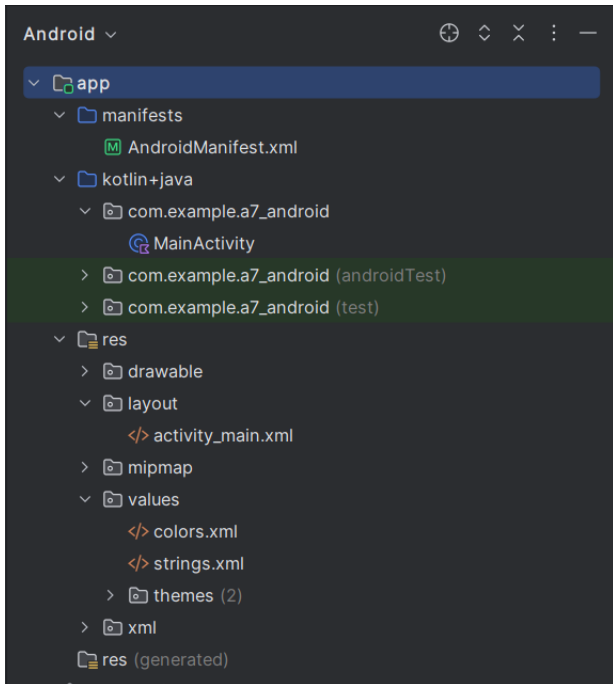


Практика 7:

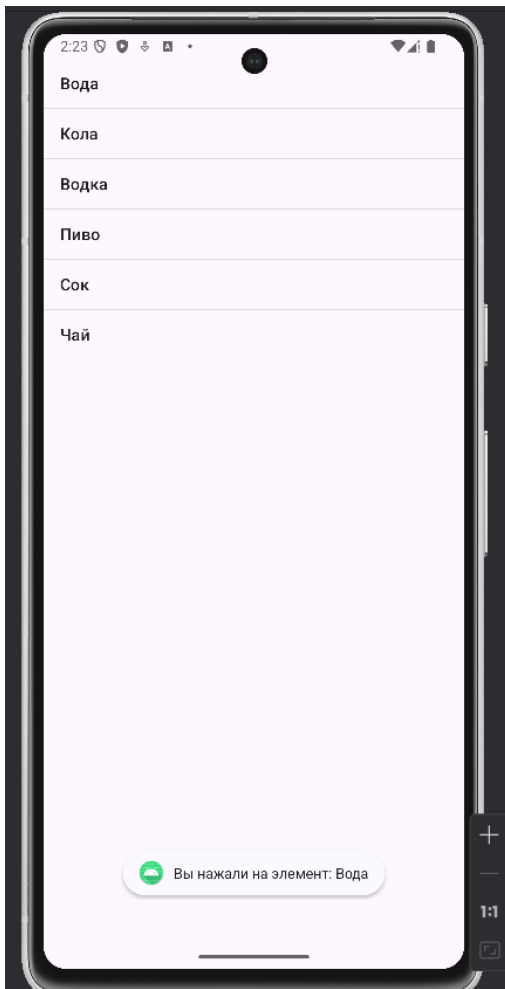
Структура:



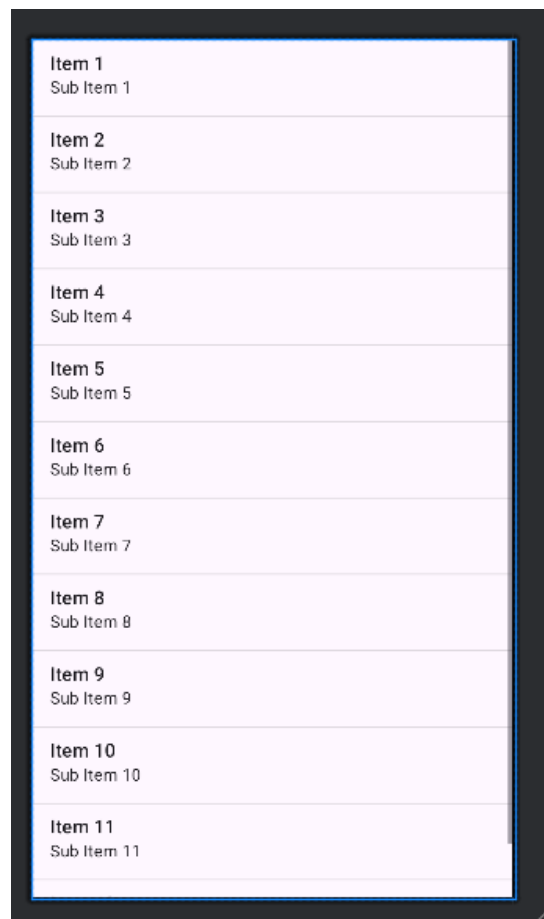
Логика:

```
class MainActivity : AppCompatActivity() {  
    @SuppressWarnings("MissingInflatedId")  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_main)  
        val listView: ListView = findViewById(R.id.lvMain)  
        val names = arrayOf("Вода", "Кола", "Водка", "Пиво", "Сок", "Чай")  
        val arrayAdapter: ArrayAdapter<String> =  
            ArrayAdapter<context> this, android.R.layout.simple_list_item_1, names)  
        listView.adapter = arrayAdapter  
        listView.setOnItemClickListener { parent, view, position, id ->  
            val selectItem = names[position]  
            Toast.makeText<context> this, <text> "Вы нажали на элемент: $selectItem", Toast.LENGTH_SHORT).show()  
        }  
        enableEdgeToEdge()  
        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 *setOnApplyWindowInsetsListener  
        }  
    }  
}
```

Итог:

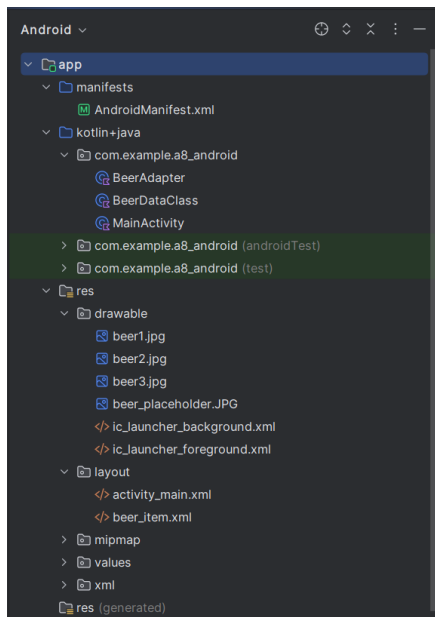


Дизайн:



Практика 8:

Структура:



MainActivity:

```
class MainActivity : AppCompatActivity() {
    private lateinit var recyclerView: RecyclerView
    private lateinit var beerArrayList: ArrayList<BeerDataClass>
    private lateinit var imageIds: Array<Int>
    private lateinit var titles: Array<String>
    private lateinit var descriptions: Array<String>
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        recyclerView = findViewById(R.id.newRecycleView)
        recyclerView.layoutManager = LinearLayoutManager(context: this)
        recyclerView.setHasFixedSize(true)
        imageIds = arrayOf(R.drawable.beer1, R.drawable.beer2, R.drawable.beer3)
        titles = resources.getStringArray(R.array.beer_titles)
        descriptions = resources.getStringArray(R.array.beer_descriptions)
        beerArrayList = arrayListOf()
        getBeerData()
        recyclerView.adapter = BeerAdapter(beerArrayList)
    }
    private fun getBeerData() {
        for (i in imageIds.indices) {
            val beer = BeerDataClass(imageIds[i], titles[i], descriptions[i])
            beerArrayList.add(beer)
        }
    }
}
```

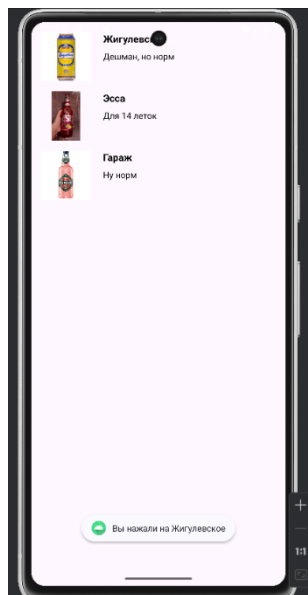
BeerAdapter:

```
class BeerAdapter(private val beerList: ArrayList<BeerDataClass>) :
    RecyclerView.Adapter<BeerAdapter.BeerViewHolder>() {
    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): BeerViewHolder {
        val itemView = LayoutInflater.from(parent.context).inflate(R.layout.beer_item, parent, attachToRoot: false)
        return BeerViewHolder(itemView)
    }
    override fun getItemCount(): Int {
        return beerList.size
    }
    override fun onBindViewHolder(holder: BeerViewHolder, position: Int) {
        val currentItem = beerList[position]
        holder.imageView.setImageResource(currentItem.imageResource)
        holder.txtTitle.text = currentItem.title
        holder.txtDescription.text = currentItem.description
        holder.itemView.setOnClickListener {
            Toast.makeText(holder.itemView.context, text: "Вы нажали на ${currentItem.title}", Toast.LENGTH_SHORT).show()
        }
    }
    class BeerViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
        val imageView: ShapeableImageView = itemView.findViewById(R.id.imageBeer)
        val txtTitle: TextView = itemView.findViewById(R.id.txtTitle)
        val txtDescription: TextView = itemView.findViewById(R.id.txtDescription)
    }
}
```

String:

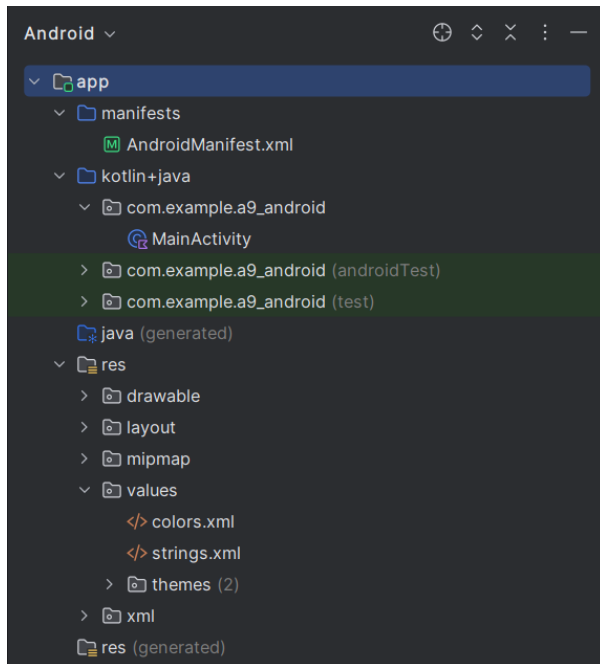
```
<resources>
<string name="app_name">8_android</string>
<string-array name="beer_titles">
    <item>Жигулевское</item>
    <item>Зсса</item>
    <item>Гараж</item>
</string-array>
<string-array name="beer_descriptions">
    <item>Дешман, но норм</item>
    <item>Для 14 леток</item>
    <item>Ну норм</item>
</string-array>
</resources>
```

Итог:

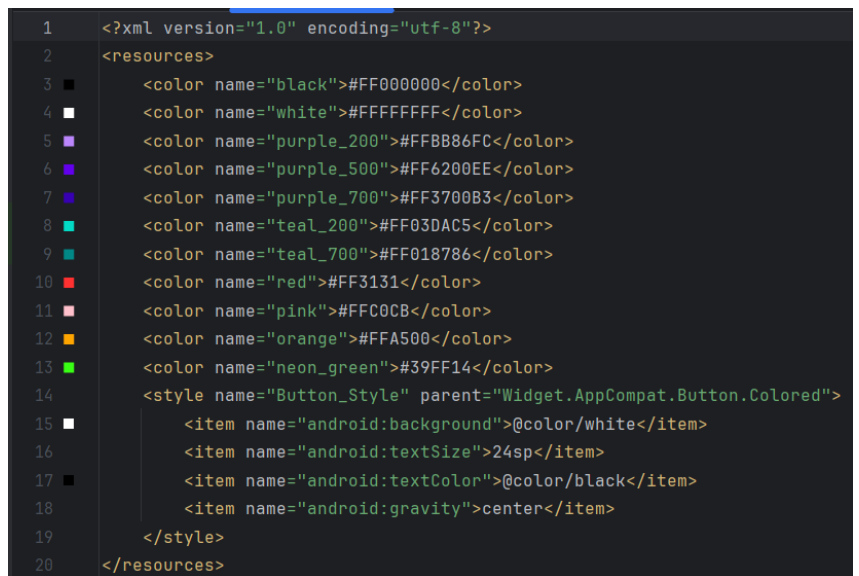


Практика 9:

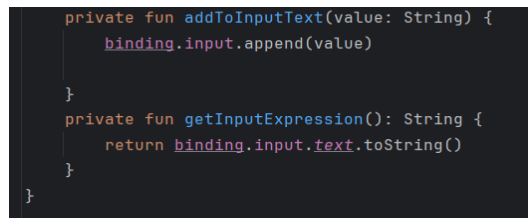
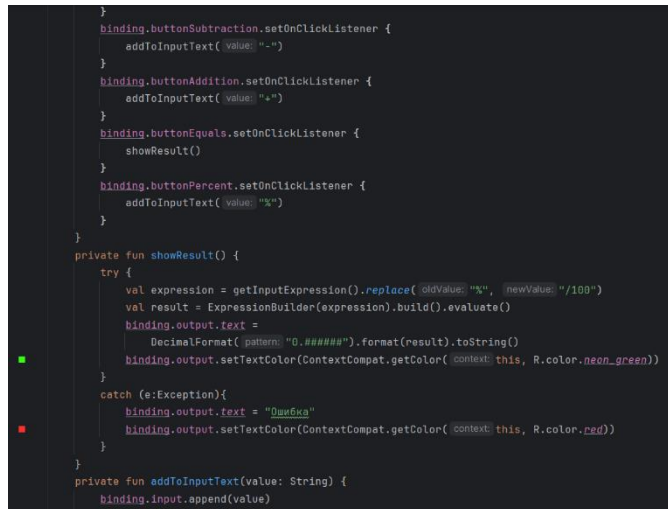
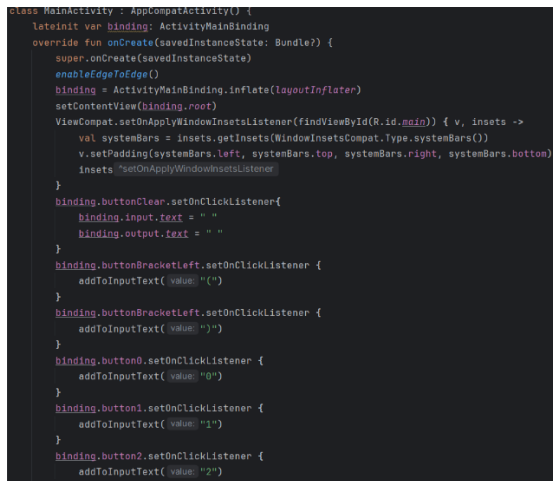
Структура:



Colors:



Логика:

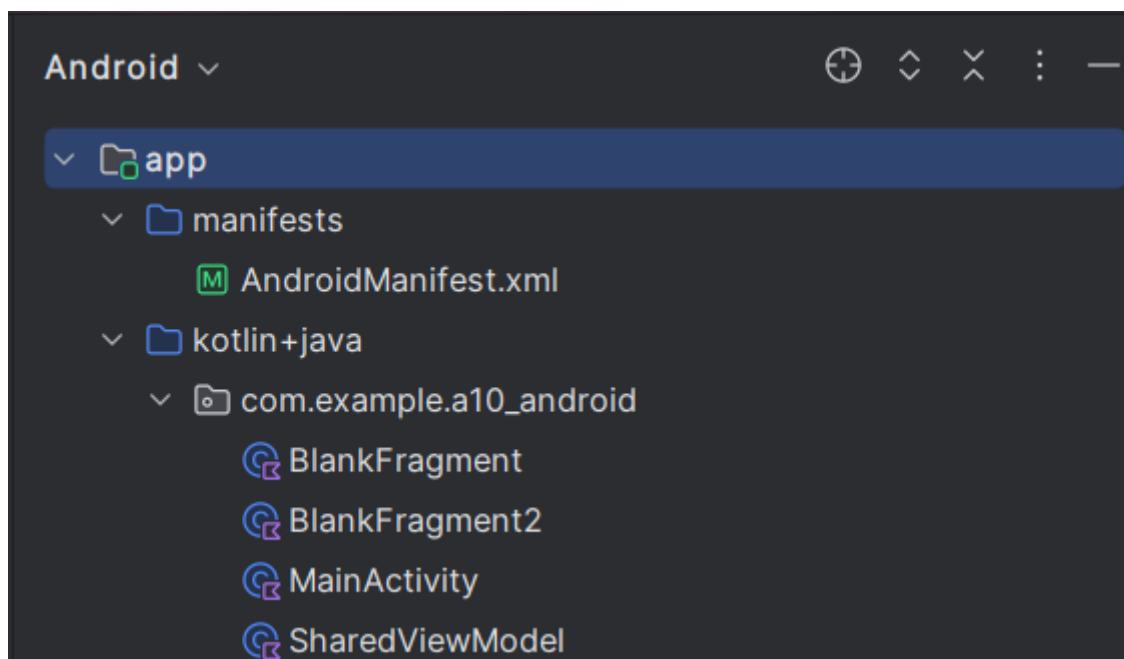


Итог:



Практика 10:

Структура:



Результат:

