**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**ВЫСШАЯ ШКОЛА ПЕЧАТИ И МЕДИАИНДУСТРИИ**

***Институт Принтмедиа и информационных технологий***

***Кафедра Информатики и информационных технологий***

**направление подготовки**

**09.03.02 «Информационные системы и технологии»**

**ЛАБОРАТОРНАЯ РАБОТА № 4**

**Дисциплина: Программирование для мобильных устройств**

**Тема:** ««Элементы управления из класса CompoundButton. Класс ImageButton»

**Выполнил(а): студент(ка) группы \_\_211-722\_**

Дерендяев Дмитрий Сергеевич\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия И.О.)

**Дата, подпись** 10.10.2023\_\_\_\_  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Проверил: \_\_*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_***

(Фамилия И.О., степень, звание) **(Оценка)**

**Дата, подпись** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Москва**

**2023**

# Введение

**Целью** лабораторной работы является знакомство с различными классами

сложных кнопок (потомков CompoundButton), представленными на Android, а также с классом ImageButton.

# Разработка MainActivity.java

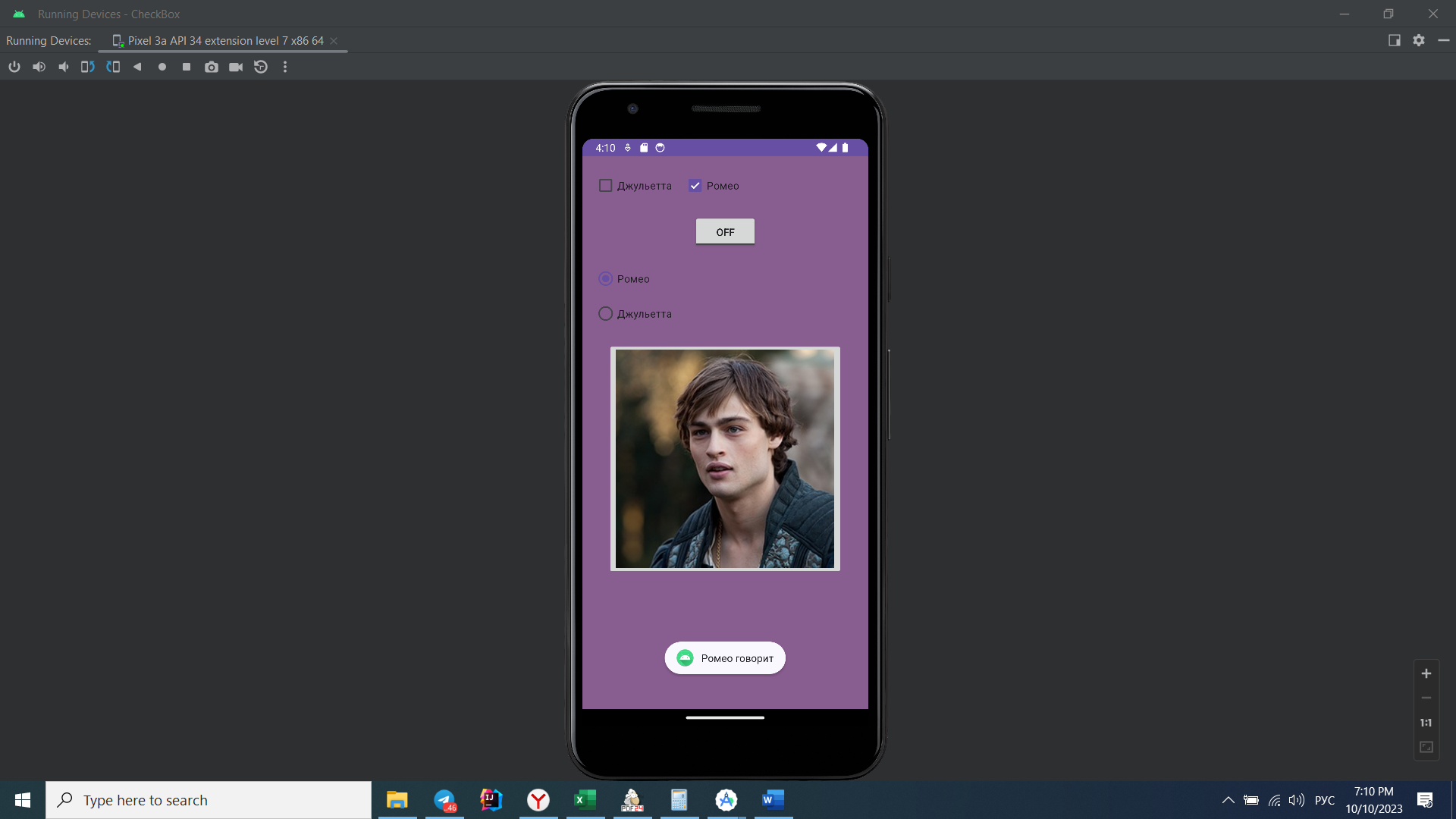
Реализуем подключение к кнопкам из activity\_main.xml и настроим обработчики событий:

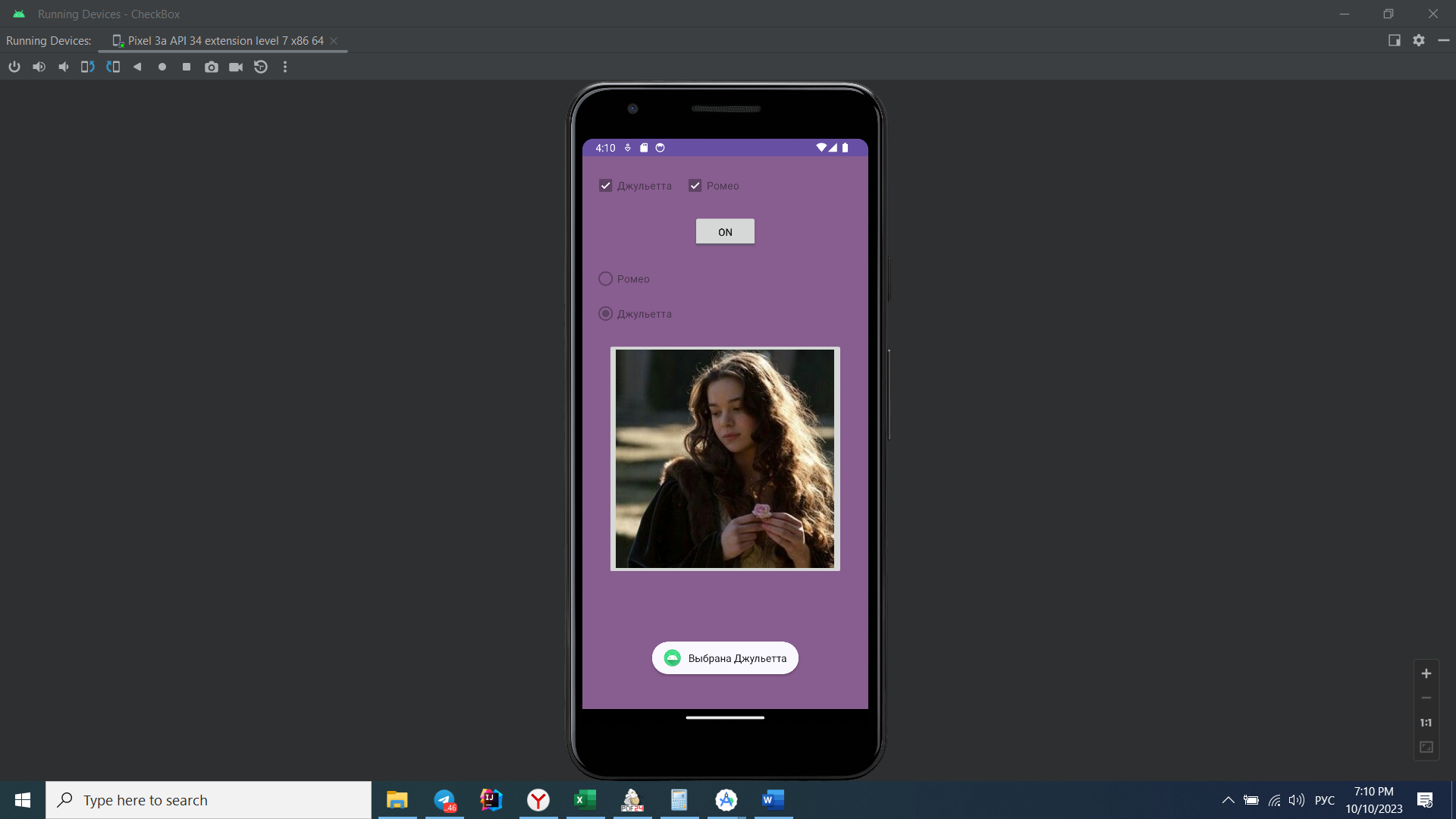
1. package com.example.checkbox;  
     
   import androidx.appcompat.app.AppCompatActivity;  
     
   import android.os.Bundle;  
   import android.view.View;  
   import android.widget.CheckBox;  
   import android.widget.CompoundButton;  
   import android.widget.ImageButton;  
   import android.widget.RadioButton;  
   import android.widget.RadioGroup;  
   import android.widget.Toast;  
   import android.widget.ToggleButton;  
     
   public class MainActivity extends AppCompatActivity {  
     
    private CheckBox checkBoxJuliet;  
    private CheckBox checkBoxRomeo;  
    private ToggleButton toggleButton;  
    private RadioGroup radioGroupCharacters;  
    private RadioButton radioRomeo;  
    private RadioButton radioJuliet;  
    private ImageButton imageButton;  
     
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.*activity\_main*);  
     
    checkBoxJuliet = findViewById(R.id.*checkbox\_juliet*);  
    checkBoxRomeo = findViewById(R.id.*checkbox\_romeo*);  
    toggleButton = findViewById(R.id.*toggle\_button*);  
    radioGroupCharacters = findViewById(R.id.*radio\_group\_characters*);  
    radioRomeo = findViewById(R.id.*radio\_romeo*);  
    radioJuliet = findViewById(R.id.*radio\_juliet*);  
    imageButton = findViewById(R.id.*image\_button*);  
     
    checkBoxJuliet.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {  
    @Override  
    public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {  
    if (isChecked) {  
    System.*out*.println("Джульетта говорит");  
    showToast("Джульетта говорит");  
    } else {  
    System.*out*.println("Джульетта молчит");  
    showToast("Джульетта молчит");  
    }  
    }  
    });  
     
    checkBoxRomeo.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {  
    @Override  
    public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {  
    if (isChecked) {  
    System.*out*.println("Ромео говорит");  
    showToast("Ромео говорит");  
    } else {  
    System.*out*.println("Ромео молчит");  
    showToast("Ромео молчит");  
    }  
    }  
    });  
     
    toggleButton.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
    boolean isEnabled = !toggleButton.isChecked();  
    checkBoxJuliet.setEnabled(isEnabled);  
    checkBoxRomeo.setEnabled(isEnabled);  
    radioRomeo.setEnabled(isEnabled);  
    radioJuliet.setEnabled(isEnabled);  
     
    if (isEnabled) {  
    showToast("Обработка включена");  
    } else {  
    showToast("Обработка выключена");  
    }  
    }  
    });  
     
    radioGroupCharacters.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {  
    @Override  
    public void onCheckedChanged(RadioGroup group, int checkedId) {  
    if (checkedId == R.id.*radio\_romeo*) {  
    showToast("Выбран Ромео");  
    imageButton.setImageResource(R.drawable.*romeo*);  
    } else if (checkedId == R.id.*radio\_juliet*) {  
    showToast("Выбрана Джульетта");  
    imageButton.setImageResource(R.drawable.*juliette*);  
    }  
    }  
    });  
     
    imageButton.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
    showToast("ImageButton нажата");  
    }  
    });  
    }  
     
    private void showToast(String message) {  
    Toast.*makeText*(this, message, Toast.*LENGTH\_SHORT*).show();  
    }  
   }

# Разработка activity\_main.xml

1. <?xml version="1.0" encoding="utf-8"?>  
   <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:layout\_width="match\_parent"  
    android:layout\_height="match\_parent"  
    tools:context=".MainActivity">  
     
    <RelativeLayout  
    android:layout\_width="match\_parent"  
    android:layout\_height="match\_parent"  
    android:background="#885D8F"  
    android:padding="16dp"  
    tools:context=".MainActivity">  
     
    <CheckBox  
    android:id="@+id/checkbox\_juliet"  
    android:layout\_width="wrap\_content"  
    android:layout\_height="wrap\_content"  
    android:layout\_alignParentStart="true"  
    android:layout\_alignParentTop="true"  
    android:text="Джульетта" />  
     
    <CheckBox  
    android:id="@+id/checkbox\_romeo"  
    android:layout\_width="wrap\_content"  
    android:layout\_height="wrap\_content"  
    android:layout\_alignParentTop="true"  
    android:layout\_marginStart="16dp"  
    android:layout\_toEndOf="@+id/checkbox\_juliet"  
    android:text="Ромео" />  
     
    <ToggleButton  
    android:id="@+id/toggle\_button"  
    android:layout\_width="wrap\_content"  
    android:layout\_height="wrap\_content"  
    android:layout\_below="@+id/checkbox\_juliet"  
    android:layout\_centerHorizontal="true"  
    android:layout\_marginTop="16dp"  
    android:text="Включить/Выключить" />  
     
    <RadioGroup  
    android:id="@+id/radio\_group\_characters"  
    android:layout\_width="match\_parent"  
    android:layout\_height="wrap\_content"  
    android:layout\_below="@+id/toggle\_button"  
    android:layout\_marginTop="16dp">  
     
    <RadioButton  
    android:id="@+id/radio\_romeo"  
    android:layout\_width="wrap\_content"  
    android:layout\_height="wrap\_content"  
    android:checked="false"  
    android:text="Ромео" />  
     
    <RadioButton  
    android:id="@+id/radio\_juliet"  
    android:layout\_width="wrap\_content"  
    android:layout\_height="wrap\_content"  
    android:checked="false"  
    android:text="Джульетта" />  
    </RadioGroup>  
     
    <ImageButton  
    android:id="@+id/image\_button"  
    android:layout\_width="wrap\_content"  
    android:layout\_height="wrap\_content"  
    android:layout\_below="@+id/radio\_group\_characters"  
    android:layout\_centerHorizontal="true"  
    android:layout\_marginTop="16dp"  
    android:src="@drawable/ic\_launcher\_foreground" />  
     
    </RelativeLayout>  
     
     
   </androidx.constraintlayout.widget.ConstraintLayout>

# Интерфейс системы





# Контрольные вопросы

1. Для программного отслеживания изменения состояния элемента CheckBox необходимо реализовать интерфейс **CompoundButton.OnCheckedChangeListener**. Этот интерфейс предоставляет метод **onCheckedChanged**, который вызывается при изменении состояния CheckBox.
2. Схожесть между CheckBox и RadioButton заключается в том, что они оба являются элементами выбора (они могут быть включены или выключены), и они используются для получения пользовательского ввода. Однако есть и различия:
   * CheckBox позволяет выбрать несколько вариантов одновременно, в то время как RadioButton позволяет выбрать только один вариант из группы RadioButton.
   * RadioButton используется, когда нужно предложить пользователю выбрать один вариант из нескольких (например, выбор пола), в то время как CheckBox используется, когда нужно предоставить пользователю возможность выбирать один или несколько вариантов (например, выбор нескольких предпочтительных опций).
3. Метод **isChecked** возвращает **true**, если элемент CheckBox включен (отмечен) пользователем, и **false**, если он выключен (не отмечен).
4. **ToggleButton** - это элемент управления, который имеет два состояния: включено и выключено (или другие два пользовательских состояния). При каждом нажатии на **ToggleButton** он меняет свое состояние между включенным и выключенным. Он часто используется для активации или деактивации какой-либо функциональности.
5. Метод **onCheckedChanged** вызывается, когда состояние элемента CheckBox или RadioButton изменяется пользователем. Этот метод передает информацию о состоянии элемента после изменения (включен или выключен) и предоставляет возможность обработать это изменение.
6. Виджет **ImageButton** отличается от **Button** тем, что он отображает изображение или иконку вместо текста. То есть, вместо того чтобы иметь текстовое содержание, как у **Button**, **ImageButton** имеет атрибут **android:src**, который определяет изображение, которое будет отображаться на кнопке. Это делает **ImageButton** удобным для создания кнопок с иконками.
7. Внутри **RadioGroup** может быть включено только одно **RadioButton** одновременно. Это свойство **RadioGroup** гарантирует, что выбор будет ограничен одним вариантом из группы **RadioButton**, и когда один **RadioButton** включается, другие автоматически выключаются.