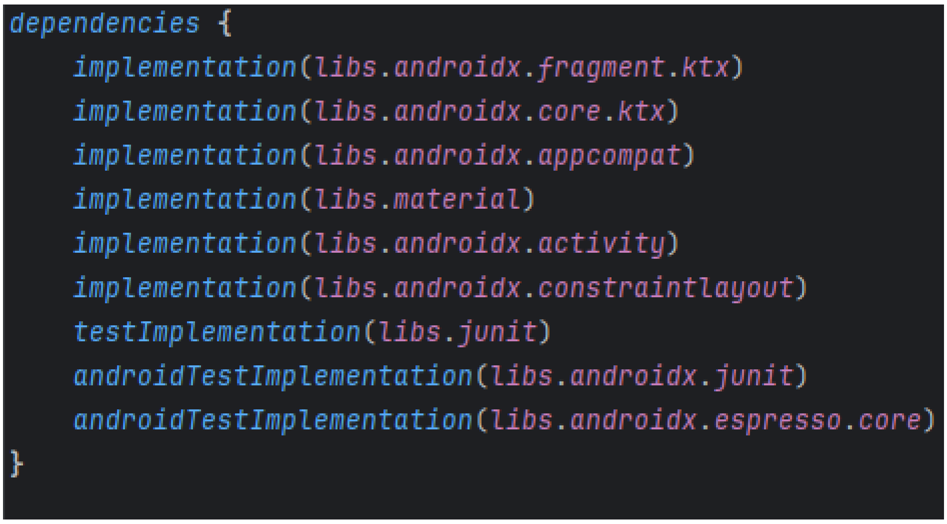
# Практическая работа номер 10

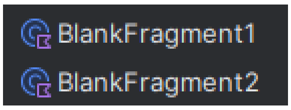
1.Создаем проект

2.Добавляем библиотеки



```
dependencies {
    implementation(libs.androidx.fragment.ktx)
    implementation(libs.androidx.core.ktx)
    implementation(libs.androidx.appcompat)
    implementation(libs.material)
    implementation(libs.androidx.activity)
    implementation(libs.androidx.constraintlayout)
    testImplementation(libs.junit)
    androidTestImplementation(libs.androidx.junit)
    androidTestImplementation(libs.androidx.espresso.core)
}
```

3.Создаем 2 BlankFragment

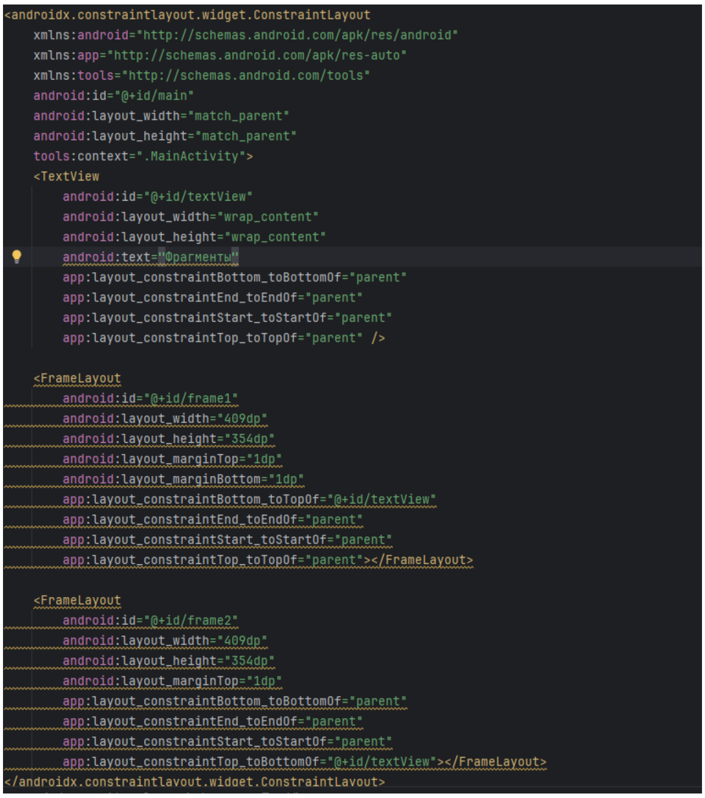


BlankFragment1

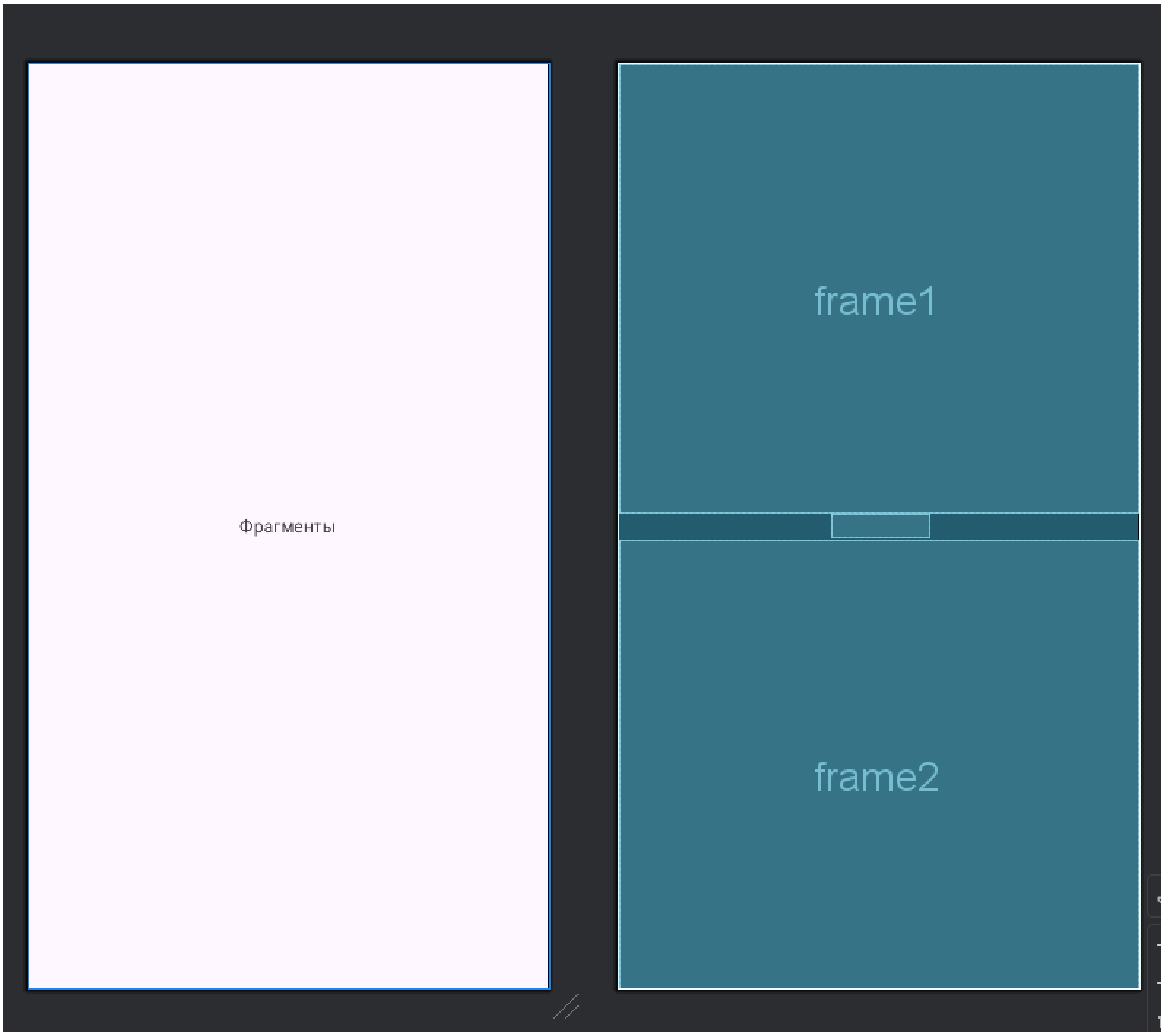
BlankFragment2

4.Оформляем mainactivity.xml и фрагменты

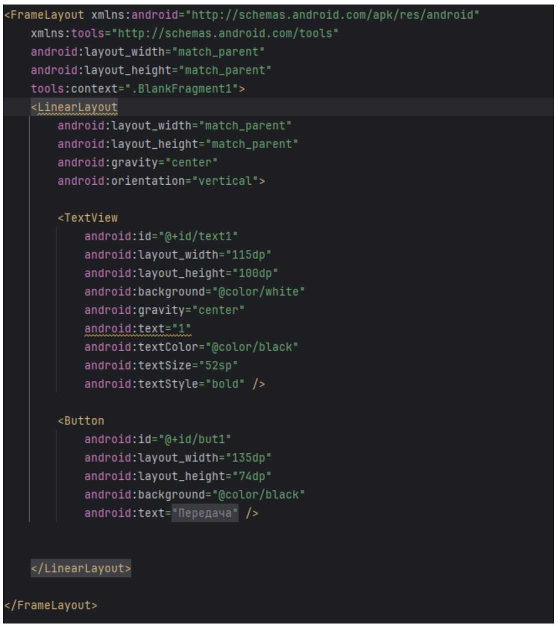
activity_main.xml



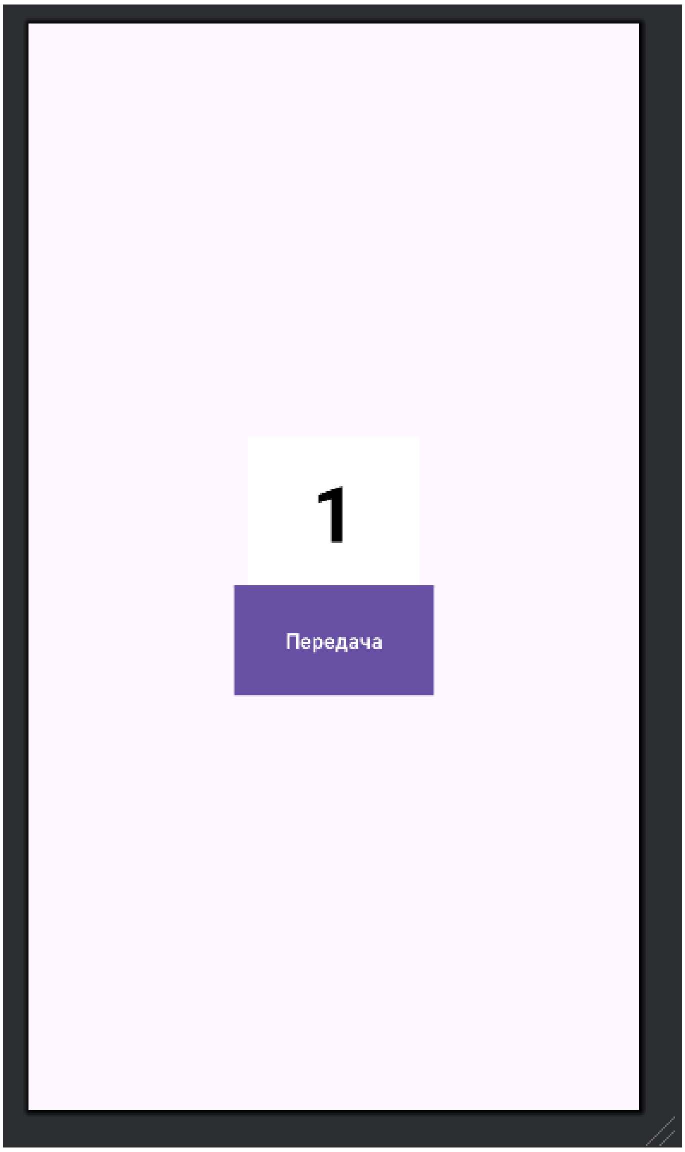
```
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Фрагменты"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <FrameLayout
        android:id="@+id/frame1"
        android:layout_width="409dp"
        android:layout_height="354dp"
        android:layout_marginTop="1dp"
        android:layout_marginBottom="1dp"
        app:layout_constraintBottom_toTopOf="@+id/textView"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"></FrameLayout>

    <FrameLayout
        android:id="@+id/frame2"
        android:layout_width="409dp"
        android:layout_height="354dp"
        android:layout_marginTop="1dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView"></FrameLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```



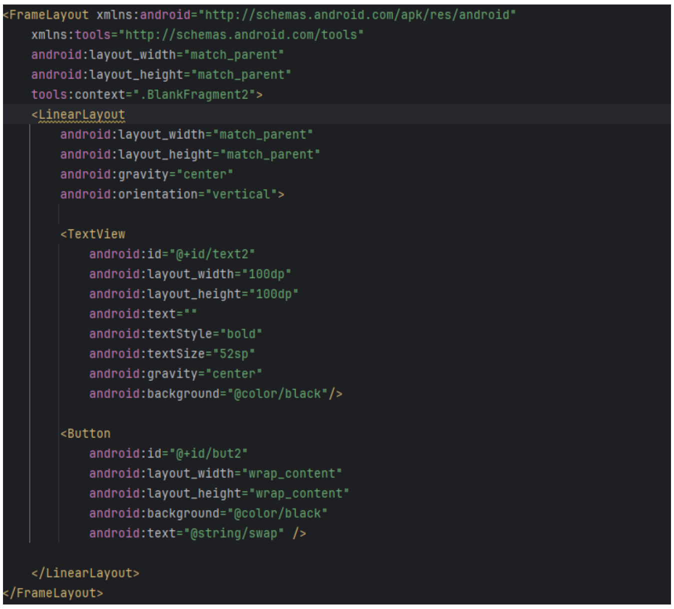
fragment_blank1.xml



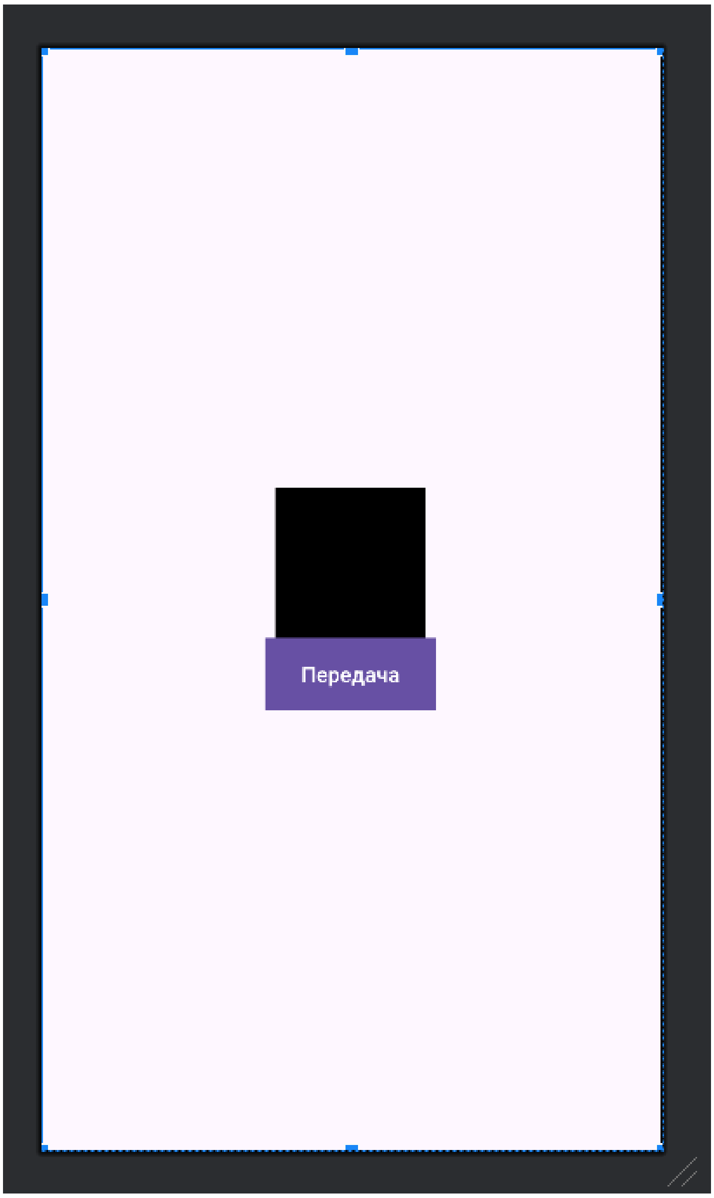
```
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".BlankFragment1">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center"
        android:orientation="vertical">

        <TextView
            android:id="@+id/text1"
            android:layout_width="115dp"
            android:layout_height="100dp"
            android:background="@color/white"
            android:gravity="center"
            android:text="1"
            android:textColor="@color/black"
            android:textSize="52sp"
            android:textStyle="bold" />

        <Button
            android:id="@+id/but1"
            android:layout_width="135dp"
            android:layout_height="74dp"
            android:background="@color/black"
            android:text="Передача" />


    </LinearLayout>

</FrameLayout>
```



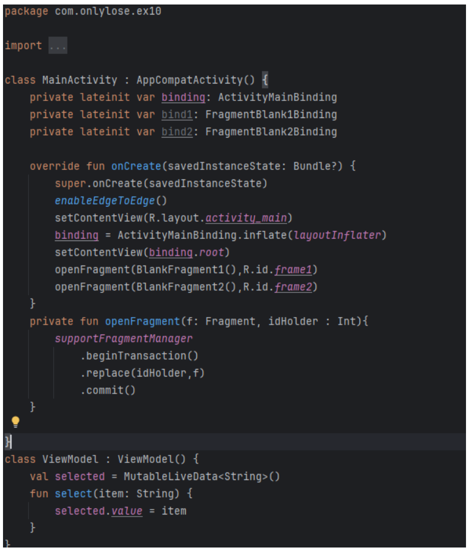
fragment_blank2.xml



```
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".BlankFragment2">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center"
        android:orientation="vertical">

        <TextView
            android:id="@+id/text2"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:text=""
            android:textStyle="bold"
            android:textSize="52sp"
            android:gravity="center"
            android:background="@color/black"/>

        <Button
            android:id="@+id/but2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:background="@color/black"
            android:text="@string/swap" />

    </LinearLayout>
</FrameLayout>
```



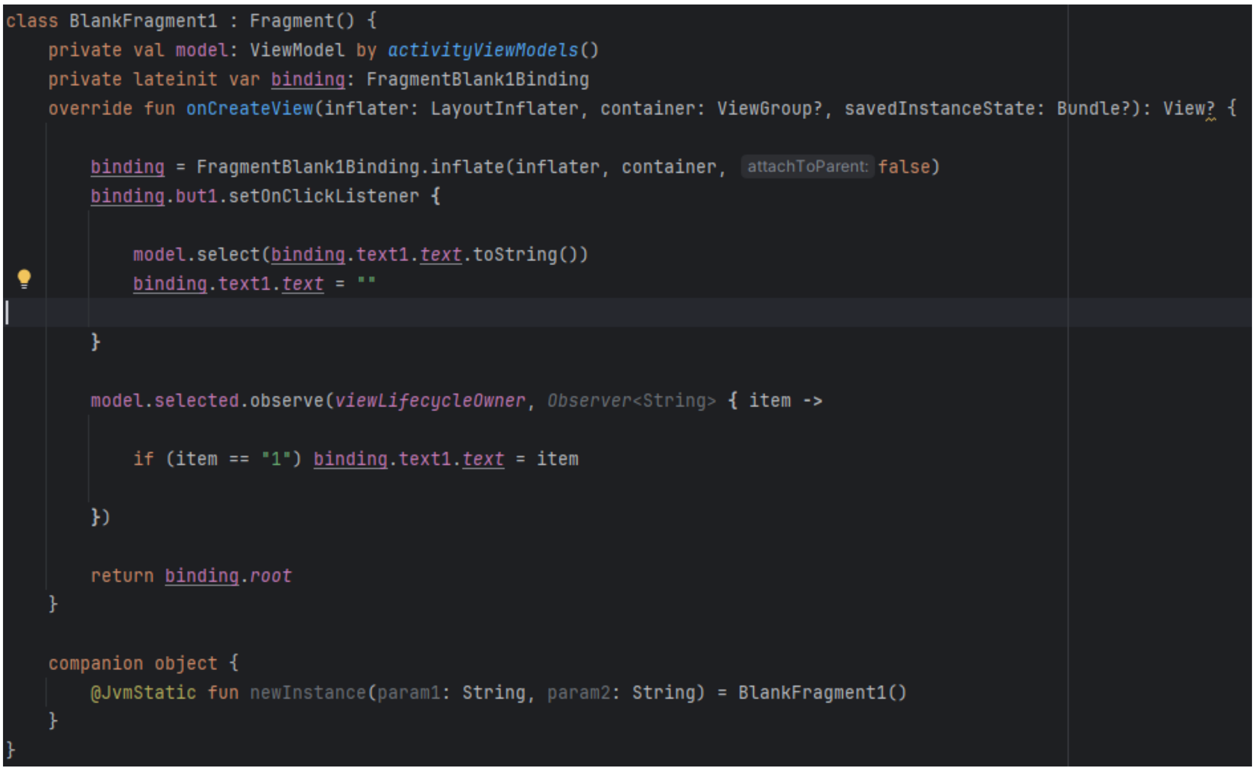
## 5.Пишем код Activity Main



```
package com.onlylose.ex10

import ...

class MainActivity : AppCompatActivity() {
    private lateinit var binding: ActivityMainBinding
    private lateinit var bind1: FragmentBlank1Binding
    private lateinit var bind2: FragmentBlank2Binding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        binding = ActivityMainBinding.inflate(layoutInflater)
        setContentView(binding.root)
        openFragment(BlankFragment1(),R.id.frame1)
        openFragment(BlankFragment2(),R.id.frame2)
    }
    private fun openFragment(f: Fragment, idHolder : Int){
        supportFragmentManager
            .beginTransaction()
            .replace(idHolder,f)
            .commit()
    }

}
class ViewModel : ViewModel() {
    val selected = MutableLiveData<String>()
    fun select(item: String) {
        selected.value = item
    }
}
```

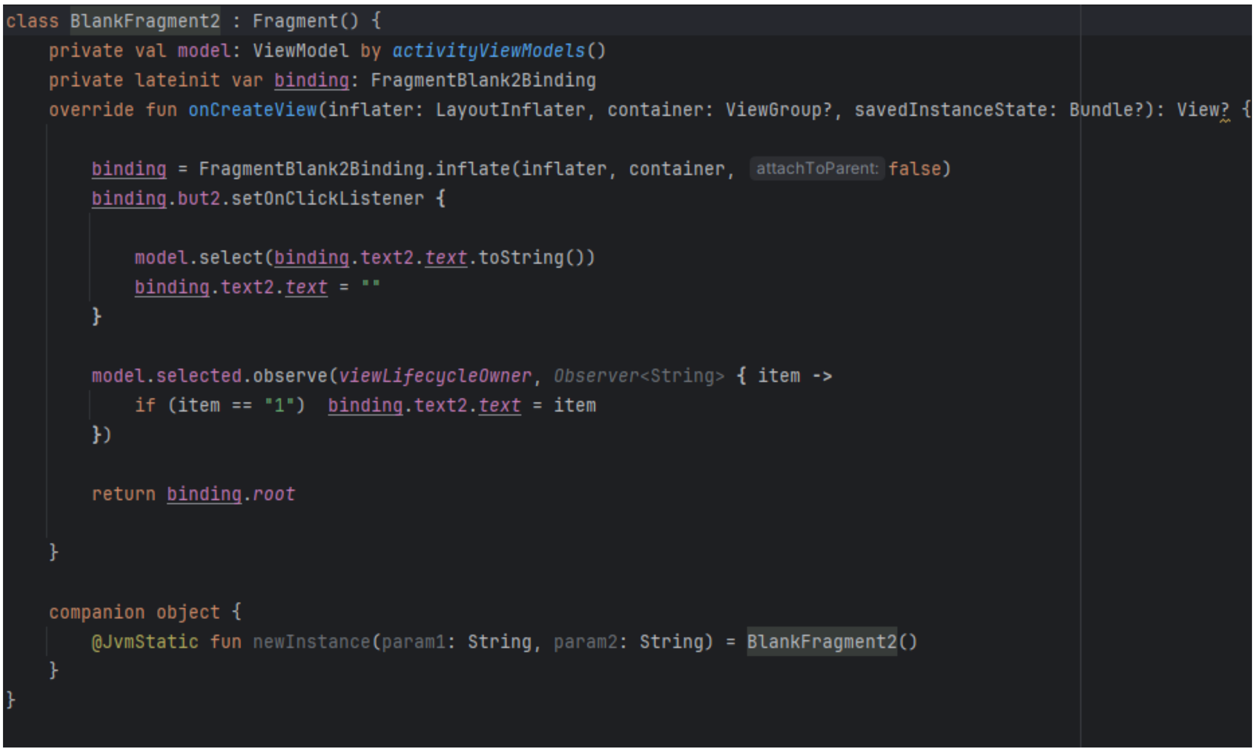
6.Передача данных между фрагментами (код)

BlankFragment1



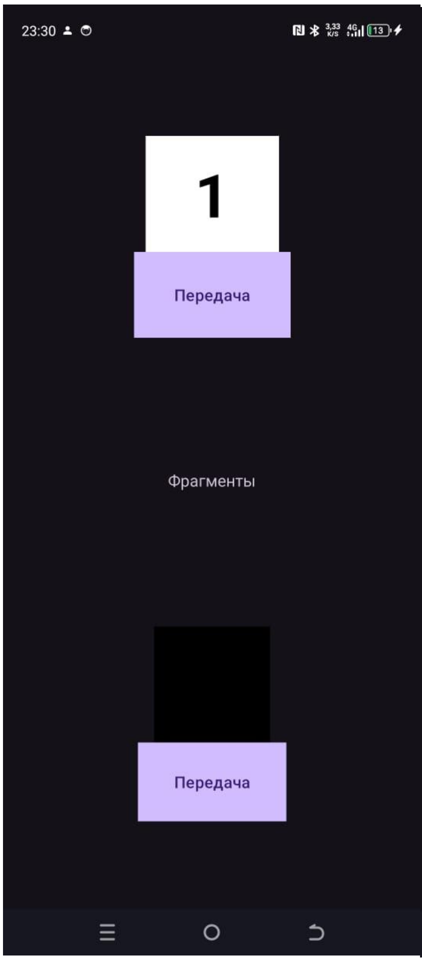
```
class BlankFragment1 : Fragment() {
    private val model: ViewModel by activityViewModels()
    private lateinit var binding: FragmentBlank1Binding
    override fun onCreateView(inflater: LayoutInflater, container: ViewGroup?, savedInstanceState: Bundle?): View? {

        binding = FragmentBlank1Binding.inflate(inflater, container,  attachToParent: false)
        binding.but1.setOnClickListener {

            model.select(binding.text1.text.toString())
            binding.text1.text = ""

        }

        model.selected.observe(viewLifecycleOwner, Observer<String> { item ->

            if (item == "1") binding.text1.text = item

        })

        return binding.root
    }

    companion object {
        @JvmStatic fun newInstance(param1: String, param2: String) = BlankFragment1()
    }
}
```

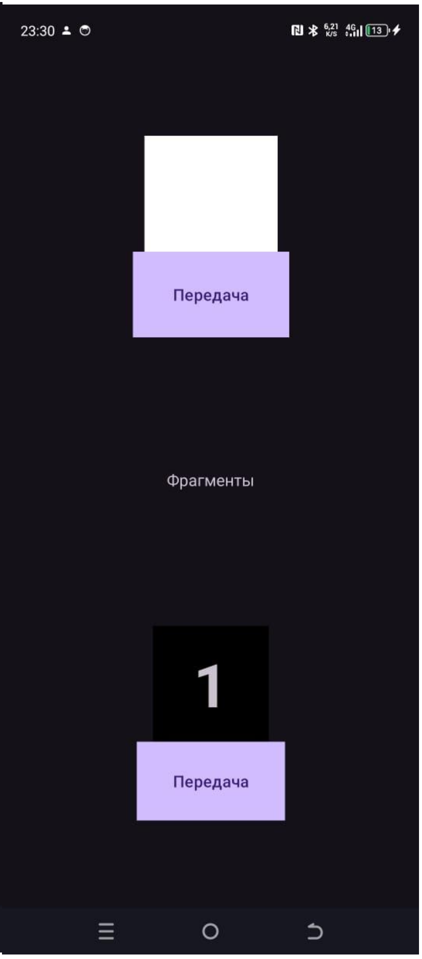
BlankFragment2



```
class BlankFragment2 : Fragment() {
    private val model: ViewModel by activityViewModels()
    private lateinit var binding: FragmentBlank2Binding
    override fun onCreateView(inflater: LayoutInflater, container: ViewGroup?, savedInstanceState: Bundle?): View? {

        binding = FragmentBlank2Binding.inflate(inflater, container,  attachToParent: false)
        binding.but2.setOnClickListener {

            model.select(binding.text2.text.toString())
            binding.text2.text = ""
        }

        model.selected.observe(viewLifecycleOwner, Observer<String> { item ->
            if (item == "1")  binding.text2.text = item
        })

        return binding.root

    }

    companion object {
        @JvmStatic fun newInstance(param1: String, param2: String) = BlankFragment2()
    }
}
```

7.Работа передачи:


23:30
1
Передача
Фрагменты
Передача


23:30
Передача
Фрагменты
1
Передача