar.setOnMenuItemClickListener(item - > {

int id = item.getItemId();

if (id == R.id.saveMenu) {

uploadData();

return true;

} else {

return false;

}

});

}

private void uploadData() {

if (edStudentName.getText().toString().trim().isEmpty()) {

edStudentName.setError("Required");

} else if (edStudentAddress.getText().toString().trim().isEmpty()) {

edStudentAddress.setError("Required");

} else if (edStudentPhoneNo.getText().toString().trim().isEmpty()) {

edStudentPhoneNo.setError("Required");

} else if (edStudentEmail.getText().toString().trim().isEmpty()) {

edStudentEmail.setError("Required");

} else if (edStudentPhoneNo.getText().toString().length() <= 9 || edStudentPhoneNo.getText().toString().length() >= 11) {

edStudentPhoneNo.setError("Phone no is 10 Digits.");

} else if (!validEmail(edStudentEmail.getText().toString().trim())) {

edStudentEmail.setError("valid e-mail");

} else if (edDateBirth.getText().toString().trim().isEmpty()) {

edDateBirth.setError("Required");

} else {

if (updateSate) {

myDB.updateStudent(edStudentName.getText().toString().trim(),

edStudentAddress.getText().toString().trim(), edStudentPhoneNo.getText().toString().trim(), edStudentEmail.getText().toString().trim(), edDateBirth.getText().toString().trim(), studentList.get(updatePos).getId());

studentList.set(updatePos, new Student(studentList.get(updatePos).getId(), edStudentName.getText().toString().trim(), edStudentAddress.getText().toString().trim(), edStudentPhoneNo.getText().toString().trim(), edStudentEmail.getText().toString().trim(), edDateBirth.getText().toString().trim(), studentList.get(updatePos).getPassword()));

studentAdapter.notifyItemChanged(updatePos);

AllNullEditText();

dismiss();

} else {

validateEditTextPassword(edPassword, txtPasswordL, "password");

if (isValidPassword) {

validateEditTextPassword(edConPassword, txtConPasswordL, "conPassword");

if (isValidConPassword) {

if (!edPassword.getText().toString().trim().equals(edConPassword.getText().toString().trim())) {

txtConPasswordL.setError("Password don't match!");

} else {

txtConPasswordL.setError(null);

String id = UUID.randomUUID().toString();

myDB.addStudent(edStudentName.getText().toString().trim(),

edPassword.getText().toString().trim(), edStudentAddress.getText().toString().trim(), edStudentPhoneNo.getText().toString().trim(), edStudentEmail.getText().toString().trim(), edDateBirth.getText().toString().trim(),

id);

studentList.add(new Student(

id, edStudentName.getText().toString().trim(), edStudentAddress.getText().toString().trim(),

edStudentPhoneNo.getText().toString().trim(), edStudentEmail.getText().toString().trim(), edDateBirth.getText().toString().trim(), edPassword.getText().toString().trim()

));

studentAdapter.notifyItemInserted(studentList.size());

AllNullEditText();

dismiss();

}

}

}

}

}

}

private boolean validEmail(String email) {

return Patterns.EMAIL\_ADDRESS.matcher(email).matches();

}

private void AllNullEditText() {

edStudentName.setText(null);

edStudentAddress.setText(null);

edStudentPhoneNo.setText(null);

edStudentEmail.setText(null);

edDateBirth.setText(null);

edPassword.setText(null);

edConPassword.setText(null);

}

private void validateEditTextPassword(EditText editText, TextInputLayout textInputLayout,

String state

) {

boolean truthState = false;

if (editText.getText().toString().trim().isEmpty()) {

textInputLayout.setError("Required");

} else if (editText.getText().toString().trim().length() < 8 || editText.getText().toString().trim().length() > 10) {

textInputLayout.setError("Password must be 8 to 10 character!");

} else {

textInputLayout.setError(null);

truthState = true;

}

if (truthState) {

if (state.equals("password")) {

isValidPassword = true;

} else if (state.equals("conPassword")) {

isValidConPassword = true;

}

} else {

if (state.equals("password")) {

isValidPassword = false;

} else if (state.equals("conPassword")) {

isValidConPassword = false;

}

}

}

}

**Student.java**

package com.android.practical4;

public class Student {

private String id;

private String studentName;

private String studentAddress;

private String studentPhoneNo;

private String studentEmail;

private String studentDOB;

private String password;

public Student(String id, String studentName, String studentAddress, String studentPhoneNo, String studentEmail, String studentDOB, String password) {

this.id = id;

this.studentName = studentName;

this.studentAddress = studentAddress;

this.studentPhoneNo = studentPhoneNo;

this.studentEmail = studentEmail;

this.studentDOB = studentDOB;

this.password = password;

}

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

public String getStudentName() {

return studentName;

}

public void setStudentName(String studentName) {

this.studentName = studentName;

}

public String getStudentAddress() {

return studentAddress;

}

public void setStudentAddress(String studentAddress) {

this.studentAddress = studentAddress;

}

public String getStudentPhoneNo() {

return studentPhoneNo;

}

public void setStudentPhoneNo(String studentPhoneNo) {

this.studentPhoneNo = studentPhoneNo;

}

public String getStudentEmail() {

return studentEmail;

}

public void setStudentEmail(String studentEmail) {

this.studentEmail = studentEmail;

}

public String getStudentDOB() {

return studentDOB;

}

public void setStudentDOB(String studentDOB) {

this.studentDOB = studentDOB;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

}

**Output:**

  

 