

● TabLayout: 탭 화면에서 탭 버튼을 다양하게 구성하기 위한 뷰이다. 단순히 탭 버튼을 동일한 간격으로 나열하는 것뿐만 아니라, 탭 버튼이 많을 때는 스크롤을 이용할 수도 있고, 다양한 방식으로 정렬할 수 있다. 또한 TabLayout 과 ViewPager2를 연동할 수도 있다.

```
// activity_main.xml

<androidx.constraintlayout.widget.ConstraintLayout>
    <com.google.android.material.tabs.TabLayout />
    <FrameLayout>
    </FrameLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

탭 화면은 TabLayout으로, 탭 버튼을 클릭했을 때 출력할 화면은 FrameLayout을 사용하였다. TabLayout의 탭 버튼이 클릭되었을 때 나오는 화면은 FrameLayout 이외에 어떤 뷰를 이용해도 상관없지만, 탭 화면의 특성상 탭 버튼이 나올 때 하나의 뷰만 출력하는 것이 일반적이므로 대부분 FrameLayout을 이용한다.

▶ 탭 버튼 구현하기

1) 코드로 탭 버튼 구성하기

```
// MainActivity.java

TabLayout.Tab tab1 = binding.tabLayout.newTab();
tab1.setText("tab1");
binding.tabLayout.addTab(tab1);

TabLayout.Tab tab1 = binding.tabLayout.newTab();
tab1.setText("tab2");
binding.tabLayout.addTab(tab2);

TabLayout.Tab tab1 = binding.tabLayout.newTab();
tab1.setText("tab3");
binding.tabLayout.addTab(tab3);
```

Tab은 TabLayout에 들어갈 탭 버튼 하나를 지칭하는 클래스이며, newTab()함수로 만든다. Tab 객체에 setText(), setIcon() 함수로 버튼을 구성하는 문자열과 이미지를 지정한다. 그리고 이렇게 만들어진 Tab 객체를 addTab() 함수의 매개변수로 전달하면, 탭 버튼이 TabLayout에 추가된다.

2) XML 파일에서 탭 버튼 구현하기: 대부분의 탭 버튼은 정적으로 구성된다. TabLayout 하위에 TabItem만 구성하면 된다.

☞ tabMode: 탭 버튼이 많을 때 스크롤을 사용할 수 있게 해주는 속성이다. 기본 값은 스크롤을 지원하지 않는다는 의미의 fixed로, 지원하고자 한다면 이 값을 scrolable로 지정해 주면 된다. scrollable로 지정하면 탭 버튼이 왼쪽부터 배치되며, 탭 버튼이 많아 화면을 벗어나면 자동으로 스크롤을 사용해 탭 화면을 움직일 수 있도록 해준다.

☞ tabgravity: 정렬 위치 지정

```
// activity_main.xml

<com.google.android.material.tabs.TabLayout
    android:id="@+id/tabLayout"
    android:layout_width="409dp"
    android:layout_height="wrap_content"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:tabMode="scrollable"
    app:tabGravity="center">

    <com.google.android.material.tabs.TabItem
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="tab1" />

    <com.google.android.material.tabs.TabItem
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="tab2" />

    <com.google.android.material.tabs.TabItem
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="tab3" />

</com.google.android.material.tabs.TabLayout>
```

- ▶ 탭 버튼을 클릭했을 때 원하는 화면 나오게 하기: OnTabSelectedListener

```
binding.tabLayout.addTabSelectedListener(new TabLayout.OnTabSelectedListener() {  
    // 탭 버튼을 선택할 때 이벤트  
    @Override  
    public void onTabSelected(TabLayout.Tab tab) {  
        FragmentTransaction transaction = getSupportFragmentManager().beginTransaction()  
        if (tab.getPosition() == 0) {  
            transaction.replace(R.id.contentLayout, oneFragment);  
        } else if (tab.getPosition() == 1) {  
            transaction.replace(R.id.contentLayout, twoFragment);  
        } else if (tab.getPosition() == 2) {  
            transaction.replace(R.id.contentLayout, threeFragment);  
        }  
        transaction.commit()  
    }  
  
    // 다른 탭 버튼을 눌러 선택된 탭 버튼이 해제될 때 이벤트  
    @Override  
    public void onTabUnselected(TabLayout.Tab tab) {  
  
    }  
  
    // 선택된 탭 버튼을 다시 선택할 때 이벤트  
    @Override  
    public void onTabReselected(TabLayout.Tab tab) {  
  
    }  
});
```



```
// activity_main.xml

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
    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">

    <com.google.android.material.tabs.TabLayout
        android:id="@+id/tabLayout"
        android:layout_width="409dp"
        android:layout_height="wrap_content"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:tabMode="scrollable"
    >

        <com.google.android.material.tabs.TabItem
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="tab1" />

        <com.google.android.material.tabs.TabItem
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="tab2" />

        <com.google.android.material.tabs.TabItem
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="tab3" />

    </com.google.android.material.tabs.TabLayout>

    <FrameLayout
        android:layout_width="409dp"
        android:layout_height="681dp"
        android:id="@+id/contentLayout"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/tabLayout">

    </FrameLayout>

</androidx.constraintlayout.widget.ConstraintLayout>
```

```

// MainActivity.java

package com.android.practicetablayout;

import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.FragmentTransaction;

import android.os.Bundle;

import com.android.practicetablayout.databinding.ActivityMainBinding;
import com.google.android.material.tabs.TabLayout;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ActivityMainBinding binding = ActivityMainBinding.inflate(getLayoutInflater());
        setContentView(binding.getRoot());

        OneFragment oneFragment = new OneFragment();
        TwoFragment twoFragment = new TwoFragment();
        ThreeFragment threeFragment = new ThreeFragment();

        FragmentTransaction transaction = getSupportFragmentManager().beginTransaction()
        transaction.add(R.id.contentLayout, oneFragment);
        transaction.commit();

        binding.tabLayout.addOnTabSelectedListener(new TabLayout.OnTabSelectedListener()
        // 탭 버튼을 선택할 때 이벤트
        @Override
        public void onTabSelected(TabLayout.Tab tab) {
            FragmentTransaction transaction = getSupportFragmentManager().beginTransaction()
            if (tab.getPosition() == 0) {
                transaction.replace(R.id.contentLayout, oneFragment);
            } else if (tab.getPosition() == 1) {
                transaction.replace(R.id.contentLayout, twoFragment);
            } else if (tab.getPosition() == 2) {
                transaction.replace(R.id.contentLayout, threeFragment);
            }
            transaction.commit();
        }

        // 다른 탭 버튼을 눌러 선택된 탭 버튼이 해제될 때 이벤트
        @Override
        public void onTabUnselected(TabLayout.Tab tab) {

        }

        // 선택된 탭 버튼을 다시 선택할 때 이벤트
        @Override
        public void onTabReselected(TabLayout.Tab tab) {

        }
    });
}

```
