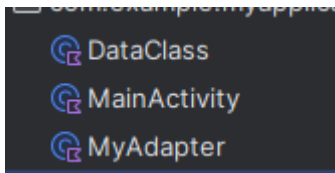


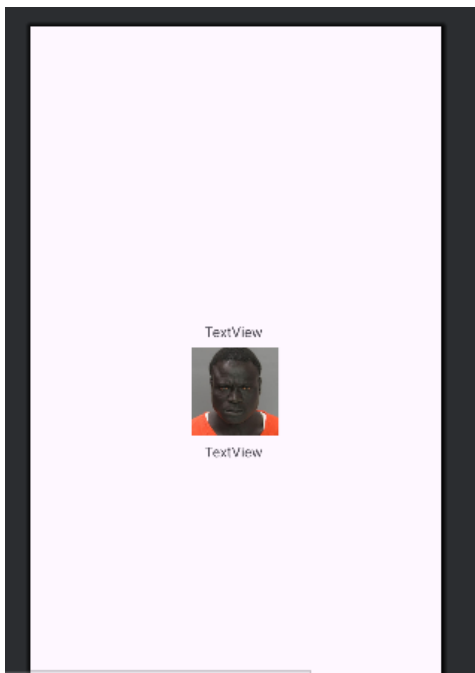
Практическая работа номер 8

1. Создаём новый проект

2. Добавляем классы MyAdapter и DataClass



3. Добавляем макет с нашим оформлением `list_item.xml`



4. Пишем код адаптера, для конвертации массивов в recycler

```
class MyAdapter(private val newList : ArrayList<DataClass>) :  
    override fun onCreateViewHolder(parent : ViewGroup, viewType: Int) :  
        MyViewHolder{  
  
        val itemView = LayoutInflater.from(parent.context).inflate(R.layout.list_item,parent, attachToRoot: false)  
        return MyViewHolder(itemView)  
    }  
  
    override fun getItemCount() : Int {  
        return newList.size  
    }  
  
    override fun onBindViewHolder(holder: MyViewHolder, position : Int){  
  
        val currentItem = newList[position]  
  
        holder.imageView.setImageResource(currentItem.titleImage)  
        holder.txtHead.text = currentItem.heading  
        holder.txtEnd.text = currentItem.negr  
  
    }  
}  
  
class MyViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView){  
  
    val imageView : ShapeableImageView = itemView.findViewById(R.id.imageItem)  
    val txtHead : TextView = itemView.findViewById(R.id.textView2)  
    val txtEnd : TextView = itemView.findViewById(R.id.textView3)  
  
}
```

5. Пишем код dataclass, для переноса данных активностей

```
data class DataClass(var titleImage : Int, var heading : String, var negr : String)
```

6. Добавляем переменные string .xml

```
<string-array name="podporka">
    <item>Всеволод</item>
    <item>Евламний</item>
    <item>Игорь</item>
    <item>Лёша</item>
    <item>Джамал</item>
</string-array>
<string-array name="op">
    <item>Великий</item>
    <item>Круглый</item>
    <item>Тэн</item>
    <item>Карлик</item>
    <item>Негр</item>
</string-array>
```

7. Пишем код для вывода в mainActivity

```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    enableEdgeToEdge()

    setContentView(R.layout.activity_main)

    imageId = arrayOf(
        R.drawable.obr,
        R.drawable.ed,
        R.drawable.asian,
        R.drawable.karlik,
        R.drawable.d,
    )

    textId = resources.getStringArray(R.array.podporka)
    textEndId = resources.getStringArray(R.array.op)

    newRecyclerView = findViewById(R.id.recycle)
    newRecyclerView.layoutManager = LinearLayoutManager(context, this)
    newRecyclerView.setHasFixedSize(true)

    newArrayList = arrayListOf<DataClass>()
```

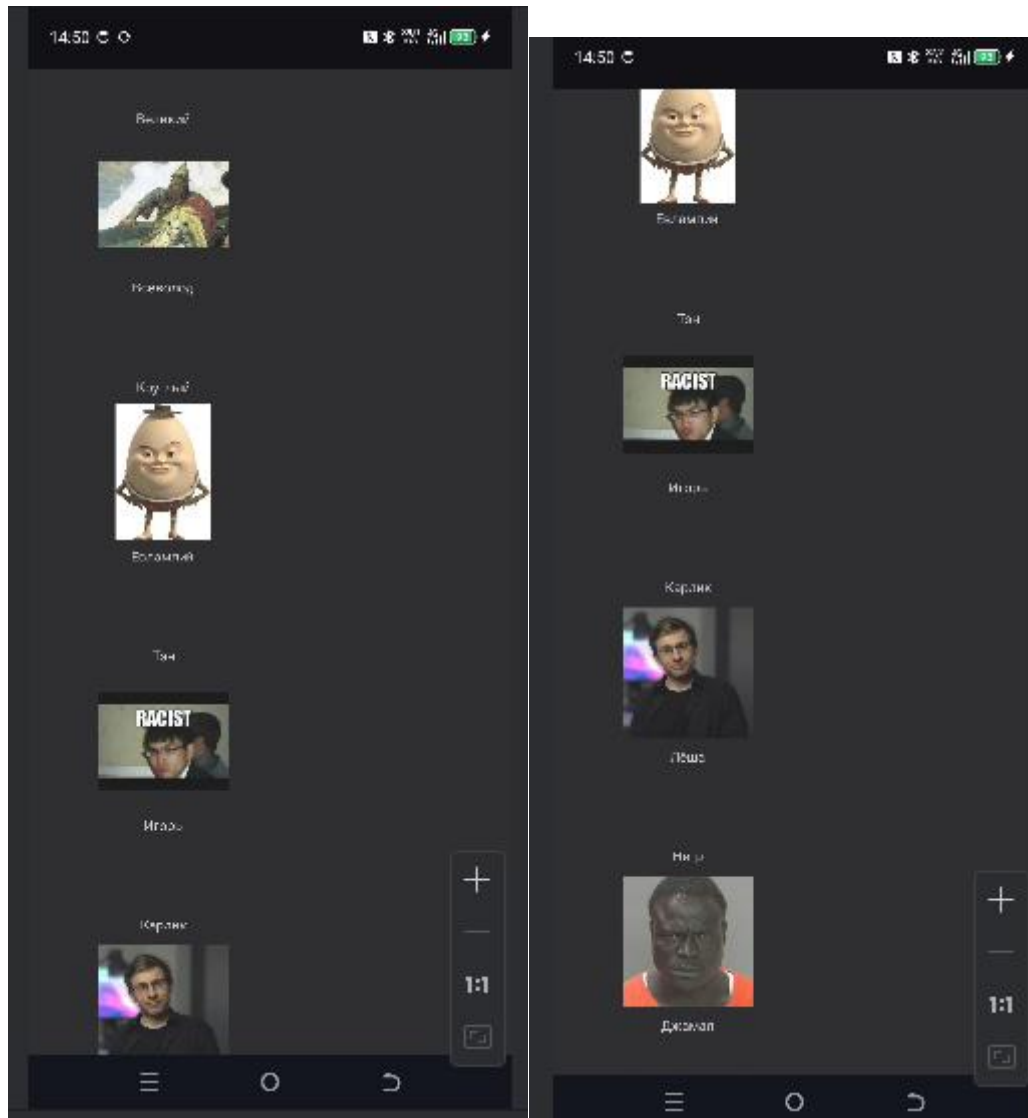
```
newArrayList = arrayListOf<DataClass>()

getUserdata()

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)
    insets
}

private fun getUserdata(){
    for(i in imageId.indices){
        val data = DataClass(imageId[i], textId[i], textEndId[i])
        newArrayList.add(data)
    }
    newRecyclerView.adapter = MyAdapter(newArrayList)
}
```

8.Результат



9.Добавляем тост оповещение при нажатии на элемент списка

```
holder.itemView.setOnClickListener{  
    if(position == 0){  
        Toast.makeText(holder.itemView.context, text: "Нажали на Элемент!", Toast.LENGTH_SHORT).show()  
    }  
}
```

