

```

1 package com.example.soundboardstarter
2
3 import android.media.AudioManager
4 import android.media.SoundPool
5 import androidx.appcompat.app.AppCompatActivity
6 import android.os.Bundle
7 import android.util.Log
8 import android.view.View
9 import android.widget.Button
10 import androidx.core.view.ViewCompat
11 import androidx.core.view.WindowInsetsCompat
12 import com.example.soundboardstarter.databinding.ActivityMainBinding
13 import com.google.gson.Gson
14 import com.google.gson.annotations.SerializedName
15 import com.google.gson.reflect.TypeToken
16 import kotlinx.coroutinesCoroutineScope
17 import kotlinx.coroutines.Dispatcher
18 import kotlinx.coroutines.delay
19 import kotlinx.coroutines.launch
20 import kotlinx.coroutines.withContext
21
22
23 class MainActivity : AppCompatActivity() {
24
25     companion object {
26         val TAG = "MainActivity"
27     }
28
29
30     lateinit var soundPool: SoundPool
31     var aNote = 0
32     var bbNote = 0
33     var bNote = 0
34     var cNote = 0
35     var cSNote = 0
36     var dNote = 0
37     var dSNote = 0
38     var eNote = 0
39     var fNote = 0
40     var fSNote = 0
41     var gNote = 0
42     var gSNote = 0
43
44     val songString = "A 500 A 0 E 500 A 0 E 250 D 250 C 250 G 500 C 0 E 500 C 0 E 250 D 250 C 250 A 500 A 0
E 500 E 250 D 250 C 250 C 0 G 500"
45     val noteMap = HashMap<String, Int>()
46
47     private lateinit var binding: ActivityMainBinding
48
49
50     override fun onCreate(savedInstanceState: Bundle?) {
51         super.onCreate(savedInstanceState)
52         binding = ActivityMainBinding.inflate(layoutInflater)
53         setContentView(binding.root)
54         var gson = Gson()
55         val inputStream = resources.openRawResource(R.raw.song)
56         val jsonString = inputStream.bufferedReader().use {
57             it.readText()
58         }
59         val type = object : TypeToken<List<Note>>(){}.type
60         val notes = gson.fromJson<List<Note>>(jsonString, type)
61         Log.d(TAG, "onCreate: $notes")
62
63
64
65
66
67         initializeSoundPool()
68         setListeners()
69
70     }

```

```

71     private fun stringConvert(song : String): ArrayList<Note> {
72         val songShorten = song.split(" ")
73         val listOfNotes = arrayListOf<Note>()
74         for(i in songShorten.indices step 2){
75             listOfNotes.add(Note(songShorten[i + 1].toLong(), songShorten[i]))
76         }
77     }
78     return listOfNotes
79 }
80
81     private suspend fun playSong(song: List<Note>){
82         for(i in song){
83             playNote(i.note)
84             delay(i.duration)
85         }
86     }
87
88     private suspend fun playSimpleSong() {
89         withContext(Dispatchers.Main) {
90             binding.buttonMainA.text = "hi"
91         }
92         playNote(aNote)
93         delay(500)
94         playNote(bNote)
95         playNote(aNote)
96         delay(500)
97         playNote(bNote)
98         playNote(aNote)
99         delay(500)
100        playNote(bNote)
101        playNote(aNote)
102        delay(500)
103        playNote(bNote)
104        playNote(aNote)
105        delay(500)
106        playNote(bNote)
107        playNote(aNote)
108        delay(500)
109        playNote(bNote)
110        withContext(Dispatchers.Main) {
111            binding.buttonMainA.text = "A"
112        }
113    }
114
115    private fun initializeSoundPool() {
116
117        this.volumeControlStream = AudioManager.STREAM_MUSIC
118        soundPool = SoundPool(10, AudioManager.STREAM_MUSIC, 0)
119 //        soundPool.setOnLoadCompleteListener(SoundPool.OnLoadCompleteListener { soundPool, sampleId,
120 //            status ->
121 //                // isSoundPoolLoaded = true
122 //            })
122        aNote = soundPool.load(this, R.raw.scalea, 1)
123        bbNote = soundPool.load(this, R.raw.scalebb, 1)
124        bNote = soundPool.load(this, R.raw.scaleb, 1)
125        cNote = soundPool.load(this, R.raw.scalec, 1)
126        cSNote = soundPool.load(this, R.raw.scalecs, 1)
127        dNote = soundPool.load(this, R.raw.scaled, 1)
128        dSNote = soundPool.load(this, R.raw.scaleds, 1)
129        eNote = soundPool.load(this, R.raw.scalee, 1)
130        fNote = soundPool.load(this, R.raw.scalef, 1)
131        fSNote = soundPool.load(this, R.raw.scalefs, 1)
132        gNote = soundPool.load(this, R.raw.scaleg, 1)
133        gSNote = soundPool.load(this, R.raw.scalegs, 1)
134
135        noteMap["A"] = aNote
136        noteMap["Bb"] = bbNote
137        noteMap["B"] = bNote
138        noteMap["C"] = cNote
139        noteMap["CS"] = cSNote
140        noteMap["D"] = dNote

```

```

141     noteMap["DS"] = dSNote
142     noteMap["E"] = eNote
143     noteMap["F"] = fNote
144     noteMap["FS"] = fSNote
145     noteMap["G"] = gNote
146     noteMap["GS"] = gSNote
147
148 }
149
150 private fun playNote(note : String){
151     playNote(noteMap[note] ?: 0)
152 }
153
154
155
156     private fun setListeners() {
157         val soundBoardListener = SoundBoardListener()
158         binding.buttonMainA.setOnClickListener(soundBoardListener)
159         binding.buttonMainB.setOnClickListener(soundBoardListener)
160         binding.buttonMainBb.setOnClickListener(soundBoardListener)
161         binding.buttonMainC.setOnClickListener(soundBoardListener)
162         binding.buttonMainCS.setOnClickListener(soundBoardListener)
163         binding.buttonMainD.setOnClickListener(soundBoardListener)
164         binding.buttonMainDS.setOnClickListener(soundBoardListener)
165         binding.buttonMainE.setOnClickListener(soundBoardListener)
166         binding.buttonMainF.setOnClickListener(soundBoardListener)
167         binding.buttonMainFS.setOnClickListener(soundBoardListener)
168         binding.buttonMainG.setOnClickListener(soundBoardListener)
169         binding.buttonMainGS.setOnClickListener(soundBoardListener)
170         binding.buttonMainPlaySong.setOnClickListener {
171             CoroutineScope(Dispatchers.IO).launch {
172                 playSong(stringConvert(songString))
173             }
174         }
175
176     }
177
178
179
180     private fun playNote(noteId: Int) {
181         soundPool.play(noteId, 1f, 1f, 1, 0, 1f)
182     }
183
184     private inner class SoundBoardListener : View.OnClickListener {
185         override fun onClick(v: View?) {
186             when (v?.id) {
187                 R.id.button_main_a -> playNote(aNote)
188                 R.id.button_main_bb -> playNote(bbNote)
189                 R.id.button_main_b -> playNote(bNote)
190                 R.id.button_main_c -> playNote(cNote)
191                 R.id.button_main_cS -> playNote(cSNote)
192                 R.id.button_main_d -> playNote(dNote)
193                 R.id.button_main_dS -> playNote(dSNote)
194                 R.id.button_main_e -> playNote(eNote)
195                 R.id.button_main_f -> playNote(fNote)
196                 R.id.button_main_fs -> playNote(fSNote)
197                 R.id.button_main_g -> playNote(gNote)
198                 R.id.button_main_gs -> playNote(gSNote)
199             }
200         }
201     }
202 }
203
204

```

```
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   tools:context=".MainActivity">
8
9   <Button
10    android:id="@+id/button_main_a"
11    android:layout_width="wrap_content"
12    android:layout_height="wrap_content"
13    android:text="@string/a_note"
14    app:layout_constraintBottom_toTopOf="@+id/button_main_bb"
15    app:layout_constraintEnd_toEndOf="parent"
16    app:layout_constraintHorizontal_bias="0.108"
17    app:layout_constraintStart_toStartOf="parent"
18    app:layout_constraintTop_toTopOf="parent" />
19
20   <Button
21    android:id="@+id/button_main_bb"
22    android:layout_width="wrap_content"
23    android:layout_height="wrap_content"
24    android:text="@string/b_flat_note"
25    app:layout_constraintBottom_toTopOf="@+id/button_main_b"
26    app:layout_constraintEnd_toEndOf="parent"
27    app:layout_constraintHorizontal_bias="0.108"
28    app:layout_constraintStart_toStartOf="parent"
29    app:layout_constraintTop_toBottomOf="@+id/button_main_a" />
30
31   <Button
32    android:id="@+id/button_main_b"
33    android:layout_width="wrap_content"
34    android:layout_height="wrap_content"
35    android:text="@string/b_note"
36    app:layout_constraintBottom_toTopOf="@+id/button_main_c"
37    app:layout_constraintEnd_toEndOf="parent"
38    app:layout_constraintHorizontal_bias="0.108"
39    app:layout_constraintStart_toStartOf="parent"
40    app:layout_constraintTop_toBottomOf="@+id/button_main_bb" />
41
42   <Button
43    android:id="@+id/button_main_c"
44    android:layout_width="wrap_content"
45    android:layout_height="wrap_content"
46    android:text="@string/c_note"
47    app:layout_constraintBottom_toBottomOf="parent"
48    app:layout_constraintEnd_toEndOf="parent"
49    app:layout_constraintHorizontal_bias="0.108"
50    app:layout_constraintStart_toStartOf="parent"
51    app:layout_constraintTop_toBottomOf="@+id/button_main_b" />
52
53   <Button
54    android:id="@+id/button_main_e"
55    android:layout_width="wrap_content"
56    android:layout_height="wrap_content"
57    android:text="@string/e_note"
58    app:layout_constraintBottom_toTopOf="@+id/button_main_dS"
59    app:layout_constraintEnd_toStartOf="@+id/button_main_f"
60    app:layout_constraintHorizontal_bias="0.5"
61    app:layout_constraintStart_toEndOf="@+id/button_main_a"
62    app:layout_constraintTop_toTopOf="parent"
63    app:layout_constraintVertical_bias="0.5" />
64
65   <Button
66    android:id="@+id/button_main_dS"
67    android:layout_width="wrap_content"
68    android:layout_height="wrap_content"
69    android:text="@string/dS_note"
70    app:layout_constraintBottom_toTopOf="@+id/button_main_d"
```

```
71     app:layout_constraintEnd_toStartOf="@+id/button_main_fs"
72     app:layout_constraintHorizontal_bias="0.5"
73     app:layout_constraintStart_toEndOf="@+id/button_main_bb"
74     app:layout_constraintTop_toBottomOf="@+id/button_main_e"
75     app:layout_constraintVertical_bias="0.5" />
76
77     <Button
78         android:id="@+id/button_main_d"
79         android:layout_width="wrap_content"
80         android:layout_height="wrap_content"
81         android:text="@string/d_note"
82         app:layout_constraintBottom_toTopOf="@+id/button_main_cS"
83         app:layout_constraintEnd_toStartOf="@+id/button_main_g"
84         app:layout_constraintHorizontal_bias="0.5"
85         app:layout_constraintStart_toEndOf="@+id/button_main_b"
86         app:layout_constraintTop_toBottomOf="@+id/button_main_dS"
87         app:layout_constraintVertical_bias="0.5" />
88
89     <Button
90         android:id="@+id/button_main_cS"
91         android:layout_width="wrap_content"
92         android:layout_height="wrap_content"
93         android:text="@string/cS_note"
94         app:layout_constraintBottom_toBottomOf="parent"
95         app:layout_constraintEnd_toStartOf="@+id/button_main_gS"
96         app:layout_constraintHorizontal_bias="0.5"
97         app:layout_constraintStart_toEndOf="@+id/button_main_c"
98         app:layout_constraintTop_toBottomOf="@+id/button_main_d"
99         app:layout_constraintVertical_bias="0.5" />
100
101    <Button
102        android:id="@+id/button_main_f"
103        android:layout_width="wrap_content"
104        android:layout_height="wrap_content"
105        android:text="@string/f_note"
106        app:layout_constraintBottom_toTopOf="@+id/button_main_fs"
107        app:layout_constraintEnd_toEndOf="parent"
108        app:layout_constraintHorizontal_bias="0.5"
109        app:layout_constraintStart_toEndOf="@+id/button_main_e"
110        app:layout_constraintTop_toTopOf="parent"
111        app:layout_constraintVertical_bias="0.5" />
112
113    <Button
114        android:id="@+id/button_main_g"
115        android:layout_width="wrap_content"
116        android:layout_height="wrap_content"
117        android:text="@string/g_note"
118        app:layout_constraintBottom_toTopOf="@+id/button_main_gS"
119        app:layout_constraintEnd_toEndOf="parent"
120        app:layout_constraintHorizontal_bias="0.5"
121        app:layout_constraintStart_toEndOf="@+id/button_main_d"
122        app:layout_constraintTop_toBottomOf="@+id/button_main_fs"
123        app:layout_constraintVertical_bias="0.5" />
124
125    <Button
126        android:id="@+id/button_main_gS"
127        android:layout_width="wrap_content"
128        android:layout_height="wrap_content"
129        android:text="@string/gS_note"
130        app:layout_constraintBottom_toBottomOf="parent"
131        app:layout_constraintEnd_toEndOf="parent"
132        app:layout_constraintHorizontal_bias="0.5"
133        app:layout_constraintStart_toEndOf="@+id/button_main_cS"
134        app:layout_constraintTop_toBottomOf="@+id/button_main_g"
135        app:layout_constraintVertical_bias="0.5" />
136
137    <Button
138        android:id="@+id/button_main_fs"
139        android:layout_width="wrap_content"
140        android:layout_height="wrap_content"
141        android:text="@string/fS_note"
```

```
142     app:layout_constraintBottom_toTopOf="@+id/button_main_g"
143     app:layout_constraintEnd_toEndOf="parent"
144     app:layout_constraintHorizontal_bias="0.5"
145     app:layout_constraintStart_toEndOf="@+id/button_main_dS"
146     app:layout_constraintTop_toBottomOf="@+id/button_main_f"
147     app:layout_constraintVertical_bias="0.5" />
148
149     <Button
150         android:id="@+id/button_main_play_song"
151         android:layout_width="wrap_content"
152         android:layout_height="wrap_content"
153         android:text="@string/play_song"
154         app:layout_constraintBottom_toBottomOf="parent"
155         app:layout_constraintEnd_toEndOf="parent"
156         app:layout_constraintHorizontal_bias="0.5"
157         app:layout_constraintStart_toStartOf="parent"
158         app:layout_constraintTop_toBottomOf="@+id/button_main_cS"
159         app:layout_constraintVertical_bias="0.5" />
160 </androidx.constraintlayout.widget.ConstraintLayout>
```