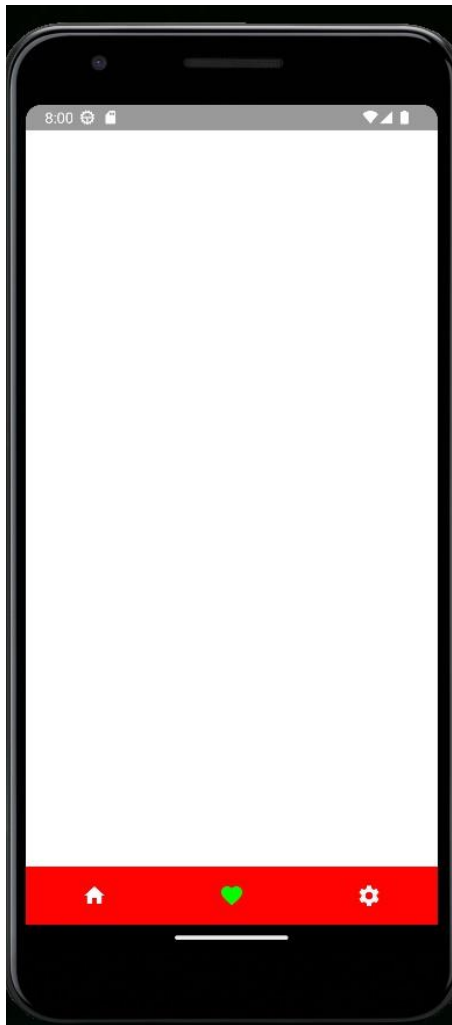


Практическая работа 9

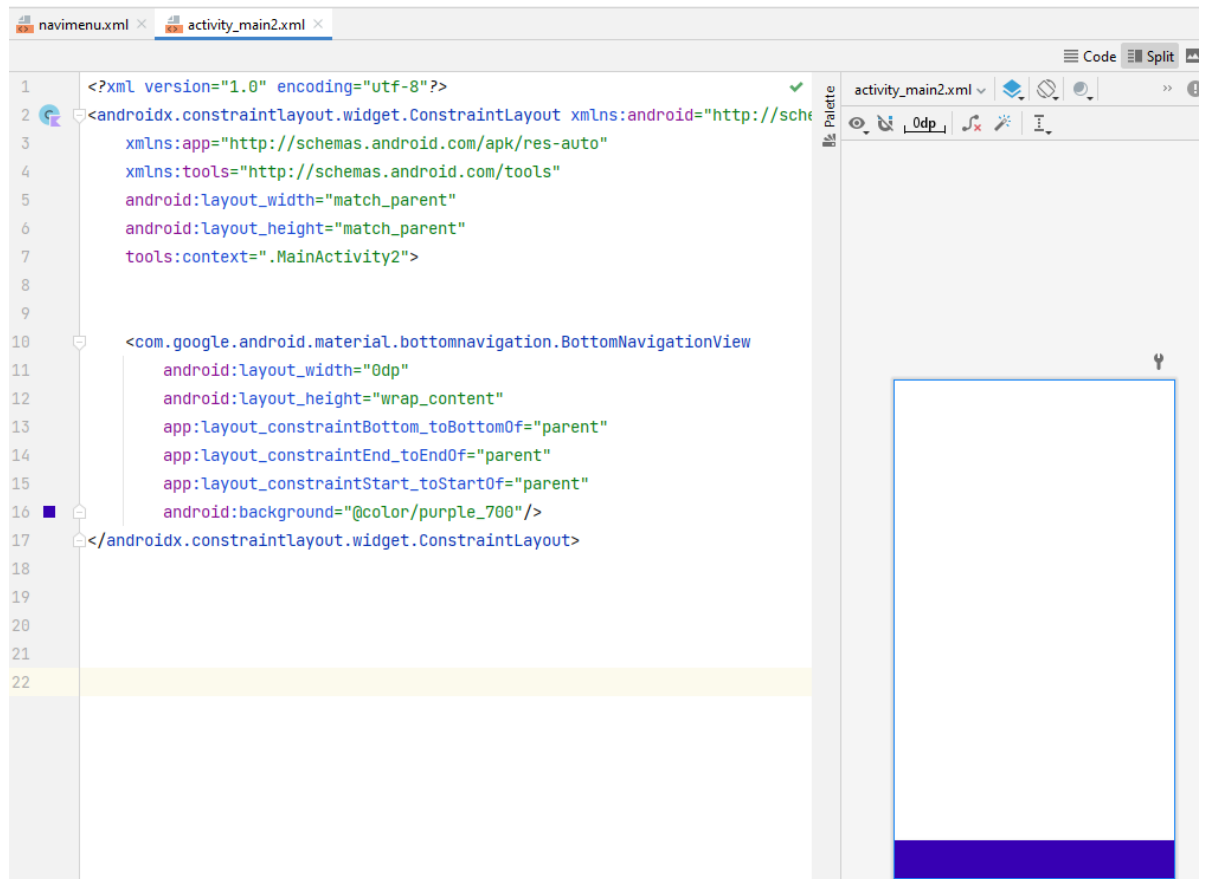
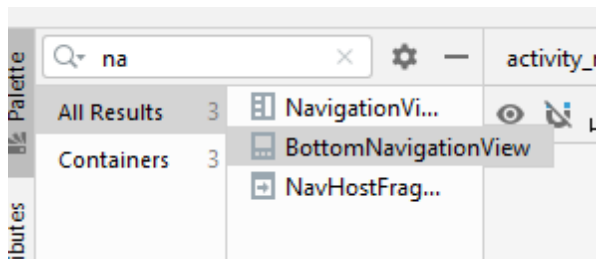
BottomNavigationView. Настройка навигационной панели



<https://developer.android.com/reference/com/google/android/material/bottomnavigation/BottomNavigationView>

BottomNavigationView - нижняя панель навигации на главном экране, в которой будут отображаться иконки - пункты меню. При нажатии каждого из них меняются экраны приложения

Создайте новый экран в любом проекте. Добавьте на экран элемент BottomNavigationView. Он займет собой весь экран. Пропишите для него следующие настройки:



После добавления навигационная панель будет пустой, меню - иконки - необходимо создать (см.практическую 6, как создается пользовательское меню). Создайте новое меню для панели навигации:

```

navimenu.xml x
1  <?xml version="1.0" encoding="utf-8"?>
2  <menu xmlns:android="http://schemas.android.com/apk/res/android">
3      <item android:icon="@drawable/ic_home"
4          android:id="@+id/ichome"
5          android:title="home"
6          android:orderInCategory="1"/>
7
8      <item android:icon="@drawable/ic_favorite"
9          android:id="@+id/icfav"
10         android:title="favorite"
11         android:orderInCategory="2"/>
12
13     <item android:icon="@drawable/ic_settings"
14         android:id="@+id/icset"
15         android:title="settings"
16         android:orderInCategory="3"/>
17 </menu>

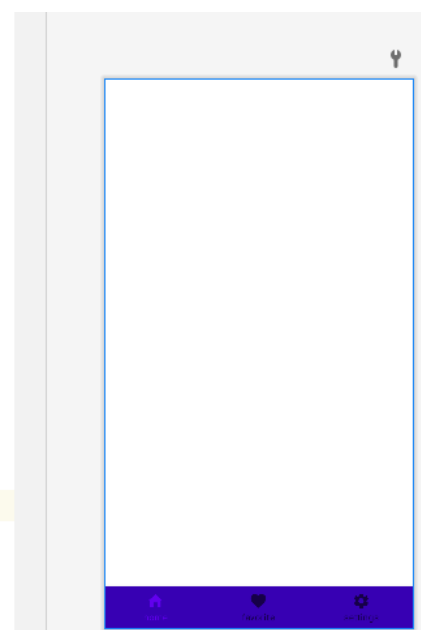
```

Вернитесь в файл разметки экрана. Для добавления иконок - созданного вами меню - на панель навигации используется параметр `app:menu="@menu/navimenu"`, где `navimenu` - название файла с вашим меню.

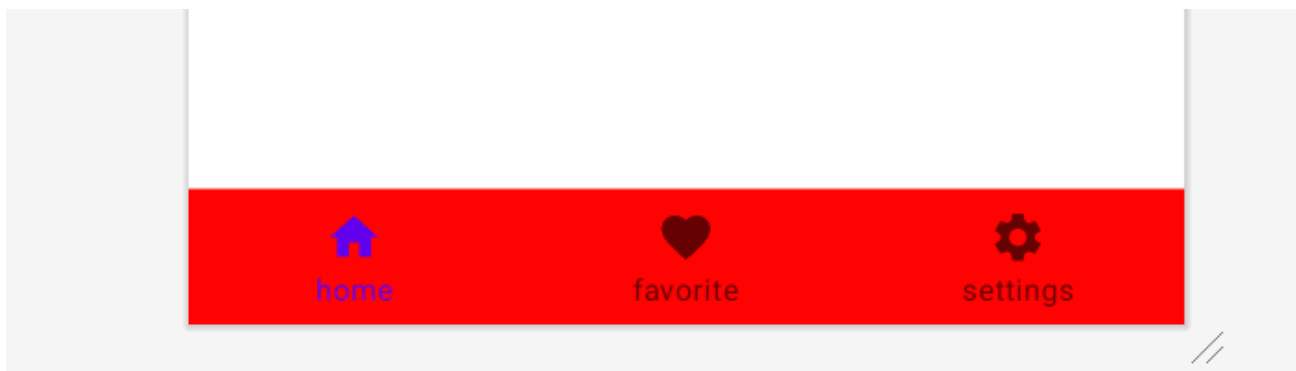
```

10  <com.google.android.material.bottomnavigation.BottomNavigationView
11      android:id="@+id/navi"
12      android:layout_width="0dp"
13      android:layout_height="wrap_content"
14      app:layout_constraintBottom_toBottomOf="parent"
15      app:layout_constraintEnd_toEndOf="parent"
16      app:layout_constraintStart_toStartOf="parent"
17      android:background="@color/purple_700"
18      app:menu="@menu/navimenu"
19  />
20  </androidx.constraintlayout.widget.ConstraintLayout>
21
22
23
24
25

```



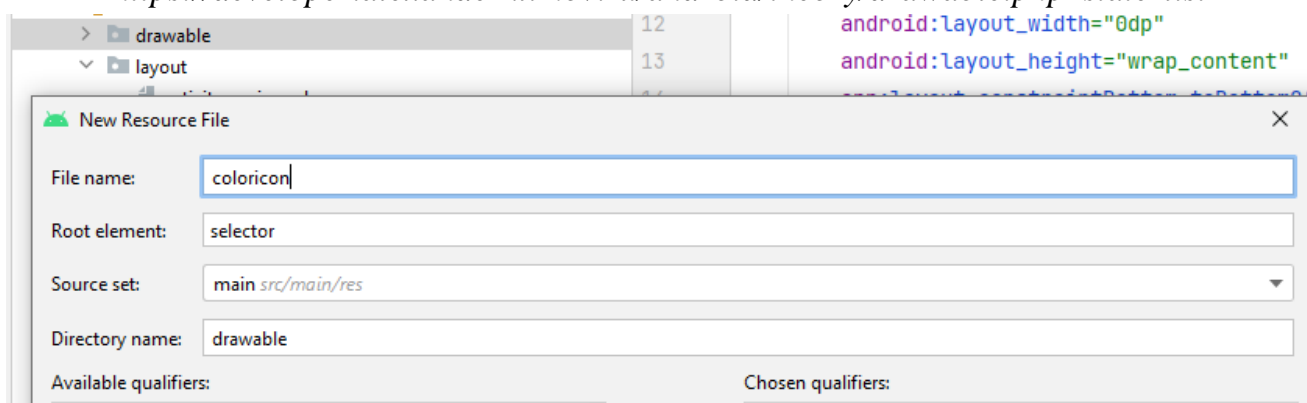
Запустите приложение, посмотрите как работают иконки.



Цвета иконок по умолчанию совершенно не сочетаются с цветом навигационной панели. Для изменения цвета иконок создайте новый Drawable Resource File тип файла **SELECTOR**:

<https://developer.android.com/guide/topics/resources/drawable-resource>

<https://developer.alexanderklimov.ru/android/theory/drawable.php#state-list>



У некоторых элементов управления есть различные состояния: нажато, выбрано и т.д.. Вы можете задавать изображения для любого из таких состояний.

Чтобы описать *StateListDrawable*, создайте файл в формате XML, в котором указываются разные ресурсы *Drawable* для каждого состояния компонента.

Каждый атрибут вида *android:state_** может принимать одно из двух значений: *true* или *false*.

Доступны следующие состояния:

- *state_focused* - в фокусе или нет
- *state_window_focused* - имеет ли фокус родительское окно
- *state_enabled* - доступен или нет
- *state_checkable* - может ли быть отмечен или нет
- *state_checked* - отмечен или нет
- *state_selected* - выделен или нет
- *state_activated* - активирован или нет
- *state_pressed* - нажат или не нажат
- *state_hovered* - находится ли курсор над компонентом или нет (API 11 и выше) и другие см.документацию разработчиков

Для списков есть также:

- *state_single*
- *state_first*
- *state_middle*
- *state_last*

```
navimenu.xml × activity_main2.xml × coloricon.xml × colors.xml ×
1 <?xml version="1.0" encoding="utf-8"?>
2 <selector xmlns:android="http://schemas.android.com/apk/res/android">
3   <item android:state_checked="true" android:color="@color/green"/>
4     <item android:state_checked="false" android:color="@color/white"/>
5   |
6 </selector>
```

3 строка - состояние - элемент отмечен/нажат (значение true), установить цвет элемента green

4 строка - состояние - элемент не отмечен/не нажат (значение false), установить цвет для элемента белый.

У всех элементов, которые не нажаты, будет состояние false.

Вернитесь в разметку экрана и установите цвет для текста под иконками в навигационной панели: `app:itemTextColor="@color/white"`

Для назначения ресурсного файла используйте параметр `app:itemIconTint="@drawable/coloricon"`

```
10 <com.google.android.material.bottomnavigation.BottomNavigationView
11   android:id="@+id/navi"
12   android:layout_width="0dp"
13   android:layout_height="wrap_content"
14   app:layout_constraintBottom_toBottomOf="parent"
15   app:layout_constraintEnd_toEndOf="parent"
16   app:layout_constraintStart_toStartOf="parent"
17   android:background="@color/red"
18   app:menu="@menu/navimenu"
19   app:itemTextColor="@color/white"
20   app:itemIconTint="@drawable/coloricon"
21   />
22 </androidx.constraintlayout.widget.ConstraintLayout>
23
```

Запустите приложение. Как вы думаете, почему получился такой результат?



Как можно исправить и получить такой:



Добавьте обработчик нажатия на иконки.

Переключитесь в файл логики экрана и пропишите функцию. Недостающие строки кода пропишите самостоятельно:

```

13  bn.setOnItemSelectedListener{ it: MenuItem
14      when (it.itemId)
15      {
16          R.id.ichome->{}
17          R.id.icfav->{}
18          R.id.icset->{}
19      }
20      true ^setOnItemSelectedListener
21  }
22  }
23  }

```

setOnItemSelectedListener - прослушивает нажатие на элемент (item) и возвращает истину, если оно было. В фигурных скобках прописывается что должно происходить. Переменная it - скрытая переменная этой функции и отвечает за обращение к id элемента меню навигации.

С работой оператора when вы уже знакомы. Пропишите любое действия для каждой иконки навигационной панели.

```

1 package com.example.myapplicationtablayout
2
3 import androidx.appcompat.app.AppCompatActivity
4 import android.os.Bundle
5 import android.widget.Toast
6 import com.google.android.material.bottomnavigation.BottomNavigationView
7 import com.google.android.material.navigation.NavigationBarView
8
9 class MainActivity2 : AppCompatActivity() {
10     ...
11     override fun onCreate(savedInstanceState: Bundle?) {
12         super.onCreate(savedInstanceState)
13         setContentView(R.layout.activity_main2)
14         ...
15         bn.setOnItemSelectedListener { it: MenuItem
16             when (it.itemId) {
17                 R.id.ichome->{
18                     Toast.makeText( context: this, text: "Паботает 1", Toast.LENGTH_SHORT).show()
19                 }
20                 R.id.icfav->{Toast.makeText( context: this, text: "Паботает 2", Toast.LENGTH_SHORT).show()}
21                 R.id.icset->{Toast.makeText( context: this, text: "Паботает 3", Toast.LENGTH_SHORT).show()}
22             }
23             true ^setOnNavigationItemSelectedListener
24             }
25         }
26     }
27
28 }

```

```

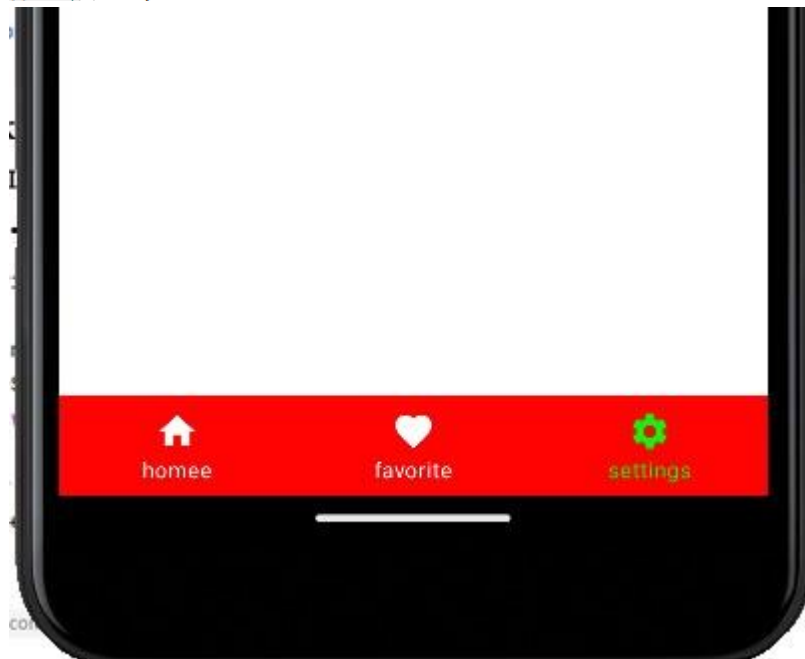
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     tools:context=".MainActivity2">
8
9
10     <com.google.android.material.bottomnavigation.BottomNavigationView
11         ...
12         android:layout_width="0dp"
13         android:layout_height="wrap_content"
14         app:layout_constraintBottom_toBottomOf="parent"
15         app:layout_constraintEnd_toEndOf="parent"
16         app:layout_constraintStart_toStartOf="parent"
17         android:background="@color/red"
18         app:menu="@menu/navimenu"
19         app:itemTextColor="@drawable/coloricon"
20         app:itemIconTint="@drawable/coloricon"
21     />
22 </androidx.constraintlayout.widget.ConstraintLayout>
23

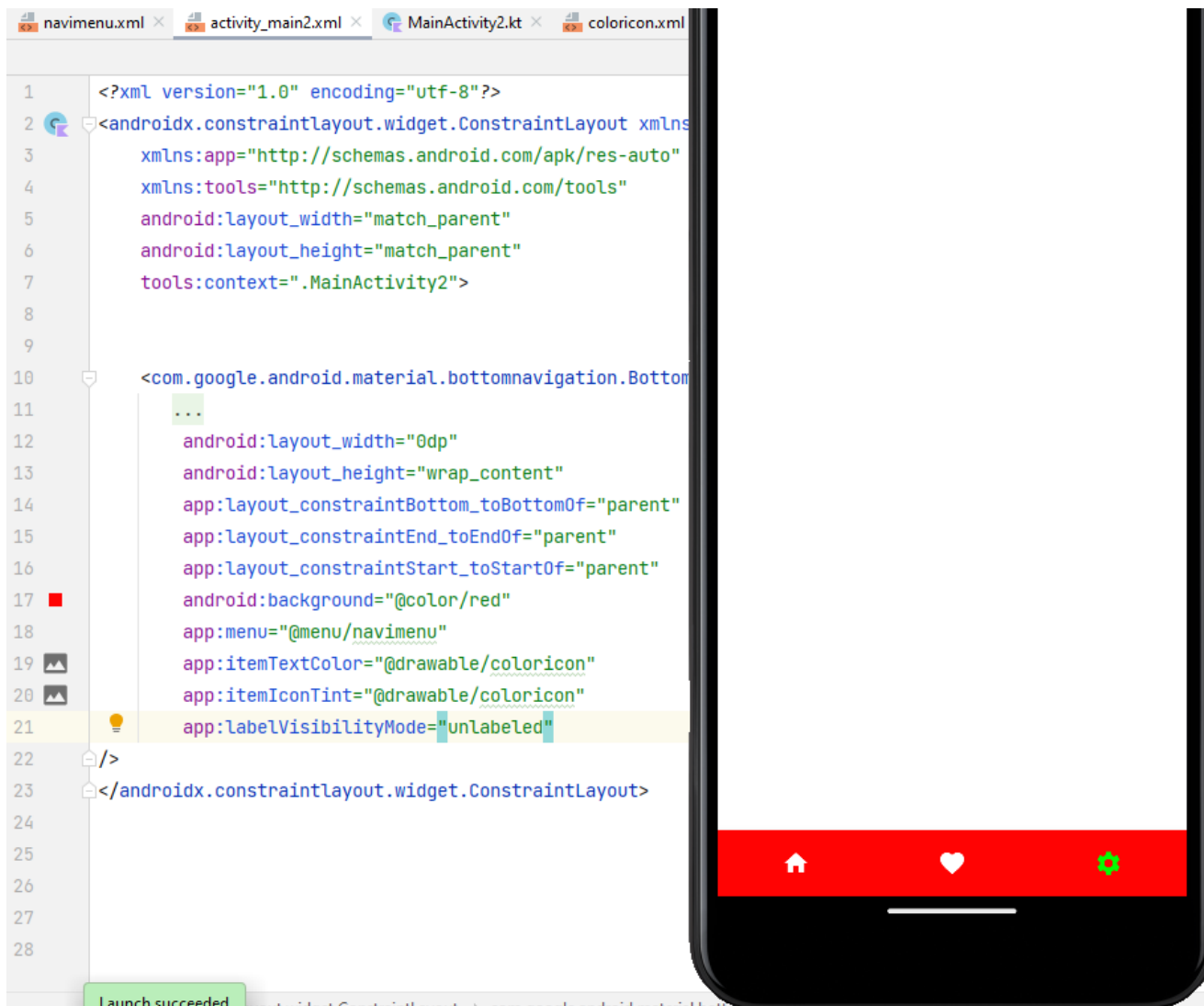
```


Если вы хотите, чтобы какой-то item в меню был выбран по умолчанию, необходимо прописать следующую строку в файле логики:

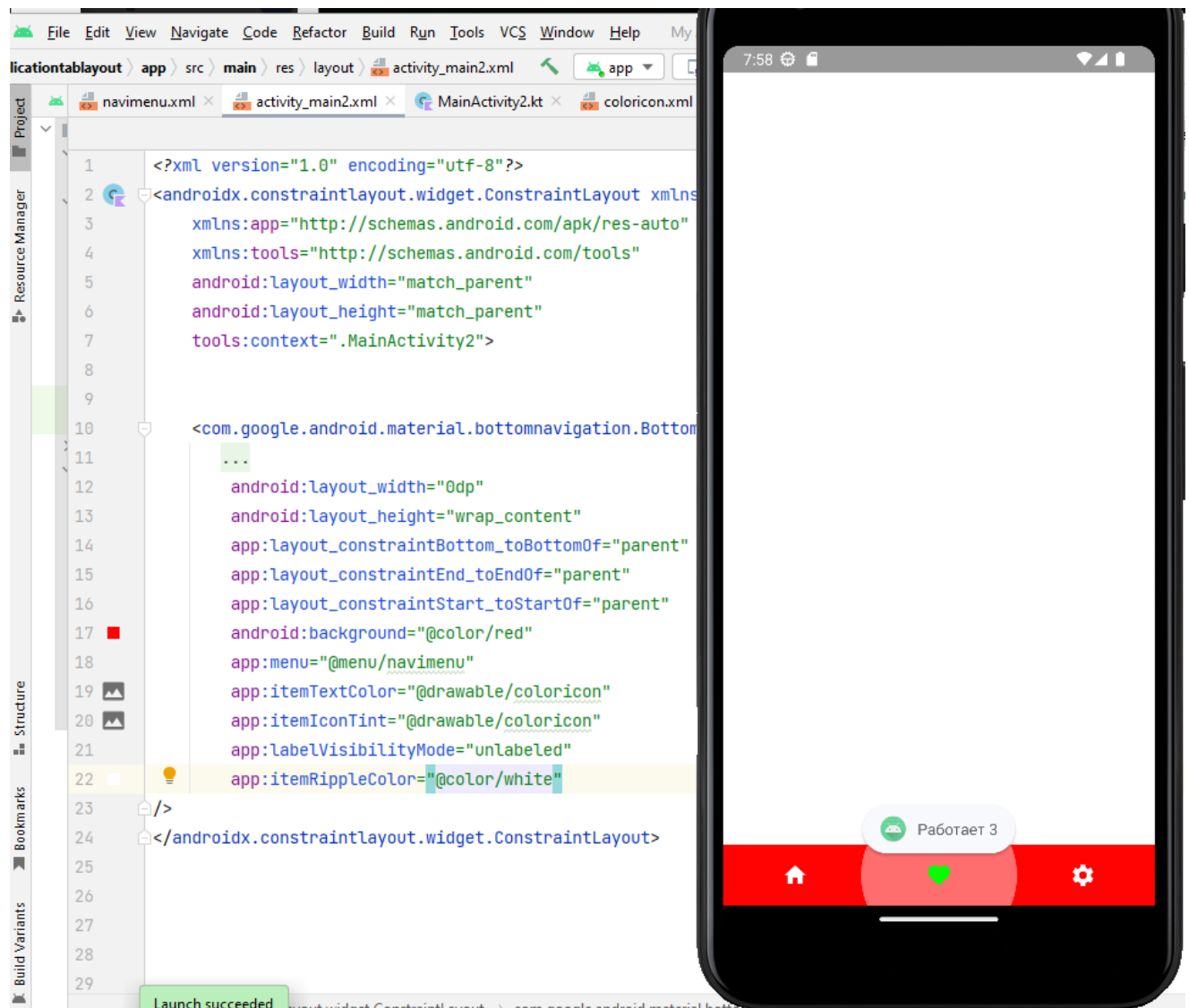
`bn.selectedItemId=R.id.icset`

```
9 class MainActivity2 : AppCompatActivity() {
10     ...
11     override fun onCreate(savedInstanceState: Bundle?) {
12         super.onCreate(savedInstanceState)
13         setContentView(R.layout.activity_main2)
14         ...
15         bn.selectedItemId=R.id.icset
16         bn.setOnItemSelectedListener { it: MenuItem
17             when (it.itemId) {
18                 R.id.ichome->{
19                     Toast.makeText( context: this, text: "Работает 1", Toast.LENGTH_SHORT).show()
20                 }
21                 R.id.icfav->{Toast.makeText( context: this, text: "Работает
22                 R.id.icset->{Toast.makeText( context: this, text: "Работает
23             }
24             true ^setOnNavigationItemSelectedListener
25         }
26     }
27 }
```

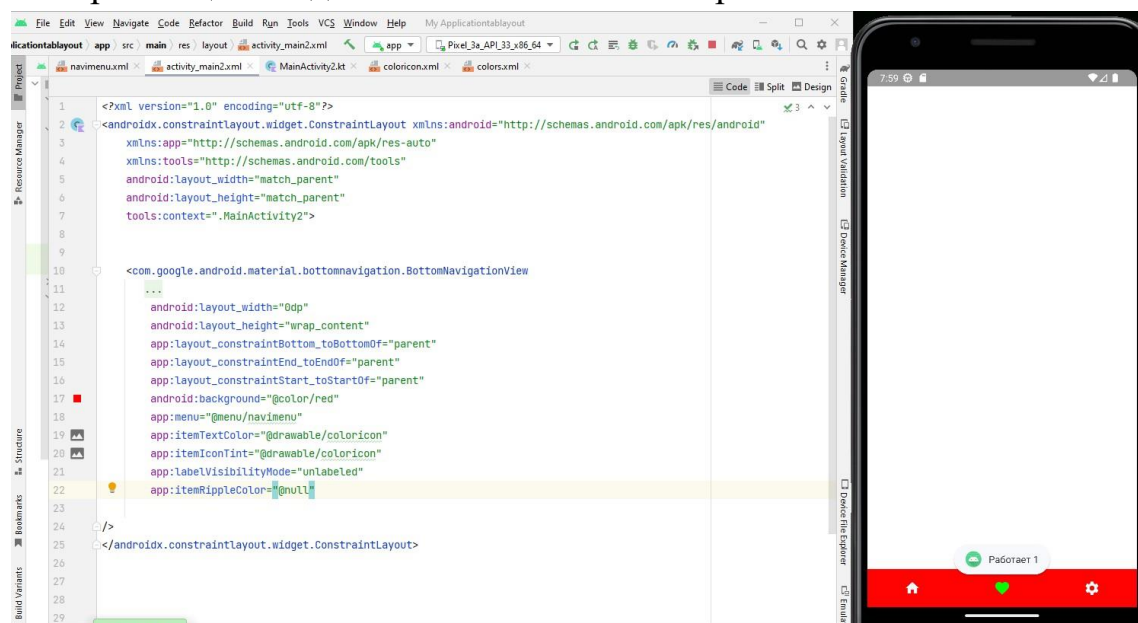




21 строка - убрать подписи иконок (попробуйте разные значения параметра)



22 строка - цвет подсвечивания области выбранной иконки отключить



Самостоятельно подключите фрагменты (как создаются фрагменты см. практическую работу 8, содержимое фрагментов может быть любым) для отображения по нажатию на иконку навигационной панели (не забудьте добавить область для отображения фрагментов – Frame Layout, значение высоты установите не match_parent, а 0dp и посмотрите как будет отображаться фрагмент):

