Print statement

```
fun main() {
println("Hello World...!")
}
```

```
Hello World...!
```

Variable declaration - val and var

```
fun main() {
val a = 5 var b = 5
a = 10
b = 10
println(a) println(b)
}
```

```
'val' cannot be reassigned.
```

Variable usage

```
fun main() {
val str = "Shinchan"
println("Hello, You got a message from $str")
}
```

```
Hello, You got a message from Shinchan
```

Assignment Options

```
fun main() { var test = 0
println("var test = 0 --> $test") test = 5
println("test = 5 -->test") test = test + 5
println("test = test + 5 --> $test") test+=5
println("test += 5 --> $test") test-=5
println("test -= 5 --> $test") test*=5
println("test *= 5 --> $test") test/=5
println("test /= 5 --> $test")
}
```

```
var test = 0 --> 0
test = 5 --> 5
test = test + 5 -->10
test += 5 --> 15
test -= 5 --> 10
test *= 5 --> 50
test /= 5 --> 10
```

Variable Declaration methods [either value/initialise or type mentioning]

```
fun main() {
val a: Int
a = 10  // 1st declared then assigned println("a = $a")
val b: Int = 20  // declared and assigned at the same time
println("b = $b")
}
```

```
a = 10
b = 20
```

// List – allows duplicate values

```
fun main() {
val lst: List<String> = listOf("Red","Yellow","Green") // :List<String> is not necessary
println(lst)
}
```

```
[Red, Yellow, Green]
```

// Mutable List

```
fun main() {
val lst: MutableList<String> = mutableListOf("Red","Yellow","Green")
println(lst)       // :MutableList<String> is not necessary lst.add("Blue")
println(lst)
}
```

```
[Red, Yellow, Green]
[Red, Yellow, Green, Blue]
```

// Set – doesn't allow duplicate values

```
fun main() {
val st: Set<String> = setOf("Monday","Tuesday","Wednesday")
println(st)
}
```

```
[Monday, Tuesday, Wednesday]
```

// Mutable Set

```
fun main() {
val st: MutableSet<String> = mutableSetOf("Monday","Tuesday","Wednesday")
println(st)
st.add("Sunday")
println(st)
st.remove("Monday")
println(st)
}
```

```
[Monday, Tuesday, Wednesday]
[Monday, Tuesday, Wednesday, Sunday]
[Tuesday, Wednesday, Sunday]
```

// Set Count

```
fun main() {
val st: Set<String> = setOf("Monday","Tuesday","Wednesday","Monday")
println("Set has ${st.count()} values.") // Counts only unique records
}
```

```
Set has 3 values.
```

```
fun main() {
val st: List<String> = listOf("Monday","Tuesday","Wednesday","Monday")
println("List has ${st.count()} values.") // Counts all records
}
```

```
List has 4 values.
```

// in keyword

```
fun main() {
val st: Set<String> = setOf("Sunday","Monday","Tuesday","Wednesday")
println("Sunday" in st)
}
```

```
true
```

// if-else statements

```
fun main() {
val a = 5
val b = 10
val max: Int
if (a > b) {max = a} else {max = b}
println("Maximum value is $max")
val new_max = if (a > b) a else b
println("Maximum value is $new_max")
}
```

```
Maximum value is 10
Maximum value is 10
```

// for loop

```
fun main() {
for (i in 1..5){
println("Value of i = $i")}}
```

```
Value of i = 1
Value of i = 2
Value of i = 3
Value of i = 4
Value of i = 5
```

MainActivity.kt

```
package com.test.practical3
import android.content.Intent
import android.net.Uri
import android.os.Bundle
import android.provider.MediaStore
import android.widget.Button
import android.widget.ImageView
import androidx.appcompat.app.AppCompatActivity
class MainActivity : AppCompatActivity() {
        lateinit var imageView: ImageView
        lateinit var button: Button
        private val pickImage = 100
        private var imageUri: Uri? = null
        override fun onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        title = "Image Chooser"
        imageView = findViewById(R.id.imageView)
        button = findViewById(R.id.buttonLoadPicture)
        button.setOnClickListener {
        val gallery = Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.INTERNAL_CONTENT_URI) startActivityForResult(gallery,
pickImage)
}
}
override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {
super.onActivityResult(requestCode, resultCode, data)
if (resultCode == RESULT_OK && requestCode == pickImage) {
        imageUri = data?.data
        imageView.setImageURI(imageUri)
        }
}
}
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="fill_parent" android:layout_height="fill_parent"
android:orientation="vertical"
android:padding="2dp">
<ImageView android:id="@+id/imageView"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_weight="1" />
```

```
<Button
android:id="@+id/buttonLoadPicture"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_gravity="center"
android:layout_weight="0"
android:text="Load Picture" />
</LinearLayout>
```

A] MainAcrtivity.kt

```
package com.test.practical2

import android.os.Bundle
import android.view.View
import android.widget.*
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val nameInput = findViewById<EditText>(R.id.name)
        val submitButton = findViewById<Button>(R.id.button)
        val displayMessage = findViewById<TextView>(R.id.textViewResponse)
        val radioGroup = findViewById<RadioGroup>(R.id.selectgender)
        val radioButton1 = findViewById<RadioButton>(R.id.radioButton1)
        val radioButton2 = findViewById<RadioButton>(R.id.radioButton2)
        val checkBox1 = findViewById<CheckBox>(R.id.checkBox1)
        val checkBox2 = findViewById<CheckBox>(R.id.checkBox2)
        val languageMessage = findViewById<TextView>(R.id.language)
        val spinner = findViewById<Spinner>(R.id.spinner)
        val classTextView = findViewById<TextView>(R.id.myclass)

        val year = arrayOf("FYCS", "SYCS", "TYCS")
        val arrayAdapter = ArrayAdapter(
            this,
            android.R.layout.simple_spinner_dropdown_item,
            year
        )
        spinner.adapter = arrayAdapter

        spinner.onItemSelectedListener = object : AdapterView.OnItemSelectedListener {
            override fun onItemSelected(
                parent: AdapterView<*>?,
                view: View?,
                position: Int,
                id: Long
            ) {
                classTextView.text = "Your Class: ${year[position]}"
            }

            override fun onNothingSelected(parent: AdapterView<*>?) {
                classTextView.text = "Please select your Class"
            }
        }

        radioGroup.setOnCheckedChangeListener { _, checkedId ->
            when (checkedId) {
                R.id.radioButton1 -> displayMessage.text = "Gender: ${radioButton1.text}"
                R.id.radioButton2 -> displayMessage.text = "Gender: ${radioButton2.text}"
            }
        }
```

```
        submitButton.setOnClickListener  {
            val name = nameInput.text.toString()
            displayMessage.text = "Welcome, $name!"
            when {
                checkBox1.isChecked && checkBox2.isChecked -> {
                    languageMessage.text = "You love both languages"}
                checkBox1.isChecked -> {
                    languageMessage.text = "You love Python"}
                checkBox2.isChecked -> {
                    languageMessage.text = "You love Kotlin"}
                else -> {languageMessage.text = "You don't like these languages"}}}}}
```

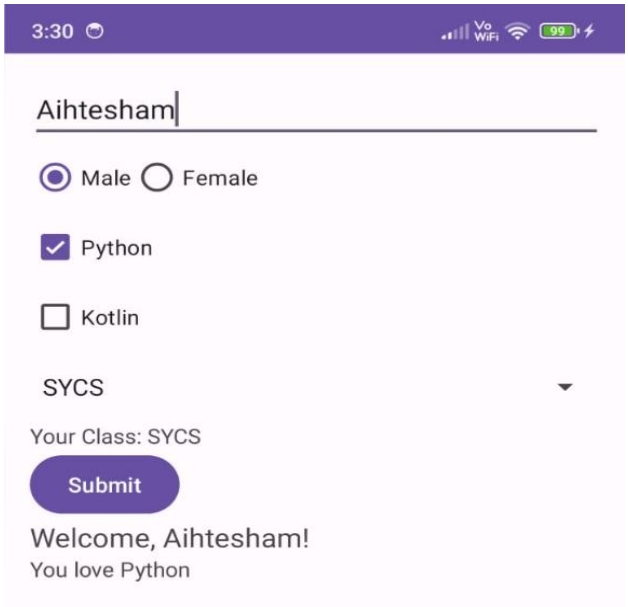
A] activity_main.xml

```
<?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">
    <EditText
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter your name" />
    <RadioGroup
        android:id="@+id/selectgender"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <RadioButton
            android:id="@+id/radioButton1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Male" />
        <RadioButton
            android:id="@+id/radioButton2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Female" />
    </RadioGroup>
    <CheckBox
        android:id="@+id/checkBox1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Python" />
    <CheckBox
        android:id="@+id/checkBox2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Kotlin" />
    <Spinner
        android:id="@+id/spinner"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
```

```
    <TextView
        android:id="@+id/myclass"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Your Class will appear here" />
    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Submit" />
    <TextView
        android:id="@+id/textViewResponse"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Response will appear here"
        android:textSize="18sp" />
    <TextView
        android:id="@+id/language"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Your language preference will appear here" />
</LinearLayout>
```

A] build:gradle.kts (:app)

```
android {
        defaultConfig {
                minSdk: 21
                targetSdk = 35
        }
}
```



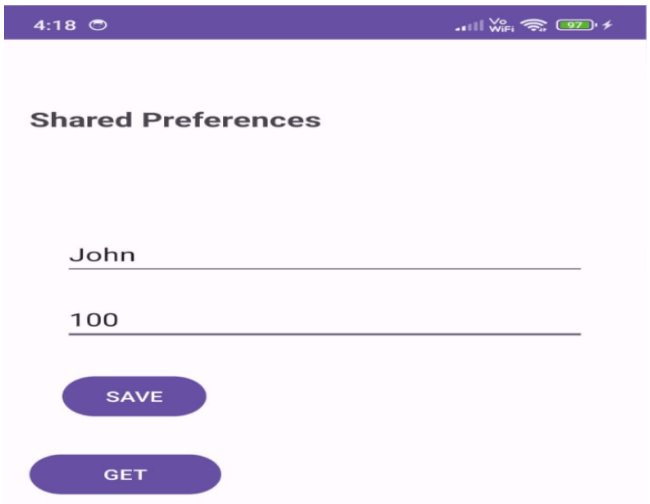
Practical 4

```
package com.example.sharedpreference

import android.content.SharedPreferences
import android.os.Bundle
import android.widget.Button import
android.widget.EditText import
android.widget.Toast
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val etName = findViewById<EditText>(R.id.name) val
        etRollNo = findViewById<EditText>(R.id.rollno) val save
        = findViewById<Button>(R.id.saveBtn)
        val get = findViewById<Button>(R.id.getBtn)

        val sharedPref = getSharedPreferences("addName", MODE_PRIVATE) val edit
        = sharedPref.edit()

        save.setOnClickListener {
            edit.putString("name", etName.text.toString())
            edit.putString("rollno", etRollNo.text.toString())
            edit.apply() // apply() is preferred over commit() for asynchronous saving
            Toast.makeText(this, "Data saved", Toast.LENGTH_SHORT).show()
        }
        get.setOnClickListener {
            val myName = sharedPref.getString("name", "default value") val
            myRoll = sharedPref.getString("rollno", "default value")
            Toast.makeText(this, "Name: $myName - Roll No: $myRoll",
Toast.LENGTH_SHORT).show()
        }}}
```

Activity.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">

    <TextView android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="50dp"
        android:text="Shared Preferences"
        android:textSize="20sp" android:textStyle="bold" />
```

```
<EditText android:id="@+id/name"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="100dp"
    android:layout_marginStart="20dp"
    android:layout_marginEnd="20dp" android:hint="Name"
    android:inputType="textPersonName" />

<EditText android:id="@+id/rollno"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/name"
    android:layout_marginTop="20dp"
    android:layout_marginStart="20dp"
    android:layout_marginEnd="20dp" android:hint="Roll
    No:" android:inputType="textPersonName" />
<Button
    android:id="@+id/saveBtn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/rollno"
    android:layout_marginTop="30dp"
    android:layout_marginStart="20dp" android:text="SAVE"
    />
    <Button
    android:id="@+id/getBtn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/rollno"
    android:layout_marginTop="30dp"
    android:layout_marginEnd="20dp"
    android:layout_alignParentEnd="true" android:text="GET"
    />
</LinearLayout>
```



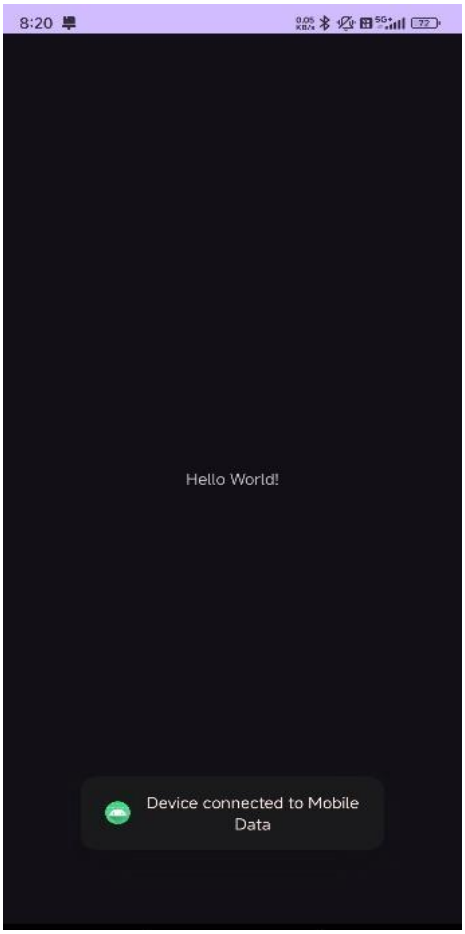
Practical 5

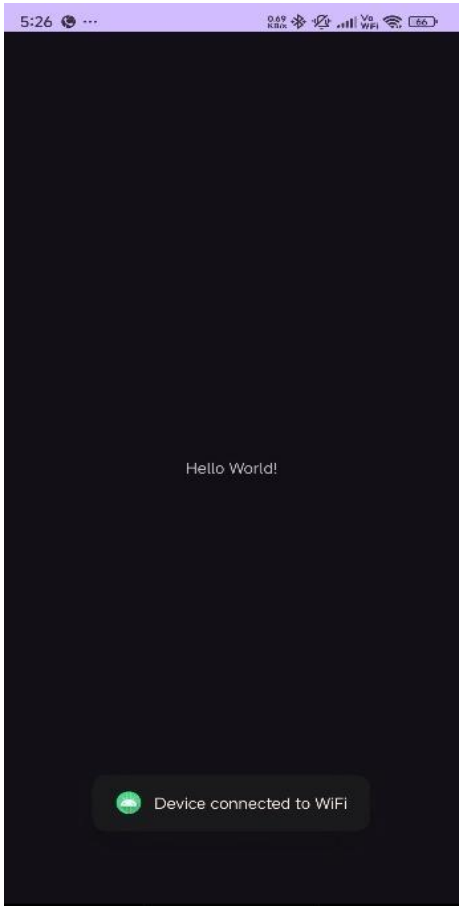
```
import android.content.Context
import android.content.Intent
import android.content.IntentFilter
import android.net.ConnectivityManager
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Toast

class MainActivity : AppCompatActivity() {
   override fun onCreate(savedInstanceState: Bundle?) {
      super.onCreate(savedInstanceState)
      setContentView(R.layout.activity_main)
      val c= applicationContext.getSystemService(Context.CONNECTIVITY_SERVICE) as
            ConnectivityManager
      val networdInfo= c.activeNetworkInfo
if(networdInfo!= null && networdInfo.isConnected)
      {
         if (networdInfo.type==ConnectivityManager.TYPE_MOBILE) {
            Toast.makeText(applicationContext, "Device connected to Mobile data",
               Toast.LENGTH_LONG).show()
         }
         if(networdInfo.type== ConnectivityManager.TYPE_WIFI)
         {
            Toast.makeText(applicationContext,"Device connected to Wifi",
               Toast.LENGTH_LONG).show()
         } }
      else {
         Toast.makeText(applicationContext,"You are Offline",
            Toast.LENGTH_LONG).show()
      } } }
```

Changes in AndroidManifest.xml file

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:tools="http://schemas.android.com/tools">
   <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE"></uses-
permission>
   <application
      android:allowBackup="true"
      android:dataExtractionRules="@xml/data_extraction_rules"
      android:fullBackupContent="@xml/backup_rules"
      android:icon="@mipmap/ic_launcher"
      android:label="@string/app_name"
      android:roundIcon="@mipmap/ic_launcher_round"
      android:supportsRtl="true"
      android:theme="@style/Theme.NetworkConnectivity"
      tools:targetApi="31">
```

```
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```



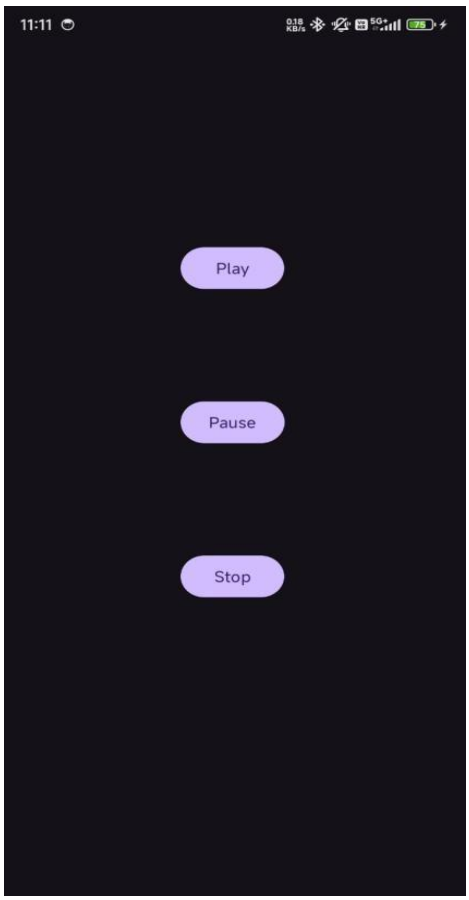


Practical 8

```
import android.media.MediaPlayer
import android.os.Bundle
import android.widget.Button
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {
    private lateinit var mediaPlayer: MediaPlayer
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        mediaPlayer = MediaPlayer.create(this, R.raw.music)

        val bPlay: Button = findViewById(R.id.playButton)
        val bPause: Button = findViewById(R.id.pauseButton)
        val bStop: Button = findViewById(R.id.stopButton)

        bPlay.setOnClickListener {
            if (!mediaPlayer.isPlaying) {
                mediaPlayer.start()
            }
        }

        bPause.setOnClickListener {
            if (mediaPlayer.isPlaying) {
                mediaPlayer.pause()
            }
        }
        bStop.setOnClickListener {
            if (mediaPlayer.isPlaying) {
                mediaPlayer.stop()
                mediaPlayer.prepare()
            }
        }
    }
    override fun onDestroy() {
        super.onDestroy()
        mediaPlayer.release()
    }
}
```

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
   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=".MainActivity"
   tools:ignore="HardcodedText">
   <TextView
      android:id="@+id/headingText"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_centerHorizontal="true"
      android:layout_marginTop="32dp"
      android:text="MEDIA PLAYER"
      android:textSize="18sp"
      android:textStyle="bold" />
   <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:layout_below="@id/headingText"
      android:layout_marginTop="16dp"
      android:gravity="center_horizontal">
      <Button
         android:id="@+id/stopButton"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:layout_marginEnd="8dp"
         android:backgroundTint="@color/colorPrimary"
         android:text="STOP"
         android:textColor="@android:color/white" />
      <Button
         android:id="@+id/playButton"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:layout_marginEnd="8dp"
         android:backgroundTint="@color/colorPrimary"
         android:text="PLAY"
         android:textColor="@android:color/white" />
      <Button
         android:id="@+id/pauseButton"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:backgroundTint="@color/colorPrimary"
         android:text="PAUSE"
         android:textColor="@android:color/white" />
   </LinearLayout>
</RelativeLayout>
```



Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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=".MainActivity">

   <Button
      android:id="@+id/button"
      android:layout_centerInParent="true"
      android:background="@color/colorPrimary"
      android:textColor="#ffffff"
      android:text="Let's Bounce"
      android:layout_width="200dp"
      android:layout_height="80dp"/>
</RelativeLayout>
```

.Kt file

```
import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity
import android.view.animation.Animation
import android.view.animation.AnimationUtils
import android.widget.Button

class MainActivity : AppCompatActivity() {
   override fun onCreate(savedInstanceState: Bundle?) {
      super.onCreate(savedInstanceState)
      setContentView(R.layout.activity_main)

      // Load the animation
      val animation: Animation = AnimationUtils.loadAnimation(this, R.anim.bounce)

      // Get the button and apply the animation on click
      val button: Button = findViewById(R.id.button)
      button.setOnClickListener {
         button.startAnimation(animation)
      }
   }
}
```

Create one more file name boumce.xml

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
   android:fillAfter="true"
   android:interpolator="@android:anim/bounce_interpolator">

   <scale
      android:pivotX="50%"
      android:pivotY="50%"
```

```
    android:fromXScale="0.5"
    android:toXScale="1.0"
    android:fromYScale="0.5"
    android:toYScale="1.0"
    android:duration="500"/>
</set>
```