Министерство науки и высшего образования Российской Федерации

**Муромский институт (филиал)**

федерального государственного бюджетного образовательного учреждения высшего образования

**«Владимирский государственный университет**

**Имени Александра Григорьевича и Николая Григорьевича Столетовых»**

**(МИВлГУ)**

Факультет ИТР

Кафедра ПИн

Практическая работа №1

По Разработке приложений для мобильных операционных систем

Тема Знакомство с Android Studio и Kotlin

Руководитель

Колпаков А.А.

(фамилия, инициалы)

(подпись) (дата)

Студент ПИн-121

(группа)

Ермилов М.В.

(фамилия, инициалы)

(подпись) (дата)

Муром 2024

Практическая работа №1

Тема: Знакомство с Android Studio и Kotlin.

Задание на практическую работу:

1. Изучить способы представления приложений в различной ориентации экрана.

2. Получить и вывести на экран текущую ориентацию экрана.

3. Подготовить программу, изменяющую ориентацию экрана при нажатии кнопки на экране.

Ход работы

Листинг:

package com.test.pr1\_mob

import android.content.pm.ActivityInfo

import android.content.res.Configuration

import android.os.Bundle

import android.view.Surface

import android.view.View

import android.widget.Button

import android.widget.EditText

import android.widget.Toast

import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

val editText = findViewById<EditText>(R.id.editText)

val orientation = getScreenOrientation()

val rotation = getRotateOrientation()

editText.setText("Ориентация экрана: $orientation\nПоворот экрана: $rotation")

val button1 = findViewById<Button>(R.id.button1)

val button2 = findViewById<Button>(R.id.button2)

val button3 = findViewById<Button>(R.id.button3)

val button4 = findViewById<Button>(R.id.button4)

val button5 = findViewById<Button>(R.id.button5)

val button6 = findViewById<Button>(R.id.button6)

button1.setOnClickListener {

showToast("Нажата первая кнопка")

}

button2.setOnClickListener {

showToast("Нажата вторая кнопка")

}

button3.setOnClickListener {

showToast("Нажата третья кнопка")

}

button4.setOnClickListener {

showToast("Нажата четвёртая кнопка")

}

button5.setOnClickListener {

showToast("Нажата пятая кнопка")

}

button6.setOnClickListener {

showToast("Нажата шестая кнопка")

}

}

private fun getScreenOrientation(): String {

return when (resources.configuration.orientation) {

Configuration.ORIENTATION\_PORTRAIT -> "Портретная ориентация"

Configuration.ORIENTATION\_LANDSCAPE -> "Альбомная ориентация"

else -> "Неизвестная ориентация"

}

}

private fun getRotateOrientation(): String {

return when (windowManager.defaultDisplay.rotation) {

Surface.ROTATION\_0 -> "Не поворачивали"

Surface.ROTATION\_90 -> "Повернули на 90 градусов по часовой стрелке"

Surface.ROTATION\_180 -> "Повернули на 180 градусов"

Surface.ROTATION\_270 -> "Повернули на 90 градусов против часовой стрелки"

else -> "Неизвестный поворот"

}

}

private fun showToast(message: String) {

Toast.makeText(this, message, Toast.LENGTH\_SHORT).show()

}

override fun onConfigurationChanged(newConfig: Configuration) {

super.onConfigurationChanged(newConfig)

val editText = findViewById<EditText>(R.id.editText)

val orientation = getScreenOrientation()

val rotation = getRotateOrientation()

editText.setText("Ориентация экрана: $orientation\nПоворот экрана: $rotation")

}

fun onMyButtonClick(view: View?) {

var rotate = getWindowManager().getDefaultDisplay().getRotation();

if(rotate == Surface.ROTATION\_0 || rotate == Surface.ROTATION\_180){

setRequestedOrientation (ActivityInfo.SCREEN\_ORIENTATION\_LANDSCAPE);

}

else{

setRequestedOrientation (ActivityInfo.SCREEN\_ORIENTATION\_PORTRAIT);

}

}

}

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

xmlns:app="http://schemas.android.com/apk/res-auto"

android:orientation="vertical">

<LinearLayout

android:orientation="vertical"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:layout\_gravity="center"

android:paddingLeft="20dp"

android:paddingRight="20dp">

<EditText

android:layout\_height="wrap\_content"

android:id="@+id/editText"

android:layout\_width="match\_parent"/>

<TableLayout

android:layout\_height="wrap\_content"

android:layout\_width="wrap\_content"

android:layout\_gravity="center"

android:stretchColumns="\*">

<TableRow> <Button

android:id="@+id/button1"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:text="Первая кнопка"/>

<Button

android:id="@+id/button2"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:text="Вторая кнопка"/>

</TableRow> <TableRow>

<Button android:id="@+id/button3"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:text="Третья кнопка"/>

<Button

android:id="@+id/button4"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:text="Четвёртая кнопка"/>

</TableRow>

<TableRow> <Button

android:id="@+id/button5"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:text="Пятая кнопка"/>

<Button

android:id="@+id/button6"

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:text="Шестая кнопка"/>

</TableRow>

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:onClick="onMyButtonClick"

android:text="Повернуть" />

</TableLayout>

</LinearLayout>

</LinearLayout>

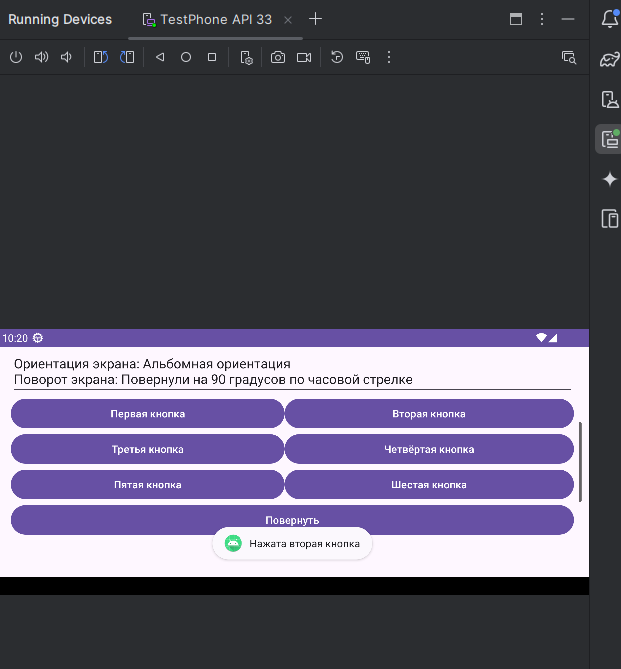


Рисунок 1 – работа приложения(альбомная ориентация)

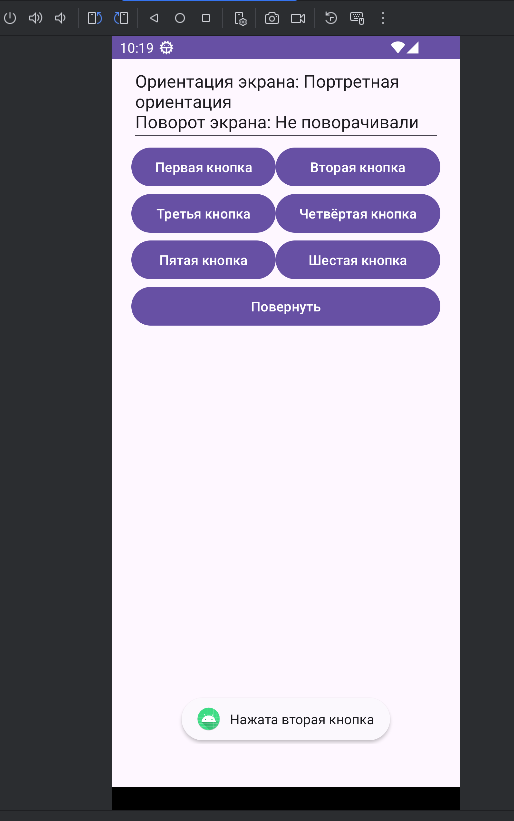


Рисунок 2 – работа приложения (портретная ориентация)

Вывод: в ходе практической работы познакомились с Android Studio и Kotlin, а также способами представления приложений в различной ориентации экрана.