

Добавил код в activity-main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/main">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/fab"
        android:layout_width="121dp"
        android:layout_height="96dp"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginStart="16dp"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="0dp"
        android:layout_marginRight="16dp"
        android:layout_marginBottom="0dp"
        android:src="@drawable/baseline_add_24"/>
</RelativeLayout>
```

В MainActivity.kt

```
package com.example.a14

import android.os.Bundle
import android.widget.EditText
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.appcompat.app.AlertDialog
import androidx.recyclerview.widget.LinearLayoutManager
import androidx.recyclerview.widget.RecyclerView
import com.google.android.material.floatingactionbutton.FloatingActionButton
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat

class MainActivity : AppCompatActivity() {

    // Входные компоненты для RecyclerView и FloatingActionButton
    private lateinit var recyclerView: RecyclerView
    private lateinit var fab: FloatingActionButton
    // Список элементов, которые будут отображаться в RecyclerView
    private val items: MutableList<Item> = mutableListOf()
    // Адаптер для управления отображением элементов в RecyclerView
    private lateinit var adapter: ItemAdapter

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge() // Для поддержки экрана без рамок

        setContentView(R.layout.activity_main)

        // Устанавливаем обработчик WindowInsets для учета системных панелей
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) {
            v, insets ->
```

```

        val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())
        v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)
    }

    // Инициализация компонентов
    recyclerView = findViewById(R.id.recyclerView)
    fab = findViewById(R.id.fab)

    // Настройка адаптера и RecyclerView
    adapter = ItemAdapter(items)
    recyclerView.layoutManager = LinearLayoutManager(this) // Устанавливаем LinearLayoutManager
    recyclerView.adapter = adapter // Устанавливаем адаптер

    // Устанавливаем обработчик нажатия на FloatingActionButton
    fab.setOnClickListener {
        showInputDialog() // Вызываем метод для отображения диалогового
        окна
    }
}

// Метод для отображения диалогового окна для ввода текста
private fun showInputDialog() {
    val builder = AlertDialog.Builder(this) // Создаем экземпляр
    AlertDialog
    builder.setTitle("Добавить элемент") // Устанавливаем заголовок
    диалогового окна
    val input = EditText(this) // Создаем EditText для ввода текста
    builder.setView(input) // Устанавливаем созданный EditText в диалог

    // Устанавливаем кнопку "Добавить" в диалоговом окне
    builder.setPositiveButton("Добавить") { dialog, _ ->
        val text = input.text.toString() // Получаем текст из EditText
        if (text.isNotEmpty()) { // Проверяем, не пустой ли он
            items.add(Item(text)) // Добавляем новый элемент в список
            adapter.notifyItemInserted(items.size - 1) // Уведомляем
            адаптер о добавлении нового элемента
        }
        dialog.dismiss() // Закрываем диалог
    }

    // Устанавливаем кнопку "Отмена" в диалоговом окне
    builder.setNegativeButton("Отмена") { dialog, _ -> dialog.cancel() }

    builder.show() // Показываем диалоговое окно
}
}

```

B Item.kt

```

package com.example.a14

data class Item (val text : String)

```

B item_rec.xml

```

<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="wrap_content">
<TextView
    android:id="@+id/txt1"
    android:layout_width="0dp"
    android:layout_height="64dp"
    android:layout_marginTop="16dp"
    android:text="test"
    android:textSize="36sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

ItemAdapter.kt

```

package com.example.a14

import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.TextView
import androidx.recyclerview.widget.RecyclerView

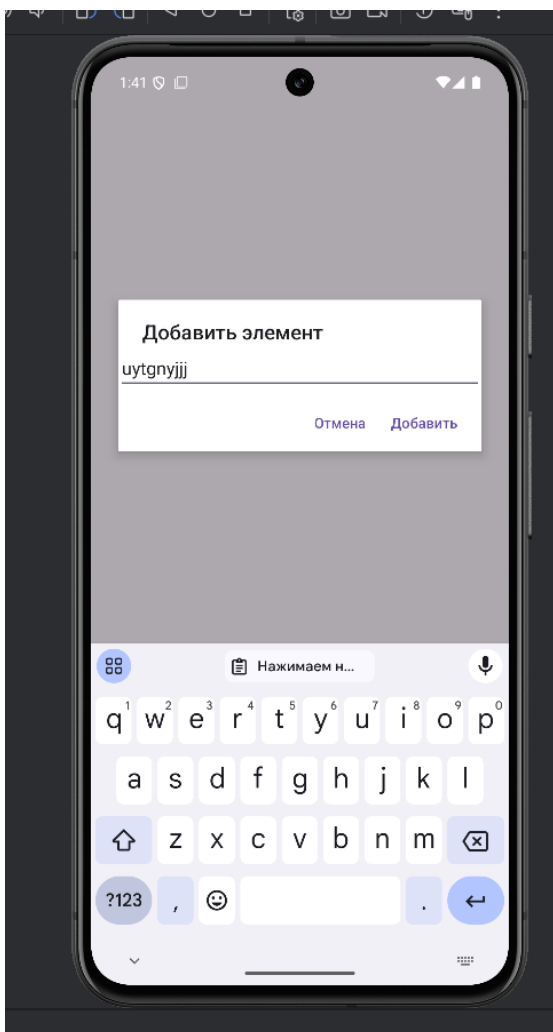
class ItemAdapter(private val item: List<Item>) :
    RecyclerView.Adapter<ItemAdapter.ItemViewHolder>() {
    class ItemViewHolder(view: View) : RecyclerView.ViewHolder(view) {
        val textView : TextView = view.findViewById(R.id.txt1)
    }
    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int):
        ItemViewHolder {
        val view =
            LayoutInflater.from(parent.context).inflate(R.layout.item_rec,
parent, false)
        return ItemViewHolder(view)
    }
    override fun onBindViewHolder(holder: ItemViewHolder, position: Int) {
        holder.textView.text = item[position].text
    }
    override fun getItemCount(): Int {
        return item.size
    }
}

```

Работа приложения:



Нажимаем на кнопку снизу, вводим нужный элемент



Нажимаем на добавить и элемент появляется в списке

