

Experiment 7

Develop an android application to implement ListView with images and text. ***I have created a Photo gallery app that uses Pixabay's API to dynamically fetch searched photos.***

Activity.kt

```
1. package com.example.listviewbasicsactivity
2.
3. import android.os.Bundle
4. import android.util.Log
5. import android.widget.Button
6. import android.widget.EditText
7. import android.widget.ListView
8. import androidx.appcompat.app.AppCompatActivity
9. import com.android.volley.Request
10. import com.android.volley.toolbox.JsonObjectRequest
11. import com.android.volley.toolbox.Volley
12. import kotlinx.android.synthetic.main.activity_main.*
13.
14.
15. class MainActivity() : AppCompatActivity(){
16.     private lateinit var listView:ListView
17.     private lateinit var searchImg:EditText
18.     private lateinit var searchBtn:Button
19.
20.     override fun onCreate(savedInstanceState: Bundle?) {
21.         super.onCreate(savedInstanceState)
22.         setContentView(R.layout.activity_main)
23.
24.         searchBtn = findViewById(R.id.searchBtn)
25.         searchImg = findViewById(R.id.searchImg)
26.         listView = findViewById(R.id.imageListCnt)
27.         fetchImages()
28.         searchBtn.setOnClickListener {
29.             val toSearch = searchImg.text.toString().trim()
30.             fetchImages(toSearch)
31.         }
32.
33.     }
34.
35.     private fun fetchImages(searchText: String=""){
36.         //create data
37.         var dataSet = arrayListOf<ListViewConstructor>() //constructor data class
38.         var imgUrls = arrayListOf<String>()
39.         var imgAuthors = arrayListOf<String>()
40.         var imgLikeCounter = arrayListOf<String>()
41.         val url = if(searchText.isNotEmpty())
42.             "https://pixabay.com/api/?key=18298258-2816622e399b7e18e43565966&image_type=photo&pretty=true&q=${searchText}&per_page=10&orientation=horizontal"
43.         else
44.             "https://pixabay.com/api/?key=18298258-2816622e399b7e18e43565966&image_type=photo&pretty=true&editors_choice=true&per_page=10&orientation=horizontal"
45.
46.
47.         val queue = Volley.newRequestQueue(this)
48.         // Request a string response from the provided URL.
49.         val jsonRequest = JsonObjectRequest(
50.             Request.Method.GET, url, null,
```

Experiment 7

Develop an android application to implement ListView with images and text. ***I have created a Photo gallery app that uses Pixabay's API to dynamically fetch searched photos.***

```
51.         { response ->
52.             // Display the first 500 characters of the response string.
53.             val images = response.getJSONArray("hits")
54.
55.             for (pos in 0 until images.length()) {
56.                 val imgObj = images.getJSONObject(pos)
57.                 imgAuthors.add(imgObj.getString("user"))
58.                 imgUrls.add(imgObj.getString("largeImageURL"))
59.                 imgLikeCounter.add(imgObj.getString("likes"))
60.             }
61.             Log.d("Pixabay Auth Req", imgAuthors.toString())
62.             Log.d("Pixabay Img req", imgUrls.toString())
63.             //0..4 cuz we are exactly fetching 5 data from the api json file
64.             for (pos in 0 until imgUrls.size) {
65.                 Log.d("Data set", pos.toString())
66.                 dataSet.add(ListViewConstructor(imgUrls[pos], imgLikeCounter[pos], i
mgAuthors[pos])) //appending results in the list
67.             }
68.             Log.d("Pixabay ds", dataSet.toString())
69.
70.             //custom adapter
71.             val customAdapter = PixabayAdapter(
72.                 applicationContext,
73.                 dataSet
74.             )
75.             imageListCnt.adapter = customAdapter
76.
77.         },
78.         {
79.             Log.d("Pixabay Req", "Request failure, image not fetched!")
80.         }
81.     )
82.     // Add the request to the RequestQueue.
83.     queue.add(jsonRequest)
84.
85. }
86.
87. }
```

ImageViewAdapter.kt

```
1. package com.example.listviewbasicsactivity
2.
3. import android.content.Context
4. import android.view.LayoutInflater
5. import android.view.View
6. import android.view.ViewGroup
7. import android.widget.BaseAdapter
8. import android.widget.ImageView
9. import android.widget.TextView
10. import com.bumptech.glide.Glide
11. import java.util.*
12.
13.
14.
15. class PixabayAdapter(private val context: Context,
```

Experiment 7

Develop an android application to implement ListView with images and text. ***I have created a Photo gallery app that uses Pixabay's API to dynamically fetch searched photos.***

```
16.                                     private val dataSource: ArrayList<ListViewCons
    tructor>) : BaseAdapter() {
17.
18.     private val inflater: LayoutInflater = context.getSystemService(Context.LAYOUT_INFL
    ATER_SERVICE) as LayoutInflater
19.
20.     override fun getCount(): Int {
21.         return dataSource.size
22.     }
23.
24.     override fun getItem(p0: Int): Any {
25.         return dataSource[p0]
26.     }
27.
28.     override fun getItemId(p0: Int): Long {
29.         return p0.toLong()
30.     }
31.
32.     override fun getView(position: Int, rowLayout: View?, generatedLayout: ViewGroup?):
    View {
33.
34.         /* Get view for row item */
35.         val rowView = inflater.inflate(R.layout.activity_row_style, generatedLayout, fa
    lse)
36.         val photographerName = rowView.findViewById(R.id.photographer) as TextView
37.         val totalLikes = rowView.findViewById(R.id.totalLikes) as TextView
38.         val thumbnailImageView = rowView.findViewById(R.id.imageCnt) as ImageView
39.         val image = getItem(position) as ListViewConstructor
40.
41.         photographerName.text = "Clicked by: "+image.photographer
42.         totalLikes.text = "Total Likes: "+image.image_likes
43.         Glide.with(context).load(image.image_url).into(thumbnailImageView)
44.         return rowView
45.     }
46. }
```

ListViewConstructor.kt

It is a data class that holds, image url and stats like the photographer name and total likes on the image. Used while creating a data source like list of received images from the API, for the adapter which further places all these info into the list view.

```
1. package com.example.listviewbasicsactivity
2.
3. data class ListViewConstructor(var image_url: String, var image_likes:String, var photo
    grapher: String)
4.
5. }
```

Experiment 7

Develop an android application to implement ListView with images and text. *I have created a Photo gallery app that uses Pixabay's API to dynamically fetch searched photos.*

Acitvity.xml

```
1. <?xml version="1.0" encoding="utf-8"?>
2. <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3.     xmlns:app="http://schemas.android.com/apk/res-auto"
4.     xmlns:tools="http://schemas.android.com/tools"
5.     android:layout_width="match_parent"
6.     android:layout_height="match_parent"
7.     android:padding="16dp"
8.     tools:context=".MainActivity">
9.
10.    <ListView
11.        android:id="@+id/imageListCnt"
12.        android:layout_width="match_parent"
13.        android:layout_height="0dp"
14.        android:divider="@android:color/black"
15.        android:dividerHeight="1dp"
16.        android:padding="16dp"
17.        app:layout_constraintBottom_toBottomOf="parent"
18.        app:layout_constraintEnd_toEndOf="parent"
19.        app:layout_constraintHorizontal_bias="0.466"
20.        app:layout_constraintStart_toStartOf="parent"
21.        app:layout_constraintTop_toBottomOf="@+id/searchImg"
22.        app:layout_constraintVertical_bias="1.0" />
23.
24.    <EditText
25.        android:id="@+id/searchImg"
26.        android:layout_width="wrap_content"
27.        android:layout_height="wrap_content"
28.        android:layout_margin="10dp"
29.        android:ems="10"
30.        android:hint="@string/searchInput_hint"
31.        android:inputType="textPersonName"
32.        app:layout_constraintBaseline_toBaselineOf="@+id/searchBtn"
33.        app:layout_constraintEnd_toStartOf="@+id/searchBtn"
34.        app:layout_constraintHorizontal_bias="0.5"
35.        app:layout_constraintStart_toStartOf="parent" />
36.
37.    <Button
38.        android:id="@+id/searchBtn"
39.        android:layout_width="wrap_content"
40.        android:layout_height="wrap_content"
41.        android:layout_margin="10dp"
42.        android:text="@string/searchBtn_txt"
43.        app:layout_constraintEnd_toEndOf="parent"
44.        app:layout_constraintHorizontal_bias="0.5"
45.        app:layout_constraintStart_toEndOf="@+id/searchImg"
46.        app:layout_constraintTop_toTopOf="parent" />
47.
48. </androidx.constraintlayout.widget.ConstraintLayout>
```

Experiment 7

Develop an android application to implement ListView with images and text. ***I have created a Photo gallery app that uses Pixabay's API to dynamically fetch searched photos.***

ListViewRowStyle.xml

```
1. <?xml version="1.0" encoding="utf-8"?>
2. <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3.     xmlns:app="http://schemas.android.com/apk/res-auto"
4.     xmlns:tools="http://schemas.android.com/tools"
5.     android:layout_width="match_parent"
6.     android:layout_height="match_parent">
7.
8.     <ImageView
9.         android:id="@+id/imageCnt"
10.        android:layout_width="0dp"
11.        android:layout_height="246dp"
12.        android:layout_marginTop="56dp"
13.        app:layout_constraintEnd_toEndOf="parent"
14.        app:layout_constraintHorizontal_bias="0.5"
15.        app:layout_constraintStart_toStartOf="parent"
16.        app:layout_constraintTop_toTopOf="parent"
17.        tools:srcCompat="@tools:sample/avatars" />
18.
19.     <TextView
20.         android:id="@+id/photographer"
21.         android:layout_width="0dp"
22.         android:layout_height="30dp"
23.         android:fontFamily="monospace"
24.         android:paddingStart="5dp"
25.         android:text="TextView"
26.         android:textColor="#152267"
27.         android:textSize="18sp"
28.         app:layout_constraintBottom_toBottomOf="parent"
29.         app:layout_constraintEnd_toEndOf="parent"
30.         app:layout_constraintHorizontal_bias="0.487"
31.         app:layout_constraintStart_toStartOf="parent"
32.         app:layout_constraintTop_toBottomOf="@+id/totalLikes"
33.         app:layout_constraintVertical_bias="0.0" />
34.
35.     <TextView
36.         android:id="@+id/totalLikes"
37.         android:layout_width="0dp"
38.         android:layout_height="30dp"
39.         android:layout_marginTop="12dp"
40.         android:fontFamily="monospace"
41.         android:paddingStart="5dp"
42.         android:text="TextView"
43.         android:textColor="#152267"
44.         android:textSize="18sp"
45.         app:layout_constraintEnd_toEndOf="parent"
46.         app:layout_constraintHorizontal_bias="0.512"
47.         app:layout_constraintStart_toStartOf="parent"
48.         app:layout_constraintTop_toBottomOf="@+id/imageCnt" />
49.
50. </androidx.constraintlayout.widget.ConstraintLayout>
```

Experiment 7

Develop an android application to implement ListView with images and text. ***I have created a Photo gallery app that uses Pixabay's API to dynamically fetch searched photos.***

Snapshots:

