

Практическая 14 Кирюшкин Артём ИС221

```
activity_main.xml baseline_add_24.xml ItemAdapter.kt item_rec.xml build.gradle.kts (:app)
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent"
5     android:id="@+id/main">
6     <androidx.recyclerview.widget.RecyclerView
7         android:id="@+id/recyclerView"
8         android:layout_width="match_parent"
9         android:layout_height="match_parent"/>
10    <com.google.android.material.floatingactionbutton.FloatingActionButton
11        android:id="@+id/fab"
12        android:layout_width="wrap_content"
13        android:layout_height="wrap_content"
14        android:layout_alignParentEnd="true"
15        android:layout_alignParentBottom="true"
16        android:layout_margin="16dp"
17        android:src="@drawable/baseline_add_24" />
18 </RelativeLayout>
```

```
activity_main.xml baseline_add_24.xml ItemAdapter.kt item_rec.xml MainActivity.kt
1 package com.example.practice14
2
3 import android.media.RouteListingPreference
4 import android.view.LayoutInflater
5 import android.view.View
6 import android.view.ViewGroup
7 import android.widget.TextView
8 import androidx.recyclerview.widget.RecyclerView
9
10 class ItemAdapter (private val item: List<RouteListingPreference.Item>) :
11     RecyclerView.Adapter<ItemAdapter.ItemViewHolder>(){
12     class ItemViewHolder(view: View) : RecyclerView.ViewHolder(view) {
13         val textView : TextView = view.findViewById(R.id.txt1)
14     }
15     override fun onCreateViewHolder(parent: ViewGroup, viewType: Int):
16         ItemViewHolder {
17         val view =
18             LayoutInflater.from(parent.context).inflate(R.layout.item_rec, parent, attachToRoot: false)
19         return ItemViewHolder(view)
20     }
21     override fun onBindViewHolder(holder: ItemViewHolder, position: Int) {
22         holder.textView.text = item[position].routeId
23     }
24     override fun getItemCount(): Int {
25         return item.size
26     }
27 }
```

```
activity_main.xml baseline_add_24.xml ItemAdapter.kt item_rec.xml MainActivity.kt
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent">
5 </androidx.constraintlayout.widget.ConstraintLayout>
```

```
activity_main.xml baseline_add_24.xml ItemAdapter.kt item_rec.xml MainActivity.kt x
1 package com.example.practice14
2
3 > import ...
14
15 class MainActivity : AppCompatActivity() {
16
17     private lateinit var recyclerView: RecyclerView
18     private lateinit var fab: FloatingActionButton
19     private val items: MutableList<RouteListingPreference.Item> = mutableListOf()
20     private lateinit var adapter: ItemAdapter
21     override fun onCreate(savedInstanceState: Bundle?) {
22         super.onCreate(savedInstanceState)
23         enableEdgeToEdge()
24         setContentView(R.layout.activity_main)
25         ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) {
26             v, insets ->
27             val systemBars =
28                 insets.getInsets(WindowInsetsCompat.Type.systemBars())
29             v.setPadding(systemBars.left, systemBars.top, systemBars.right,
30                 systemBars.bottom)
31             insets
32         }
33         recyclerView = findViewById(R.id.recyclerView)
34         fab = findViewById(R.id.fab)
35         adapter = ItemAdapter(items)
36         recyclerView.layoutManager = LinearLayoutManager(context: this)
37         recyclerView.adapter = adapter
38         adapter = ItemAdapter(items)
39         recyclerView.layoutManager = LinearLayoutManager(context: this)
40         recyclerView.adapter = adapter
41         fab.setOnClickListener {
42             showInputDialog()
43         }
44     }
45     private fun showInputDialog() {
46         val builder = AlertDialog.Builder(context: this)
47         builder.setTitle("Добавить элемент")
48         val input = EditText(context: this)
49         builder.setView(input)
50         builder.setPositiveButton(text: "Добавить") { dialog, _ ->
51             val text = input.text.toString()
52             if (text.isNotEmpty()) {
53                 items.add(error(text))
54                 adapter.notifyItemInserted(position: items.size - 1)
55             }
56             dialog.dismiss()
57         }
58         builder.setNegativeButton(text: "Отмена") { dialog, _ -> dialog.cancel() }
59         builder.show()
60     }
61 }
```

