

На учебной практике вам предстоит решить несколько заданий. Весь неизвестный функционал, который будет использоваться в представленном коде, на учебной практике является материалом для самостоятельного изучения

#### Задание 4. Дополнение

##### Ход выполнения работы:

##### 1 способ создания выпадающего списка

1. Проверьте тему вашего приложения

```
3 <style name="Base.Theme.DarkT" parent="Theme.Material3.Light.NoActionBar">
```

2. Создайте новый файл разметки (только разметки) и пропишите следующий код:

```
list_item.xml x
1 <?xml version="1.0" encoding="utf-8"?>
2 <TextView xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="wrap_content"
4     android:layout_height="wrap_content"
5     android:padding="16dp"
6     android:ellipsize="end"
7     android:maxLines="1"
8     android:textAppearance="@style/TextAppearance.MaterialComponents.Subtitle1"
9     >
10 </TextView>
```

*ellipsize - ставит три точки, если текст не помещается в поле*

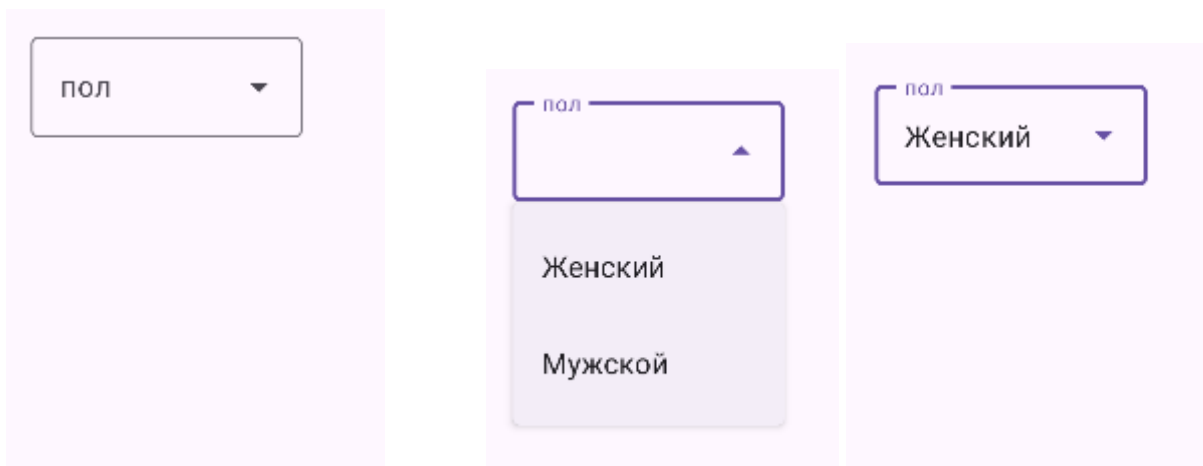
3. Измените тип текстового поля (если вы до этого уже реализовали скрывание пароля, то ваши текстовые поля уже оформлены так как нужно для реализации выпадающего списка):

```
9 <com.google.android.material.textfield.TextInputLayout
10     android:layout_width="wrap_content"
11     android:layout_height="wrap_content"
12     android:layout_margin="16dp"
13     style="@style/Widget.Material3.TextInputLayout.OutlinedBox.ExposedDropdownMenu">
14     <AutoCompleteTextView
15         android:id="@+id/dropdown_list"
16         android:layout_width="match_parent"
17         android:layout_height="wrap_content"
18         android:hint="пол"
19         android:inputType="none"/>
20 </com.google.android.material.textfield.TextInputLayout>
21 </androidx.appcompat.widget.LinearLayoutCompat>
```

- В файле логики экрана создайте список пунктов, которые будет содержать выпадающий список.
- Используйте биндинги или команду поиска по ID для изменения компонента разметки autoCompleteTextView через программный код.
- Создайте переменную adapter, которая свяжет содержимое списка с разметкой, созданной ранее.
- Назначьте текстовому полю адаптер.

```
class MainActivity2 : AppCompatActivity() {
    lateinit var binding: ActivityMain2Binding
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        binding = ActivityMain2Binding.inflate(layoutInflater)
        setContentView(binding.root)
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
            val systemBars : Insets = insets.getInsets(WindowInsetsCompat.Type.systemBars())
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)
            insets
        }
        val items : List<String> = listOf("Женский", "Мужской")
        val adapter : ArrayAdapter<String!> = ArrayAdapter<String!>(context: this, R.layout.list_item, items)
        binding.dropdownList.setAdapter(adapter)
    }
}
```

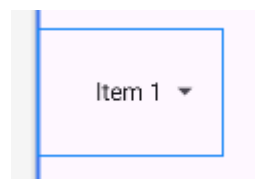
- 
- Протестируйте как работает:



## 2 способ создания выпадающего списка

- Настройте разметку, добавьте новый компонент:

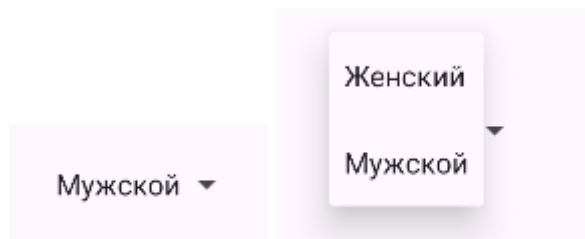
```
<Spinner
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/sp"
    android:padding="30dp"/>
```



- Создайте массив значений, которые будет отображать Spinner

12.Создайте адаптер (мост, между элементом пользовательского интерфейса и источником данных. Он преобразует данные из источника в элемент просмотра, чтобы их можно было отобразить)

```
val adapter2 : ArrayAdapter<String!> =ArrayAdapter( context: this, com.google.android.material.R.layout.support_simple_spinner_dropdown_item, items)  
binding.sp.adapter = adapter2
```



13.Помимо указанных, существуют еще способы для создания выпадающих списков. Вы можете использовать представленные или найти свой).