

Mobile Application Development Labs

MAD LAB 1

MainActivity.java

```
package com.example.mad_lab_1;

import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

activity_main.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="match_parent"
    android:gravity="center" android:orientation="vertical">

    <TextView android:id="@+id/helloText" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:text="Hello, World!"
        android:textSize="24sp" />
</LinearLayout>
```

MAD LAB 2

MainActivity.java

```
package com.example.mad_lab_2;

import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
```

```

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    EditText edtNumber1, edtNumber2;
    Button btnAdd, btnSubtract, btnMultiply, btnDivide;
    TextView txtResult;

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

        // Initialize UI components
        edtNumber1 = findViewById(R.id.edtNumber1);
        edtNumber2 = findViewById(R.id.edtNumber2);
        btnAdd = findViewById(R.id.btnAdd);
        btnSubtract = findViewById(R.id.btnSubtract);
        btnMultiply = findViewById(R.id.btnMultiply);
        btnDivide = findViewById(R.id.btnDivide);
        txtResult = findViewById(R.id.txtResult);

        // Set event listeners for button clicks
        btnAdd.setOnClickListener(view -> calculate('+'));
        btnSubtract.setOnClickListener(view -> calculate('-'));
        btnMultiply.setOnClickListener(view -> calculate('*'));
        btnDivide.setOnClickListener(view -> calculate('/'));
    }

    // Method to perform calculations
    private void calculate(char operator) {
        String num1Str = edtNumber1.getText().toString();
        String num2Str = edtNumber2.getText().toString();

        if (num1Str.isEmpty() || num2Str.isEmpty()) {
            txtResult.setText("Please enter both numbers!");
            return;
        }

        double num1 = Double.parseDouble(num1Str);
        double num2 = Double.parseDouble(num2Str);
        double result = 0;

        switch (operator) {
            case '+':
                result = num1 + num2;
                break;
            case '-':
                result = num1 - num2;
                break;
            case '*':
                result = num1 * num2;
                break;
            case '/':
                if (num2 == 0) {

```

```

        txtResult.setText("Cannot divide by zero!");
        return;
    }
    result = num1 / num2;
    break;
}

txtResult.setText("Result: " + result);
}
}

```

activity_main.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="match_parent"
    android:orientation="vertical" android:padding="16dp"
    android:gravity="center">

    <EditText android:id="@+id/edtNumber1" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:hint="Enter first number"
        android:inputType="numberDecimal" />

    <EditText android:id="@+id/edtNumber2" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:hint="Enter second number"
        android:inputType="numberDecimal" />

    <LinearLayout android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal" android:gravity="center">

        <Button android:id="@+id/btnAdd" android:layout_width="wrap_content"
            android:layout_height="wrap_content" android:text="+" />

        <Button android:id="@+id/btnSubtract" android:layout_width="wrap_content"
            android:layout_height="wrap_content" android:text="-" />

        <Button android:id="@+id/btnMultiply" android:layout_width="wrap_content"
            android:layout_height="wrap_content" android:text="x" />

        <Button android:id="@+id/btnDivide" android:layout_width="wrap_content"
            android:layout_height="wrap_content" android:text="÷" />
    </LinearLayout>

    <TextView android:id="@+id/txtResult" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:text="Result: "
        android:textSize="20sp"
        android:textStyle="bold" android:gravity="center"
        android:paddingTop="16dp" />
</LinearLayout>

```

MAD LAB 3

MainActivity.java

```
package com.example.mad_lab_3;

import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

activity_main.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="match_parent"
    android:orientation="vertical" android:gravity="center">

    <com.example.mad_lab_3.DrawingView android:id="@+id/drawingView"
        android:layout_width="match_parent" android:layout_height="match_parent"
    />

</LinearLayout>
```

DrawingView.java

```
package com.example.mad_lab_3;

import android.content.Context;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.util.AttributeSet;
import android.view.View;

import androidx.annotation.NonNull;
```

```

public class DrawingView extends View {
    private final Paint paint;

    public DrawingView(Context context, AttributeSet attrs) {
        super(context, attrs);
        paint = new Paint();
        paint.setAntiAlias(true);
        paint.setStrokeWidth(5);
    }

    @Override
    protected void onDraw(@NonNull Canvas canvas) {
        super.onDraw(canvas);

        // Set black background
        canvas.drawColor(Color.BLACK);

        // Draw a blue filled circle
        paint.setColor(Color.BLUE);
        paint.setStyle(Paint.Style.FILL);
        canvas.drawCircle(150, 150, 50, paint); // (x=150, y=150)

        // Draw a red outlined rectangle
        paint.setColor(Color.RED);
        paint.setStyle(Paint.Style.STROKE);
        canvas.drawRect(300, 100, 400, 300, paint); // (left=300, top=100,
right=400, bottom=300)

        // Draw a smaller blue outlined rectangle
        paint.setColor(Color.BLUE);
        paint.setStyle(Paint.Style.STROKE);
        canvas.drawRect(100, 400, 200, 500, paint); // (left=100, top=400,
right=200, bottom=500)

        // Draw a yellow horizontal line
        paint.setColor(Color.YELLOW);
        paint.setStyle(Paint.Style.STROKE);
        canvas.drawLine(300, 400, 500, 400, paint); // (startX=300, startY=400,
stopX=500, stopY=400)

        // Draw a yellow vertical line
        paint.setColor(Color.YELLOW);
        paint.setStyle(Paint.Style.STROKE);
        canvas.drawLine(600, 100, 600, 300, paint); // (startX=600, startY=100,
stopX=600, stopY=300)

        // Draw a smaller blue outlined circle
        paint.setColor(Color.BLUE);
        paint.setStyle(Paint.Style.STROKE);
        canvas.drawCircle(450, 450, 40, paint); // (x=450, y=450)
    }
}

```

MAD LAB 4

MainActivity.java

```
package com.example.mad_lab_4;

import android.database.Cursor;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    DatabaseHelper dbHelper;
    EditText etName, etAge;
    Button btnSave, btnView;
    TextView tvResult;

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

        dbHelper = new DatabaseHelper(this);

        etName = findViewById(R.id.etName);
        etAge = findViewById(R.id.etAge);
        btnSave = findViewById(R.id.btnSave);
        btnView = findViewById(R.id.btnView);
        tvResult = findViewById(R.id.tvResult);

        // Save data
        btnSave.setOnClickListener(view -> {
            String name = etName.getText().toString();
            String ageText = etAge.getText().toString();

            if (name.isEmpty() || ageText.isEmpty()) {
                Toast.makeText(this, "Please fill all fields",
                    Toast.LENGTH_SHORT).show();
            } else {
                int age = Integer.parseInt(ageText);
                boolean isInserted = dbHelper.insertData(name, age);
                if (isInserted) {
                    Toast.makeText(this, "Data saved successfully",
                        Toast.LENGTH_SHORT).show();
                } else {
                    Toast.makeText(this, "Failed to save data",
                        Toast.LENGTH_SHORT).show();
                }
            }
        })
    }
}
```

```

    }
});

// View data
btnView.setOnClickListener(view -> {
    Cursor cursor = dbHelper.getAllData();
    if (cursor.getCount() == 0) {
        tvResult.setText("No data found");
        return;
    }

    StringBuilder builder = new StringBuilder();
    while (cursor.moveToNext()) {
        builder.append("ID: ").append(cursor.getInt(0)).append("\n");
        builder.append("Name: ").append(cursor.getString(1)).append("\n");
        builder.append("Age: ").append(cursor.getInt(2)).append("\n\n");
    }
    tvResult.setText(builder.toString());
});
}
}

```

activity_main.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="match_parent"
    android:orientation="vertical" android:gravity="center"
    android:padding="16dp">

    <EditText android:id="@+id/etName" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:hint="Enter Name"
    android:inputType="text"
        android:layout_marginBottom="10dp" />

    <EditText android:id="@+id/etAge" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:hint="Enter Age"
    android:inputType="number"
        android:layout_marginBottom="20dp" />

    <Button android:id="@+id/btnSave" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:text="Save Data"
        android:layout_marginBottom="10dp" />

    <Button android:id="@+id/btnView" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:text="View Data"
        android:layout_marginBottom="20dp" />

    <TextView android:id="@+id/tvResult" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:text="Result will appear

```

```
here"
        android:padding="10dp" />
</LinearLayout>
```

DatabaseHelper.java

```
package com.example.mad_lab_4;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DatabaseHelper extends SQLiteOpenHelper {

    public DatabaseHelper(Context context) {
        super(context, "UserDB", null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("CREATE TABLE Users (ID INTEGER PRIMARY KEY AUTOINCREMENT, Name TEXT, Age INTEGER)");
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS Users");
        onCreate(db);
    }

    // Method to insert data
    public boolean insertData(String name, int age) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put("Name", name);
        contentValues.put("Age", age);
        long result = db.insert("Users", null, contentValues);
        return result != -1; // Return true if data is inserted successfully
    }

    // Method to fetch all data
    public Cursor getAllData() {
        SQLiteDatabase db = this.getReadableDatabase();
        return db.rawQuery("SELECT * FROM Users", null);
    }
}
```


MAD LAB 5

MainActivity.java

```
package com.example.mad_lab_5;

import android.app.Notification;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.content.Context;
import android.os.Bundle;
import android.os.Handler;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.NotificationCompat;

public class MainActivity extends AppCompatActivity {
    private static final String CHANNEL_ID = "alarm_channel";
    private EditText secondsInput;
    private Button setAlarmButton;
    private Handler handler;

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

        // Initialize UI components
        secondsInput = findViewById(R.id.seconds_input);
        setAlarmButton = findViewById(R.id.set_alarm_button);

        // Set up handler
        handler = new Handler();

        // Create a notification channel (needed for Android 8.0 and above)
        createNotificationChannel();

        // Button click listener
        setAlarmButton.setOnClickListener(v -> {
            String secondsStr = secondsInput.getText().toString();
            if (!secondsStr.isEmpty()) {
                int seconds = Integer.parseInt(secondsStr);
                Toast.makeText(MainActivity.this, "Alarm Set for " + seconds + "
seconds", Toast.LENGTH_SHORT).show();
                startAlarm(seconds);
            } else {
                Toast.makeText(MainActivity.this, "Please enter a number of
seconds", Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

```

    }

    private void startAlarm(int seconds) {
        // Delay the alarm by the entered number of seconds
        handler.postDelayed(this::showToastAndNotification, seconds * 1000L);
    }

    private void showToastAndNotification() {
        // Show a Toast message
        Toast.makeText(MainActivity.this, "Alarm triggered!",
            Toast.LENGTH_SHORT).show();

        // Show a Notification
        NotificationManager notificationManager = (NotificationManager)
            getSystemService(Context.NOTIFICATION_SERVICE);
        Notification notification = new
            NotificationCompat.Builder(MainActivity.this, CHANNEL_ID)
                .setContentTitle("Alarm")
                .setContentText("Your alarm has been triggered!")
                .setSmallIcon(R.mipmap.ic_launcher_foreground)
                .build();

        // Display the notification
        notificationManager.notify(1, notification);
    }

    private void createNotificationChannel() {
        // Notification Channel for Android 8.0 and above
        if (android.os.Build.VERSION.SDK_INT >= android.os.Build.VERSION_CODES.O)
        {
            CharSequence name = "Alarm Channel";
            String description = "Channel for alarm notifications";
            int importance = NotificationManager.IMPORTANCE_DEFAULT;
            NotificationChannel channel = new NotificationChannel(CHANNEL_ID,
                name, importance);
            channel.setDescription(description);
            NotificationManager notificationManager =
                getSystemService(NotificationManager.class);
            notificationManager.createNotificationChannel(channel);
        }
    }
}

```

activity_main.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="match_parent"
    android:orientation="vertical" android:gravity="center"
    android:padding="16dp">

```

```
<EditText android:id="@+id/seconds_input" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:hint="Enter seconds"
android:inputType="number"
    android:textAlignment="center" />

<Button android:id="@+id/set_alarm_button" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:text="Set Alarm"
    android:layout_marginTop="16dp" />

</LinearLayout>
```

MAD LAB 6

MainActivity.java

```
package com.example.mad_lab_6;

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

public class MainActivity extends AppCompatActivity {

    private TextView countdownText;
    private Button showButton;
    private int countdownValue = 10;

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

        // Initialize views
        countdownText = findViewById(R.id.countdownText);
        showButton = findViewById(R.id.showButton);

        // Start the countdown on a separate thread
        startCountdown();

        // Set up button click listener to show a toast when clicked
        showButton.setOnClickListener(v -> {
            // Show a toast when the button is clicked
            Toast.makeText(MainActivity.this, "Button Clicked!",
                Toast.LENGTH_SHORT).show();
        });
    }
}
```

```

}

private void startCountdown() {
    // Create a Handler to update the UI
    Handler handler = new Handler();

    // Create a new thread for the countdown
    Thread countdownThread = new Thread(() -> {
        while (countdownValue > 0) {
            try {
                // Wait for 1 second
                Thread.sleep(1000);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }

            // Update the countdown value
            countdownValue--;

            // Post the UI update to the main thread using Handler
            handler.post(() ->
countdownText.setText(String.valueOf(countdownValue)));
        }

        // Once countdown is complete, show the button
        handler.post(() -> showButton.setVisibility(View.VISIBLE));
    });

    // Start the countdown thread
    countdownThread.start();
}
}

```

activity_main.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="match_parent"
    android:orientation="vertical" android:gravity="center"
    android:padding="16dp">

    <!-- TextView to display countdown -->
    <TextView android:id="@+id/countdownText" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:text="10"
    android:textSize="48sp" />

    <!-- Button that will be shown after countdown -->
    <Button android:id="@+id/showButton" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:text="Click Me!"
    android:visibility="gone" />

```

```
</LinearLayout>
```

MAD LAB 7

MainActivity.java

```
package com.example.mad_lab_7;

import android.Manifest;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import com.google.android.gms.location.FusedLocationProviderClient;
import com.google.android.gms.location.LocationServices;

public class MainActivity extends AppCompatActivity {

    private FusedLocationProviderClient fusedLocationClient;
    private TextView locationText;

    // Request code for location permission
    private static final int REQUEST_LOCATION_PERMISSION = 1;

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

        // Initialize the fused location client
        fusedLocationClient =
            LocationServices.getFusedLocationProviderClient(this);

        // Initialize TextView for displaying the location
        locationText = findViewById(R.id.locationText);

        // Check for location permission
        if (ActivityCompat.checkSelfPermission(this,
            Manifest.permission.ACCESS_FINE_LOCATION) !=
            PackageManager.PERMISSION_GRANTED) {
            // Request permission if not granted
            ActivityCompat.requestPermissions(this, new String[] {
                Manifest.permission.ACCESS_FINE_LOCATION },
                REQUEST_LOCATION_PERMISSION);
        } else {
```

```

        // Permission granted, get the last location
        getLastLocation();
    }
}

private void getLastLocation() {
    // Get the last known location
    if (ActivityCompat.checkSelfPermission(this,
        Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED
        && ActivityCompat.checkSelfPermission(this,
            Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
        return;
    }
    fusedLocationClient.getLastLocation()
        .addOnSuccessListener(this, location -> {
            if (location != null) {
                // Successfully received location, display latitude and
longitude
                double latitude = location.getLatitude();
                double longitude = location.getLongitude();
                locationText.setText("Latitude: " + latitude +
"\nLongitude: " + longitude);
            } else {
                // Handle case where location is null (e.g., location
services are disabled)
                Toast.makeText(MainActivity.this, "Location not
available", Toast.LENGTH_SHORT).show();
            }
        });
}

@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[]
permissions,
    @NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
    if (requestCode == REQUEST_LOCATION_PERMISSION) {
        if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
            // Permission granted, get the location
            getLastLocation();
        } else {
            // Permission denied
            Toast.makeText(this, "Permission denied",
Toast.LENGTH_SHORT).show();
        }
    }
}
}

```

activity_main.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="match_parent"
    android:orientation="vertical" android:gravity="center"
    android:padding="16dp">

    <!-- TextView to display latitude and longitude -->
    <TextView android:id="@+id/locationText" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:text="Waiting for
location..."
        android:textSize="18sp" />

</LinearLayout>
```

MAD LAB 8

MainActivity.java

```
package com.example.mad_lab_8;

import android.os.Bundle;
import android.widget.Button;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;

public class MainActivity extends AppCompatActivity {

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

        Button writeButton = findViewById(R.id.writeButton);

        // Button click to write data to SD card (app-specific directory)
        writeButton.setOnClickListener(v -> writeDataToSDCard());
    }

    private void writeDataToSDCard() {
        // Get the app-specific directory on external storage
        File dir = getExternalFilesDir(null); // this returns a directory where
```

```

app data is stored
    if (dir != null) {
        File file = new File(dir, "examplefile.txt");

        try (FileOutputStream fos = new FileOutputStream(file)) {
            String data = "This is some sample data to be written to the SD
card. Hello World!";
            fos.write(data.getBytes());
            fos.flush();
            Toast.makeText(this, "Data written to SD card!",
Toast.LENGTH_SHORT).show();
        } catch (IOException e) {
            e.printStackTrace();
            Toast.makeText(this, "Error writing to SD card",
Toast.LENGTH_SHORT).show();
        }
    } else {
        Toast.makeText(this, "External storage not available",
Toast.LENGTH_SHORT).show();
    }
}
}

```

activity_main.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="match_parent"
    android:orientation="vertical" android:padding="16dp"
    android:gravity="center">

    <!-- Button to trigger writing to SD card -->
    <Button android:id="@+id/writeButton" android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/write_to_sd_card" />
</LinearLayout>

```

MAD LAB 9

MainActivity.java

```

package com.example.mad_lab_9;

import android.Manifest;
import android.os.Bundle;
import android.telephony.SmsManager;

```



```

import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

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

public class MainActivity extends AppCompatActivity {
    private EditText phoneNumber, messageText;

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

        phoneNumber = findViewById(R.id.phoneNumber);
        messageText = findViewById(R.id.messageText);
        Button sendButton = findViewById(R.id.sendButton);

        ActivityCompat.requestPermissions(this, new String[]
{Manifest.permission.SEND_SMS}, 1);

        sendButton.setOnClickListener(v -> sendSMS());
    }

    private void sendSMS() {
        String phoneNo = phoneNumber.getText().toString().trim();
        String message = messageText.getText().toString().trim();

        if (phoneNo.isEmpty() || message.isEmpty()) {
            Toast.makeText(this, "Please enter phone number and message",
Toast.LENGTH_SHORT).show();
            return;
        }

        try {
            SmsManager.getDefault().sendTextMessage(phoneNo, null, message, null,
null);
            Toast.makeText(this, "SMS Sent", Toast.LENGTH_SHORT).show();
        } catch (Exception e) {
            Toast.makeText(this, "SMS failed, try again!",
Toast.LENGTH_SHORT).show();
        }
    }
}

```

activity_main.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="match_parent"
android:orientation="vertical"
android:gravity="center"
android:padding="16dp">

<EditText
    android:id="@+id/phoneNumber"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:hint="Enter phone number"
    android:inputType="phone" />

<EditText
    android:id="@+id/messageText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:hint="Enter message"
    android:inputType="text" />

<Button
    android:id="@+id/sendButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="16dp"
    android:text="Send SMS" />

</LinearLayout>
```

MAD LAB 10

MainActivity.java

```
package com.example.mad_lab_10;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.os.Handler;
import android.os.Looper;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ListView;
import android.widget.ProgressBar;

import androidx.appcompat.app.AppCompatActivity;
```

```

import org.xmlpull.v1.XmlPullParser;
import org.xmlpull.v1.XmlPullParserFactory;

import java.io.InputStream;
import java.lang.ref.WeakReference;
import java.net.HttpURLConnection;
import java.net.URL;
import java.util.ArrayList;
import java.util.concurrent.ExecutorService;
import java.util.concurrent.Executors;

public class MainActivity extends AppCompatActivity {
    ListView rssListView;
    ProgressBar progressBar;
    ArrayList<String> titles = new ArrayList<>();
    ArrayList<String> links = new ArrayList<>();
    private final ExecutorService executorService =
Executors.newSingleThreadExecutor();
    private final Handler uiHandler = new Handler(Looper.getMainLooper());

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

        rssListView = findViewById(R.id.rssListView);
        progressBar = findViewById(R.id.progressBar);

        new
FetchRSSFeed(this).fetchRSS("https://feeds.bbc.co.uk/news/technology/rss.xml");
// BBC Tech News

        rssListView.setOnItemClickListener((parent, view, position, id) -> {
            Intent intent = new Intent(Intent.ACTION_VIEW,
Uri.parse(links.get(position)));
            startActivity(intent);
        });
    }

    @Override
    protected void onDestroy() {
        super.onDestroy();
        executorService.shutdown();
    }

    private static class FetchRSSFeed {
        private final WeakReference<MainActivity> activityRef;

        public FetchRSSFeed(MainActivity activity) {
            this.activityRef = new WeakReference<>(activity);
        }

        public void fetchRSS(String urlString) {
            MainActivity activity = activityRef.get();

```

```

        if (activity == null) return;

        activity.progressBar.setVisibility(View.VISIBLE);

        activity.executorService.execute(() -> {
            ArrayList<String> fetchedTitles = new ArrayList<>();
            ArrayList<String> fetchedLinks = new ArrayList<>();

            try {
                URL url = new URL(urlString);
                HttpURLConnection connection = (HttpURLConnection)
url.openConnection();
                InputStream inputStream = connection.getInputStream();

                XmlPullParserFactory factory =
XmlPullParserFactory.newInstance();
                XmlPullParser parser = factory.newPullParser();
                parser.setInput(inputStream, null);

                boolean insideItem = false;
                String title = "", link = "";
                int eventType = parser.getEventType();

                while (eventType != XmlPullParser.END_DOCUMENT) {
                    String tagName = parser.getName();
                    if (eventType == XmlPullParser.START_TAG) {
                        if ("item".equalsIgnoreCase(tagName)) insideItem =
true;

                        else if (insideItem &&
"title".equalsIgnoreCase(tagName)) title = parser.nextText();
                        else if (insideItem &&
"link".equalsIgnoreCase(tagName)) link = parser.nextText();
                    } else if (eventType == XmlPullParser.END_TAG &&
"item".equalsIgnoreCase(tagName)) {
                        fetchedTitles.add(title);
                        fetchedLinks.add(link);
                        insideItem = false;
                    }
                    eventType = parser.next();
                }

                inputStream.close();
            } catch (Exception e) {
                Log.e("RSS Fetch Error", "Error fetching RSS feed", e);
            }

            activity.uiHandler.post(() -> {
                MainActivity activityPost = activityRef.get();
                if (activityPost == null) return;

                activityPost.progressBar.setVisibility(View.GONE);
                activityPost.titles.clear();
                activityPost.links.clear();
                activityPost.titles.addAll(fetchedTitles);
                activityPost.links.addAll(fetchedLinks);
            });
        });
    }
}

```

```

        ArrayAdapter<String> adapter = new ArrayAdapter<>
(activityPost, android.R.layout.simple_list_item_1, activityPost.titles);
        activityPost.rssListView.setAdapter(adapter);
    });
});
}
}
}

```

activity_main.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="match_parent"
    android:orientation="vertical">

    <ProgressBar
        android:id="@+id/progressBar"
        style="?android:attr/progressBarStyleLarge"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:visibility="gone" />

    <ListView
        android:id="@+id/rssListView"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

</LinearLayout>

```

MAD LAB 11

MainActivity.java

```

package com.example.mad_lab_11;

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

```

```

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    private EditText emailInput, subjectInput, messageInput;

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

        emailInput = findViewById(R.id.email_input);
        subjectInput = findViewById(R.id.subject_input);
        messageInput = findViewById(R.id.message_input);
    }

    public void sendEmail(View view) {
        String email = emailInput.getText().toString().trim();
        String subject = subjectInput.getText().toString().trim();
        String message = messageInput.getText().toString().trim();

        if (email.isEmpty() || subject.isEmpty() || message.isEmpty()) {
            Toast.makeText(this, "All fields are required",
                Toast.LENGTH_SHORT).show();
            return;
        }

        // Constructing mailto URI properly
        Uri mailtoUri = Uri.parse("mailto:" + email +
            "?subject=" + Uri.encode(subject) +
            "&body=" + Uri.encode(message));

        Intent emailIntent = new Intent(Intent.ACTION_SENDTO, mailtoUri);

        if (emailIntent.resolveActivity(getPackageManager()) != null) {
            startActivity(emailIntent);
        } else {
            Toast.makeText(this, "No email client found",
                Toast.LENGTH_SHORT).show();
        }
    }
}

```

activity_main.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="match_parent"
    android:orientation="vertical"
    android:gravity="center"

```

```
android:padding="16dp">
```

```
<EditText
    android:id="@+id/email_input"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:autofillHints="emailAddress"
    android:hint="Recipient Email"
    android:minWidth="200dp"
    android:inputType="textEmailAddress" />
```

```
<EditText
    android:id="@+id/subject_input"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:hint="Subject"
    android:minWidth="200dp"
    android:inputType="text" />
```

```
<EditText
    android:id="@+id/message_input"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:hint="Message"
    android:minWidth="200dp"
    android:inputType="textMultiLine"
    android:minLines="4" />
```

```
<Button
    android:id="@+id/send_email_button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Send Email"
    android:layout_marginTop="16dp"
    android:onClick="sendEmail" />
```

```
</LinearLayout>
```
