|  |  |  |
| --- | --- | --- |
| **Sr.**  **No.** | **Program Statement** | **Pag**  **e No.** |
| 1 | Demonstrate different Layouts with different views in android LayoutsConstraintLayout, RelativeLayout, TableLayout Views- Button, TextView, EditText, WebView, CheckBox, RadioButton, ToggleButton, ImageButton,  RatingBar, ProgressBar, SeekBar, VideoView, DatePicker, CalendarView, Spinner |  |
| 2 | Write an android code to make phone call using Intent. |  |
| 3 | Write an android code to turn ON/OFF Bluetooth. |  |
| 4 | Write an android code to turn ON /OFF the Wi-Fi. |  |
| 5 | Design android application for login activity. Write android code to check login  credentials with username = "mca" and password = "android". Display appropriate toast message to the user. |  |
| 6 | Create a fragment that has its own UI and enable your activities to communicate  with fragments. |  |
| 7 | Demonstrate Array Adapter using List View to display list of fruits. |  |
| 8 | Write an application to demonstrate Alert Dialog Box in android. |  |
| 9 | Demonstrate Options Menu, Context Menu and Popup Menu in android. |  |
| 10 | Write an application to produce Notification. |  |
| 11 | Write an android application using SQLite to create table and perform CRUD  operations (Example. COURSE table (ID, Name, Duration, Description), perform ADD, UPDATE, DELETE and READ operations) |  |
| 12 | Create an Android app, powered by Firebase Realtime database that supports: Adding Data to Firebase Realtime database, Retrieving Data from Firebase and  Deleting data from firebase data. |  |
| 13 | Demonstrate WebView to display the web pages in an android application. |  |
| 14 | Write an android app to write JSON data into a file and read JSON data from  created file. |  |
| 15 | Write an application to display a PDF as an image in React app using URL. |  |
| 16 | Develop simple flutter application to open a browser using Android SDK. |  |

**Q.2**

**Manifest.xml**

<uses-permission android:name="android.permission.CALL\_PHONE" />

## activity\_main.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" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="center" android:orientation="vertical" tools:context=".MainActivity">

<EditText android:id="@+id/mobNo" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:hint="Enter mobile no" android:inputType="number" android:maxEms="10" android:padding="11dp" />

<androidx.appcompat.widget.AppCompatButton android:id="@+id/callBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Call" />

</LinearLayout>

**MainActivity.java** package com.subhdroid.lab\_j6;

//6. Write an android code to make phone call using Intent import androidx.appcompat.app.AppCompatActivity; import androidx.appcompat.widget.AppCompatButton;

import android.content.Intent; import android.net.Uri; import android.os.Bundle; import android.widget.EditText;

public class MainActivity extends AppCompatActivity {

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

EditText mobNo = findViewById(R.id.mobNo);

AppCompatButton callBtn = findViewById(R.id.callBtn);

callBtn.setOnClickListener(view -> {

Intent intent = new Intent(Intent.ACTION\_CALL); intent.setData(Uri.parse("tel:" + mobNo.getText().toString())); startActivity(intent);

});

}

}

Q.3

## Manifest.xml

<uses-permission android:name="android.permission.BLUETOOTH"/>

<uses-permission

android:name="android.permission.BLUETOOTH\_ADMIN"/>

<uses-permission

android:name="android.permission.BLUETOOTH\_CONNECT" />

## activity\_main.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" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="center" android:orientation="vertical" tools:context=".MainActivity">

<androidx.appcompat.widget.AppCompatButton android:id="@+id/onBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Turn On Bluetooth" /> <androidx.appcompat.widget.AppCompatButton android:id="@+id/offBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Turn Off Bluetooth" />

</LinearLayout>

**MainActivity.java** package com.subhdroid.lab\_j7;

//7. Write an android code to turn ON/OFF Bluetooth

import androidx.appcompat.app.AppCompatActivity; import androidx.appcompat.widget.AppCompatButton; import androidx.core.app.ActivityCompat;

import android.Manifest; import android.bluetooth.BluetoothAdapter; import android.content.pm.PackageManager; import android.os.Bundle;

public class MainActivity extends AppCompatActivity

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

AppCompatButton onBtn, offBtn;

onBtn = findViewById(R.id.onBtn); offBtn = findViewById(R.id.offBtn);

final BluetoothAdapter bluetAdapter = BluetoothAdapter.getDefaultAdapter();

onBtn.setOnClickListener(view -> { if (!bluetAdapter.isEnabled()) {

if (ActivityCompat.checkSelfPermission(MainActivity.this,

Manifest.permission.BLUETOOTH\_CONNECT) !=

PackageManager.PERMISSION\_GRANTED) {

bluetAdapter.enable();

}

}

});

offBtn.setOnClickListener(view -> { if (bluetAdapter.isEnabled()) {

if (ActivityCompat.checkSelfPermission(MainActivity.this,

Manifest.permission.BLUETOOTH\_CONNECT) !=

PackageManager.PERMISSION\_GRANTED) { bluetAdapter.disable();

}

}

});

}

}

Q.4

## Manifest.xml

<uses-permission

android:name="android.permission.ACCESS\_WIFI\_STATE"/>

<uses-permission

android:name="android.permission.CHANGE\_WIFI\_STATE"/>

## activity\_main.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" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="center" android:orientation="vertical" tools:context=".MainActivity"> <androidx.appcompat.widget.AppCompatButton android:id="@+id/onBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Turn On Wi-Fi" />

<androidx.appcompat.widget.AppCompatButton android:id="@+id/offBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Turn Off Wi-Fi" />

</LinearLayout>

**MainActivity.java** package com.subhdroid.lab\_j8;

//8. Write an android code to turn ON /OFF the Wi-Fi

import androidx.appcompat.app.AppCompatActivity; import androidx.appcompat.widget.AppCompatButton;

import android.content.Context; import android.net.wifi.WifiManager; import android.os.Bundle;

public class MainActivity extends AppCompatActivity { @Override protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

AppCompatButton onBtn, offBtn;

onBtn = findViewById(R.id.onBtn); offBtn = findViewById(R.id.offBtn);

WifiManager wifiManager =

(WifiManager)

getApplicationContext().getSystemService(Context.WIFI\_SERVICE);

onBtn.setOnClickListener(view -> { if (!wifiManager.isWifiEnabled()) { wifiManager.setWifiEnabled(true);

}

});

offBtn.setOnClickListener(view -> { if (wifiManager.isWifiEnabled()) { wifiManager.setWifiEnabled(false);

}

});

}

}

Q.5

**Manifest.xml**

## activity\_main.xml

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

<TableLayout 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:gravity="center" tools:context=".MainActivity">

<TableRow>

<TextView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_column="1" android:text="Username : " android:textSize="18sp" />

<EditText android:id="@+id/username" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_column="2" android:hint="Enter username" android:inputType="text" />

</TableRow>

<TableRow>

<TextView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_column="1" android:text="Password : " android:textSize="18sp" />

<EditText android:id="@+id/password" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_column="2" android:hint="Enter password" android:inputType="textPassword" />

</TableRow>

<TableRow>

<androidx.appcompat.widget.AppCompatButton android:id="@+id/loginBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_column="2" android:text="Login" />

</TableRow>

</TableLayout>

## MainActivity.java

package com.subhdroid.lab\_j9;

//9. Design android application for login activity by using TableLayout. Write android code to

// check login credentials with username = "mca" and password = "android". Display appropriate

// toast message to the user

import androidx.appcompat.app.AppCompatActivity; import androidx.appcompat.widget.AppCompatButton; import android.os.Bundle; import android.widget.EditText; import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

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

EditText username, password;

AppCompatButton loginBtn;

username = findViewById(R.id.username); password = findViewById(R.id.password); loginBtn = findViewById(R.id.loginBtn);

loginBtn.setOnClickListener(view -> { if (username.getText().toString().equals("mca") && password.getText().toString().equals("android")) {

Toast.makeText(MainActivity.this, "Login Successfully",

Toast.LENGTH\_SHORT).show();

} else {

Toast.makeText(MainActivity.this, "Login failed",

Toast.LENGTH\_SHORT).show();

}

});

}

}

Q.6

## activity\_main.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" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" tools:context=".MainActivity" android:orientation="vertical" android:weightSum="10">

<FrameLayout android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:id="@+id/containerFrame" android:layout\_weight="0.7"/> <LinearLayout android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="bottom" android:layout\_weight="9.3">

<ImageButton android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:src="@drawable/ic\_baseline\_home\_24" android:layout\_weight="1" android:id="@+id/homeBtn"/>

<ImageButton android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:src="@drawable/ic\_baseline\_ondemand\_video\_24" android:layout\_weight="1" android:id="@+id/reelsBtn"/>

<ImageButton android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:src="@drawable/ic\_baseline\_notifications\_24" android:layout\_weight="1" android:id="@+id/notificationBtn"/>

<ImageButton android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:src="@drawable/ic\_baseline\_person\_24" android:layout\_weight="1" android:id="@+id/profileBtn"/>

</LinearLayout>

</LinearLayout>

## home\_fragment.xml

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" tools:context=".HomeFragment" android:background="#9CCC65">

<TextView android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:text="Home Page" android:gravity="center" android:textStyle="bold" android:textSize="25sp"/>

</LinearLayout>

## profile\_fragment.xml

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" tools:context=".ProfileFragment" android:background="#FF7043">

<TextView android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:text="Profile Page" android:textSize="25sp" android:textStyle="bold" android:gravity="center"/>

</LinearLayout>

## reels\_fragment.xml

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" tools:context=".ReelsFragment" android:background="#29B6F6">

<TextView android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:text="Reels Page" android:textStyle="bold" android:textSize="25sp" android:gravity="center"/>

</LinearLayout>

## notification\_fragment.xml

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" tools:context=".NotificationsFragment"

android:background="#FFEE58">

<TextView android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:text="Notification Page" android:gravity="center" android:textStyle="bold" android:textSize="25sp"/>

</LinearLayout>

**MainActivity.java** package com.subhdroid.LAB\_J10;

//10. Create a fragment that has its own UI and enable your activities to communicate with

// fragments.

import androidx.appcompat.app.AppCompatActivity; import androidx.fragment.app.Fragment; import androidx.fragment.app.FragmentManager; import androidx.fragment.app.FragmentTransaction;

import android.os.Bundle; import android.widget.ImageButton;

public class MainActivity extends AppCompatActivity {

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

ImageButton homeBtn, reelsBtn, notificationBtn, profileBtn; homeBtn = findViewById(R.id.homeBtn); reelsBtn = findViewById(R.id.reelsBtn); notificationBtn = findViewById(R.id.notificationBtn); profileBtn = findViewById(R.id.profileBtn);

loadFragment(new NotificationsFragment(), 0);

homeBtn.setOnClickListener(view -> loadFragment(new HomeFragment(), 1));

reelsBtn.setOnClickListener(view -> loadFragment(new ReelsFragment(), 1));

notificationBtn.setOnClickListener(view -> loadFragment(new NotificationsFragment(), 1));

profileBtn.setOnClickListener(view -> loadFragment(new ProfileFragment(), 1));

}

public void loadFragment(Fragment fragment, int flag) { FragmentManager fm = getSupportFragmentManager(); FragmentTransaction ft = fm.beginTransaction();

if (flag == 0) ft.add(R.id.containerFrame, fragment); else ft.replace(R.id.containerFrame, fragment);

ft.commit();

}

}

**HomeFragment.java** package com.subhdroid.LAB\_J10;

import android.os.Bundle;

import androidx.fragment.app.Fragment;

import android.view.LayoutInflater; import android.view.View; import android.view.ViewGroup;

public class HomeFragment extends Fragment {

@Override public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) { // Inflate the layout for this fragment return inflater.inflate(R.layout.fragment\_home, container, false);

}

}

**ProfileFragment.java** package com.subhdroid.LAB\_J10;

import android.os.Bundle;

import androidx.fragment.app.Fragment;

import android.view.LayoutInflater; import android.view.View; import android.view.ViewGroup;

public class ProfileFragment extends Fragment {

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) { // Inflate the layout for this fragment return inflater.inflate(R.layout.fragment\_profile, container, false);

}

}

**ReelsFragment.java** package com.subhdroid.LAB\_J10;

import android.os.Bundle;

import androidx.fragment.app.Fragment;

import android.view.LayoutInflater; import android.view.View; import android.view.ViewGroup;

public class ReelsFragment extends Fragment {

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) { // Inflate the layout for this fragment return inflater.inflate(R.layout.fragment\_reels, container, false);

}

}

**NotificationFragment.java** package com.subhdroid.LAB\_J10;

import android.os.Bundle;

import androidx.fragment.app.Fragment;

import android.view.LayoutInflater; import android.view.View; import android.view.ViewGroup;

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) { // Inflate the layout for this fragment return inflater.inflate(R.layout.fragment\_notifications, container, false);

}

}

public class NotificationsFragment extends Fragment {

Q.7

## activity\_main.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" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:orientation="vertical" android:gravity="center" tools:context=".MainActivity">

<ListView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:id="@+id/lstView"/>

</LinearLayout>

**MainActivity.java** package com.subhdroid.lab\_j11;

//11. Demonstrate Array Adapter using List View to display list of fruits.

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle; import android.widget.ArrayAdapter; import android.widget.ListView; public class MainActivity extends AppCompatActivity {

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

ListView lstView = findViewById(R.id.lstView);

String fruits[] = {"Apple","Banana","Orange","Mango","Dragan"};

ArrayAdapter adapter = new ArrayAdapter(MainActivity.this, android.R.layout.simple\_list\_item\_1,fruits);

lstView.setAdapter(adapter);

}

}

Q.8

## activity\_main.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" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="center" android:orientation="vertical" tools:context=".MainActivity">

<TextView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Alert Dialog Box" />

</LinearLayout>

**MainActivity.java** package com.subhdroid.lab\_j12;

//12. Write an application to demonstrate Alert Dialog Box in android

import androidx.appcompat.app.AlertDialog; import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

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

}

@Override

public void onBackPressed() {

AlertDialog.Builder alertBox = new AlertDialog.Builder(MainActivity.this); alertBox.setTitle("Exit");

alertBox.setMessage("Are you sure want to exit?"); alertBox.setPositiveButton("Yes", (dialogInterface, i) -> finishAffinity());

alertBox.setNegativeButton("No", (dialogInterface, i) -> dialogInterface.dismiss()); alertBox.show();

}

}

Q.9

## activity\_main.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" xmlns:tools="http://schemas.android.com/tools" android:id="@+id/ll" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="center" android:orientation="vertical" tools:context=".MainActivity">

<TextView android:id="@+id/contextMenuTxt" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Context Menu(Long press me)" />

<androidx.appcompat.widget.AppCompatButton android:id="@+id/popupMenuBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_marginTop="21dp" android:text="Popup Menu" />

</LinearLayout>

## menu.xml

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

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

<item android:id="@+id/save" android:title="Save" />

<item android:id="@+id/open" android:title="Open" />

<item android:id="@+id/close" android:title="Close" />

<item android:id="@+id/exit" android:title="Exit" />

</menu>

**MainActivity.java** package com.subhdroid.lab\_j13;

//13. Demonstrate Options Menu, Context Menu and Popup Menu in android

import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity; import androidx.appcompat.widget.AppCompatButton; import androidx.appcompat.widget.PopupMenu;

import android.graphics.Color; import android.os.Bundle; import android.view.ContextMenu; import android.view.Menu; import android.view.MenuInflater; import android.view.MenuItem; import android.view.View; import android.widget.LinearLayout; import android.widget.TextView; import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

LinearLayout ll;

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

AppCompatButton popupMenuBtn = findViewById(R.id.popupMenuBtn); TextView contextMenuTxt = findViewById(R.id.contextMenuTxt); ll = findViewById(R.id.ll); registerForContextMenu(contextMenuTxt); popupMenuBtn.setOnClickListener(view -> {

PopupMenu popupMenu = new PopupMenu(MainActivity.this, popupMenuBtn); popupMenu.getMenuInflater().inflate(R.menu.menus\_items, popupMenu.getMenu());

popupMenu.setOnMenuItemClickListener(item -> {

Toast.makeText(MainActivity.this, item.getTitle() + " clicked", Toast.LENGTH\_SHORT).show(); return true;

});

popupMenu.show();

});

}

@Override

public boolean onCreateOptionsMenu(Menu menu) { MenuInflater inflater = getMenuInflater(); inflater.inflate(R.menu.menus\_items, menu); return true;

}

@Override public boolean onOptionsItemSelected(@NonNull MenuItem item) { if (item.getItemId() == R.id.close) {

Toast.makeText(this, "Close clicked",

Toast.LENGTH\_SHORT).show();

}

if (item.getItemId() == R.id.save) {

Toast.makeText(this, "Save clicked", Toast.LENGTH\_SHORT).show();

}

if (item.getItemId() == R.id.open) {

Toast.makeText(this, "Open clicked",

Toast.LENGTH\_SHORT).show();

} return true;

}

@Override

public void onCreateContextMenu(ContextMenu menu, View v, ContextMenu.ContextMenuInfo menuInfo) { super.onCreateContextMenu(menu, v, menuInfo); menu.setHeaderTitle("Set Background color"); menu.add(0, v.getId(), 0, "Grey"); menu.add(0, v.getId(), 0, "Yellow"); menu.add(0, v.getId(), 0, "Red");

}

@Override public boolean onContextItemSelected(@NonNull MenuItem item) { if (item.getTitle().equals("Grey")) { ll.setBackgroundColor(Color.GRAY);

}

if (item.getTitle().equals("Yellow")) { ll.setBackgroundColor(Color.YELLOW);

}

if (item.getTitle().equals("Red")) {

ll.setBackgroundColor(Color.RED);

} return true;

}

}

Q.10

## activity\_main.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" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="center" tools:context=".MainActivity">

<androidx.appcompat.widget.AppCompatButton android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Notify me" android:id="@+id/btn"/>

</LinearLayout>

## activity\_new.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" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="center" tools:context=".NewActivity">

<TextView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="New Activity" />

</LinearLayout>

**MainActivity.java** package com.subhdroid.lab\_j14;

//14. Write an application to produce Notification

import androidx.appcompat.app.AppCompatActivity;

import androidx.core.app.NotificationCompat;

import android.app.NotificationChannel; import android.app.NotificationManager; import android.app.PendingIntent; import android.content.Intent; import android.os.Build; import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

private static final String CHANNEL\_ID = "Notification Channel"; private static final int REQ\_CODE = 100;

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

findViewById(R.id.btn).setOnClickListener(view -> {

NotificationManager nm = (NotificationManager) getSystemService(NOTIFICATION\_SERVICE);

NotificationCompat.Builder nb = new NotificationCompat.Builder(this,

CHANNEL\_ID);

Intent intent = new Intent(MainActivity.this, NewActivity.class);

PendingIntent pendingIntent = PendingIntent.getActivity(this, REQ\_CODE, intent,

PendingIntent.FLAG\_UPDATE\_CURRENT);

if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.O) { nb.setSmallIcon(R.drawable.ic\_launcher\_background)

.setContentTitle("New Message Title")

.setContentText("Context text")

.setSubText("Subtext")

.setContentIntent(pendingIntent)

.setChannelId(CHANNEL\_ID)

.build();

nm.createNotificationChannel((new

NotificationChannel(CHANNEL\_ID, "Channel One",

NotificationManager.IMPORTANCE\_HIGH)));

} else {

nb.setSmallIcon(R.drawable.ic\_launcher\_background)

.setContentTitle("New Message Title")

.setContentText("Context text")

.setSubText("Subtext")

.setContentIntent(pendingIntent)

.setChannelId(CHANNEL\_ID)

.build();

}

nm.notify(1, nb.build());

});

}

}

**NewActivity.java** package com.subhdroid.lab\_j14;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class NewActivity extends AppCompatActivity {

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

}

}

Q.11

## activity\_main.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" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="center" android:orientation="vertical" tools:context=".MainActivity">

<EditText android:id="@+id/name" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_marginTop="11dp" android:hint="Course name" android:inputType="text" />

<EditText android:id="@+id/duration" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_marginTop="11dp" android:hint="Duration (in year)" android:inputType="number" />

<EditText android:id="@+id/description" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_marginTop="11dp" android:hint="Description" />

<androidx.appcompat.widget.AppCompatButton android:id="@+id/addBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Add Course" />

<LinearLayout android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:gravity="center" android:orientation="horizontal">

<androidx.appcompat.widget.AppCompatButton android:id="@+id/updateBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Update" />

<androidx.appcompat.widget.AppCompatButton android:id="@+id/deleteBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Delete" />

<androidx.appcompat.widget.AppCompatButton android:id="@+id/displayBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Display" />

</LinearLayout>

<TextView android:id="@+id/txtView" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" />

</LinearLayout>

**MainActivity.java** package com.subhdroid.lab\_j15;

//15. Write an android application using SQLite to create table and perform CRUD operations

// (Example. COURSE table (ID, Name, Duration, Description), perform

ADD, UPDATE,

// DELETE and READ operations)

import androidx.appcompat.app.AppCompatActivity;

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

import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {

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

MyDBClass mydb = new MyDBClass(this);

EditText name, duration, description; TextView txt = findViewById(R.id.txtView); name = findViewById(R.id.name); duration = findViewById(R.id.duration); description = findViewById(R.id.description);

findViewById(R.id.addBtn).setOnClickListener(view -> mydb.addRecord(name.getText().toString(), duration.getText().toString(), description.getText().toString()));

findViewById(R.id.updateBtn).setOnClickListener(view -> mydb.updateRecord(duration.getText().toString(), name.getText().toString())); findViewById(R.id.deleteBtn).setOnClickListener(view -> mydb.deleteRecord(name.getText().toString()));

findViewById(R.id.displayBtn).setOnClickListener(view -> {

ArrayList<CourseModel> list = mydb.getRecords();

String str = "ID Name Duration Description";

for (int i = 0; i < list.size(); i++) {

str += "\n" + list.get(i).id + " " + list.get(i).name + " " + list.get(i).duration + " " + list.get(i).description;

}

txt.setText(str);

});

}

}

**CourseModel.java** package com.subhdroid.lab\_j15;

public class CourseModel {

String name, duration, description;

int id;

public CourseModel() {

}

}

**MyDBClass.java** package com.subhdroid.lab\_j15;

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;

import androidx.annotation.Nullable;

import java.util.ArrayList;

public class MyDBClass extends SQLiteOpenHelper { private static final String DBName = "LabDB"; private static final int DB\_VERSION = 1;

Context context;

public MyDBClass(@Nullable Context context) { super(context, DBName, null, DB\_VERSION); this.context = context;

}

@Override public void onCreate(SQLiteDatabase sqLiteDatabase) {

sqLiteDatabase.execSQL("CREATE TABLE course(id INTEGER PRIMARY KEY AUTOINCREMENT,name " +

"TEXT,duration TEXT,description TEXT)");

}

@Override public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {

}

public void addRecord(String name, String duration, String description) {

SQLiteDatabase database = this.getWritableDatabase();

ContentValues values = new ContentValues(); values.put("name", name); values.put("duration", duration); values.put("description", description);

database.insert("course", null, values);

Toast.makeText(context, "Added successfully",

Toast.LENGTH\_SHORT).show();

// database.close();

}

public ArrayList<CourseModel> getRecords() { SQLiteDatabase db = this.getReadableDatabase();

Cursor cursor = db.rawQuery("SELECT \* FROM course", null);

ArrayList<CourseModel> recordList = new ArrayList<>();

while (cursor.moveToNext()) {

CourseModel model = new CourseModel();

model.id = cursor.getInt(0); model.name = cursor.getString(1); model.duration = cursor.getString(2); model.description = cursor.getString(3);

recordList.add(model);

} return recordList;

}

public void updateRecord(String duration, String name) { SQLiteDatabase db = this.getWritableDatabase();

ContentValues cv = new ContentValues(); cv.put("duration", duration);

db.update("course", cv, "name=?", new String[]{name});

Toast.makeText(context, "Updated successfully",

Toast.LENGTH\_SHORT).show();

}

public void deleteRecord(String courseName) { SQLiteDatabase database = this.getWritableDatabase();

database.delete("course", "name=?", new String[]{courseName});

Toast.makeText(context, "Deleted successfully",

Toast.LENGTH\_SHORT).show();

}

}

Q.12

**Manifest.xml**

## activity\_main.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" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="center" android:orientation="vertical" tools:context=".MainActivity">

<EditText android:id="@+id/name" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_marginTop="11dp" android:hint="Course name" android:inputType="text" />

<EditText android:id="@+id/duration" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content"

android:layout\_marginTop="11dp" android:hint="Duration (in year)" android:inputType="number" />

<EditText android:id="@+id/description" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_marginTop="11dp" android:hint="Description" />

<androidx.appcompat.widget.AppCompatButton android:id="@+id/addBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Add Course" />

<LinearLayout android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:gravity="center" android:orientation="horizontal">

<androidx.appcompat.widget.AppCompatButton android:id="@+id/updateBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Update" />

<androidx.appcompat.widget.AppCompatButton android:id="@+id/deleteBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Delete" />

<androidx.appcompat.widget.AppCompatButton android:id="@+id/displayBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Display" />

</LinearLayout>

<TextView android:id="@+id/txtView" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" />

</LinearLayout>

**MainActivity.java** package com.subhdroid.lab\_j16;

import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle; import android.os.Handler; import android.util.Log; import android.widget.EditText; import android.widget.TextView; import android.widget.Toast;

import com.google.firebase.database.DataSnapshot; import com.google.firebase.database.DatabaseError; import com.google.firebase.database.DatabaseReference; import com.google.firebase.database.FirebaseDatabase; import com.google.firebase.database.ValueEventListener;

import java.sql.Array; import java.util.HashMap;

public class MainActivity extends AppCompatActivity {

DatabaseReference courseRef =

FirebaseDatabase.getInstance().getReference("course");

EditText name, duration, description;

TextView txt;

String record = "";

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

txt = findViewById(R.id.txtView); name = findViewById(R.id.name); duration = findViewById(R.id.duration); description = findViewById(R.id.description);

findViewById(R.id.addBtn).setOnClickListener(view -> addRecord());

findViewById(R.id.updateBtn).setOnClickListener(view -> updateRecord());

findViewById(R.id.deleteBtn).setOnClickListener(view -> deleteRecord());

findViewById(R.id.displayBtn).setOnClickListener(view -> { getAllCourse();

Handler handler = new Handler(); handler.postDelayed(new Runnable() {

@Override public void run() { txt.setText(record);

}

}, 3000);

});

}

private void addRecord() {

CourseModel courseModel = new CourseModel(name.getText().toString(), duration.getText().toString(), description.getText().toString());

String courseID = courseRef.push().getKey();

courseRef.child(courseID).setValue(courseModel);

Toast.makeText(this, "Course added", Toast.LENGTH\_SHORT).show();

}

private void deleteRecord() { courseRef.addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

HashMap<String, Array> dataMap = (HashMap<String, Array>) dataSnapshot.getValue(); for (String key : dataMap.keySet()) {

courseRef.child(key).addValueEventListener(new ValueEventListener() {

@Override public void onDataChange(@NonNull DataSnapshot snapshot) {

CourseModel course = snapshot.getValue(CourseModel.class);

if (name.getText().toString().equals(course.getName())) { snapshot.getRef().removeValue();

Toast.makeText(MainActivity.this, "Record deleted",

Toast.LENGTH\_SHORT).show();

}

}

@Override

public void onCancelled(@NonNull DatabaseError error) {

Log.d("DB Error : ", error.toString());

}

});

}

}

@Override

public void onCancelled(@NonNull DatabaseError error) {

Toast.makeText(getApplicationContext(), "Fail to get data.",

Toast.LENGTH\_SHORT).show();

}

});

}

private void updateRecord() { courseRef.addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

HashMap<String, Array> dataMap = (HashMap<String, Array>) dataSnapshot.getValue(); for (String key : dataMap.keySet()) {

courseRef.child(key).addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot snapshot) {

CourseModel course = snapshot.getValue(CourseModel.class); if (name.getText().toString().equals(course.getName())) {

courseRef.child(key).child("duration").setValue(duration.getText().toString());

Toast.makeText(MainActivity.this, "Record Updated",

Toast.LENGTH\_SHORT).show();

}

}

@Override

public void onCancelled(@NonNull DatabaseError error) { Log.d("DB Error : ", error.toString());

}

});

}

}

@Override

public void onCancelled(@NonNull DatabaseError error) {

Toast.makeText(getApplicationContext(), "Fail to get data.",

Toast.LENGTH\_SHORT).show();

}

});

}

private void getAllCourse() {

courseRef.addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

HashMap<String, Array> dataMap = (HashMap<String, Array>) dataSnapshot.getValue();

record = ""; for (String key : dataMap.keySet()) {

courseRef.child(key).addValueEventListener(new ValueEventListener() {

@Override public void onDataChange(@NonNull DataSnapshot snapshot) {

CourseModel course = snapshot.getValue(CourseModel.class);

String str = "\n" + course.getName() + " " + course.getDuration() +

" " + course.getDescription();

record += str;

}

@Override

public void onCancelled(@NonNull DatabaseError error) {

Log.d("DB Error : ", error.toString());

}

});

}

}

@Override

public void onCancelled(@NonNull DatabaseError error) {

Toast.makeText(getApplicationContext(), "Fail to get data.",

Toast.LENGTH\_SHORT).show();

}

});

}

}

**CourseModel.java** package com.subhdroid.lab\_j16;

public class CourseModel {

String name, duration, description;

CourseModel(String name, String duration, String description) { this.name = name; this.duration = duration; this.description = description;

}

public CourseModel() {

}

public String getName() { return name;

}

public void setName(String name) { this.name = name;

}

public String getDuration() { return duration;

}

public void setDuration(String duration) { this.duration = duration;

}

public String getDescription() { return description;

}

public void setDescription(String description) { this.description = description;

}

}

Q.13

**Manifest.xml**

<uses-permission android:name="android.permission.INTERNET"/>

## activity\_main.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" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="center" android:orientation="vertical" tools:context=".MainActivity">

<WebView android:id="@+id/webView" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" /> <androidx.appcompat.widget.AppCompatButton android:id="@+id/btn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Go to Google" />

<ProgressBar android:id="@+id/pgBar" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:visibility="gone" />

</LinearLayout>

**MainActivity.java** package com.subhdroid.lab\_j17;

//17. Demonstrate WebView to display the web pages in an android application.

import androidx.appcompat.app.AppCompatActivity; import androidx.appcompat.widget.AppCompatButton;

import android.graphics.Bitmap; import android.os.Bundle; import android.view.View; import android.webkit.WebView; import android.webkit.WebViewClient; import android.widget.ProgressBar;

public class MainActivity extends AppCompatActivity {

WebView webView;

ProgressBar pgBar;

AppCompatButton btn;

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

webView = findViewById(R.id.webView); pgBar = findViewById(R.id.pgBar); btn = findViewById(R.id.btn); btn.setOnClickListener(new View.OnClickListener() {

@Override public void onClick(View view) { webView.loadUrl("https://www.google.com"); pgBar.setVisibility(View.VISIBLE); webView.setWebViewClient(new WebViewClient() {

@Override public void onPageStarted(WebView view, String url, Bitmap favicon) { super.onPageStarted(view, url, favicon);

}

@Override

public void onPageFinished(WebView view, String url) { pgBar.setVisibility(View.GONE); btn.setVisibility(View.GONE); super.onPageFinished(view, url);

}

});

}

});

}

@Override

public void onBackPressed() { if (webView.canGoBack()) { webView.goBack();

}

else

{

super.onBackPressed();

}

}

}

Q.14

## activity\_main.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" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="center" android:orientation="vertical" tools:context=".MainActivity">

<EditText android:id="@+id/name" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:hint="Enter name" />

<EditText android:id="@+id/mobile" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:hint="Enter phone no" />

<androidx.appcompat.widget.AppCompatButton android:id="@+id/setDataBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Set Data" />

<androidx.appcompat.widget.AppCompatButton android:id="@+id/getDataBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Get Data" />

<TextView android:id="@+id/data" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" />

</LinearLayout>

## MainActivity.java

package com.subhdroid.lab\_j18;

//18. Write an android app to write JSON data into a file and read JSON data from created file.

import androidx.appcompat.app.AppCompatActivity;

import androidx.appcompat.widget.AppCompatButton;

import android.os.Bundle; import android.widget.EditText; import android.widget.TextView; import android.widget.Toast;

import org.json.JSONException; import org.json.JSONObject;

import java.io.BufferedReader; import java.io.BufferedWriter; import java.io.File; import java.io.FileReader; import java.io.FileWriter; import java.io.IOException;

public class MainActivity extends AppCompatActivity {

AppCompatButton setDataBtn, getDataBtn;

EditText name, mobile;

TextView data;

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

name = findViewById(R.id.name); mobile = findViewById(R.id.mobile); data = findViewById(R.id.data);

setDataBtn = findViewById(R.id.setDataBtn); getDataBtn = findViewById(R.id.getDataBtn);

setDataBtn.setOnClickListener(view -> setData());

getDataBtn.setOnClickListener(view -> getData());

}

private void setData() {

JSONObject jsonObject = new JSONObject(); try {

jsonObject.put("Name", name.getText().toString()); jsonObject.put("Phone", mobile.getText().toString()); } catch (JSONException e) { e.printStackTrace();

}

String userString = jsonObject.toString(); try {

File file = new File(getApplicationContext().getFilesDir(),

"LAB\_J18.json");

FileWriter fileWriter = new FileWriter(file);

BufferedWriter bufferedWriter = new BufferedWriter(fileWriter); bufferedWriter.write(userString);

bufferedWriter.close(); } catch (IOException e) { e.printStackTrace();

}

Toast.makeText(this, "Data Set", Toast.LENGTH\_SHORT).show();

}

private void getData() {

try {

File file = new File(getApplicationContext().getFilesDir(), "LAB\_J18.json");

FileReader fileReader = new FileReader(file);

BufferedReader bufferedReader = new BufferedReader(fileReader);

StringBuilder stringBuilder = new StringBuilder(); String line = bufferedReader.readLine(); while (line != null) { stringBuilder.append(line).append("\n"); line = bufferedReader.readLine();

}

bufferedReader.close();

String response = stringBuilder.toString();

JSONObject jsonObject = new JSONObject(response); String rec = "Name : " + jsonObject.get("Name"); rec += "\nPhone : " + jsonObject.get("Phone"); data.setText(rec);

} catch (IOException e) { e.printStackTrace();

} catch (JSONException e) { e.printStackTrace();

}

}

**}**