**Использование командных элементов управления**

**Кнопки-переключатели**

В Android существуют несколько видов кнопок-переключателей, например: СheckBox, Switch или ToggleButton. Все эти элементы при нажатии меняют свое состояние, меняется изображение или текст на кнопке.

**CheckBox**

CheckBox или Флажок это кнопка с галочкой и поясняющим текстом. CheckBox используется при представлении пользователям группы выбираемых параметров, которые не являются взаимоисключающими. Качественный текст может исключить возможность неоднозначного толкования флажка. Однако этот же поясняющий текст вынуждает пользователя замедляться для прочтения, а также занимает значительное экранное пространство.

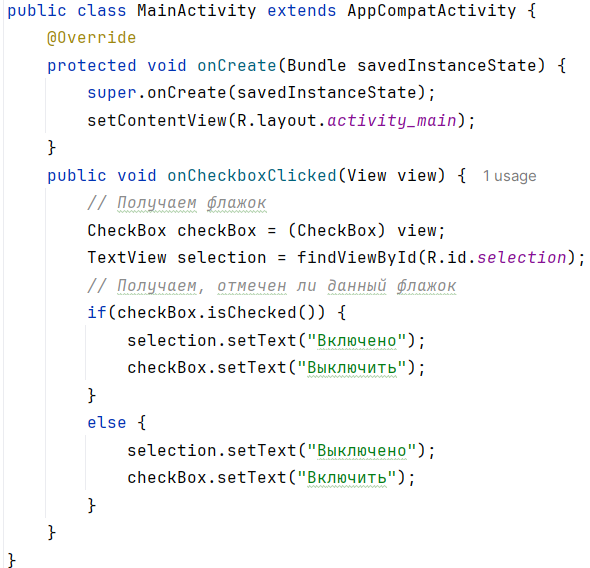
В Android флажки используются очень часто. Например, для проведения опроса в приложении для Android мы можем перечислить несколько вариантов и разрешить пользователю выбирать с помощью флажка. Пользователь просто установит эти флажки, а не введет свой собственный вариант в EditText. Еще одно очень распространенное использование флажка - это опция “запомнить меня” в форме входа в систему.

Традиционно флажки имеют квадратную форму. Пользователи распознают визуальные объекты по форме, и квадратная форма флажков - важный стандарт.

Для создания флажков используется элемент **Checkbox**



Атрибут **android:onClick**, как и в случае с простыми кнопками, позволяет задать обработчик нажатия на флажок. Определим обработчик нажатия в коде **MainActivity**:



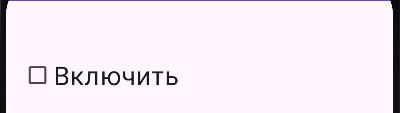
В качестве параметра в обработчик нажатия **onCheckboxClicked** передается нажатый флажок. Обработчик срабатывает при каждом нажатии на **checkBox**. То есть и когда мы устанавливаем флажок, и когда мы снимем отметку. С помощью метода **isChecked()** можно узнать, выделен ли флажок - в этом случае метод возвращает true.

Так же мы можем использовать слушатель **OnCheckedChangeListener**. Этот метод срабатывает, когда мы устанавливаем или убираем отметку на флажке.



В коде MainActivity подключим обработчик изменения состояния:







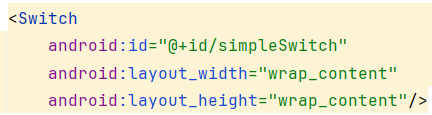
Вы можете проверить текущее состояние флажка программно, используя метод IsChecked(). Этот метод возвращает логическое значение true или false, если флажок установлен, то он возвращает значение true, в противном случае возвращает значение false.



**Switch**

Switch или Тумблер это кнопка-переключатель использующийся для отображения включенного и выключенного состояния, которая указывает текущее состояние переключателя.

Переключатель позволяет пользователям переключать настройки между двумя состояниями, например, включать / выключать Wi-Fi, Bluetooth и т.д., в меню настроек вашего телефона.





**ToggleButton**

ToggleButton или Триггеры используется для отображения состояния кнопки "включено" и "не включено".

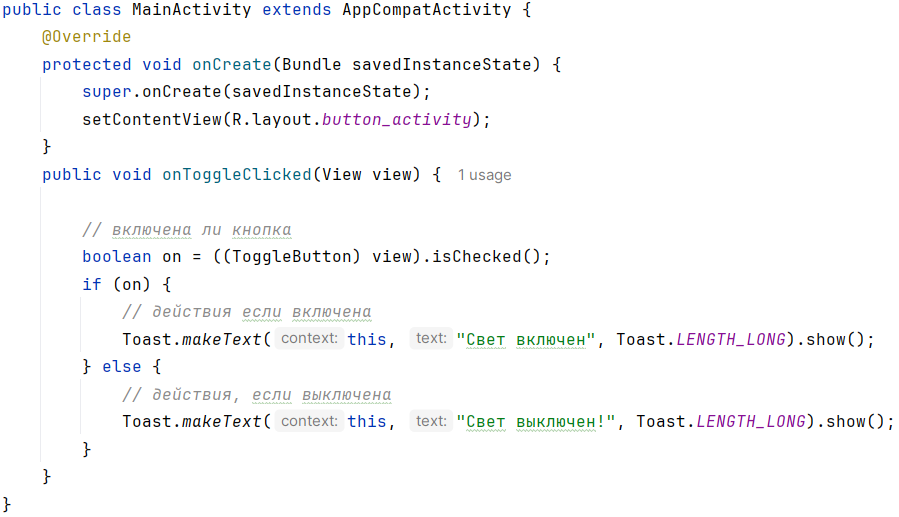
Они призваны экономить экранное пространство, к сожалению, ценой значительной дезориентации пользователя. Классический пример - размещение на одной кнопке функций воспроизведения и паузы для музыкального проигрывателя. Подводным камнем такого подхода является то, что элемент управления можно ошибочно посчитать индикатором состояния проигрывателя ("на паузе" или "идет воспроизведение").



Для создания триггеров используется элемент **ToggleButton**

****

Атрибуты **android:textOn** и **android:textOff** задают текст кнопки в отмеченном и не отмеченном состоянии соответственно. И также, как и для других кнопок, мы можем обработать нажатие на элемент с помощью события **onClick**. В этом случае определим в классе Activity обработчик события:







**RadioButton и RadioGroup**

RadioButton или Радиокнопка в основном используются в RadioGroup. В RadioGroup активация одного переключателя из нескольких переключателей, автоматически снимает флажки со всех остальных. Это означает, что одновременно мы можем установить только один переключатель из группы переключателей, принадлежащих к одной RadioGroup.

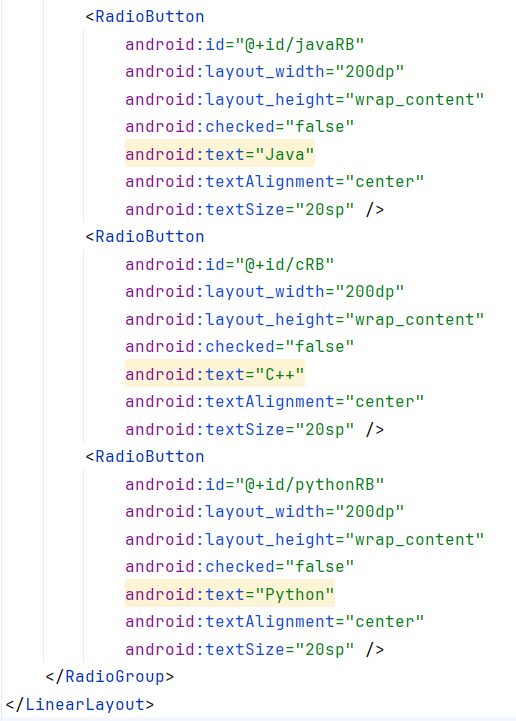
Радиокнопки внешне похожи на флажки, но являются взаимоисключающими, то есть выбор одного из вариантов автоматически аннулирует предыдущий выбор. В каждый момент времени может быть выбрана только одна кнопка. Радиокнопки всегда объединяются в группы из двух или более радиокнопок, причем в каждой группе одна радиокнопка всегда выбрана. Радиокнопки всегда круглые по той же причине, по которой флажки всегда имеют квадратную форму: именно такими они были изначально.

Радиокнопки занимают даже больше места, чем флажки, однако в некоторых случаях такой расход экранного пространства оправдан.

Для создания радиокнопок используется элемент **RadioButton**

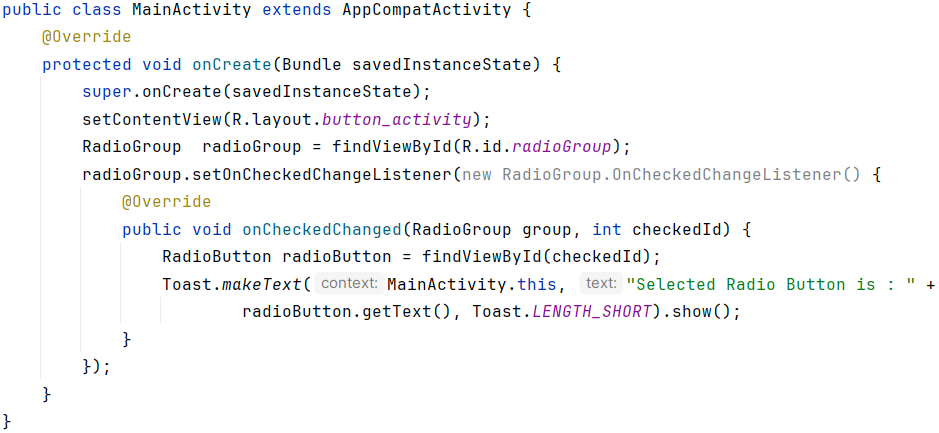
Чтобы создать список переключателей для выбора, вначале надо создать объект **RadioGroup**, который будет включать в себя все переключатели:





Вы можете обрабатывать нажатие на каждую радиокнопку как на обычную кнопку, но и установить слушатель OnCheckedChangeListener на RadioGroup.

В классе **MainActivity** определим обработку выбора переключателей:



Слушатель **RadioGroup.OnCheckedChangeListener** определяет метод **onCheckedChanged()**, в который передается объект **RadioGroup** и **id** выделенного переключателя. Далее также вы можете проверить **id** и выполнить определенную обработку.



**Задание для самостоятельной работы:**

1. Создайте приложение по заданному макету используя изученные элементы интерфейса.

