

스마트 앱 프로젝트

Navigation Drawer



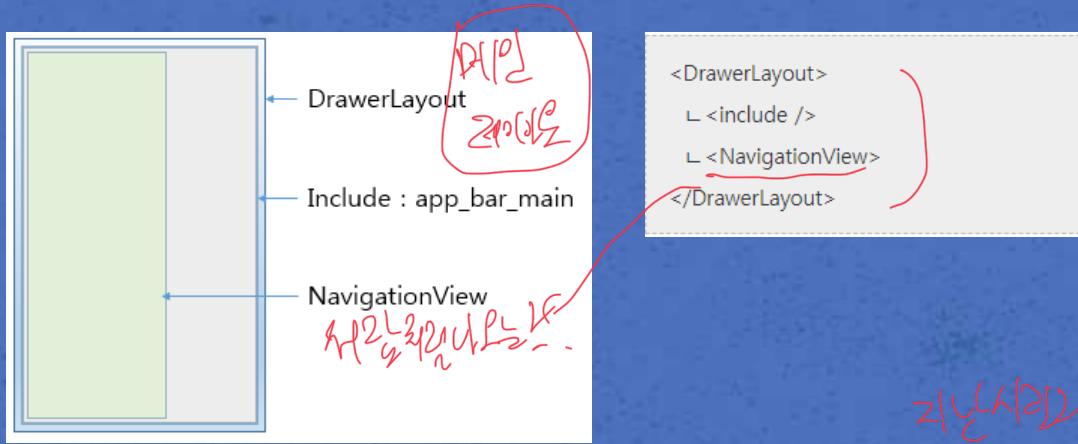
Navigation Drawer

- Navigation
 - 사용자가 앱 내의 여러 콘텐츠를 탐색하고, 그곳에 들어갔다 나올 수 있게 하는 상호작용을 의미
 - Navigation Component : 안드로이드 앱 안에서 화면 간의 이동을 도와주는 구성 요소
- Navigation Drawer
 - 앱 내의 컨텐츠간 탐색을 위하여 메뉴를 제공하는 화면 배울 당시
 - DrawerLayout / NavigationView
 - NavigationUI / AppBarConfiguration 이전 내용
 - NavController / ViewModel



% Navigation Drawer

- 구성원리

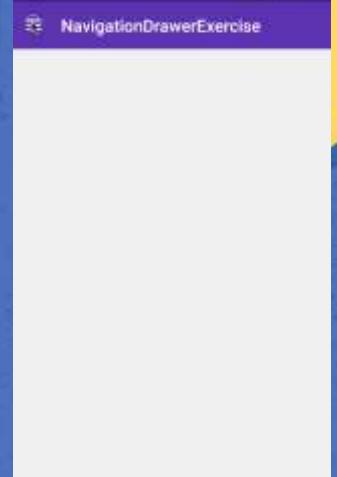


- DrawerLayout : 메인 뷰와 NavigationView를 감싸는 구조
- 메인뷰 : fragment로 구성
 - Navigation을 위한 Navigation Graph 구성
 - Fragment 클래스를 상속하여 각 화면을 위한 Fragment 클래스 정의
- NavigationView : 이동할 화면을 선택할 수 있는 탐색 menu 파일(xml 형식)이 필요. menu의 <item>의 id와 Navigation Graph의 이동할 목적지 fragment의 id가 같아야 함.
- AppBarConfiguration 객체와 NavigationUI 객체를 이용하여 탐색 메뉴의 동작 관리

지난시간과 다음
흐름 연결

Navigation Drawer

- Navigation Drawer 구현 절차
 1. 각 화면별 layout 및 Fragment 클래스 *(시작과 끝 내용)*
 2. Navigation Graph 생성
 3. DrawerLayout과 NavigationView를 사용하여 메인화면 레이아웃 구성하고 Nav Host 설정
 4. Navigation 메뉴 설정
 5. AppBarConfiguration과 NavigationUI 객체를 사용하여 *탐색 동작 설정*



Navigation Drawer 구현 절차

1. 각 화면별 layout 및 Fragment 클래스 정의

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    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=".ui.slideshow.SlideshowFragment">

    <TextView
        android:id="@+id/text_slideshow"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="8dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="8dp"
        android:textAlignment="center"
        android:textSize="20sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

```
public class SlideshowFragment extends Fragment {

    public View onCreateView(@NonNull LayoutInflater inflater,
                           ViewGroup container, Bundle savedInstanceState) {
        View root = inflater.inflate(R.layout.fragment_slideshow, container, false);
        return root;
    }

    public void onViewCreated(@NonNull View view, Bundle savedInstanceState) {
        super.onViewCreated(view, savedInstanceState);
        view.findViewById(R.id.text_slideshow).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Toast.makeText(getContext(), "slideshow", Toast.LENGTH_LONG).show();
            }
        });
    }
}
```

(D) 허여경 / 부별정동재미

Navigation Drawer 구현 절차

2. Navigation Graph 생성

- navigation id : Nav Host 설정시 사용
- startDestination : 시작 화면
- fragment id : navigation menu에서 사용
- name : fragment class
- label : 화면 이동시 앱바에서 보여줄 화면 타이틀
- layout : 화면 구성 정보 명시한 레이아웃

```
<navigation 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:id="@+id/mobile_navigation"  
    app:startDestination="@+id/nav_home">  
  
    <fragment  
        android:id="@+id/nav_home"  
        android:name="com.example.myapplication.ui.home.HomeFragment"  
        android:label="Home"  
        tools:layout="@layout/fragment_home" >  
    </fragment>  
  
    <fragment  
        android:id="@+id/nav_gallery"  
        android:name="com.example.myapplication.ui.gallery.GalleryFragment"  
        android:label="Gallery"  
        tools:layout="@layout/fragment_gallery" />  
  
    <fragment  
        android:id="@+id/nav_slideshow"  
        android:name="com.example.myapplication.ui.slideshow.SlideshowFragment"  
        android:label="Slideshow"  
        tools:layout="@layout/fragment_slideshow" />  
</navigation>
```

NavigationGraph에요? ?

HomeFragment는 헬퍼필터 X

설정은 이동할 때 온다.

HomeFragment는 표시될 때 온다.

GalleryFragment는 표시될 때 온다.

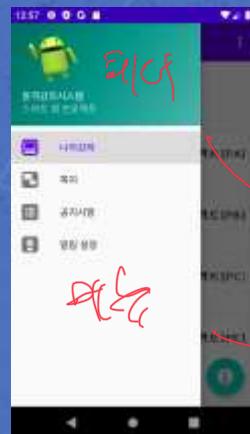
SlideshowFragment는 표시될 때 온다.

Navigation Drawer 구현 절차

2019.5.21

3.1 DrawerLayout과 NavigationView로 전체 레이아웃 구성 (activity_main.xml)

- fitsSystemWindows
- openDrawer
- layout_gravity
- headerLayout
- menu



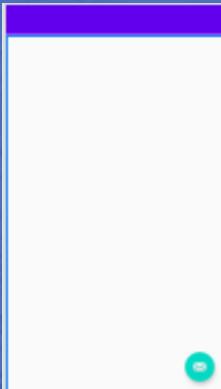
```
<androidx.drawerlayout.widget.DrawerLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fitsSystemWindows="true" 사각형 깊이가
    tools:openDrawer="start" 여기까지 디자인상 깊이가 3

    <include
        layout="@layout/app_bar_main"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

    <com.google.android.material.navigation.NavigationView
        android:id="@+id/nav_view"
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:layout_gravity="start" =left 왼쪽에 위치
        android:fitsSystemWindows="true"
        app:headerLayout="@layout/nav_header_main"
        app:menu="@menu/activity_main_drawer" />
</androidx.drawerlayout.widget.DrawerLayout>
```

Navigation Drawer 구현 절차

3.2 MainLayout(app_bar_main)과 Nav Host 구성



```
<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=".MainActivity">  
    <com.google.android.material.appbar.AppBarLayout  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:theme="@style/AppTheme.AppBarOverlay">  
            <androidx.appcompat.widget.Toolbar  
                android:id="@+id/toolbar"  
                android:layout_width="match_parent"  
                android:layout_height="?attr/actionBarSize"  
                android:background="?attr/colorPrimary"  
                app:popupTheme="@style/AppTheme.PopupOverlay" />  
        </com.google.android.material.appbar.AppBarLayout>  
  
        <include layout="@layout/content_main" />  
  
        <com.google.android.material.floatingactionbutton.FloatingActionButton  
            android:id="@+id/fab"  
            android:layout_width="wrap_content"  
            android:layout_height="wrap_content"  
            android:layout_gravity="bottom|end"  
            android:layout_margin="16dp"  
            app:srcCompat="@android:drawable/ic_dialog_email" />  
    </androidx.coordinatorlayout.widget.CoordinatorLayout>
```

```
<androidx.constraintlayout.widget.ConstraintLayout  
    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"  
    app:layout_behavior="@string/appbar_scrolling_view_behavior"  
    tools:showIn="@layout/app_bar_main">  
  
<fragment  
    android:id="@+id/nav_host_fragment"  
    android:name="androidx.navigation.fragment.NavHostFragment"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    app:defaultNavHost="true"  
    app:layout_constraintLeft_toLeftOf="parent"  
    app:layout_constraintRight_toRightOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    app:navGraph="@navigation/mobile_navigation" />  
</androidx.constraintlayout.widget.ConstraintLayout>
```

(이거 고우지)

Navigation Drawer 구현 절차

4. Navigation 메뉴 설정

- <menu> : navigation 메뉴의 최상위 엘리먼트
- <group>
 - <item> 엘리먼트들을 묶어서 한꺼번에 속성을 부여할 때 사용
 - checkableBehavior 속성 : 하나씩만 선택 가능하도록 single로 설정
- <item>
 - id 속성 : nav graph의 fragment id와 동일하게 설정해야 함



The XML code defines a navigation menu with three items, each representing a fragment in the navigation graph. The first item is checked by default due to the 'single' value in its checkableBehavior attribute.

```
<menu xmlns:android="http://schemas.android.com/apk/res/android"  
      xmlns:tools="http://schemas.android.com/tools"  
      tools:showIn="navigation_view">  
    <group android:checkableBehavior="single">  
      <item  
          android:id="@+id/nav_home"  
          android:icon="@drawable/ic_menu_camera"  
          android:title="Home" />  
      <item  
          android:id="@+id/nav_gallery"  
          android:icon="@drawable/ic_menu_gallery"  
          android:title="Gallery" />  
      <item  
          android:id="@+id/nav_slideshow"  
          android:icon="@drawable/ic_menu_slideshow"  
          android:title="Slideshow" />  
    </group>  
</menu>
```

Annotations in red:

- A red circle highlights the `checkableBehavior="single"` attribute with the handwritten note "한번선택" (single selection).
- A red circle highlights the `single` value with the handwritten note "한번선택하면" (once selected).
- A red bracket groups the three `item` elements under the `group` element, with the handwritten note "묶어쓰기" (grouping).
- A red bracket groups the three `item` elements under the `group` element, with the handwritten note "묶어쓰기" (grouping).

Navigation Drawer 구현 절차

5. AppBarConfiguration과 NavigationUI 객체를 사용하여 탐색 동작 설정

```
AppBarConfiguration mAppBarConfiguration;
```

Handwritten notes in red:

- Top right: '설정' (Settings)
- Bottom right: '다음' (Next) and '이전' (Previous)
- Bottom left: '제작' (Created by)

```
DrawerLayout drawer = findViewById(R.id.drawer_layout);
NavigationView navigationView = findViewById(R.id.nav_view);
// Passing each menu ID as a set of IDs because each
// menu should be considered as top level destinations.
mAppBarConfiguration = new AppBarConfiguration.Builder(
    R.id.nav_home, R.id.nav_gallery, R.id.nav_slideshow)
    .setOpenableLayout(drawer)//.setDrawerLayout(drawer)
    .build();
NavController navController = Navigation.findNavController(this, R.id.nav_host_fragment);
NavigationUI.setupActionBarWithNavController(this, navController, mAppBarConfiguration);
NavigationUI.setupWithNavController(navigationView, navController);
```

Handwritten annotations on the code:

- A red circle highlights the line `mAppBarConfiguration = new AppBarConfiguration.Builder(...)`.
- A red arrow points from the handwritten note 'DrawerLayout' to the line `.setOpenableLayout(drawer)`.
- A red arrow points from the handwritten note 'NavigationUI' to the line `NavigationUI.setupWithNavController(navigationView, navController);`.

```
@Override
public boolean onSupportNavigateUp() {
    NavController navController = Navigation.findNavController(this, R.id.nav_host_fragment);
    return NavigationUI.navigateUp(navController, mAppBarConfiguration)
        || super.onSupportNavigateUp();
}
```

Navigation Drawer 구현 절차

5. AppBarConfiguration과 NavigationUI 객체를 사용하여 탐색 동작 설정

5.1 AppBarConfiguration 객체타입의 멤버변수 선언

5.2 DrawerLayout 객체 찾아서 얻어오기

5.3 NavigationView 객체 찾아서 얻어오기

5.4 AppBarConfiguration.Builder 객체 생성한 후 build 메소드 호출하여 AppBarConfiguration 객체 생성

5.5 NavController 객체 생성

5.6 setupActionBarWithNavController 메소드 호출하여 NavController와 AppBarConfiguration 객체 연결

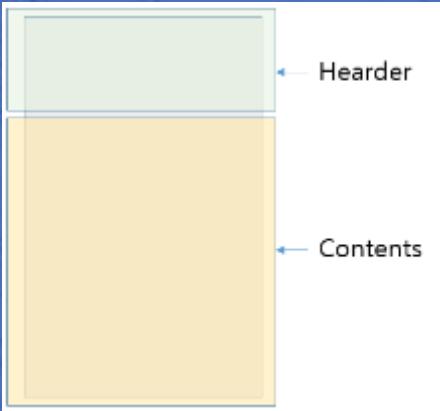
5.7 setupWithNavController 메소드로 NavigationView와 NavController 객체 연결

DrawerLayout

- 속성
 - android:fitsSystemWindow : 상태바와 소프트키보드 사이에 NavigationView가 보이도록 함.
 - tools: openDrawer : 디자인창에서 Drawer처럼 열리는 설정값 NavigationView가 어느 위치에서 열릴지 보여주기 위해 사용됨
- 메소드
 - void openDrawer(int gravity) / openDrawer(View drawerView)
 - Gravity.LEFT / Gravity.RIGHT
 - GravityCompat.START / GravityCompat.END
 - void closeDrawer(int gravity) / closeDrawer(View drawerView)

NavigationView

- 구성



```
<com.google.android.material.navigation.NavigationView  
    android:id="@+id/nav_view"  
    android:layout_width="wrap_content"  
    android:layout_height="match_parent"  
    android:layout_gravity="start"  
    android:fitsSystemWindows="true"  
    app:headerLayout="@layout/nav_header_main"  
    app:menu="@menu/activity_main_drawer" />  
  
<menu xmlns:android="http://schemas.android.com/apk/res/android"  
      xmlns:tools="http://schemas.android.com/tools"  
      tools:showIn="navigation_view">  
  <group android:checkableBehavior="single">  
    <item  
        android:id="@+id/nav_home"  
        android:icon="@drawable/ic_menu_camera"  
        android:title="Home" />  
    <item  
        android:id="@+id/nav_gallery"  
        android:icon="@drawable/ic_menu_gallery"  
        android:title="Gallery" />  
    <item  
        android:id="@+id/nav_slideshow"  
        android:icon="@drawable/ic_menu_slideshow"  
        android:title="Slideshow" />  
  </group>  
</menu>
```

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:app="http://schemas.android.com/apk/res-auto"  
    android:layout_width="match_parent"  
    android:layout_height="176dp"  
    android:background="@drawable/side_nav_bar"  
    android:gravity="bottom"    android:orientation="vertical"  
    android:padding="16dp"  
    android:theme="@style/ThemeOverlay.AppCompat.Dark">  
  <ImageView  
      android:id="@+id/imageView"  
      android:layout_width="wrap_content"  
      android:layout_height="wrap_content"  
      android:contentDescription="Navigation header"  
      android:paddingTop="8dp"  
      app:srcCompat="@mipmap/ic_launcher_round" />  
  <TextView  
      android:layout_width="match_parent"  
      android:layout_height="wrap_content"  
      android:paddingTop="8dp"  
      android:text="Android Studio"  
      android:textAppearance="@style/TextAppearance.AppCompat.Body1" />  
  <TextView  
      android:id="@+id/textView"  
      android:layout_width="wrap_content"  
      android:layout_height="wrap_content"  
      android:text="android.studio@android.com" />  
</LinearLayout>
```

Navigation View

- 속성
 - app:headerLayout – 헤더 레이아웃 설정
 - app:menu – 메뉴 리소스 설정
 - android:layout_gravity – navigation view의 위치 설정. start(left)만 가능
- 메소드
 - void setNavigationItemSelectedListener
(NavigationView.OnNavigationItemSelectedListener listener) :
NavigationView에서 나열해서 보여주는 항목 중 하나를 선택했을 때
발생하는 이벤트를 처리하기 위한 이벤트 리스너를 등록하는 메소드
- NavigationView.OnNavigationItemSelectedListener
 - 이동할 화면을 선택하기 위해 클릭했을 때 화면선택 이벤트를 처리하기
위한 이벤트 리스너 인터페이스

Navigation View 사용해보기

Nav

VI

가

자

기

제

작

성



NavigationUI

관련주제



- NavigationUI 클래스

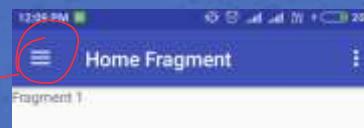
- 사용자가 앱을 탐색할 때 Toolbar, Navigation Menu, NavigationView등의 행동들이 연결되어 실행될 수 있도록 지원 (예: 앱바 타이틀의 변경, Navigation Menu 클릭시 페이지 이동 등)
- AppBarConfiguration 객체를 이용해 앱바의 왼쪽 상단 모서리에 있는 탐색 버튼의 동작을 관리
- Navigaion View와 Navigation controller를 연결
- 메소드
 - `void setupActionBarWithNavController(activity, navController, appBarConfiguration)` : activity에 포함된 ActionBar가 navController를 사용할 수 있도록 설정해 줌. 대상 페이지가 변경될 때마다 자동으로 ActionBar 타이틀이 변경되고 탐색버튼이 생성
 - `void setupWithNavController(navigationView, navController)` : navigationView가 navController를 사용할 수 있도록 설정. navigationView의 메뉴항목이 선택되면 자동으로 해당 페이지로 이동함

AppBarConfiguration

- AppBarConfiguration 클래스
 - 앱 표시 영역의 왼쪽 상단 모서리에 있는 탐색 버튼의 동작을 관리
- AppBarConfiguration.Builder 클래스
 - AppBarConfiguration 객체 생성에 사용되는 객체 *→ 가능성이 매우 많다*
 - AppBarConfiguration.Builder(int... topLevelDestinationIds) : 생성자 매개변수로 최상위 대상 fragment의 id(navigation graph에 정의되어 있는 id)들을 사용
↳ 아직 배운 3가지 대상은 없음
 - AppBarConfiguration.Builder setDrawerLayout(DrawerLayout drawerLayout) / setOpenableLayout(Openable openableLayout) : 사용되는 drawer layout 객체를 설정
 - AppBarConfiguration build() : AppBarConfiguration 객체를 생성하여 반환해 줌

MainActivity (AppCompatActivity)

- onSupportNavigateUp 메소드
 - 상단 앱바의 탐색 버튼에 대한 클릭이벤트 처리하는 메소드
 - NavigationUI 클래스의 navigateUp 메소드 호출
 - boolean NavigationUI.navigateUp(navController, appBarConfiguration)
 - 최상위 탐색 페이지가 아닌 경우 이전 페이지로 이동. (Navigation Menu에 의해 이동한 경우는 start page로 이동)



onSupportNavigateUp 메소드

- NavigationUI 객체에 의한 화면 이동 흐름 관리

