

Практическая работа 1. Создание сворачиваемой панели инструментов

Вы знаете, как заставить панель инструментов уходить с экрана. Сделаем следующий шаг и заменим ее несколько иной разновидностью панелей инструментов: сворачиваемой панелью инструментов. Сворачиваемая панель инструментов изначально имеет обычные размеры, уменьшается при прокрутке экрана вверх и снова принимает нормальные размеры при прокрутке вниз. К ней даже можно добавить изображение, которое исчезает при достижении панелью минимальной высоты и снова становится видимым при ее расширении:

На нескольких ближайших страницах вы узнаете, как превратить простую панель инструментов в сворачиваемую, для этого нужно добавить ее в макет `OrderFragment`. Начнем с создания простой сворачиваемой панели инструментов, а затем создадим панель с графическим изображением. Итак, за дело.

Создание простой сворачиваемой панели инструментов

Панель инструментов с прокруткой относительно просто преобразуется в сворачиваемую панель инструментов. Для этого достаточно заключить панель инструментов в макет сворачиваемой панели инструментов и настроить атрибуты панели инструментов. Базовая структура кода выглядит так:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout ...>
  <com.google.android.material.appbar.AppBarLayout ...>

    <com.google.android.material.appbar.CollapsingToolbarLayout
      android:layout_width="match_parent"
      android:layout_height="300dp"
      app:layout_scrollFlags="scroll|exitUntilCollapsed">

      <com.google.android.material.appbar.MaterialToolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="?attr/actionBarSize"
        app:layout_collapseMode="pin"/>
      </com.google.android.material.appbar.CollapsingToolbarLayout>
    </com.google.android.material.appbar.AppBarLayout>

    ...
  </androidx.coordinatorlayout.widget.CoordinatorLayout>
```

Как видите, макет сворачиваемой панели инструментов определяется элементом

`<...CollapsingToolbarLayout>` из библиотеки `com.google.android.material`. Максимальная высота определяется его атрибутом `layout_height`, а строка:

```
app:layout_scrollFlags="scroll|exitUntilCollapsed"
```

отдает команду панели сворачиваться при прокрутке вверх, пока остается возможность для уменьшения, и расширяться при прокрутке вниз, пока панель не достигнет своей полной высоты. Также необходимо позаботиться о том, чтобы при сворачивании панели инструментов все, что на ней отображается, оставалось на экране (например, кнопка **Up** и любые элементы меню). Для этого в элемент `Toolbar` добавляется следующий атрибут:

```
app:layout_collapseMode="pin"
```

Как добавить изображение к сворачиваемой панели инструментов

После того как вы создадите простую сворачиваемую панель инструментов, к ней можно добавить изображение — для этого добавьте в макет сворачиваемой панели инструментов элемент `<ImageView>` и укажите изображение, которое вы хотите использовать. Код имеет следующую структуру:

```
<androidx.coordinatorlayout.widget.CoordinatorLayout ...>
  <com.google.android.material.appbar.AppBarLayout ...>
    <com.google.android.material.appbar.CollapsingToolbarLayout
      android:layout_width="match_parent"
      android:layout_height="300dp"
      app:layout_scrollFlags="scroll|exitUntilCollapsed"
      app:contentScrim="?attr/colorPrimary">
      <ImageView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:scaleType="centerCrop"
        android:src="@drawable/image"
        app:layout_collapseMode="parallax"/>
      < com.google.android.material.appbar.MaterialToolbar .../>
    </com.google.android.material.appbar.CollapsingToolbarLayout>
  </com.google.android.material.appbar.AppBarLayout>
  ...
</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

Строка

```
app:contentScrim="?attr/colorPrimary"
```

добавленная в `<...CollapsingToolbarLayout>`, назначает панели инструментов простой цвет фона в свернутом состоянии. Также к изображению была добавлена параллаксная анимация, для чего использовалась следующая строка:

```
app:layout_collapseMode="parallax"
```

Этот атрибут не является обязательным; просто с ним изображение прокручивается со скоростью, отличающейся от скорости прокрутки остальных частей панели инструментов. Итак, теперь вы знаете, как создаются сворачиваемые панели инструментов. Добавим такую панель в макет `OrderFragment`.

Добавление изображения

Мы хотим, чтобы сворачиваемая панель инструментов `OrderFragment` включала изображение ресторана, поэтому давайте начнем с добавления ее в проект.

Убедитесь в том, что в проекте присутствует папка с именем `app/src/main/res/drawable`, затем загрузите файл `restaurant.webp` и добавьте его в папку `drawable`. Тем самым вы включаете изображение в свой проект как графический ресурс. А теперь добавим сворачиваемую панель инструментов.

Полный код `fragment_order.xml`

Приведенный ниже код добавляет сворачиваемую панель инструментов в макет `OrderFragment`. Обновите код `fragment_order.xml`:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    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=".OrderFragment">
    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/ThemeOverlay.MaterialComponents.Dark.ActionBar">
        <com.google.android.material.appbar.CollapsingToolbarLayout
            android:layout_width="match_parent"
            android:layout_height="300dp"
            app:layout_scrollFlags="scroll|exitUntilCollapsed"
            app:contentScrim="?attr/colorPrimary">
            <ImageView
                android:layout_width="match_parent"
```

```
        android:layout_height="match_parent"
        android:scaleType="centerCrop"
        android:src="@drawable/restaurant"
        app:layout_collapseMode="parallax"/>
    <com.google.android.material.appbar.MaterialToolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="?attr/actionBarSize"
        app:layout_scrollFlags="scroll|enterAlways"
        app:layout_collapseMode="pin" />
    </com.google.android.material.appbar.CollapsingToolbarLayout>
</com.google.android.material.appbar.AppBarLayout>
<androidx.core.widget.NestedScrollView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    app:layout_behavior="@string/appbar_scrolling_view_behavior">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="This is the order fragment" />
    </androidx.core.widget.NestedScrollView>
</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

И это все изменения, которые были необходимы для создания сворачиваемой панели инструментов. Проведем тест-драйв приложения и посмотрим, как оно выглядит.

При запуске приложения отображается фрагмент `OrderFragment`. Он включает сворачиваемую панель инструментов с изображением. При прокрутке вверх панель инструментов сворачивается, изображение растворяется, а фон панели инструментов заменяется основным цветом приложения. При прокрутке вниз панель расширяется и изображение появляется снова.



Alt text