



## 2.7. Элементы пользовательского интерфейса

Курс по программированию от  
IT Академии Samsung

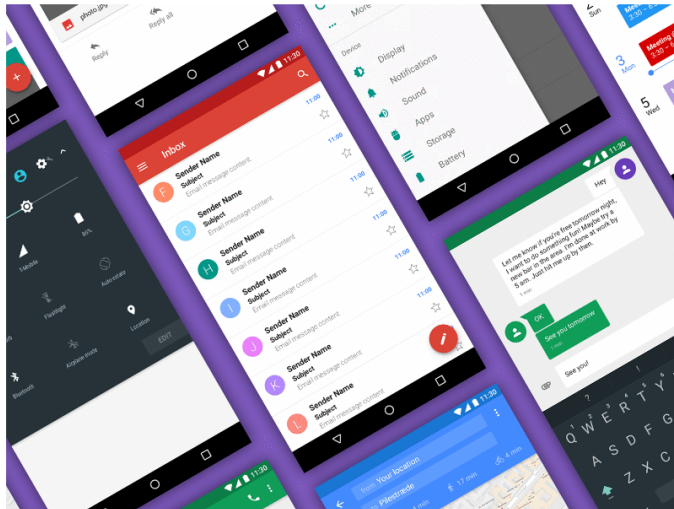


## 2.7.1 Продолжение знакомства с пользовательским интерфейсом

SAMSUNG



- Пользовательский интерфейс (UI) приложения – это ключевая составляющая в мобильной разработке. UI – это то, что пользователь видит на экране и с чем взаимодействует. Как вы уже знаете, средства Android-разработки предоставляют большое количество готовых решений для пользовательского интерфейса, которые позволяют создавать графический интерфейс приложений. Средствами Android можно создавать поля ввода, чекбоксы, радио кнопки, спиннеры, переключатели, всплывающие подсказки, всплывающие сообщения (pop-up message). В этом уроке приведем описание некоторых элементов UI.





## 2.7.2 Поля ввода EditText

SAMSUNG



```
<EditText android:id="@+id/plain_text_input"  
    android:layout_height="wrap_content"  
    android:layout_width="match_parent"  
    android:inputType="text"  
    android:inputType:@+id/textView1/>
```

Hello It Academy Samsung!

Hello It Academy Samsung!

- px — пиксели;
- dp — независимые от плотности пиксели. Это абстрактная единица измерения, основанная на физической плотности экрана;
- sp — независимые от масштабирования пиксели, т.е. зависят от пользовательских настроек размеров шрифтов;
- in — дюймы, базируются на физических размерах экрана;
- pt — 1/72 дюйма, базируются на физических размерах экрана;
- mm — миллиметры, также базируются на физических размерах экрана.

```
var editTextHello = findViewById(R.id.editTextHello)
```





## 2.7.3 Checkbox

SAMSUNG



```
<CheckBox  
    android:id="@+id/checkBox"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:padding="10dp"  
    android:text="@string/check_it"/>
```



Кликни меня

```
override fun onCreate(savedInstanceState: Bundle?) {  
    super.onCreate(savedInstanceState)  
    setContentView(R.layout.activity_main)  
    val checkBox = findViewById<CheckBox>(R.id.checkBox)  
    checkBox?.setOnCheckedChangeListener {  
        buttonView, isChecked -> val msg = "You have " + (if (isChecked) "checked" else "unchecked") + " Checkbox"  
        Toast.makeText(this@MainActivity, msg, Toast.LENGTH_SHORT).show()  
    }  
}
```



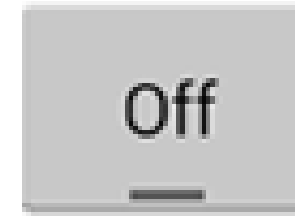


## 2.7.5 ToggleButton

SAMSUNG



```
<ToggleButton android:id="@+id/toggleButton2"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:textOff="NIGHT"  
    android:checked="true"  
    android:textOn="DAY" />
```



```
toggle_button.setOnCheckedChangeListener { buttonView, isChecked ->  
    if (isChecked) {  
        // The toggle is enabled/checked  
        Toast.makeText(applicationContext, "Toggle on", Toast.LENGTH_SHORT).show()  
    }  
    else {  
        // The toggle is disabled  
        Toast.makeText(applicationContext, "Toggle off", Toast.LENGTH_SHORT).show()  
    }  
}
```





## 2.7.4 Переключатели/включатели

SAMSUNG



☐ Нажми меня

☐ Нет меня

<RadioGroup

android:layout\_width="match\_parent"  
android:layout\_height="wrap\_content"  
android:orientation="vertical">

<RadioButton

android:id="@+id/b1"  
android:layout\_width="wrap\_content"  
android:layout\_height="wrap\_content"  
android:text="1"  
android:onClick="onClicked1"/>

<RadioButton

android:id="@+id/b2"  
android:layout\_width="wrap\_content"  
android:layout\_height="wrap\_content"  
android:text="2"  
android:onClick="onClicked2"/>

</RadioGroup>

```
fun onRadioButtonClicked(view: View) {  
    if (view is RadioButton) {  
        // Is the button now checked?  
        val checked = view.isChecked  
        // Check which radio button was clicked  
        when (view.getId()) {  
            R.id.b1 ->  
                if (checked) {  
                    // You pressed firstbtn  
                }  
            R.id.b2 ->  
                if (checked) {  
                    // You pressed second btn  
                }  
        }  
    }  
}
```



## 2.7.6 Всплывающие окна

SAMSUNG



- Всплывающее уведомление (Toast Notification) является сообщением, которое появляется на поверхности активности, заполняя необходимое ему количество пространства, требуемого для сообщения. При этом приложение не останавливается.



```
val text = "Hello!"  
val duration = Toast.LENGTH_SHORT  
val toast = Toast.makeText(applicationContext, text, duration)  
toast.show()
```



```
Toast.makeText(applicationContext, text, duration).show()
```

- LENGTH\_SHORT — показывает уведомление на короткий промежуток времени;
- LENGTH\_LONG — показывает уведомление в течение длительного периода времени.

```
toast.setGravity(Gravity.TOP or Gravity.LEFT, 0, 0)
```





## 2.7.8 Всплывающие сообщения (pop-up message)

SAMSUNG

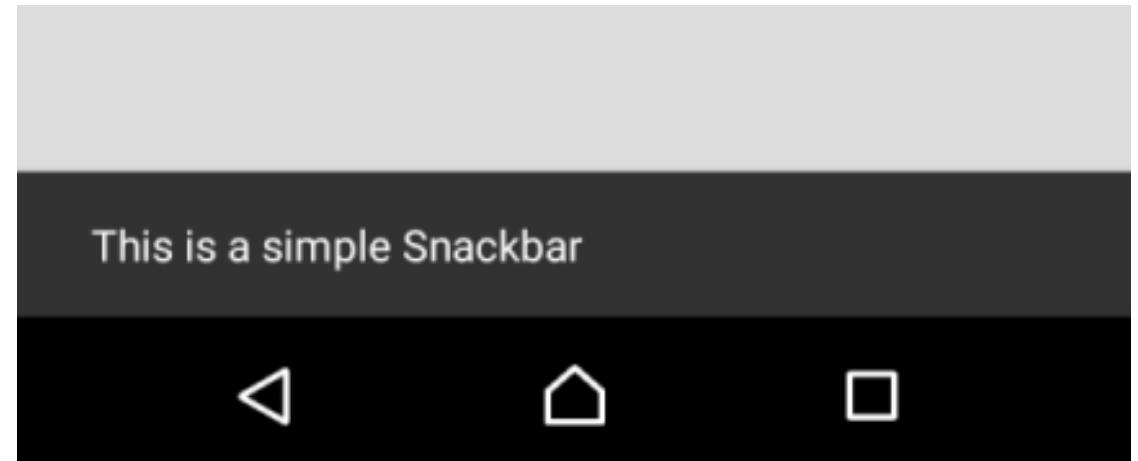


Snackbar предоставляет пользователю быстрое всплывающее сообщение (pop-up message). При этом работа с приложением продолжается. Через некоторое время Snackbar исчезает автоматически



```
android {  
    compileSdkVersion 26  
    ...  
}  
  
dependencies {  
    ...  
    implementation  
    'com.android.support:design:26.1.0'  
    ...  
}
```

```
val snack = Snackbar.make(it, "This is a simple Snackbar",  
                           Snackbar.LENGTH_LONG)  
snack.show()
```







## Упражнение 2.7

SAMSUNG



- Пусть у нас имеется поле для ввода, две RadioButton, TuggleButton и чекбокс. С помощью радиокнопок будем вставлять в текстовое поле разный текст, с помощью TuggleButton менять задний фон EditText, а с помощью CheckBox показывать Toast-сообщение - активна она или нет.



**SAMSUNG**



**Спасибо за внимание!**

Курс по программированию от  
IT Академии Samsung