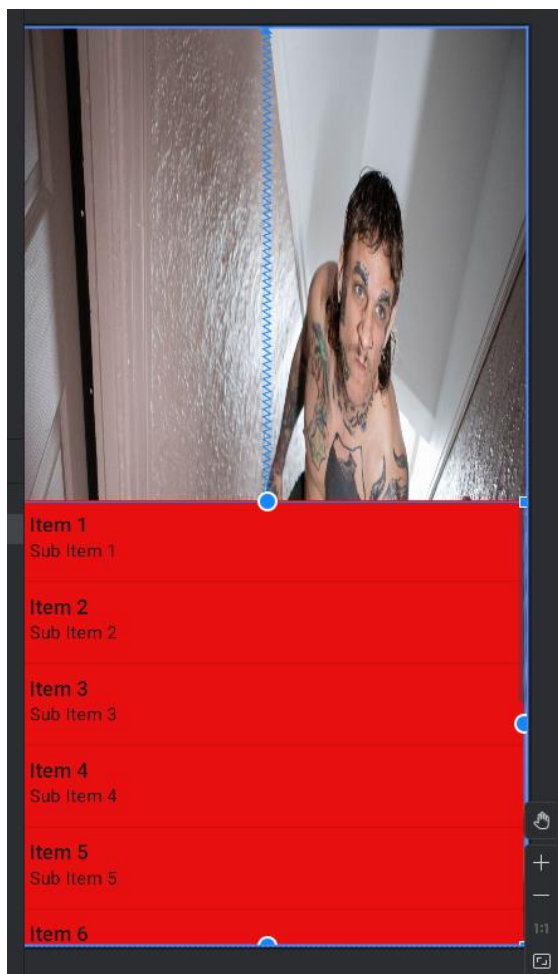


## Практическая работа номер 7

Создаем emptyactivity и оформляем дизайн и добавляем listview



Вставляем первый блок кода

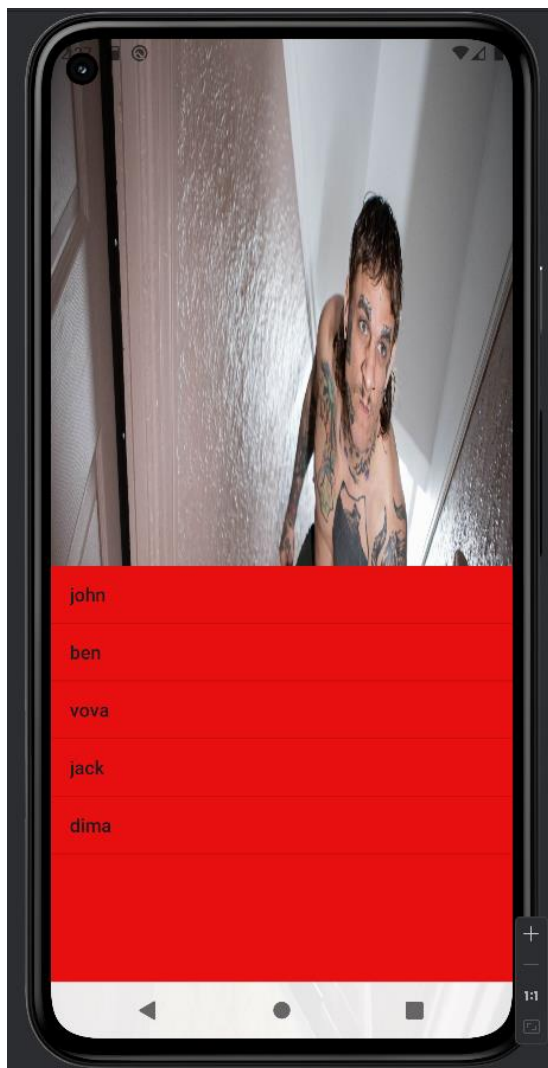
```
val list:ListView = findViewById(R.id.listv)

val arrayAdapter : ArrayAdapter<*>

val name = arrayOf("john","ben","vova","jack","dima")

arrayAdapter = ArrayAdapter(context: this, android.R.layout.simple_list_item_1,name)
list.adapter = arrayAdapter
```

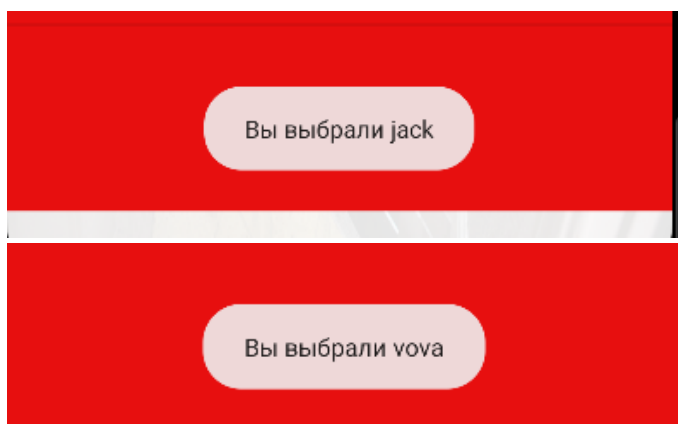
Проверяем работу программы



Работает!!!

Добавляем оповещение при нажатии

```
list.setOnItemClickListener{parent,view,position,id ->
    val Select = name[position]
    Toast.makeText(context: this, text: "Вы выбрали ${name[position]}", Toast.LENGTH_SHORT).show()
}
```



Задание:

Новый дизайн:



Добавляем дата класс и класс для адаптера массива

```
data class ListItem(val imageR : Int,val text:String,val action:String){  
      
}  
  
class AdapterList(context : Context,private val items: List<ListItem>) : ArrayAdapter<ListItem>(context, resource: 0,items){  
    override fun getView(position : Int, convertView: View?, parent: ViewGroup): View{  
        val item = getItem(position)  
  
        val view = convertView ?: LayoutInflater.from(context).inflate(R.layout.list_view,parent, attachToRoot: false)  
  
        val imageView:  ImageView = view.findViewById(R.id.imageView)  
        val textView : TextView = view.findViewById(R.id.tvList)  
  
        item?.let {  
            imageView.setImageResource(it.imageR)  
            textView.text =it.text  
        }  
        return view  
    }  
}
```

Код activity main:

```

class MainActivity : AppCompatActivity() {

    private lateinit var mediaPlayer: MediaPlayer
    private lateinit var mediaPlayer2: MediaPlayer
    private lateinit var mediaPlayer3: MediaPlayer
    private lateinit var mediaPlayer4: MediaPlayer
    private lateinit var mediaPlayer5: MediaPlayer

    override fun onCreate(savedInstanceState: Bundle?) {

        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)

        val listView : ListView = findViewById(R.id.listView)

        val items = listOf(
            ListItem(R.drawable.pudge, text: "Dota", action: "Action1"),
            ListItem(R.drawable.sad, text: "Sad", action: "Action2"),
            ListItem(R.drawable.mumble, text: "Fun", action: "Action3"),
            ListItem(R.drawable.kanve, text: "FbALLS", action: "Action4"),
            ListItem(R.drawable.praize, text: "Praize", action: "Action5")
        )

        val adapter = AdapterList( context: this,items)
        listView.adapter = adapter

        listView.setOnItemClickListener{parent,view,position,id ->
            val selectedItem = items[position]

            when(selectedItem.action){

                "Action1" -> {
                    Toast.makeText( context: this, text: "Dota music",Toast.LENGTH_SHORT).show()

                    if(!this::mediaPlayer.isInitialized) {

```

Я добавил медиафайлы для включения треков

```

        val adapter = AdapterList( context: this,items)
        listView.adapter = adapter

        listView.setOnItemClickListener{parent,view,position,id ->
            val selectedItem = items[position]

            when(selectedItem.action){

                "Action1" -> {
                    Toast.makeText( context: this, text: "Dota music",Toast.LENGTH_SHORT).show()

                    if(!this::mediaPlayer.isInitialized) {
                        mediaPlayer = MediaPlayer.create( context: this,R.raw.dota)
                    }
                    if(mediaPlayer.isPlaying){
                        mediaPlayer.pause()
                        mediaPlayer.seekTo( msec: 0)
                        return@setOnItemClickListener
                    }
                    mediaPlayer.start()
                }
                "Action2" -> {
                    Toast.makeText( context: this, text: "sad music",Toast.LENGTH_SHORT).show()

                    if(!this::mediaPlayer2.isInitialized) {
                        mediaPlayer2 = MediaPlayer.create( context: this,R.raw.sad)
                    }
                    if(mediaPlayer2.isPlaying){
                        mediaPlayer2.pause()
                        mediaPlayer2.seekTo( msec: 0)
                        return@setOnItemClickListener
                    }
                    mediaPlayer2.start()
                }
            }
        }
    }
}

```

```

}

"Action3" -> {
    Toast.makeText(context: this, text: "Fun", Toast.LENGTH_SHORT).show()

    if(!this::mediaPlayer3.isInitialized) {
        mediaPlayer3 = MediaPlayer.create(context: this, R.raw.yonkers)
    }
    if(mediaPlayer3.isPlaying){
        mediaPlayer3.pause()
        mediaPlayer3.seekTo(msec: 0)
        return@setOnItemClickListener
    }
    mediaPlayer3.start()
}

"Action4" -> {
    Toast.makeText(context: this, text: "FbALLS", Toast.LENGTH_SHORT).show()

    if(!this::mediaPlayer4.isInitialized) {
        mediaPlayer4 = MediaPlayer.create(context: this, R.raw.fballs)
    }
    if(mediaPlayer4.isPlaying){
        mediaPlayer4.pause()
        mediaPlayer4.seekTo(msec: 0)
        return@setOnItemClickListener
    }
    mediaPlayer4.start()
}

"Action5" -> {
    Toast.makeText(context: this, text: "Praise", Toast.LENGTH_SHORT).show()

    if(!this::mediaPlayer5.isInitialized) {
        mediaPlayer5 = MediaPlayer.create(context: this, R.raw.praise)
    }
    if(mediaPlayer5.isPlaying){
        mediaPlayer5.pause()
        mediaPlayer5.seekTo(msec: 0)
        return@setOnItemClickListener
    }
}

```

```

override fun onDestroy() {
    mediaPlayer.stop()
    mediaPlayer.release()
    super.onDestroy()
}

data class ListItem(val imageR : Int, val text:String, val action:String){

}

class AdapterList(context : Context, private val items: List<ListItem>) : ArrayAdapter<ListItem>(context, R.layout.list_item) {
    override fun getView(position : Int, convertView: View?, parent: ViewGroup): View {
        val item = getItem(position)

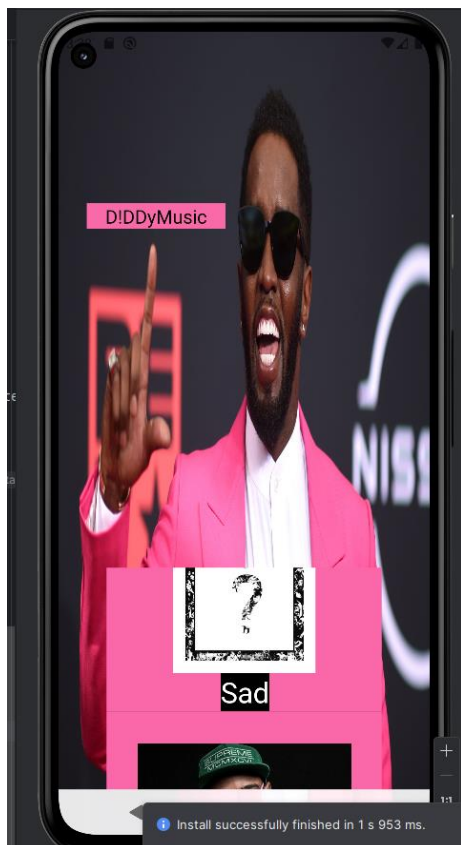
        val view = convertView ?: LayoutInflater.from(context).inflate(R.layout.list_view, parent, false)

        val imageView: ImageView = view.findViewById(R.id.imageView)
        val textView: TextView = view.findViewById(R.id.tvList)

        item?.let {
            imageView.setImageResource(it.imageR)
            textView.text = it.text
        }
        return view
    }
}

```

Результат:



Работает!!