Main Kotlin File.

```
package com.example.kotlinyoutube

import android.annotation.SuppressLint
import android.app.Activity
import android.content.Intent
import android.os.Bundle
import android.view.View
import android.view.inputmethod.InputMethodManager
import androidx.appcompat.app.ActionBar
import androidx.appcompat.app.AppCompatActivity
import androidx.fragment.app.FragmentTransaction
import com.example.kotlinyoutube.ui.HomeFragment
import kotlinx.android.synthetic.main.activity_main.*
import com.firebase.ui.auth.AuthUI
import com.google.firebase.auth.FirebaseAuth

class MainActivity: AppCompatActivity() {

    private lateinit var homeFragment: HomeFragment

    companion object {
        private const val RC_SIGN_IN = 123

        val PLAYLIST_HTTP_URL =
"https://www.googleapis.com/youtube/v3/playlistItems?&maxResults=100" +
                "&playlistId=PLx0sYbCqOb8TBPRdmBHs5Iftvv9TPboYG" +
                "&key=${BuildConfig.MY_SECRET_API_KEY}" +
                "&fields=items(snippet(publishedAt,title,description,channelTitle,position,thumbnails))" +
                "&part=snippet"
    }

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        // set up action bar
        setSupportActionBar(toolbar)
        supportActionBar?.let{
            initActionBar(it)
        }
        createSignInIntent()
    }

    // Source: https://github.com/firebase/snippets-android/blob/82c214e66547830dd9f8cbe66019e616f22881e6/auth/app/src/main/java/com/google/firebase/quickstart/auth/kotlin/FirebaseUIActivity.kt#
    private fun createSignInIntent() {
        // Choose authentication providers
        val providers = arrayListOf(
            AuthUI.IdpConfig.EmailBuilder().build(),
        )

        // Create and launch sign-in intent
        startActivityForResult(
            AuthUI.getInstance()
                .createSignInIntentBuilder()
                .setIsSmartLockEnabled(false)
                .setAvailableProviders(providers)
                .build(),
            RC_SIGN_IN
        )
    }
```

```
override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {
  super.onActivityResult(requestCode, resultCode, data)

  if (requestCode == RC_SIGN_IN && resultCode == Activity.RESULT_OK) {
    // Successfully signed in
    val user = FirebaseAuth.getInstance().currentUser
    if (user != null) {
      // initialize home fragment
      homeFragment = HomeFragment.newInstance()
      initHomeFragment()
    }
  }
}

fun hideKeyboard() {
  val imm = getSystemService(INPUT_METHOD_SERVICE) as InputMethodManager
  imm.hideSoftInputFromWindow(window.decorView.rootView.windowToken, 0);
}

@SuppressLint("InflateParams")
private fun initActionBar(actionBar: ActionBar) {
  // Disable the default and enable the custom
  actionBar.setDisplayShowTitleEnabled(false)
  actionBar.setDisplayShowCustomEnabled(true)
  val customView: View =
    layoutInflater.inflate(R.layout.action_bar, null)
  // Apply the custom view
  actionBar.customView = customView
}
```

```
        private fun initHomeFragment() {
          supportFragmentManager
            .beginTransaction()
            // No back stack for home
            .add(R.id.main_frame, homeFragment)
            // TRANSIT_FRAGMENT_FADE calls for the Fragment to fade away
            .setTransition(FragmentTransaction.TRANSIT_FRAGMENT_FADE)
            .commit()
        }
}
```