# ExNo1: MyApp

// File: MainActivity.kt

package com.example.myapp  
import android.app.Activity  
import android.os.Bundle  
import android.widget.Button  
import android.widget.TextView  
import android.widget.Toast  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.activity.enableEdgeToEdge  
import androidx.compose.foundation.layout.fillMaxSize  
import androidx.compose.foundation.layout.padding  
import androidx.compose.material3.Button  
import androidx.compose.material3.Scaffold  
import androidx.compose.material3.Text  
import androidx.compose.runtime.Composable  
import androidx.compose.ui.Modifier  
import android.graphics.Typeface  
import android.graphics.Color  
//import androidx.compose.ui.graphics.Color  
//import androidx.compose.ui.text.font.Typeface  
import androidx.compose.ui.tooling.preview.Preview  
import com.example.myapp.ui.theme.MyAppTheme  
class MainActivity : Activity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_main)  
 val t1 = findViewById<TextView>(R.id.text1)  
 val b1 = findViewById<Button>(R.id.font1)  
 val b2 = findViewById<Button>(R.id.color1)  
 var fontIndex = 0  
 val fonts = arrayOf(Typeface.DEFAULT\_BOLD, Typeface.MONOSPACE,Typeface.DEFAULT\_BOLD)  
 var currentIndex = 0  
 val fontSizes = arrayOf(16f, 20f, 24f, 28f, 32f)  
 var colorIndex = 0  
 val colors = arrayOf(Color.RED, Color.BLUE, Color.GREEN, Color.MAGENTA, Color.BLACK)  
 b1.setOnClickListener {  
 fontIndex = (fontIndex + 1) % fonts.size  
 t1.typeface = fonts[fontIndex]  
 currentIndex = (currentIndex + 1) % fontSizes.size  
 t1.textSize = fontSizes[currentIndex]  
 //t1.textSize = 30F  
 Toast.makeText(this, "Font and Size Changed", Toast.LENGTH\_SHORT).show()  
 }  
 b2.setOnClickListener {  
 colorIndex = (colorIndex + 1) % colors.size  
 t1.setTextColor(colors[colorIndex])  
 //t2.setTextColor(Color.RED)  
 Toast.makeText(this,"Font Color Changed",Toast.LENGTH\_SHORT).show()  
 }  
 }  
}

**Output**

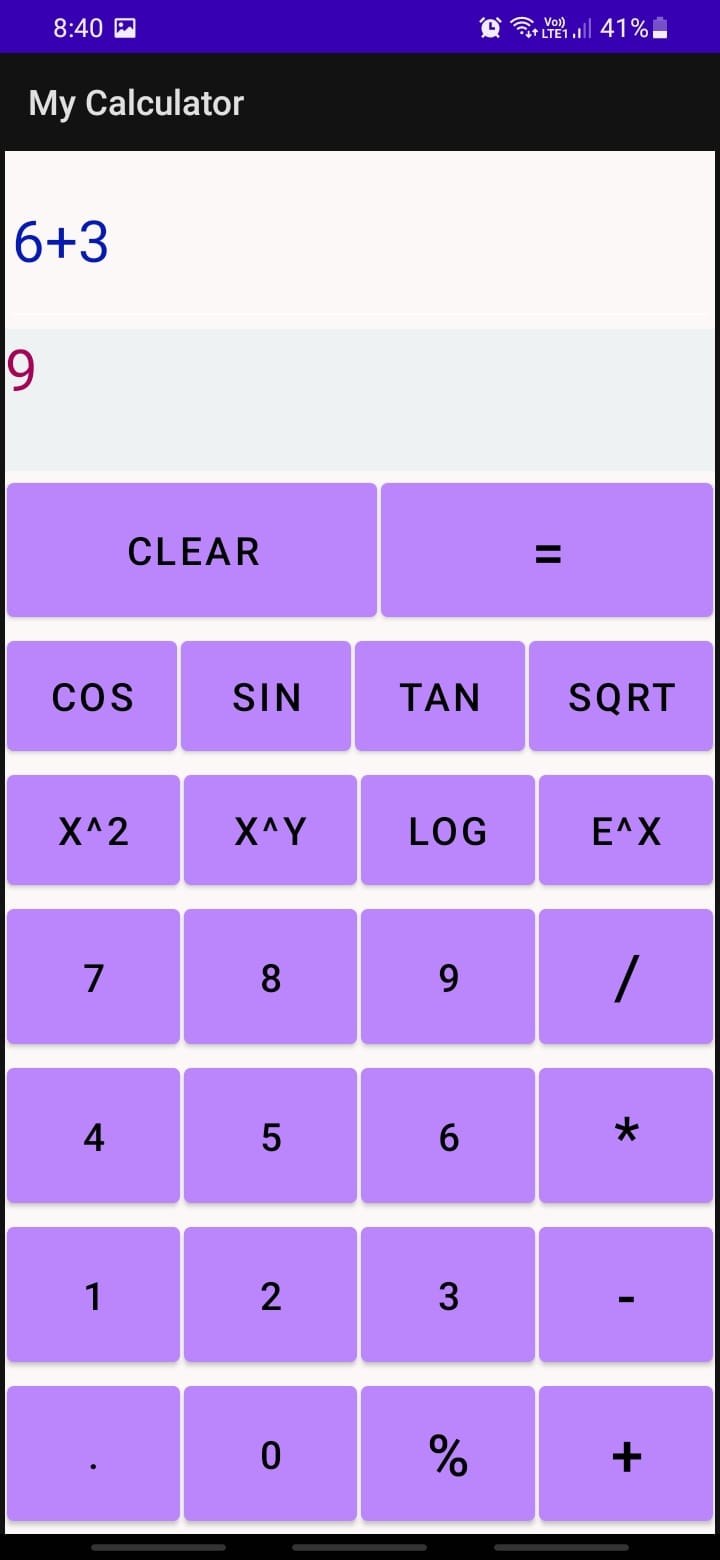


# ExNo:2 MyCalculator

// File: MainActivity.kt

package com.example.mycalculator  
import android.os.Bundle  
import android.os.PersistableBundle  
import android.widget.Button  
import android.widget.EditText  
import android.widget.TextView  
import android.widget.Toast  
import androidx.appcompat.app.AppCompatActivity  
  
class MainActivity : AppCompatActivity() {  
  
 var input1: Double = 0.0  
 var input2: Double = 0.0  
 var Add: Boolean = false  
 var Sub: Boolean = false  
 var Mul: Boolean = false  
 var Div: Boolean = false  
 var Rem: Boolean = false  
 var dec: Boolean = false  
 var cos: Boolean = false  
 var sin: Boolean = false  
 var tan: Boolean = false  
 var pow: Boolean = false  
 var sq: Boolean = false  
 var sqrt: Boolean = false  
 var log: Boolean = false  
 var exp: Boolean = false  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_main)  
 val button0 = findViewById<Button>(R.id.button0)  
 val button1 = findViewById<Button>(R.id.button1)  
 val button2 = findViewById<Button>(R.id.button2)  
 val button3 = findViewById<Button>(R.id.button3)  
 val button4 = findViewById<Button>(R.id.button4)  
 val button5 = findViewById<Button>(R.id.button5)  
 val button6 = findViewById<Button>(R.id.button6)  
 val button7 = findViewById<Button>(R.id.button7)  
 val button8 = findViewById<Button>(R.id.button8)  
 val button9 = findViewById<Button>(R.id.button9)  
 val buttonDot = findViewById<Button>(R.id.buttondot)  
 val buttonAdd = findViewById<Button>(R.id.buttonadd)  
 val buttonSub = findViewById<Button>(R.id.buttonsub)  
 val buttonMul = findViewById<Button>(R.id.buttonmul)  
 val buttonDiv = findViewById<Button>(R.id.buttondiv)  
 val buttonRem = findViewById<Button>(R.id.buttonrem)  
 val buttonCos = findViewById<Button>(R.id.buttoncos)  
 val buttonSin = findViewById<Button>(R.id.buttonsin)  
 val buttonTan = findViewById<Button>(R.id.buttontan)  
 val buttonSqrt = findViewById<Button>(R.id.buttonsqrt)  
 val buttonPow = findViewById<Button>(R.id.buttonpow)  
 val buttonLog = findViewById<Button>(R.id.buttonlog)  
 val buttonExp = findViewById<Button>(R.id.buttonexp)  
 val buttonSq = findViewById<Button>(R.id.buttonsq)  
 val buttonClr = findViewById<Button>(R.id.buttoncir)  
 val buttonEqual = findViewById<Button>(R.id.buttoneql)  
 val edtInput = findViewById<EditText>(R.id.input)  
 val editDisplay = findViewById<TextView>(R.id.display)  
 button1.setOnClickListener() {  
 edtInput.setText("${edtInput.text}1")  
 }  
 button2.setOnClickListener() {  
 edtInput.setText("${edtInput.text}2")  
 }  
 button3.setOnClickListener() {  
 edtInput.setText("${edtInput.text}3")  
 }  
 button4.setOnClickListener() {  
 edtInput.setText("${edtInput.text}4")  
 }  
 button5.setOnClickListener() {  
 edtInput.setText("${edtInput.text}5")  
 }  
 button6.setOnClickListener() {  
 edtInput.setText("${edtInput.text}6")  
 }  
 button7.setOnClickListener() {  
 edtInput.setText("${edtInput.text}7")  
 }  
 button8.setOnClickListener() {  
 edtInput.setText("${edtInput.text}8")  
 }  
 button9.setOnClickListener() {  
 edtInput.setText("${edtInput.text}9")  
 }  
 button0.setOnClickListener() {  
 edtInput.setText("${edtInput.text}0")  
 }  
 buttonAdd.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 Add = true  
 dec = false  
 edtInput.setText(null)  
 }  
 }  
 buttonSub.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 Sub = true  
 dec = false  
 edtInput.setText(null)  
 }  
 }  
 buttonMul.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 Mul = true  
 dec = false  
 edtInput.setText(null)  
 }  
 }  
 buttonDiv.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 Div = true  
 dec = false  
 edtInput.setText(null)  
 }  
 }  
 buttonRem.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 Rem = true  
 dec = false  
 edtInput.setText(null)  
  
 }  
 }  
 buttonCos.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 cos = true  
 dec = false  
 edtInput.setText(null)  
 }  
 }  
 buttonSin.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 sin = true  
 dec = false  
 edtInput.setText(null)  
 }  
 }  
 buttonTan.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 tan = true  
 dec = false  
 edtInput.setText(null)  
 }  
 }  
 buttonPow.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 pow = true  
 dec = false  
 edtInput.setText(null)  
 }  
 }  
 buttonSq.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 sq = true  
 dec = false  
 edtInput.setText(null)  
 }  
 }  
 buttonSqrt.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 sqrt = true  
 dec = false  
 edtInput.setText(null)  
 }  
 }  
 buttonLog.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 log = true  
 dec = false  
 edtInput.setText(null)  
 }  
 }  
 buttonExp.setOnClickListener() {  
 if (edtInput.getText().length != 0) {  
 input1 = "${edtInput.text}".toDouble()  
 exp = true  
 dec = false  
 edtInput.setText(null)  
 }  
 }  
 buttonDot.setOnClickListener() {  
 if (dec) {  
 // do nothing or you can show the error  
 } else {  
 edtInput.setText("${edtInput.text}.")  
 dec = true  
 }  
 }  
 buttonClr.setOnClickListener() {  
 editDisplay.setText("")  
 input1 = 0.0  
 input2 = 0.0  
 }  
 buttonEqual.setOnClickListener() {  
 if (Add || Sub || Mul || Div || Rem || pow) {  
 input2 = "${edtInput.text}".toDouble()  
 if (Add) {  
edtInput.setText("${input1.toInt()}+${input2.toInt()}")  
 val radd: Double = input1 + input2  
 editDisplay.setText("${radd.toInt()}")  
 Add = false  
 }  
 if (Sub) {  
 edtInput.setText("${input1.toInt()}-${input2.toInt()}")  
 val rsub: Double = input1 - input2  
 editDisplay.setText("${rsub.toInt()}")  
 Sub = false  
 }  
 if (Mul) {  
 edtInput.setText("${input1.toInt()}\*${input2.toInt()}")  
 val rmul: Double = input1 \* input2  
 editDisplay.setText("${rmul.toInt()}")  
 Mul = false  
 }  
 if (Div) {  
edtInput.setText("${input1.toInt()}/${input2.toInt()}")  
 val rdiv: Double = input1 / input2  
 editDisplay.setText("${rdiv.toInt()}")  
 Div = false  
 }  
 if (Rem) {  
 edtInput.setText("${input1.toInt()}%${input2.toInt()}")  
 val rrem: Double = input1 % input2  
 editDisplay.setText("${rrem.toInt()}")  
 Rem = false  
 }  
 if (cos) {  
 edtInput.setText("cos(${input1.toInt()})")  
 val rcos: Double = Math.cos(Math.toRadians(input1))  
 editDisplay.setText("${rcos.toInt()}")  
 cos = false  
 }  
 if (sin) {  
 edtInput.setText("sin(${input1.toInt()})")  
 val rsin: Double = Math.sin(Math.toRadians(input1))  
 editDisplay.setText("${rsin.toInt()}")  
 sin = false  
 }  
 if (tan) {  
 edtInput.setText("tan(${input1.toInt()})")  
 val rtan: Double = Math.tan(Math.toRadians(input1))  
 editDisplay.setText("${rtan.toInt()}")  
 tan = false  
 }  
 if (sqrt) {  
 edtInput.setText("sqrt(${input1.toInt()})")  
 val sqrteql: Double = Math.sqrt(input1)  
 editDisplay.setText("${sqrteql}")  
 sqrt = false  
 }  
 if (sq) {  
 edtInput.setText("${input1.toInt()}^2")  
 val sqeql: Double = Math.pow(input1, 2.0)  
 editDisplay.setText("${sqeql.toInt()}")  
 log = false  
 }  
 if (pow) {  
 edtInput.setText("${input1.toInt()}^${input2.toInt()}")  
 val peql: Double = Math.pow(input1, input2)  
 editDisplay.setText("${peql.toInt()}")  
 pow = false  
 }  
 if (log) {  
 edtInput.setText("log(${input1.toInt()})")  
 val lgeql: Double = Math.log10(input1)  
 editDisplay.setText("${lgeql.toInt()}")  
 log = false  
 }  
 if (exp) {  
 edtInput.setText("e^${input1.toInt()}")  
 val expeql: Double = Math.exp(input1)  
 editDisplay.setText("${expeql.toInt()}")  
 exp = false  
 }  
 }  
 }  
 }  
}

**Output**



# ExNo:3 Graphical Primitives

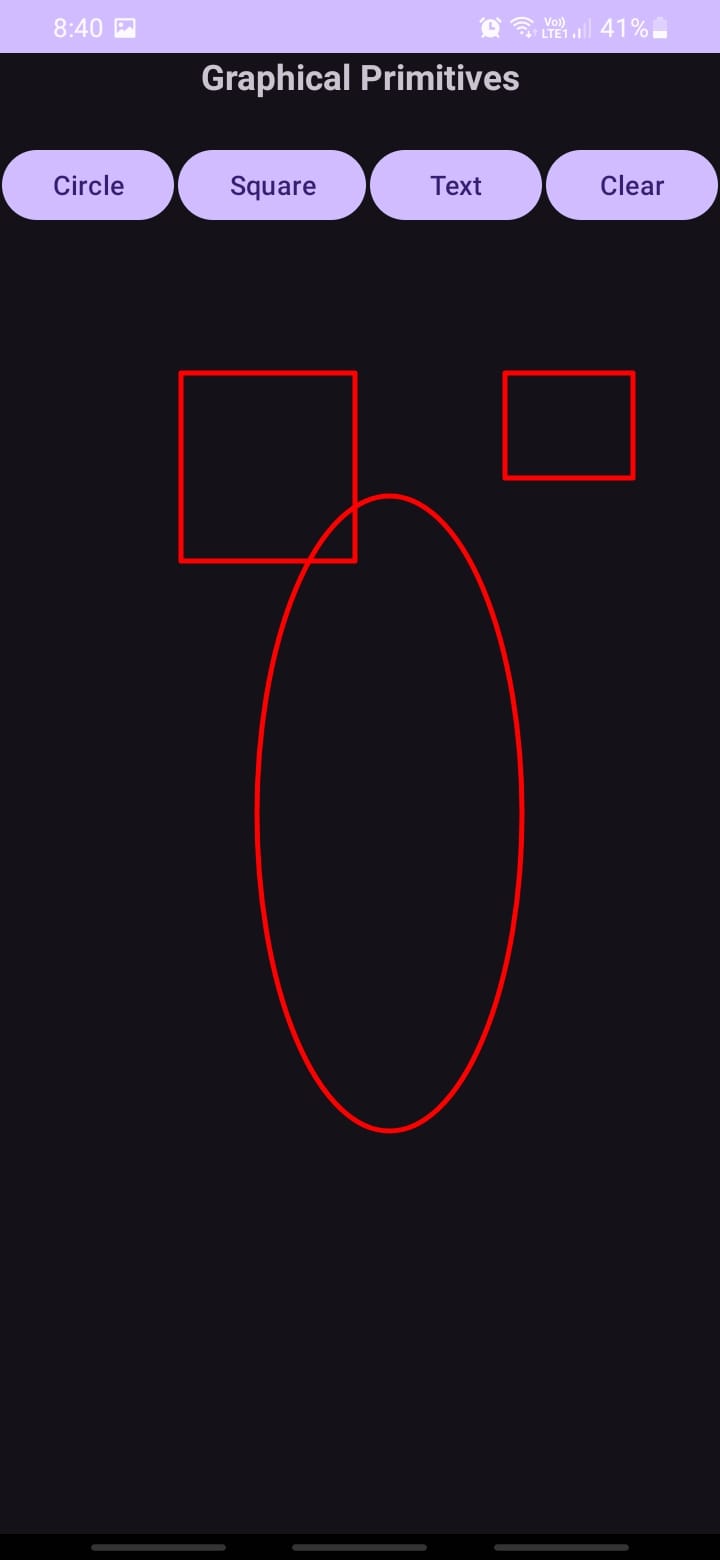
// File: MainActivity.kt

package com.example.graphicalprimitives  
import android.os.Bundle  
import androidx.appcompat.app.AppCompatActivity  
import android.widget.Button  
class MainActivity : AppCompatActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_main)  
 val tl = findViewById<TouchScreen>(R.id.t1) // Fixed `R\_id` to `R.id`  
 val bl = findViewById<Button>(R.id.b1)  
 val b2 = findViewById<Button>(R.id.b2)  
 val b3 = findViewById<Button>(R.id.b3)  
 val b4 = findViewById<Button>(R.id.b4)  
 bl.setOnClickListener {  
 tl.setDrawingMode(0) // Fixed method name from `setDrawint` to `setDrawingMode`  
 }  
 b2.setOnClickListener {  
 tl.setDrawingMode(1) // Fixed `l` (which was undefined) to `1`  
 }  
 b3.setOnClickListener {  
 tl.setDrawingMode(2) // Fixed `t1` (which was undefined) to `tl`  
 }  
 b4.setOnClickListener {  
 tl.startDrawing() // Fixed `t1` (which was undefined) to `tl`  
 }  
 }  
}

// File: TouchScreen.kt

package com.example.graphicalprimitives  
import android.content.Context  
import android.graphics.Canvas  
import android.graphics.Color  
import android.graphics.Paint  
import android.graphics.Path  
import android.graphics.RectF  
import android.util.AttributeSet  
import android.view.MotionEvent  
import android.view.View  
class TouchScreen : View {  
 var paint = Paint()  
 var path = Path()  
 var flag = 0  
 var x1: Float = 0.0f  
 var y1: Float = 0.0f  
 var x2: Float = 0.0f  
 var y2: Float = 0.0f  
 constructor(context: Context, attributeSet: AttributeSet) : super(context, attributeSet) {  
 paint.color = Color.RED  
 paint.isAntiAlias = true  
 paint.strokeJoin = Paint.Join.ROUND  
 paint.style = Paint.Style.STROKE  
 paint.strokeWidth = 5f  
 }  
  
 override fun onDraw(canvas: Canvas) {  
 super.onDraw(canvas)  
 canvas.drawPath(path, paint)  
 }  
 override fun onTouchEvent(event: MotionEvent): Boolean {  
 if (flag == 2) {  
 val X: Float = event.x  
 val Y: Float = event.y  
 when (event.action) {  
 MotionEvent.ACTION\_DOWN -> {  
 path.moveTo(X, Y)  
 return true  
 }  
 MotionEvent.ACTION\_MOVE -> {  
 path.lineTo(X, Y)  
 }  
 MotionEvent.ACTION\_UP -> {  
 // You can add additional logic here if needed  
 }  
 else -> return false  
 }  
 }  
 if (flag == 0 || flag == 1) {  
 when (event.action) {  
 MotionEvent.ACTION\_DOWN -> {  
 x1 = event.x  
 y1 = event.y  
 return true  
 }  
 MotionEvent.ACTION\_UP -> {  
 x2 = event.x  
 y2 = event.y  
 val rectF = RectF(x1, y1, x2, y2)  
 if (flag == 0) {  
 path.addOval(rectF, Path.Direction.CCW)  
 }  
 if (flag == 1) {  
 path.addRect(rectF, Path.Direction.CCW)  
 }  
 invalidate()  
 return true  
 }  
 else -> return false  
 }  
 }  
 return false  
 }  
 fun setDrawingMode(F: Int) {  
 flag = F  
 }  
 fun startDrawing() {  
 path.rewind()  
 invalidate()  
 }  
}

**Output**



# ExNo:4 Fragment

// File: MainActivity.kt

package com.example.ex4\_fragment  
import android.os.Bundle  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.activity.enableEdgeToEdge  
import androidx.compose.foundation.layout.fillMaxSize  
import androidx.compose.foundation.layout.padding  
import androidx.compose.material3.Scaffold  
import androidx.compose.material3.Text  
import androidx.compose.runtime.Composable

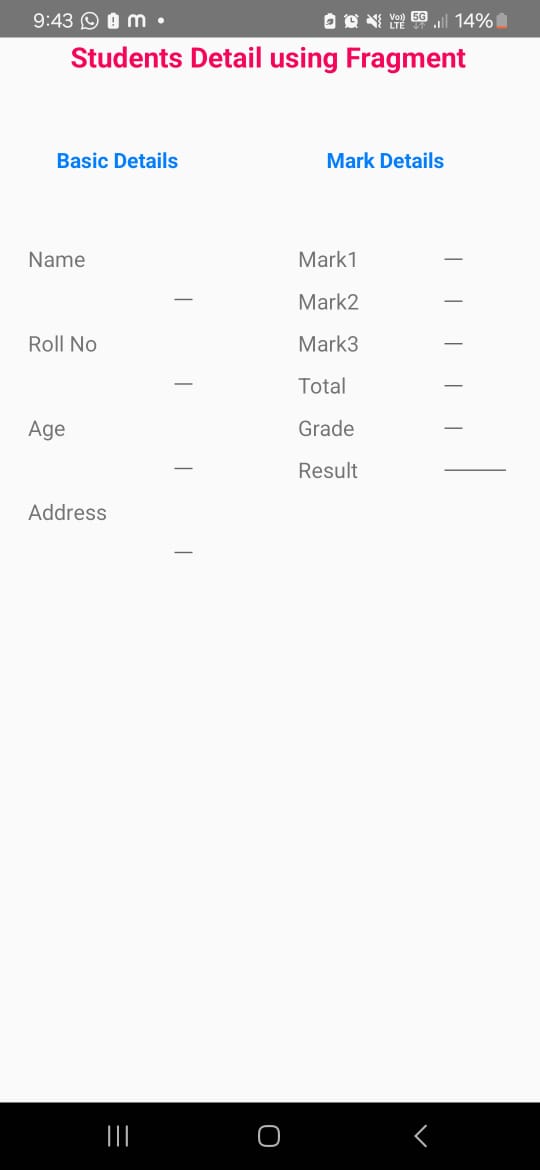
import androidx.compose.ui.Modifier  
import androidx.compose.ui.tooling.preview.Preview  
import com.example.ex4\_fragment.ui.theme.Ex4\_FragmentTheme  
class MainActivity : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_main)  
 }  
}

// File: StudentBasicDetailsActivity.kt

package com.example.ex4\_fragment  
import android.app.Fragment  
import android.os.Bundle  
//import androidx.fragment.app.Fragment  
import android.view.LayoutInflater  
import android.view.View  
import android.view.ViewGroup  
class StudentBasicDetailsActivity: Fragment(){  
 override fun onCreateView(  
 inflater: LayoutInflater,  
 container: ViewGroup?,  
 savedInstanceState: Bundle?  
 ): View? {  
 return inflater.inflate(R.layout.fragment\_student\_basic\_details\_activity, container, false)  
 }  
}// File: StudentMarkActivity.kt

package com.example.ex4\_fragment  
import android.app.Fragment  
import android.os.Bundle  
//import androidx.fragment.app.Fragment  
import android.view.LayoutInflater  
import android.view.View  
import android.view.ViewGroup  
class StudentMarkActivity : Fragment(){  
 override fun onCreateView(  
 inflater: LayoutInflater,  
 container: ViewGroup?,  
 savedInstanceState: Bundle?  
 ): View? {  
 return inflater.inflate(R.layout.fragment\_student\_mark\_activity, container, false)  
 }  
}

**Output**

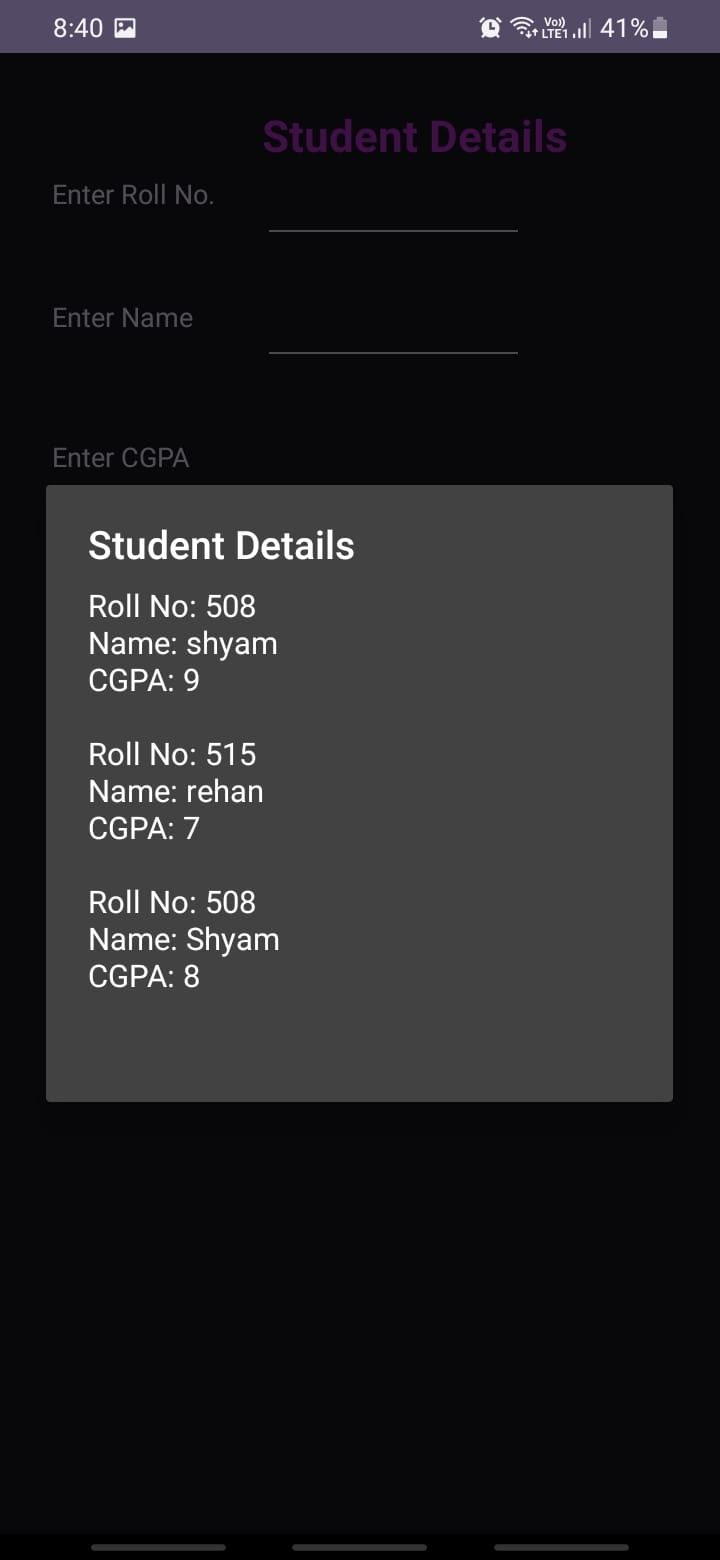


**Ex.NO:5 SQLite**

// File: MainActivity.kt

package com.example.sqlite  
import android.app.AlertDialog  
import android.content.Context  
import android.database.Cursor  
import android.database.sqlite.SQLiteDatabase  
import android.os.Bundle  
import android.view.View  
import android.widget.Button  
import android.widget.EditText  
import android.widget.Toast  
import androidx.appcompat.app.AppCompatActivity  
class MainActivity : AppCompatActivity(), View.OnClickListener {  
 private lateinit var srollno: EditText  
 private lateinit var sname: EditText  
 private lateinit var scgpa: EditText  
 private lateinit var insert: Button  
 private lateinit var update: Button  
 private lateinit var delete: Button  
 private lateinit var view: Button  
 private lateinit var viewAll: Button  
 private lateinit var db: SQLiteDatabase  
 private lateinit var c: Cursor  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_main)  
 // Initialize views  
 srollno = findViewById(R.id.editRollno)  
 sname = findViewById(R.id.editName)  
 scgpa = findViewById(R.id.editCGPA)  
 insert = findViewById(R.id.btnInsert)  
 update = findViewById(R.id.btnUpdate)  
 delete = findViewById(R.id.btnDelete)  
 view = findViewById(R.id.btnView)  
 viewAll = findViewById(R.id.btnViewAll)  
 // Set OnClickListener for buttons  
 insert.setOnClickListener(this)  
 update.setOnClickListener(this)  
 delete.setOnClickListener(this)  
 view.setOnClickListener(this)  
 viewAll.setOnClickListener(this)  
 // Initialize database  
 db = openOrCreateDatabase("StudentDB", Context.MODE\_PRIVATE, null)  
 db.execSQL("CREATE TABLE IF NOT EXISTS student(rollno VARCHAR, name VARCHAR, marks VARCHAR);")  
 }  
 override fun onClick(v: View?) {  
 when (v?.id) {  
 R.id.btnInsert -> {  
 if (srollno.text.toString().isEmpty() || sname.text.toString().isEmpty()) {  
 showMessage("Error", "Please enter all values")  
 return  
 }  
 db.execSQL("INSERT INTO student VALUES('${srollno.text}', '${sname.text}', '${scgpa.text}')")  
 showMessage("Success", "Record added")  
 clearText()  
 }  
 R.id.btnUpdate -> {  
 if (srollno.text.toString().isEmpty()) {  
 showMessage("Error", "Please enter roll no.")  
 return  
 }  
 c = db.rawQuery("SELECT \* FROM student WHERE rollno='${srollno.text}'", null)  
 if (c.moveToFirst()) {  
 db.execSQL("UPDATE student SET name='${sname.text}', marks='${scgpa.text}' WHERE rollno='${srollno.text}'")  
 showMessage("Success", "Record updated")  
 } else {  
 showMessage("Error", "Invalid roll no.")  
 }  
 clearText()  
 c.close()  
 }  
 R.id.btnDelete -> {  
 if (srollno.text.toString().isEmpty()) {  
 showMessage("Error", "Please enter roll no.")  
 return  
 }  
 c = db.rawQuery("SELECT \* FROM student WHERE rollno='${srollno.text}'", null)  
 if (c.moveToFirst()) {  
 db.execSQL("DELETE FROM student WHERE rollno='${srollno.text}'")  
 showMessage("Success", "Record deleted")  
 } else {  
 showMessage("Error", "Invalid roll no.")  
 }  
 clearText()  
 c.close()  
 }  
 R.id.btnView -> {  
 if (srollno.text.toString().isEmpty()) {  
 showMessage("Error", "Please enter roll no.")  
 return  
 }  
 c = db.rawQuery("SELECT \* FROM student WHERE rollno='${srollno.text}'", null)  
 if (c.moveToFirst()) {  
 sname.setText(c.getString(1))  
 scgpa.setText(c.getString(2))  
 } else {  
 showMessage("Error", "Invalid roll no.")  
 }  
 c.close()  
 }  
 R.id.btnViewAll -> {  
 c = db.rawQuery("SELECT \* FROM student", null)  
 if (c.count == 0) {  
 showMessage("Error", "No records found")  
 return  
 }  
 val buffer = StringBuffer()  
 while (c.moveToNext()) {  
 buffer.append("Roll No: ${c.getString(0)}\n")  
 buffer.append("Name: ${c.getString(1)}\n")  
 buffer.append("CGPA: ${c.getString(2)}\n\n")  
 }  
 showMessage("Student Details", buffer.toString())  
 c.close()  
 }  
 }  
 }  
 private fun showMessage(title: String, message: String) {  
 val builder = AlertDialog.Builder(this)  
 builder.setCancelable(true)  
 builder.setTitle(title)  
 builder.setMessage(message)  
 builder.show()  
 }  
 private fun clearText() {  
 srollno.setText("")  
 sname.setText("")  
 scgpa.setText("")  
 }  
}

**Output**



**ExNo:6 Form Validation**

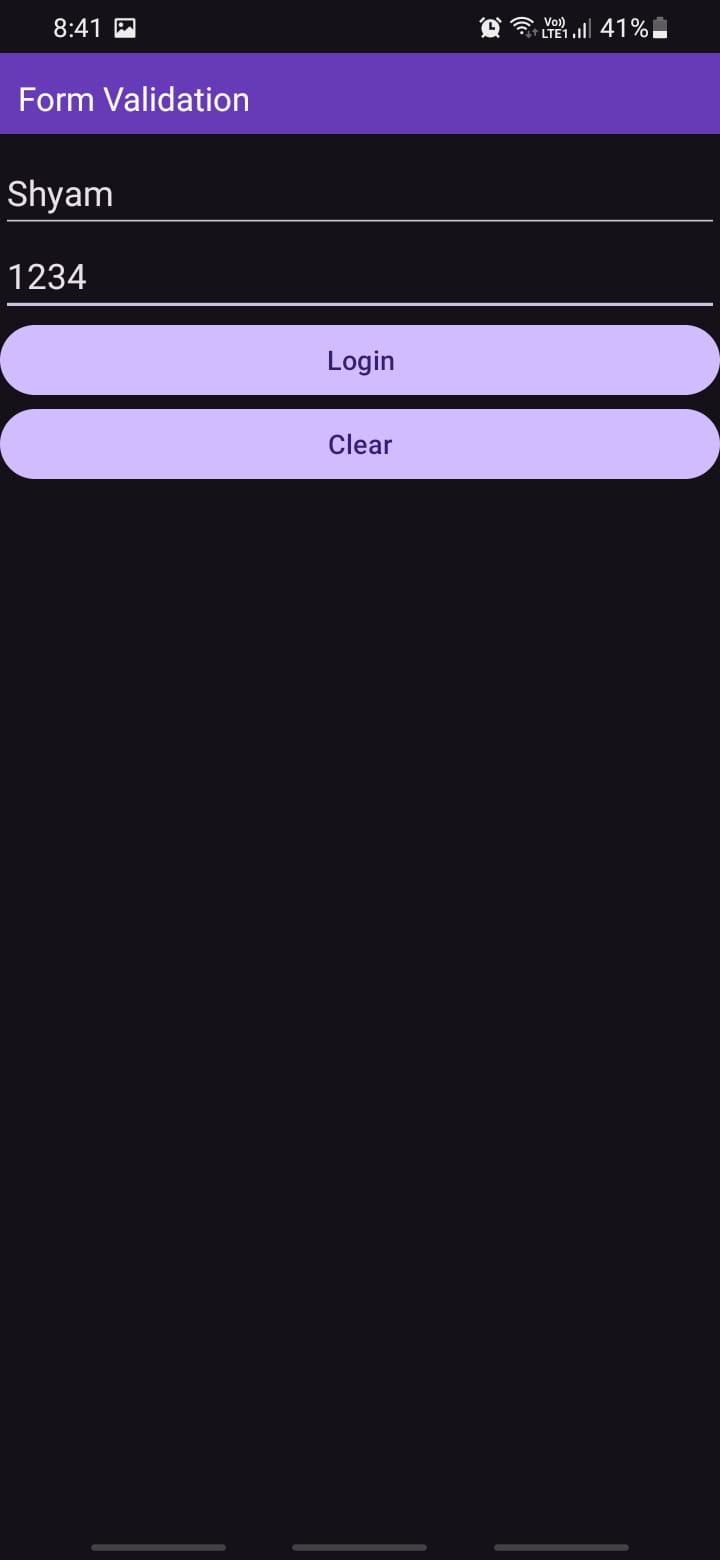
// File: MainActivity.kt

package com.example.ex6\_formvalidation  
import android.content.Intent  
import android.os.Bundle  
import android.widget.Button  
import android.widget.EditText  
import android.widget.Toast  
import androidx.activity.enableEdgeToEdge  
import androidx.appcompat.app.AppCompatActivity  
import androidx.core.view.ViewCompat  
import androidx.core.view.WindowInsetsCompat  
class MainActivity : AppCompatActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 enableEdgeToEdge()  
 setContentView(R.layout.activity\_main)  
 val etUserName : EditText=findViewById(R.id.etUserName)  
 val etPinNumber: EditText =findViewById(R.id.etPinNumber)  
 val btLogin: Button =findViewById(R.id.btLogin)  
 val btClear:Button=findViewById(R.id.btClear)  
 btLogin.setOnClickListener{  
 val userName=etUserName.text.toString().trim()  
 val pinNumber=etPinNumber.text.toString().trim()  
 if(userName.isEmpty() || pinNumber.isEmpty())  
 {  
 Toast.makeText(this,"All field are mandatory",Toast.LENGTH\_LONG).show()  
 return@setOnClickListener  
 }  
 if(!userName.matches(Regex("[a-zA-Z]+$")))  
 {  
 Toast.makeText(this,"Invalid Username",Toast.LENGTH\_LONG).show()  
 return@setOnClickListener  
 }  
 if(!pinNumber.matches(Regex("^[0-9]{4}$")))  
 {  
 Toast.makeText(this,"Invalid Pin Number",Toast.LENGTH\_LONG).show()  
 return@setOnClickListener  
 }  
 val intent: Intent =Intent(this,MainActivity2::class.java)  
 startActivity(intent)  
 }  
ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->  
 val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())  
 v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)  
 insets  
 }  
 }  
}

// File: MainActivity2.kt

package com.example.ex6\_formvalidation  
import android.os.Bundle  
import androidx.activity.enableEdgeToEdge  
import androidx.appcompat.app.AppCompatActivity  
import androidx.core.view.ViewCompat  
import androidx.core.view.WindowInsetsCompat  
class MainActivity2 : AppCompatActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 enableEdgeToEdge()  
 setContentView(R.layout.activity\_main2) ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->  
 val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())  
 v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)  
 insets  
 }  
 }  
}

**Output**

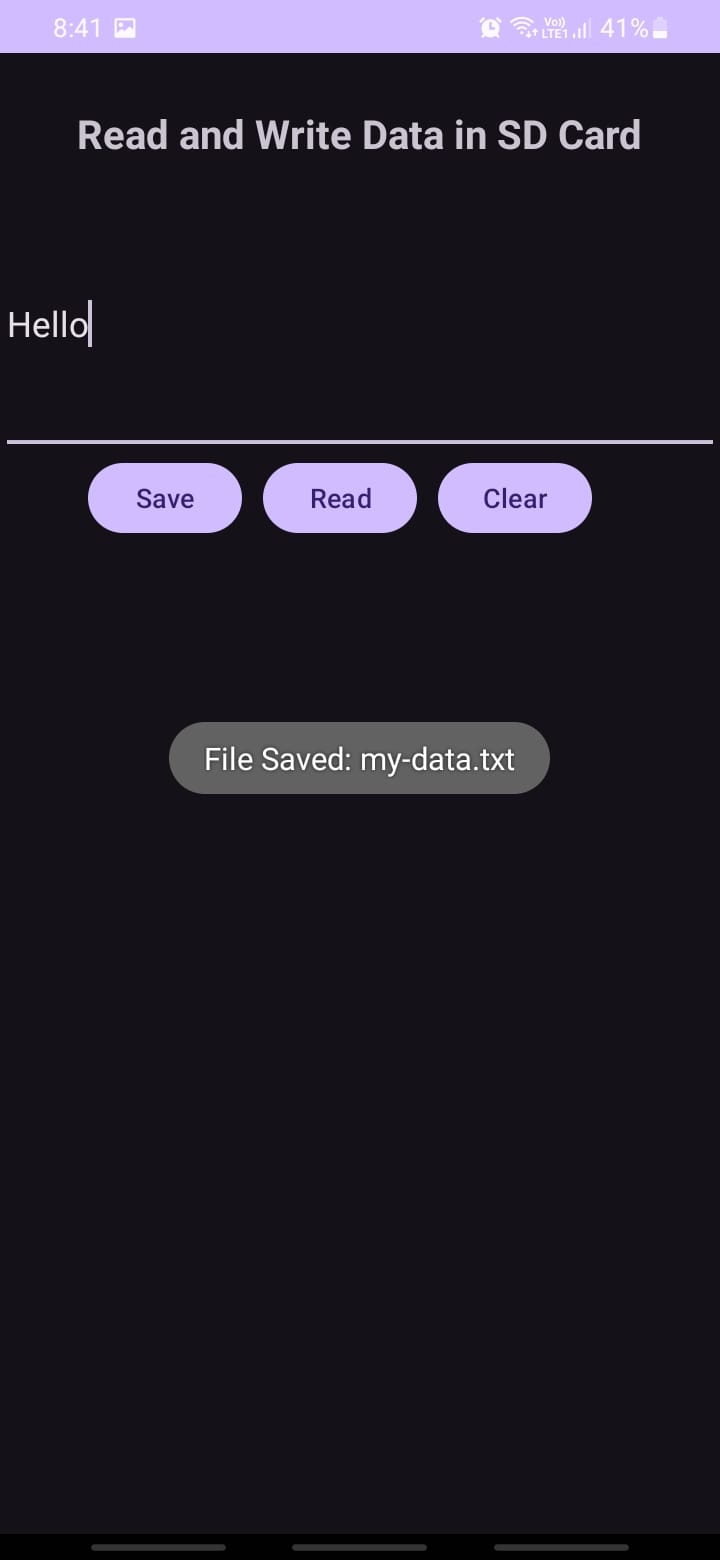


# ExNo:7 SDCard

// File: MainActivity.kt

package com.example.sdcard  
import android.os.Bundle  
import android.view.View  
import android.widget.Button  
import android.widget.EditText  
import android.widget.Toast  
import androidx.appcompat.app.AppCompatActivity  
import java.io.FileInputStream  
import java.io.FileOutputStream  
class MainActivity : AppCompatActivity() {  
 var E1: EditText? = null  
 var B1: Button? = null  
 var B2: Button? = null  
 var B3: Button? = null  
 var data: String = ""  
 var filename = "my-data.txt"  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_main)  
 E1 = findViewById<View>(R.id.E1) as EditText  
 B1 = findViewById<View>(R.id.B1) as Button  
 B2 = findViewById<View>(R.id.B2) as Button  
 B3 = findViewById<View>(R.id.B3) as Button  
 E1!!.hint = "Enter Some Text Here"  
 B1!!.setOnClickListener {  
 writeData()  
 }  
 B2!!.setOnClickListener {  
 readData()  
 }  
 B3!!.setOnClickListener {  
 E1!!.setText("")  
 }  
 }  
 private fun writeData() {  
 data = E1!!.text.toString()  
 try {  
 val fos: FileOutputStream = openFileOutput(filename, MODE\_PRIVATE)  
 fos.write(data!!.toByteArray())  
 fos.close()  
 Toast.makeText(applicationContext, "File Saved: $filename", Toast.LENGTH\_LONG).show()  
 } catch (e: Exception) {  
 Toast.makeText(applicationContext, e.message, Toast.LENGTH\_LONG).show()  
 }  
 }  
 private fun readData() {  
 var c: Int  
 var temp = ""  
 try {  
 val fis: FileInputStream = openFileInput(filename)  
 while ((fis.read().also { c = it }) != -1) {  
 temp += c.toChar().toString()  
 }  
 E1!!.setText(temp)  
 Toast.makeText(applicationContext, "File Read: $filename", Toast.LENGTH\_LONG).show()  
 } catch (e: Exception) {  
 Toast.makeText(applicationContext, e.message, Toast.LENGTH\_LONG).show()  
 }  
 }  
}

**Output**



# ExNo:8 AlertDialogBox

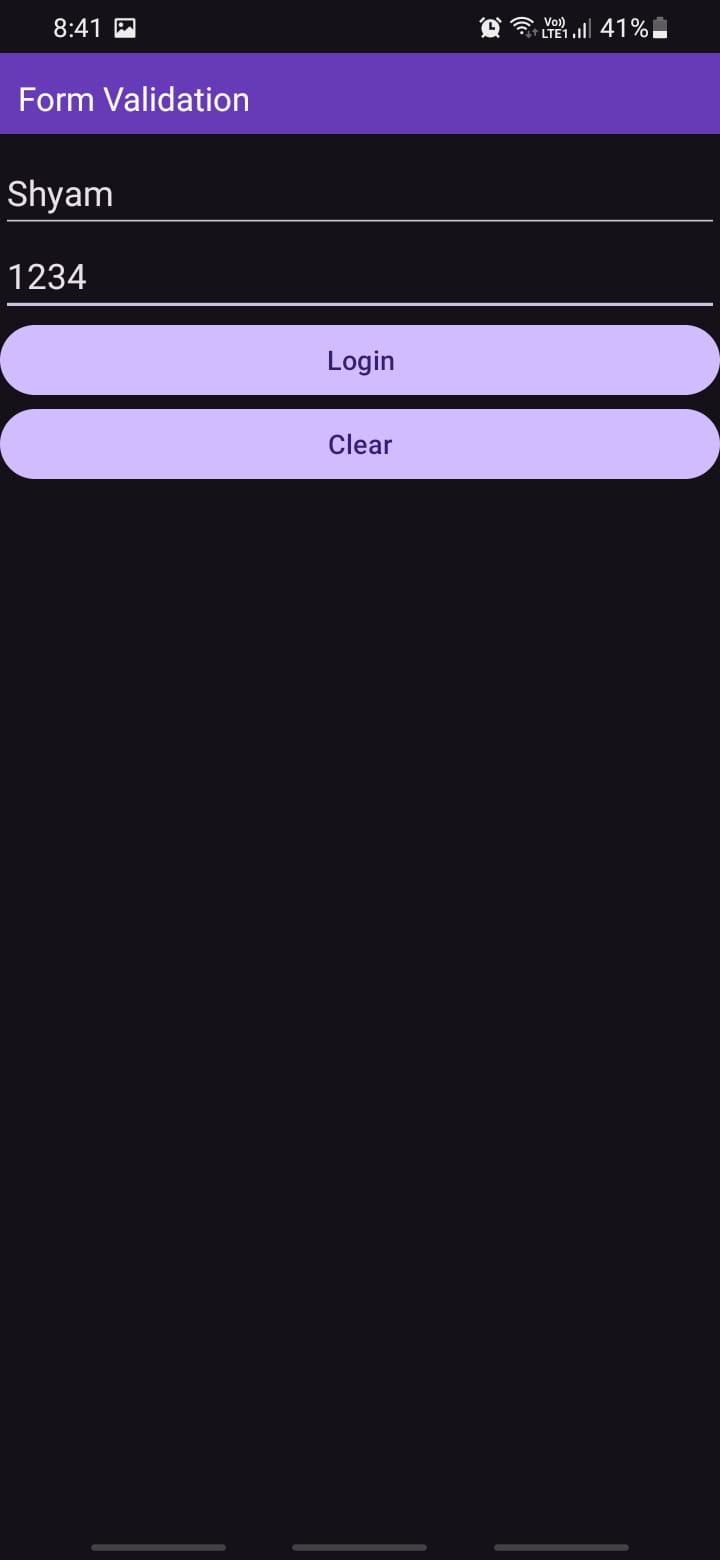
// File: MainActivity.kt

package com.example.alertdialogbox  
import android.os.Bundle  
import androidx.activity.enableEdgeToEdge  
import androidx.appcompat.app.AppCompatActivity  
import androidx.core.view.ViewCompat  
import androidx.core.view.WindowInsetsCompat  
import android.widget.Button  
import android.widget.EditText  
import android.widget.Toast  
import android.view.View  
import android.content.Intent  
import android.app.AlertDialog  
import kotlin.jvm.java  
class MainActivity : AppCompatActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_main)  
 val alert = findViewById<Button>(R.id.alert\_Button)  
 val clear = findViewById<Button>(R.id.clear\_Button)  
 val sms = findViewById<EditText>(R.id.edit\_Text)  
 val builder = AlertDialog.Builder(this)  
 alert.setOnClickListener {  
 val message = sms.text.toString()  
 if (message != "") {  
 builder.setMessage(message).setTitle("New Message")  
 builder.setMessage(message)  
 builder.setCancelable(false)  
 builder.setPositiveButton("OK") { dialogInterface, which ->  
 Toast.makeText(applicationContext, "AlertDialog Closed", Toast.LENGTH\_LONG).show()  
 }  
 val smsIntent = Intent(this, SmsAlert::class.java)  
 smsIntent.putExtra(message, 1)  
 val alertBox: AlertDialog = builder.create()  
 alertBox.setTitle("New Message")  
 alertBox.show()  
 startActivity(smsIntent)  
 } else {  
 Toast.makeText(applicationContext, "Type Message in Message Box", Toast.LENGTH\_LONG).show()  
 }  
 }  
 clear.setOnClickListener {  
 sms.setText("")  
 }  
 }  
}

// File: SmsAlert.kt

package com.example.alertdialogbox  
import android.os.Bundle  
import androidx.activity.enableEdgeToEdge  
import androidx.appcompat.app.AppCompatActivity  
import androidx.core.view.ViewCompat  
import androidx.core.view.WindowInsetsCompat  
import android.widget.TextView  
class SmsAlert : AppCompatActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_sms\_alert)  
 val showmsg = findViewById<TextView>(R.id.showmsg)  
 val extras = intent.extras  
 showmsg.text = extras!!.getString("sms")  
 }  
}

**Output**



# ExNo:9 Alarm

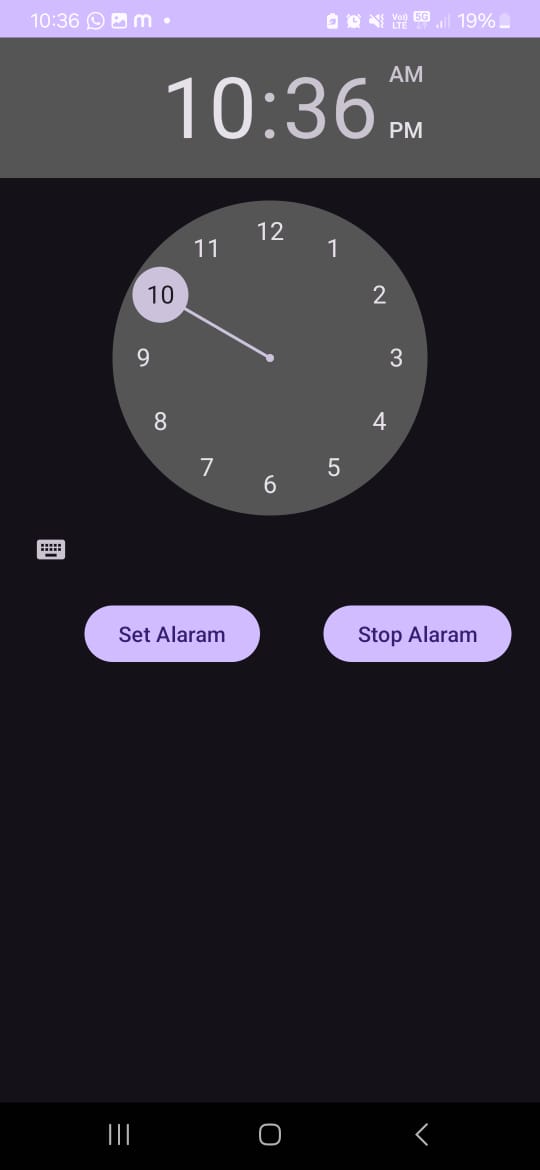
// File: AlarmReceiver.kt

package com.example.alertdialogbox  
import android.content.BroadcastReceiver  
import android.content.Context  
import android.content.Intent  
import android.media.Ringtone  
import android.media.RingtoneManager  
import android.net. Uri  
import android.widget.Toast  
class AlarmReceiver: BroadcastReceiver() {  
 lateinit var ringtone: Ringtone  
 override fun onReceive(context: Context, intent: Intent) {  
 Toast.makeText(context, "Alarm! Ringing", Toast.LENGTH\_LONG).show()  
 val alarmUri: Uri =  
 RingtoneManager.getDefaultUri(RingtoneManager.TYPE\_ALARM)  
 ringtone = RingtoneManager.getRingtone(context, alarmUri)  
 ringtone.play()  
  
 }  
}

// File: MainActivity.kt

package com.example.alertdialogbox  
import android.os.Bundle  
import android.widget.Button  
import android.widget.EditText  
import android.widget.Toast  
import androidx.activity.enableEdgeToEdge  
import androidx.appcompat.app.AlertDialog  
import androidx.appcompat.app.AppCompatActivity  
import androidx.core.view.ViewCompat  
import androidx.core.view.WindowInsetsCompat  
import android.app.AlarmManager  
import android.app.PendingIntent  
import android.content.Intent  
import android.widget. TimePicker  
import java.util .\*  
class MainActivity : AppCompatActivity() {  
 private lateinit var alarmTimePicker: TimePicker  
 private lateinit var pendingIntent: PendingIntent  
 private lateinit var alarmManager: AlarmManager  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_main)  
 val btnSet = findViewById<Button>(R.id.btnSet)  
 val btnStop = findViewById<Button>(R.id.btnStop)  
 alarmTimePicker = findViewById(R.id.timePicker)  
 alarmManager = getSystemService(ALARM\_SERVICE) as AlarmManager  
 btnSet.setOnClickListener {  
 Toast.makeText(this, "ALARM ON", Toast.LENGTH\_SHORT).show()  
 val calendar: Calendar = Calendar.getInstance()  
 calendar.set(Calendar.HOUR\_OF\_DAY, alarmTimePicker.hour)  
 calendar.set(Calendar.MINUTE, alarmTimePicker.minute)  
 val intent = Intent(this, AlarmReceiver::class.java)  
 pendingIntent = PendingIntent.getBroadcast(  
 this.applicationContext, 2, intent,  
 PendingIntent.FLAG\_CANCEL\_CURRENT  
 )  
 val time: Long = calendar.timeInMillis - (calendar.timeInMillis % 60000)  
 alarmManager.setRepeating(  
 AlarmManager.RTC\_WAKEUP, time, 10000,  
 pendingIntent  
 )  
 btnStop.setOnClickListener {  
 alarmManager.cancel(pendingIntent)  
 Toast.makeText(this, "ALARM OFF", Toast.LENGTH\_SHORT).show()  
 }  
 }  
 }  
}

**Output**

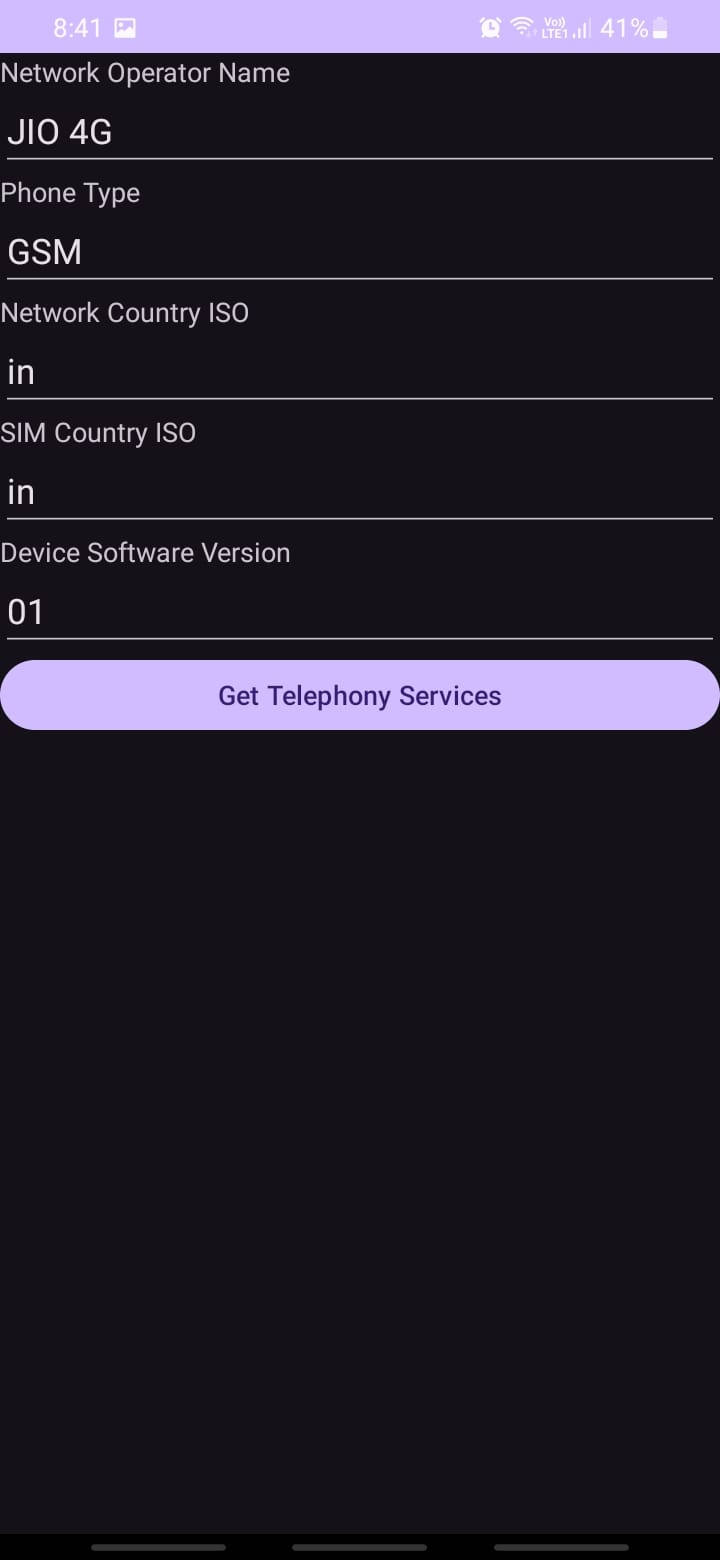


# ExNo:10 Telephony Service

// File: MainActivity.kt

package com.example.telephonyservices  
import android.content.Context  
import android.content.pm.PackageManager  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.telephony.TelephonyManager  
import android.widget.Button  
import android.widget.EditText  
import androidx.core.app.ActivityCompat  
class MainActivity : AppCompatActivity() {  
 private val REQUEST\_CODE\_PHONE\_STATE = 1000  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_main)  
 val etNetworkOperatorName : EditText =  
 findViewById(R.id.etNetworkOperatorName)  
 val etPhoneType : EditText = findViewById(R.id.etPhoneType)  
 val etNetworkCountryISO : EditText =  
 findViewById(R.id.etNetworkCountryISO)  
 val etSIMCountryISO : EditText = findViewById(R.id.etSIMCountryISO)  
 val etDeviceSoftwareVersion : EditText =  
 findViewById(R.id.etDeviceSoftwareVersion)  
 val btGetTelephonyServices : Button =  
 findViewById(R.id.btGetTelephonyServices)  
 val telephonyManager = getSystemService(Context.TELEPHONY\_SERVICE) as  
 TelephonyManager  
 if (ActivityCompat.checkSelfPermission(this,  
 android.Manifest.permission.READ\_PHONE\_STATE) !=  
 PackageManager.PERMISSION\_GRANTED ) {  
 ActivityCompat.requestPermissions(this,  
 arrayOf(android.Manifest.permission.READ\_PHONE\_STATE), REQUEST\_CODE\_PHONE\_STATE)  
 }  
 btGetTelephonyServices.setOnClickListener {  
 android.util.Log.d("MainActivity", "Get Telephony Services button clicked")  
 val networkOperatorName = telephonyManager.networkOperatorName  
 val phoneType: Int = telephonyManager.getPhoneType()  
 var strphoneType : String = ""  
 val networkCountryISO: String =  
 telephonyManager.getNetworkCountryIso()  
 val SIMCountryISO: String = telephonyManager.getSimCountryIso()  
 val deviceSoftwareVersion: String? =  
 telephonyManager.getDeviceSoftwareVersion()  
 when (phoneType) {  
 TelephonyManager.PHONE\_TYPE\_CDMA -> strphoneType = "CDMA"  
 TelephonyManager.PHONE\_TYPE\_GSM -> strphoneType = "GSM"  
 TelephonyManager.PHONE\_TYPE\_NONE -> strphoneType = "NONE"  
 }  
 etNetworkOperatorName.setText(networkOperatorName)  
 etPhoneType.setText(strphoneType)  
 etNetworkCountryISO.setText(networkCountryISO)  
 etSIMCountryISO.setText(SIMCountryISO)  
 etDeviceSoftwareVersion.setText(deviceSoftwareVersion)  
 }  
 }  
}

**Output**

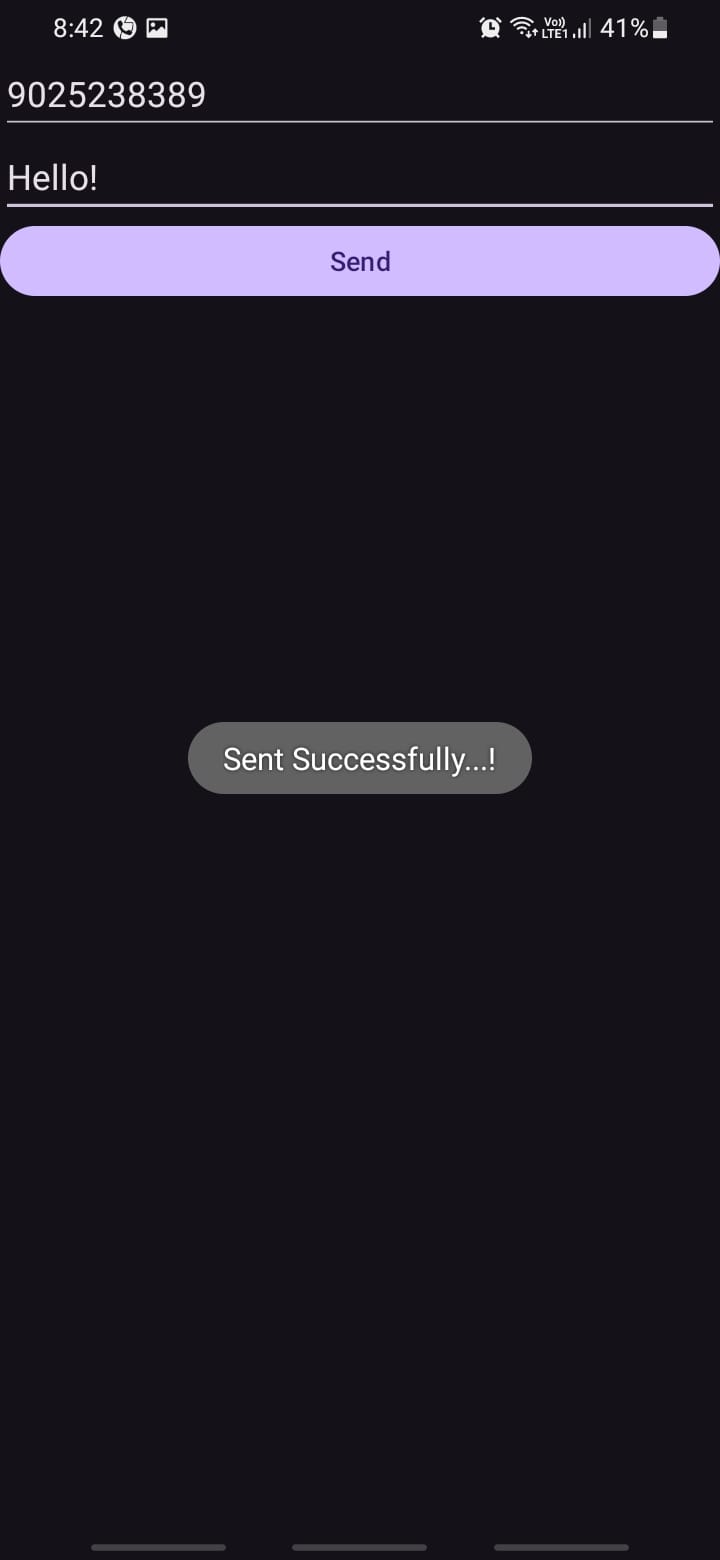


# ExNo:11 Send SMS

// File: MainActivity.kt

package com.example.sendsms  
import android.content.pm.PackageManager  
import android.os.Bundle  
import android.telephony.SmsManager  
import android.widget.Button  
import android.widget.EditText  
import android.widget.Toast  
import androidx.activity.enableEdgeToEdge  
import androidx.appcompat.app.AppCompatActivity  
import androidx.core.app.ActivityCompat  
import androidx.core.view.ViewCompat  
import androidx.core.view.WindowInsetsCompat  
class MainActivity : AppCompatActivity() {  
 private val REQUEST\_PERMISSION\_CODE =1000  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 enableEdgeToEdge()  
 setContentView(R.layout.activity\_main)  
 val etPhoneNo : EditText = findViewById(R.id.etPhoneNo)  
 val etMessage : EditText = findViewById(R.id.etMessage)  
 val btSend : Button = findViewById(R.id.btSend)  
 if(ActivityCompat.checkSelfPermission(this, android.Manifest.permission.SEND\_SMS) !=  
 PackageManager.PERMISSION\_GRANTED)  
 {  
 ActivityCompat.requestPermissions(this, arrayOf(android.Manifest.permission.SEND\_SMS), REQUEST\_PERMISSION\_CODE)  
 }  
 btSend.setOnClickListener {  
 val phoneNo = etPhoneNo.text.toString()  
 val message = etMessage.text.toString()  
 val smsManager : SmsManager  
 smsManager = SmsManager.getDefault()  
 smsManager.sendTextMessage(phoneNo,null,message, null,null)  
 Toast.makeText(this,"Sent Successfully...!",Toast.LENGTH\_LONG).show()  
 } ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->  
 val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())  
 v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)  
 insets  
 }  
 }  
}

**Output**

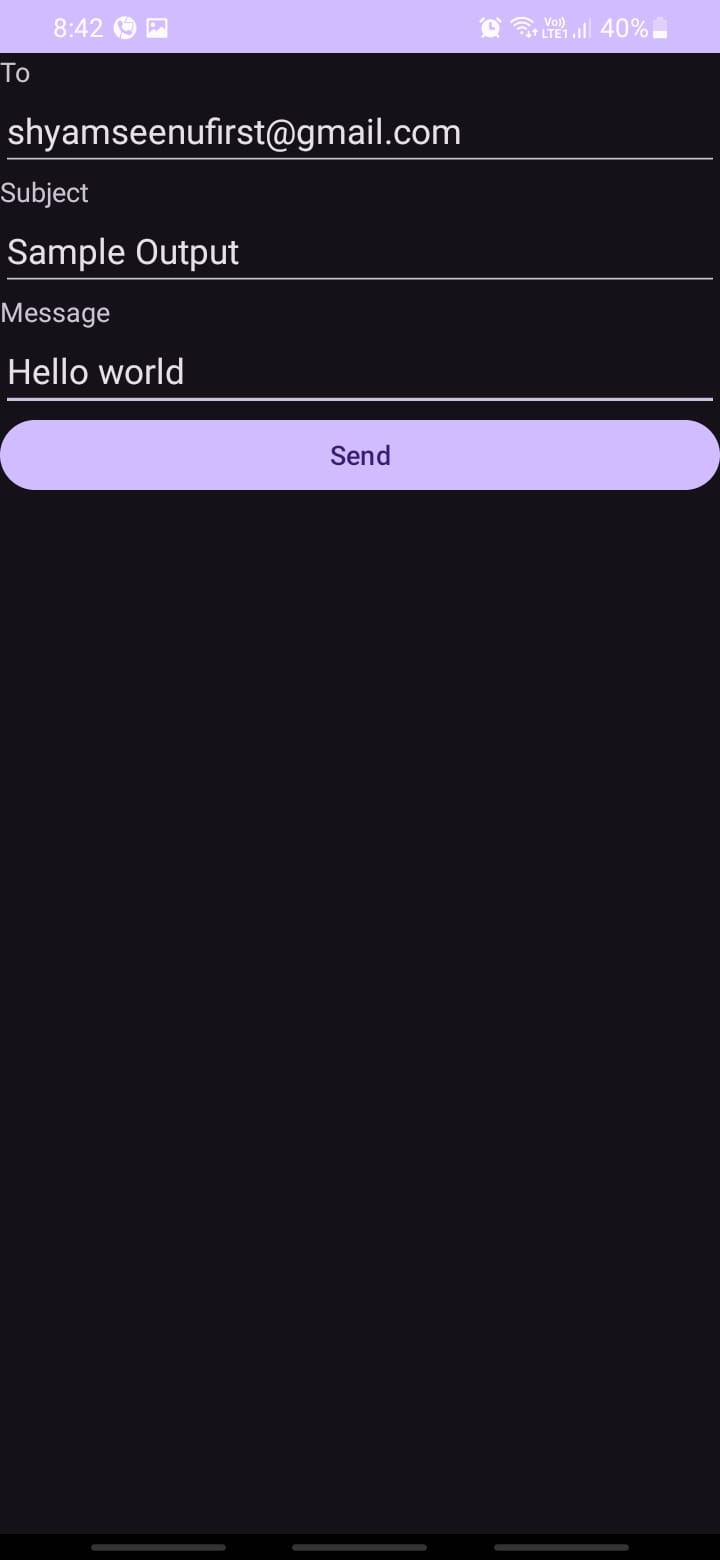


# ExNO:12 Send Email

// File: MainActivity.kt

package com.example.sendemail  
import android.os.Bundle  
import androidx.activity.enableEdgeToEdge  
import androidx.appcompat.app.AppCompatActivity  
import androidx.core.view.ViewCompat  
import androidx.core.view.WindowInsetsCompat  
import android.content.Intent  
import android.widget.Button  
import android.widget.EditText  
import android.widget.TextView  
class MainActivity : AppCompatActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_main)  
 val etEmail : EditText = findViewById(R.id.etEmail)  
 val etSubject : EditText = findViewById(R.id.etSubject)  
 val etMessage : EditText = findViewById(R.id.etMessage)  
 val btSend : Button = findViewById(R.id.btSend)  
 btSend.setOnClickListener {  
 val email = etEmail.text.toString()  
 val subject = etSubject.text.toString()  
 val message = etMessage.text.toString()  
 val intent = Intent(Intent.ACTION\_SEND)  
 intent.putExtra(Intent.EXTRA\_EMAIL, arrayOf(email))  
 intent.putExtra(Intent.EXTRA\_SUBJECT, subject)  
 intent.putExtra(Intent.EXTRA\_TEXT, message)  
 intent.type = "message/rfc822"  
 startActivity(Intent.createChooser(intent, "Choose an Email client:"))  
 }  
 }  
}

**Output**

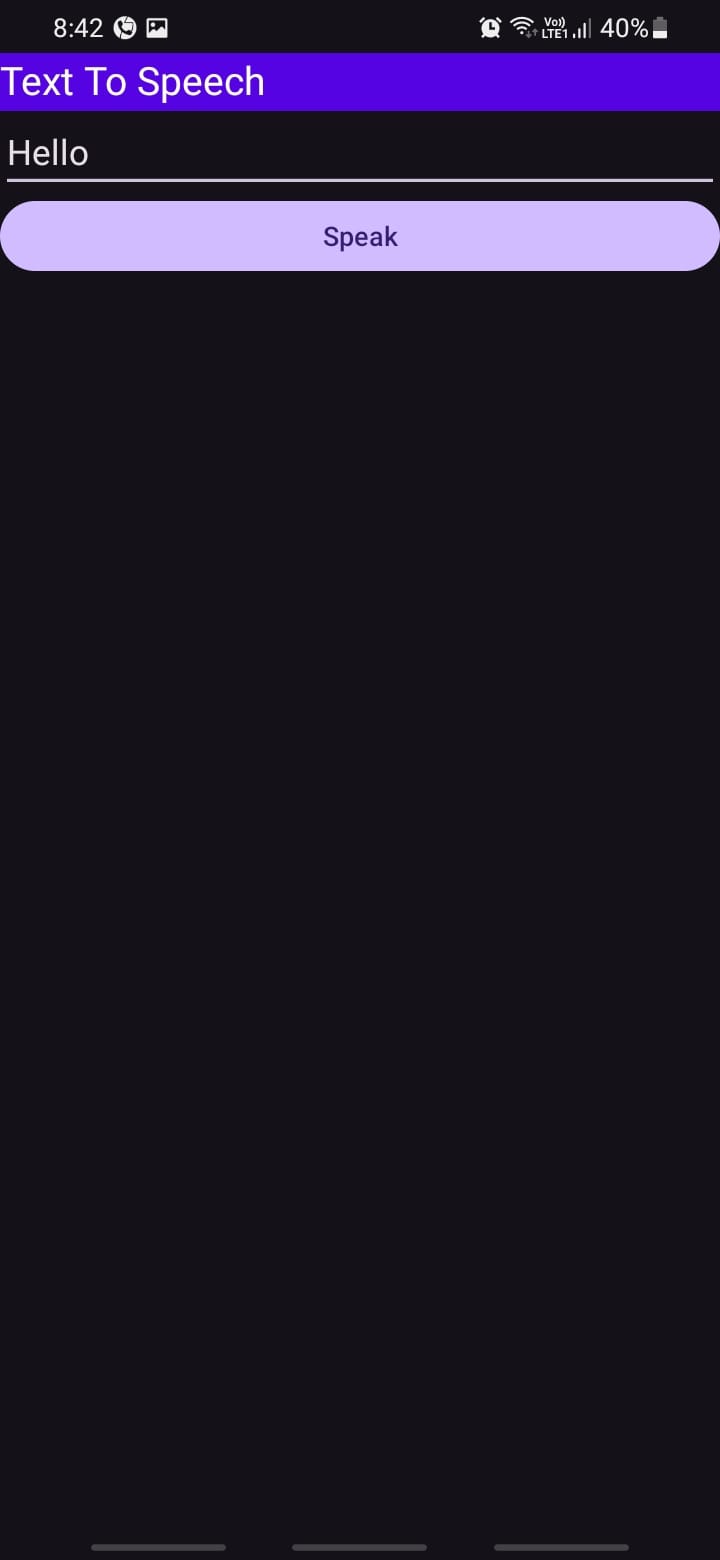


# ExNo:13 Text to Speech

// File: MainActivity.kt

package com.example.texttospeech  
import android.os.Bundle  
import android.speech.tts.TextToSpeech  
import android.widget.Button  
import android.widget.EditText  
import android.widget.Toast  
import androidx.activity.enableEdgeToEdge  
import androidx.appcompat.app.AppCompatActivity  
import androidx.core.view.ViewCompat  
import androidx.core.view.WindowInsetsCompat  
import org.w3c.dom.Text  
import java.util.Locale  
class MainActivity : AppCompatActivity(), TextToSpeech.OnInitListener {  
 private lateinit var etText: EditText  
 private lateinit var btSpeak: Button  
 private lateinit var textToSpeech: TextToSpeech  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 enableEdgeToEdge()  
 setContentView(R.layout.activity\_main)  
 etText = findViewById(R.id.etText)  
 btSpeak = findViewById(R.id.btSpeak)  
 textToSpeech = TextToSpeech(this,this)  
 btSpeak.setOnClickListener{  
 val text = etText.text!!.toString()  
 textToSpeech!!.speak(text, TextToSpeech.QUEUE\_FLUSH, null, "")  
 } ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->  
 val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())  
 v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)  
 insets  
 }  
 }  
 override fun onInit(status: Int) {  
 if (status == TextToSpeech.SUCCESS) {  
 val result = textToSpeech!!.setLanguage(Locale.US)  
 if (result == TextToSpeech.LANG\_MISSING\_DATA || result ==  
 TextToSpeech.LANG\_NOT\_SUPPORTED) {  
 Toast.makeText(applicationContext, "The Language not supported...!", Toast.LENGTH\_LONG).show()  
 }  
 else {  
 btSpeak!!.isEnabled = true  
 }  
 }  
 }  
}

**Output**



# ExNo:14 Speech to Text

// File: MainActivity.kt

package com.example.speechtotext  
import android.os.Bundle  
import androidx.activity.enableEdgeToEdge  
import androidx.appcompat.app.AppCompatActivity  
import androidx.core.view.ViewCompat  
import androidx.core.view.WindowInsetsCompat  
import android.content.Intent  
import android.speech.RecognizerIntent  
import android.widget.ImageView  
import android.widget.TextView  
import java.util.\*  
  
class MainActivity : AppCompatActivity() {  
 lateinit var tvText : TextView  
 private val REQUEST\_CODE\_SPEECH\_INPUT = 1000  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_main)  
 var imgMic : ImageView = findViewById(R.id.imgMic)  
 tvText = findViewById(R.id.tvText)  
 imgMic.setOnClickListener {  
 val intent = Intent(RecognizerIntent.ACTION\_RECOGNIZE\_SPEECH)  
 intent.putExtra(RecognizerIntent.EXTRA\_LANGUAGE\_MODEL,  
 RecognizerIntent.LANGUAGE\_MODEL\_FREE\_FORM)  
 intent.putExtra(RecognizerIntent.EXTRA\_LANGUAGE,  
 Locale.getDefault())  
 intent.putExtra(RecognizerIntent.EXTRA\_PROMPT, "Speak...!")  
 startActivityForResult(intent, REQUEST\_CODE\_SPEECH\_INPUT)  
 }  
 }  
 override fun onActivityResult(requestCode: Int, resultCode: Int, data:  
 Intent?) {  
 super.onActivityResult(requestCode, resultCode, data)  
 if(requestCode == REQUEST\_CODE\_SPEECH\_INPUT && resultCode == RESULT\_OK  
 && data != null) {  
 var res : ArrayList<String> =  
 data.getStringArrayListExtra(RecognizerIntent.EXTRA\_RESULTS) as  
 ArrayList<String>  
 tvText.setText( Objects.requireNonNull(res)[0])  
 }  
 }  
}

**Output**



# ExNo:15 Image Capture

// File: MainActivity.kt

package com.example.imagecapture  
import android.app.ComponentCaller  
import android.content.Intent  
import android.graphics.Bitmap  
import android.os.Bundle  
import android.provider.ContactsContract.CommonDataKinds.Photo  
import android.provider.MediaStore  
import android.widget.Button  
import android.widget.ImageView  
import androidx.activity.enableEdgeToEdge  
import androidx.appcompat.app.AppCompatActivity  
import androidx.core.view.ViewCompat  
import androidx.core.view.WindowInsetsCompat  
class MainActivity : AppCompatActivity() {  
 lateinit var imgViewPhoto: ImageView  
 private val CAMERA\_PERMISSION =1000  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 enableEdgeToEdge()  
 setContentView(R.layout.activity\_main)  
 imgViewPhoto = findViewById(R.id.imgViewPhoto)  
 val btTakePicture : Button = findViewById(R.id.btTakePicture)  
 btTakePicture.setOnClickListener {  
 val intent = Intent(MediaStore.ACTION\_IMAGE\_CAPTURE)  
 startActivityForResult(intent, CAMERA\_PERMISSION)  
 } ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->  
 val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())  
 v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)  
 insets  
 }  
 }  
 override fun onActivityResult(  
 requestCode: Int,  
 resultCode: Int,  
 data: Intent?,  
 caller: ComponentCaller  
 ) {  
 super.onActivityResult(requestCode, resultCode, data, caller)  
 if(requestCode == CAMERA\_PERMISSION && resultCode == RESULT\_OK)  
 {  
 val photo = data!!.extras?.get("data") as Bitmap  
 imgViewPhoto.setImageBitmap(photo)  
 }  
 }  
}

**Output**

