

1. Write an android application to use of Shared Preferences. (Add, Delete)

activity shared preference a e s.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".SharedPreferenceAES">
    <TextView
        android:id="@+id/textview"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_gravity="center"
        android:layout_marginTop="32dp"
        android:text="Shared Preferences Demo"
        android:textColor="@android:color/black"
        android:textSize="24sp" />

    <!--EditText to take the data from the user
    and save the data in SharedPreferences-->
    <EditText
        android:id="@+id/edname"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textview"
        android:layout_marginStart="16dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="16dp"
        android:hint="Enter your Name"
        android:padding="10dp" />

    <!--EditText to take the data from the user and
    save the data in SharedPreferences-->
    <EditText
        android:id="@+id/edage"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/edit1"
        android:layout_marginStart="16dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="16dp"
        android:hint="Enter your Age"
        android:padding="10dp"
        android:inputType="number" />

    <LinearLayout
```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:orientation="horizontal">

```

```

<Button
    android:id="@+id/btnsave"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="SAVE" />

```

```

<Button
    android:id="@+id/btnupdate"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="Update" />

```

```

<Button
    android:id="@+id/btndelete"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="DELETE" />

```

```

<Button
    android:id="@+id/btnclear"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="Clear" />

```

```

</LinearLayout>
</LinearLayout>

```

SharedPreferenceAES.kt

```

package com.example.ch5
import android.content.Context
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Toast
import kotlinx.android.synthetic.main.activity_shared_preference_demo.*
import kotlinx.android.synthetic.main.activity_shared_preference_demo.btnsave
import kotlinx.android.synthetic.main.activity_shared_preference_demo.edage
import kotlinx.android.synthetic.main.activity_shared_preference_demo.edname
import kotlinx.android.synthetic.main.activity_shared_preference_a_e_s.*
class SharedPreferenceAES : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_shared_preference_a_e_s)
    }
}

```

```

btnsave.setOnClickListener {
    var sharedPref =
        getSharedPreferences("MySharedPrefAES", Context.MODE_PRIVATE)
        val myEdit = sharedPref.edit()
        myEdit.putString("name", edname.text.toString())
        myEdit.putInt("age", edage.text.toString().toInt())
        myEdit.apply()
        clear()
    }
btndelete.setOnClickListener {
    val sharedPreferences =
        getSharedPreferences("MySharedPrefAES", Context.MODE_PRIVATE)
    val sharededit=sharedPreferences.edit()
    val sharedIdValue = sharedPreferences.getInt("age",0)
    val sharedNameValue = sharedPreferences.getString("name","defaultname")
    if(edage.text.toString().toInt() == sharedIdValue &&
sharedNameValue.equals(edname.text.toString())){
        sharededit.remove("name").toString()
        sharededit.remove("age").toString()
        sharededit.apply()
        Toast.makeText(this,"Delete Your $sharedNameValue
Data",Toast.LENGTH_SHORT).show()
        clear()
    }else if (edname.text.toString() != sharedNameValue.toString()) {
        Toast.makeText(this,"Please Enter Valid Data",Toast.LENGTH_SHORT).show()
        clear()
    }else if (edage.text.toString().toInt() != sharedIdValue){
        Toast.makeText(this,"Please Enter Valid Data",Toast.LENGTH_SHORT).show()
        clear()
    }
    }
    }
btnclear.setOnClickListener {
    clear()
    }
    btn
}
fun clear(){
    edname.setText("").toString()
    edage.setText("").toString()
}
}

```

2. Write an android application to use of Google Map Location.

activity_map_demo.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
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=".MapDemo">
        <fragment xmlns:android="http://schemas.android.com/apk/res/android"
            xmlns:tools="http://schemas.android.com/tools"
            android:id="@+id/map"
            android:name="com.google.android.gms.maps.SupportMapFragment"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            tools:context=".MapsActivity" />
    </androidx.constraintlayout.widget.ConstraintLayout>

```

MapDemo.kt

```
package com.example.ch_5
```

```

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import androidx.core.app.ActivityCompat
import android.Manifest
import android.content.pm.PackageManager
import android.location.Location
import com.google.android.gms.location.FusedLocationProviderClient
import com.google.android.gms.location.LocationServices
import com.google.android.gms.maps.CameraUpdateFactory
import com.google.android.gms.maps.GoogleMap
import com.google.android.gms.maps.OnMapReadyCallback
import com.google.android.gms.maps.SupportMapFragment
import com.google.android.gms.maps.model.LatLng
import com.google.android.gms.maps.model.MarkerOptions

class MapDemo : AppCompatActivity(), OnMapReadyCallback {

    private var currentLocation : Location? = null
    private var fusedLocationProviderClient: FusedLocationProviderClient? = null
    private val REQUEST_CODE = 101

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_map_demo)
        fusedLocationProviderClient = LocationServices.getFusedLocationProviderClient(this)

        fetchLocation()
    }
    private fun fetchLocation() {
        if
(ActivityCompat.checkSelfPermission(this,Manifest.permission.ACCESS_FINE_LOCATION)
        != PackageManager.PERMISSION_GRANTED &&
        ActivityCompat.checkSelfPermission(this,Manifest.permission.ACCESS_COARSE_LOCATION)

```

```

        != PackageManager.PERMISSION_GRANTED)
    {
        ActivityCompat.requestPermissions(this,
arrayOf(Manifest.permission.ACCESS_FINE_LOCATION), REQUEST_CODE)
        return
    }

    val task = fusedLocationProviderClient!!.lastLocation
    task.addOnSuccessListener { location ->
        if (location != null){
            currentLocation = location
            val supportMapFragment =
(supportFragmentManager.findFragmentById(R.id.map) as SupportMapFragment?)
            supportMapFragment!!.getMapAsync(this)
        }
    }
}

override fun onMapReady(googleMap: GoogleMap) {
    val latLng = LatLng(currentLocation!!.latitude, currentLocation!!.longitude)
    val markerOptions = MarkerOptions().position(latLng).title("I Am Here!")
    googleMap.animateCamera(CameraUpdateFactory.newLatLng(latLng))
    googleMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLng,15f))
    googleMap.addMarker(markerOptions)
}

override fun onRequestPermissionsResult(requestCode: Int, permissions: Array<out
String>, grantResults: IntArray) {
    when(requestCode){
        REQUEST_CODE -> {
            if (grantResults.isNotEmpty() && grantResults[0] ==
PackageManager.PERMISSION_GRANTED){
                fetchLocation()
            }
        }
    }
    super.onRequestPermissionsResult(requestCode, permissions, grantResults)
}
}

```

3. Write an android application to use of WebView Control.

activity_web_view_demo.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    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=".WebViewDemo">
    <WebView
        android:id="@+id/webView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

WebViewDemo.kt

```

package com.example.ch_3_afternoon

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.webkit.WebViewClient
import kotlinx.android.synthetic.main.activity_web_view_demo.*

class WebViewDemo : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_web_view_demo)
        // WebViewClient allows you to handle
        // onPageFinished and override Url loading.
        webView.webViewClient = WebViewClient()
        // this will load the url of the website
        webView.loadUrl("https://www.mjkacc.com/")
        // this will enable the javascript settings, it can also allow xss vulnerabilities
        webView.settings.javaScriptEnabled = true
        // if you want to enable zoom feature
        webView.settings.setSupportZoom(true)
    }
    // if you press Back button this code will work
    override fun onBackPressed() {
        // if your webview can go back it will go back
        if (webView.canGoBack())
            webView.goBack()
        // if your webview cannot go back
        // it will exit the application
        else
            super.onBackPressed()
    }
}

```

4. Write an android application to save contact in application. (Telephony API)

activity_contact_demo.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=".ContactDemo"
    android:padding="16dp"
    android:orientation="vertical">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:padding="8dp"
        android:text="Add new contacts to phone.."
        android:textAlignment="center"
        android:textColor="@android:color/holo_blue_dark"
        android:textSize="24sp"
        android:textStyle="bold" />

    <EditText
        android:id="@+id/etName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:inputType="text" />

    <EditText
        android:id="@+id/etNumber"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="number" />

    <Button
        android:id="@+id/buttonSave"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Save contact" />

</LinearLayout>

```

ContactDemo.kt

```

package com.example.ch_5
import android.app.Activity
import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.provider.ContactsContract
import android.widget.EditText
import android.widget.Toast
import kotlinx.android.synthetic.main.activity_contact_demo.*
class ContactDemo : AppCompatActivity() {

```

```

override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_contact_demo)
    buttonSave.setOnClickListener {
        addContact()
    }
}

fun addContact() {
    val etContactName: EditText = findViewById(R.id.etName)
    val etContactNumber: EditText = findViewById(R.id.etNumber)
    val name: String = etContactName.text.toString()
    val phone = etContactNumber.text.toString()
    val intent = Intent(ContactsContract.Intents.Insert.ACTION)
    intent.setType(ContactsContract.RawContacts.CONTENT_TYPE)
    intent.putExtra(ContactsContract.Intents.Insert.NAME, name)
    intent.putExtra(ContactsContract.Intents.Insert.PHONE, phone)
    startActivityForResult(intent, 1)
}

override fun onActivityResult(
    requestCode: Int,
    resultCode: Int,
    intent: Intent?
) {
    super.onActivityResult(requestCode, resultCode, intent)
    if (requestCode == 1) {
        if (resultCode == Activity.RESULT_OK) {
            Toast.makeText(this, "Added Contact", Toast.LENGTH_SHORT).show()
        }
        if (resultCode == Activity.RESULT_CANCELED) {
            Toast.makeText(this, "Cancelled Added Contact",
Toast.LENGTH_SHORT).show()
        }
    }
}
}

```

5. Write an android application to use of Notification.

activity_notification_demo.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".NotificationDemo">
    <Button
        android:id="@+id/btn"

```



```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_centerInParent="true"
    android:text="Show Notification" />
```

```
<Button
    android:id="@+id/btndownload"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_centerInParent="true"
    android:text="Download Notification" />
```

```
<Button
    android:id="@+id/btnexpand"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_centerInParent="true"
    android:text="Expand Notification" />
```

```
</LinearLayout>
```

NotificationDemo.kt

```
package com.example.ch5
import android.app.Notification
import android.app.NotificationChannel
import android.app.NotificationManager
import android.app.PendingIntent
import android.content.Context
import android.content.Intent
import android.graphics.BitmapFactory
import android.graphics.Color
import android.os.Build
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.os.SystemClock
import android.view.View
import android.widget.Button
import android.widget.RemoteViews
import android.widget.Toast
import androidx.core.app.NotificationBuilderWithBuilderAccessor
import androidx.core.app.NotificationCompat
import androidx.core.app.NotificationManagerCompat
import kotlinx.android.synthetic.main.activity_notification_demo.*
class NotificationDemo : AppCompatActivity() {
    lateinit var notificationChannel: NotificationChannel
    lateinit var notificationManager: NotificationManager
    lateinit var builder: Notification.Builder
    private val channelId = "12345"
    private val description = "Test Notification"

    override fun onCreate(savedInstanceState: Bundle?) {
```

```

super.onCreate(savedInstanceState)
setContentView(R.layout.activity_notification_demo)

notificationManager = getSystemService(Context.NOTIFICATION_SERVICE) as
NotificationManager

btn.setOnClickListener {
    // pendingIntent is an intent for future use i.e after
    // the notification is clicked, this intent will come into action
    val intent = Intent(this, AfterNotification::class.java)
    val pendingIntent = PendingIntent.getActivity(this, 0, intent,
PendingIntent.FLAG_UPDATE_CURRENT)
    // RemoteViews are used to use the content of
    // some different layout apart from the current activity layout
    val contentView = RemoteViews(packageName, R.layout.activity_after_notification)

    // checking if android version is greater than oreo(API 26) or not
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
        notificationChannel = NotificationChannel(channelId, description,
NotificationManager.IMPORTANCE_HIGH)
        notificationChannel.enableLights(true)
        notificationChannel.lightColor = Color.BLUE
        notificationChannel.enableVibration(false)
        notificationManager.createNotificationChannel(notificationChannel)

        builder = Notification.Builder(this, channelId)
            .setContent(contentView)
            .setContentTitle("Notification Demo")
            .setContentText("Hello")
            .setSmallIcon(R.drawable.ic_launcher_background)
            .setLargeIcon(BitmapFactory.decodeResource(this.resources,
R.drawable.ic_launcher_background))
            .setContentIntent(pendingIntent)
    } else {

        builder = Notification.Builder(this)
            .setContent(contentView)
            .setContentTitle("Notification Demo")
            .setContentText("Hello")
            .setSmallIcon(R.drawable.ic_launcher_background)
            .setLargeIcon(BitmapFactory.decodeResource(this.resources,
R.drawable.ic_launcher_background))
            .setContentIntent(pendingIntent)
    }
    notificationManager.notify(1234, builder.build())
}

btndownload.setOnClickListener {
    //Start() is called when the buttons is pressed.
    fun start(view: View){

```

```

val intent = Intent(this, MainActivity::class.java).apply{
    flags = Intent.FLAG_ACTIVITY_NEW_TASK or
        Intent.FLAG_ACTIVITY_CLEAR_TASK
}

val pendingIntent: PendingIntent = PendingIntent.getActivity(
    this, 0, intent, 0)

//Sets the maximum progress as 100
val progressMax = 100
//Creating a notification and setting its various attributes
val notification =
    NotificationCompat.Builder(this, channelId)
        .setContentTitle("GeeksforGeeks")
        .setContentText("Downloading")
        .setPriority(NotificationCompat.PRIORITY_LOW)
        .setOngoing(true)
        .setOnlyAlertOnce(true)
        .setProgress(progressMax, 0, true)
        .setContentIntent(pendingIntent)
        .setAutoCancel(true)

//Initial Alert
notificationManager.notify(1, notification.build())

Thread(Runnable{
    SystemClock.sleep(2000)
    var progress = 0
    while (progress <= progressMax) {
        SystemClock.sleep(
            1000
        )
        progress += 20
        //Use this to make it a Fixed-duration progress indicator notification

        //notification.setContentText(progress.toString()+"%")
        //.setProgress(progressMax, progress, false)

        //notificationManager.notify(1, notification.build())
    }

    notification.setContentText("Download complete")
        .setProgress(0, 0, false)
        .setOngoing(false)
    notificationManager.notify(1, notification.build())
}).start()
}
}

```

```

        btnexpand.setOnClickListener {
            var notification = NotificationCompat.Builder(this, channelId)
                .setSmallIcon(R.drawable.espresso)
                .setContentTitle("5 New mails from ")
                .setContentText("imageDescription")
                .setStyle(NotificationCompat.InboxStyle()
                    .addLine("messageSnippet1")
                    .addLine("messageSnippet2"))
            notification.build()
            //Initial Alert
            notificationManager.notify(1, notification.build())
        }
    }
}

```

activity_after_notification.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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=".AfterNotifcation">
    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:text="Welcome To Notifcation"
        android:textSize="15sp"
        android:textStyle="bold" />
</RelativeLayout>

```

6. Write an android application to use of JSON.

activity_json_demo.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    android:orientation="vertical"
    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=".JSONDemo">
    <ListView
        android:id="@+id/user_list"

```

```

        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:dividerHeight="1dp" />
</LinearLayout>

```

JSONDemo.kt

```

package com.example.ch_3_afternoon
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.util.Log
import android.widget.ListAdapter
import android.widget.ListView
import android.widget.SimpleAdapter
import org.json.JSONException
import org.json.JSONObject
class JSONDemo : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_j_s_o_n_demo)
        // private string declare in the latter section of the program
        val jsonStr = listData
        try {
            // Create a userList string hashmap arraylist
            val userList = ArrayList<HashMap<String, String?>>()
            // Declaring the listView from the layout file
            val lv = findViewById<ListView>(R.id.user_list)
            // Initializing the JSON object and extracting the information
            val jsonObj = JSONObject(jsonStr)
            val jsonArray = jsonObj.getJSONArray("users")
            for (i in 0 until jsonArray.length()) {
                val user = HashMap<String, String?>()
                val obj = jsonArray.getJSONObject(i)
                user["name"] = obj.getString("name")
                user["designation"] = obj.getString("designation")
                user["location"] = obj.getString("location")
                userList.add(user)
            }
            // ListAdapter to broadcast the information to the list elements
            val adapter: ListAdapter = SimpleAdapter(
                this, userList, R.layout.list_row,
                arrayOf("name", "designation", "location"), intArrayOf(
                    R.id.name,
                    R.id.designation, R.id.location
                )
            )
            lv.adapter = adapter
        } catch (ex: JSONException) {
            Log.e("JsonParser Example", "unexpected JSON exception", ex)
        }
    }
}

```

```

    }
    /*{
        users:[
            {
                "name":"Ace",
                "designation":"Engineer",
                "location":"New Yourk"
            },
        ]
    },*/
    // JSON object in the form of input stream
    private val listData: String
    get() = ("{" + "\"users\"":[" +
        "\"name\":\"Kishan\", \"designation\":\"Engineer\", \"location\":\"Ahemdabad\"" +
        "\", {\"name\":\"Aarti\", \"designation\":\"Director\", \"location\":\"Gandhinagar\"" +
        "\", {\"name\":\"Pooja\", \"designation\":\"Chartered Accountant\", \"location\":\"Baroda\"}] }")
}

```

list_row.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="5dip">

    <!--TextView to display the name-->
    <TextView
        android:id="@+id/name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="17dp"
        android:textStyle="bold" />

    <!--TextView to display the designation-->
    <TextView
        android:id="@+id/designation"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/name"
        android:layout_marginTop="7dp"
        android:textColor="#343434"
        android:textSize="14dp" />

    <!--TextView to display the location-->
    <TextView
        android:id="@+id/location"
        android:layout_width="wrap_content"

```

```
        android:layout_height="wrap_content"
        android:layout_alignBaseline="@+id/designation"
        android:layout_alignBottom="@+id/designation"
        android:layout_alignParentRight="true"
        android:textColor="#343434"
        android:textSize="14dp" />
</RelativeLayout>
```