**Министерство науки и высшего образования Российской Федерации**

Федеральное государственное автономное образовательное учреждение

высшего образования «Уральский федеральный университет

имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий - РТФ

Центр ускоренного обучения

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ 14**

по дисциплине «Разработка мобильных приложений»

**Тема: Unit 2 Пользовательский интерфейс “Навигация пользователя”**

Студент группы РИЗ-300028у: А.С. Севостьянов

Преподаватель: С.И.Тимошенко,

доц., к.т.н.

**Екатеринбург 2022**

содержание

[1 Постановка задачи 3](#_Toc118041527)

[2 Результаты проделанной работы 4](#_Toc118041528)

[3 Анализ полученных результатов 5](#_Toc118041529)

## 1 Постановка задачи

Создать новое приложение с навигационным ящиком. Когда пользователь нажимает на выбор навигационного ящика, закройте ящик и отобразите Toast сообщение, показывающее, какой выбор был выбран.

## Результаты проделанной работы

Программный код проекта, файл MainActivity.

package com.example.android.navdrawerexperiment;  
  
import android.os.Bundle;  
import android.support.design.widget.FloatingActionButton;  
import android.support.design.widget.Snackbar;  
import android.view.View;  
import android.support.design.widget.NavigationView;  
import android.support.v4.view.GravityCompat;  
import android.support.v4.widget.DrawerLayout;  
import android.support.v7.app.ActionBarDrawerToggle;  
import android.support.v7.app.AppCompatActivity;  
import android.support.v7.widget.Toolbar;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.widget.Toast;  
  
*/\*\*  
 Это приложение настроило панель навигации со следующими вариантами: Импорт, галерея,  
 Слайд-шоу, инструменты, общий доступ и отправка. Выбор любого варианта закрывает панель навигации  
 и отображает тостовое сообщение, показывающее, какой выбор был выбран.  
 Это приложение было создано с использованием шаблона действий в навигационном ящике.  
 \*/*public class MainActivity extends AppCompatActivity  
 implements NavigationView.OnNavigationItemSelectedListener {  
  
 */\*\*  
 Создает представление содержимого и панель инструментов, настраивает расположение ящика и переключатель  
 панель действий, а также настраивает вид навигации  
 \*/* @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.*toolbar*);  
 setSupportActionBar(toolbar);  
  
 FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.*fab*);  
 if (fab != null) {  
 fab.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Snackbar.*make*(view, "Загружаем", Snackbar.*LENGTH\_LONG*)  
 .setAction("Action", null).show();  
 }  
 });  
 }  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.*drawer\_layout*);  
 ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(  
 this, drawer, toolbar, R.string.*navigation\_drawer\_open*,  
 R.string.*navigation\_drawer\_close*);  
 if (drawer != null) {  
 drawer.addDrawerListener(toggle);  
 }  
 toggle.syncState();  
  
 NavigationView navigationView = (NavigationView) findViewById(R.id.*nav\_view*);  
 if (navigationView != null) {  
 navigationView.setNavigationItemSelectedListener(this);  
 }  
 }  
  
 */\*\*  
 \* Управляет кнопкой "Назад": закрывает навигационный ящик.  
 \*/* @Override  
 public void onBackPressed() {  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.*drawer\_layout*);  
 if (drawer != null) {  
 if (drawer.isDrawerOpen(GravityCompat.*START*)) {  
 drawer.closeDrawer(GravityCompat.*START*);  
 } else {  
 super.onBackPressed();  
 }  
 }  
 }  
  
 */\*\*  
 Раздувает меню опций.  
 \*/* @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
// Раздуть меню; это добавляет элементы на панель действий, если она присутствует.  
 getMenuInflater().inflate(R.menu.*main*, menu);  
 return true;  
 }  
  
 */\*\*  
 Обрабатывает щелчок по пункту Настройки в меню опций.  
 \*/* @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
// Обработайте элемент панели действий, нажав здесь. Панель действий будет  
// автоматически обрабатывать нажатия на кнопку "Домой"/"Вверх", пока  
// как вы указываете родительское действие в AndroidManifest.xml .  
 int id = item.getItemId();  
  
//отсутствие проверки, упрощаемое утверждение  
 if (id == R.id.*action\_settings*) {  
 return true;  
 }  
  
 return super.onOptionsItemSelected(item);  
 }  
  
 */\*\*  
 \* Обрабатывает щелчок по элементу навигационного ящика. Он определяет, какой элемент был  
 \* щелкнул и отобразил всплывающее сообщение, показывающее, какой элемент.  
 \*/* @SuppressWarnings("StatementWithEmptyBody")  
 @Override  
 public boolean onNavigationItemSelected(MenuItem item) {  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.*drawer\_layout*);  
// Обработайте элемент просмотра навигации, нажав здесь.  
 switch (item.getItemId()) {  
 case R.id.*nav\_camera*:  
// Обработайте действие импорта камеры (пока отобразите тост).  
 drawer.closeDrawer(GravityCompat.*START*);  
 displayToast(getString(R.string.*chose\_camera*));  
 return true;  
 case R.id.*nav\_gallery*:  
// Обработайте действие галереи (пока отобразите тост).  
 drawer.closeDrawer(GravityCompat.*START*);  
 displayToast(getString(R.string.*chose\_gallery*));  
 return true;  
 case R.id.*nav\_slideshow*:  
// Обработайте действие слайд-шоу (пока отобразите тост).  
 drawer.closeDrawer(GravityCompat.*START*);  
 displayToast(getString(R.string.*chose\_slideshow*));  
 return true;  
 case R.id.*nav\_manage*:  
// Обработайте действие tools (пока отобразите тост).  
 drawer.closeDrawer(GravityCompat.*START*);  
 displayToast(getString(R.string.*chose\_tools*));  
 return true;  
 case R.id.*nav\_share*:  
// Обработайте действие "Поделиться" (пока отобразите тост).  
 drawer.closeDrawer(GravityCompat.*START*);  
 displayToast(getString(R.string.*chose\_share*));  
 return true;  
 case R.id.*nav\_send*:  
// Обработайте действие отправки (пока отобразите тост).  
 drawer.closeDrawer(GravityCompat.*START*);  
 displayToast(getString(R.string.*chose\_send*));  
 return true;  
 default:  
 return false;  
 }  
 }  
  
 */\*\*  
 \* Отображает всплывающее сообщение.  
 \*/* public void displayToast(String message) {  
 Toast.*makeText*(getApplicationContext(), message, Toast.*LENGTH\_SHORT*).show();  
 }  
}

На рисунке 1 показано первое состояние приложения, в верхней части приложения есть навигационная панель, у нее есть название, в правом верхнем углу три точки Toolbar, он в себе содержит список настроек приложения. Слева в верхнем углу SideBar – содержит в себе выедающую панель со списком.

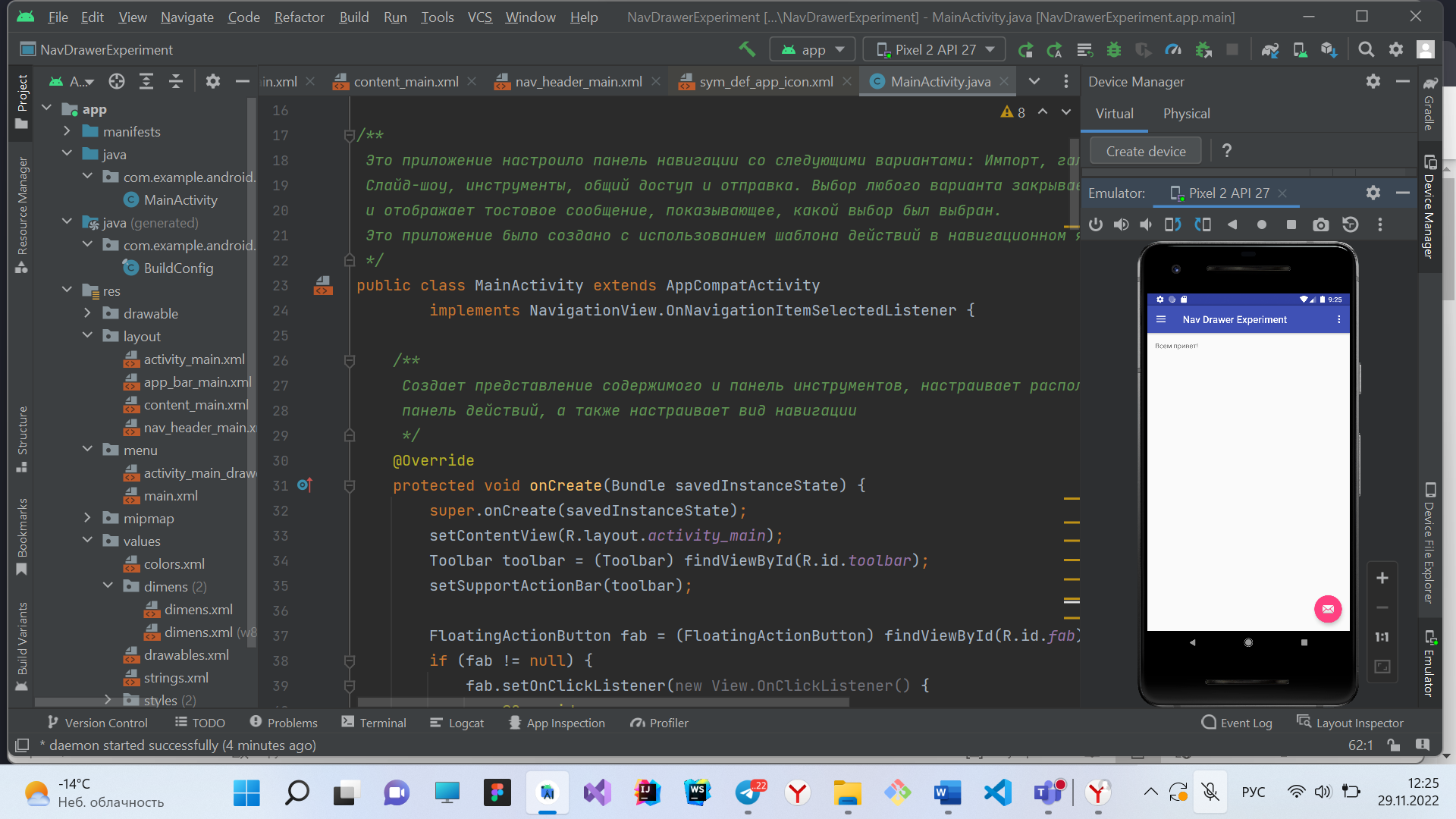


Рисунок 1 – первое состояние приложния

На рисунке 2 показано активное состояние выезжающей панели SideBar со списком возможных функций.

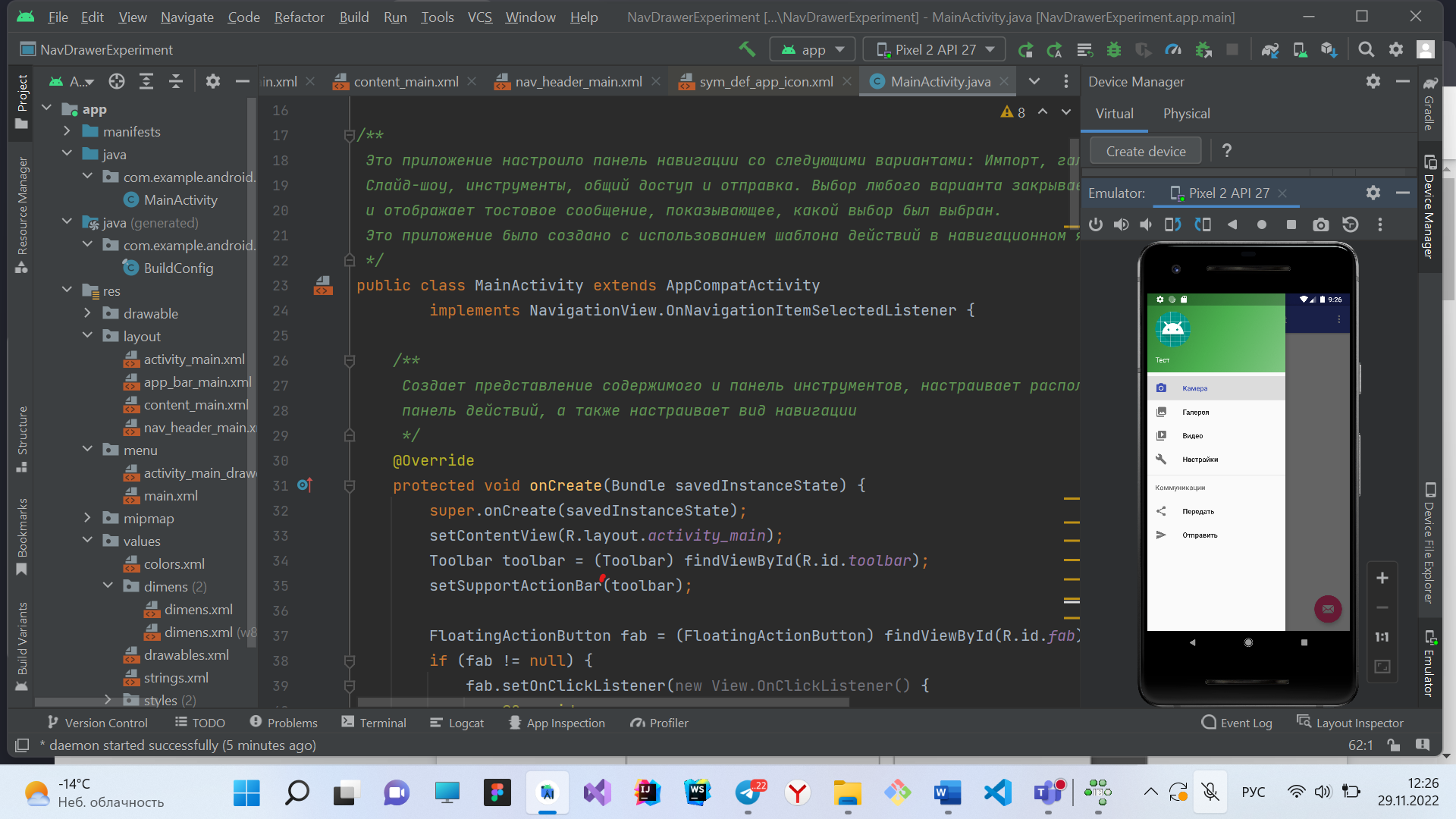


Рисунок 2 – Панель SideBar

На рисунке 3 показан результат клика на функцию в SideBar.

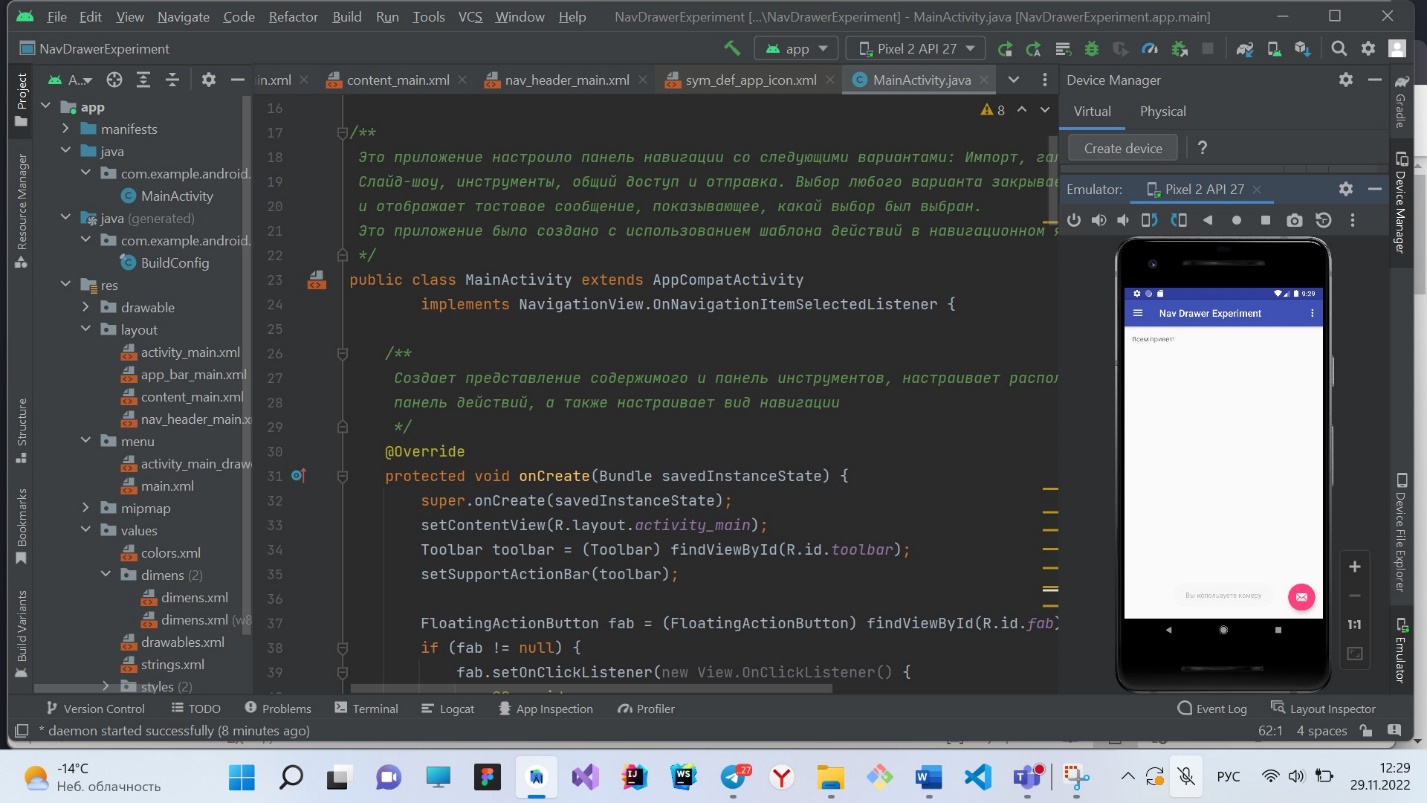


Рисунок 3 – Toast уведомление

## **Анализ полученных результатов**

В результате выполнения данной лабораторной работы мной был разработан проект, в котором я научился, создавать навигационную панель, пользователь которой может нажимать на выбор навигационного ящика, ящик закроется и отобразится уведомление Toast сообщение, показывающее какой выбор был сделан.