

1. Buka kembali file projek week 3 kalian kemarin atau boleh juga kalau mau membuat new project
2. Karena di week ini kita akan menggunakan navigation maka kita harus import gradle navigation terlebih dahulu di build.gradle(module:app)

Bisa diliat dibawah ini:

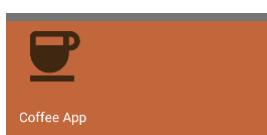
```
dependencies {

    implementation("androidx.core:core-ktx:1.9.0")
    implementation("androidx.appcompat:appcompat:1.6.1")
    implementation("com.google.android.material:material:1.9.0")
    implementation("androidx.constraintlayout:constraintlayout:2.1.4")
    testImplementation("junit:junit:4.13.2")
    androidTestImplementation("androidx.test.ext:junit:1.1.5")
    androidTestImplementation("androidx.test.espresso:espresso-core:3.5.1")
    implementation("androidx.navigation:navigation-fragment-ktx:2.5.3")
    implementation("androidx.navigation:navigation-ui-ktx:2.5.3")
    implementation("androidx.viewpager2:viewpager2:1.0.0")
}
```

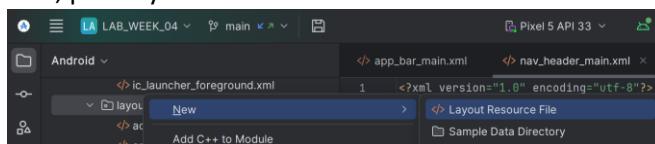
Question: Kenapa perlu import gradle nya kak?

Answer: Supaya bisa digunakan, ini seperti konsep di react setiap kita ingin menggunakan module apapun kita harus import terlebih dahulu.

3. Untuk buat navbar dibawah ini



Kita akan membuat file nav_header_main.xml di folder layout, lalu klik kanan mouse, pilih new, pilih layout resource file



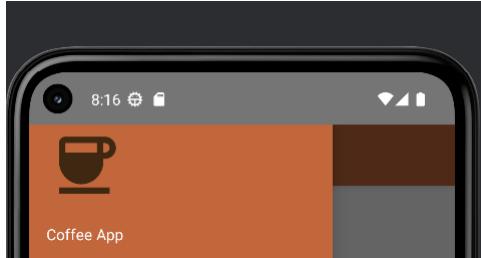
4. Lalu tambahkan kode dibawah ini di file nav_header_main.xml

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     android:layout_width="match_parent"
6     android:layout_height="176dp"
7     android:background="@color/light_brown"
8     android:gravity="bottom"
9     android:orientation="vertical"
10    android:paddingStart="16dp"
11    android:paddingTop="16dp"
12    android:paddingEnd="16dp"
13    android:paddingBottom="16dp"
14    android:theme="@style/ThemeOverlay.AppCompat.Dark">

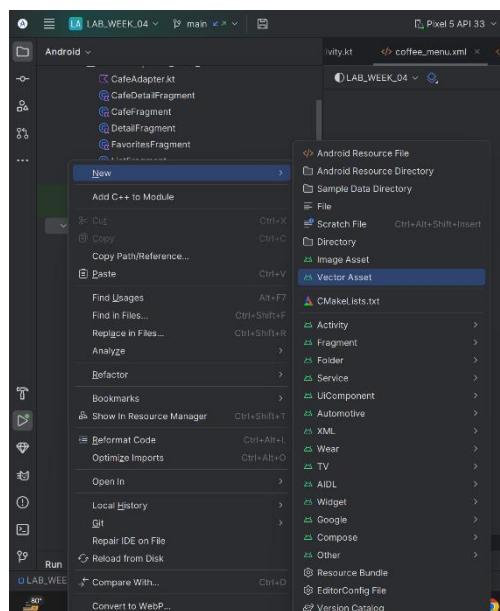
```

5. Selanjutnya, kita akan ke bagian gambar kopi di navigation dibawah ini:

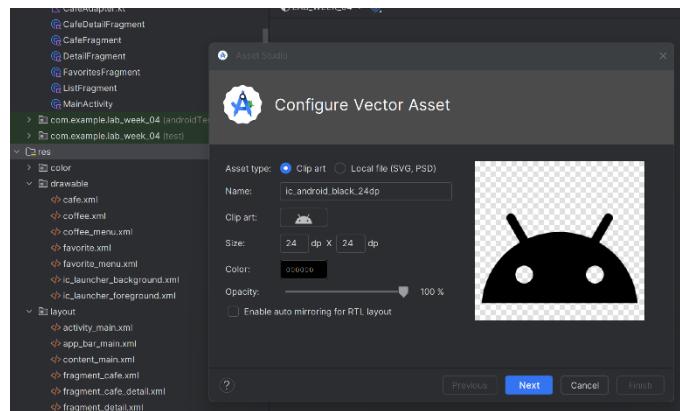


Cara untuk mendapatkan icon coffee nya adalah:

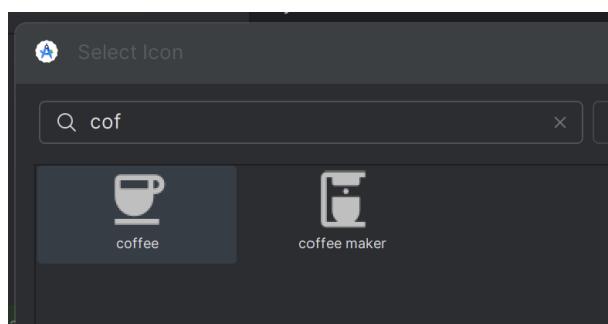
- Ke folder res -> new -> vector asset



- Klik Clip Art



- Lalu klik bagian clip art, lalu search "cofee"



- Lalu klik ok
- Setelah kalian klik ok, maka akan muncul cofee.xml

```

1 <vector android:height="24dp" android:tint="#3E2811"
2   android:viewportHeight="24" android:viewportWidth="24"
3     android:width="24dp" xmlns:android="http://schemas.android.com/apk/res/android">
4       <path android:fillColor="@android:color/white" android:pathData="M18.5,3H6C4.9,3
5         </vector>
6

```

6. Kita akan panggil icon coffee ke file nav_header_main.xml

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      android:layout_width="match_parent"
6      android:layout_height="176dp"
7      android:background="@color/light_brown"
8      android:gravity="bottom"
9      android:orientation="vertical"
10     android:paddingStart="16dp"
11     android:paddingTop="16dp"
12     android:paddingEnd="16dp"
13     android:paddingBottom="16dp"
14     android:theme="@style/ThemeOverlay.AppCompat.Dark">
15     <ImageView
16         android:id="@+id/imageView"
17         android:layout_width="70dp"
18         android:layout_height="100dp"
19         android:contentDescription="Navigation header"
20         android:paddingTop="8dp"
21         app:srcCompat="@drawable/coffee" />

```

- Dibawah icon coffee ada tulisan Coffee App, caranya pada string.xml masukkan kode dibawah ini:

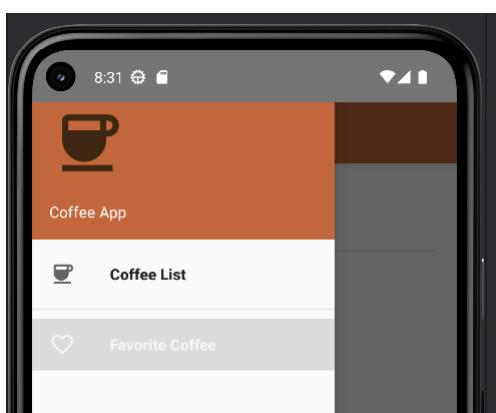
```

:resources>
    <string name="app_name">LAB_WEEK_04</string>

    <string name="app_title">Coffee App</string>

```

- Kita akan lanjutkan slicing UI setelah Coffee App, kita akan membuat dua menu pada navigation nya yaitu Coffee List dan Favorite coffee:



Caranya tambahkan kode dibawah ini di file string.xml:

```

1  :resources>
2      <string name="app_name">LAB_WEEK_04</string>
3
4      <string name="app_title">Coffee App</string>
5
6      <string name="nav_header_desc">Navigation header</string>
7
8      <string name="menu_coffee">Coffee List</string>
9      <string name="menu_favorites">Favorite Coffee</string>

```

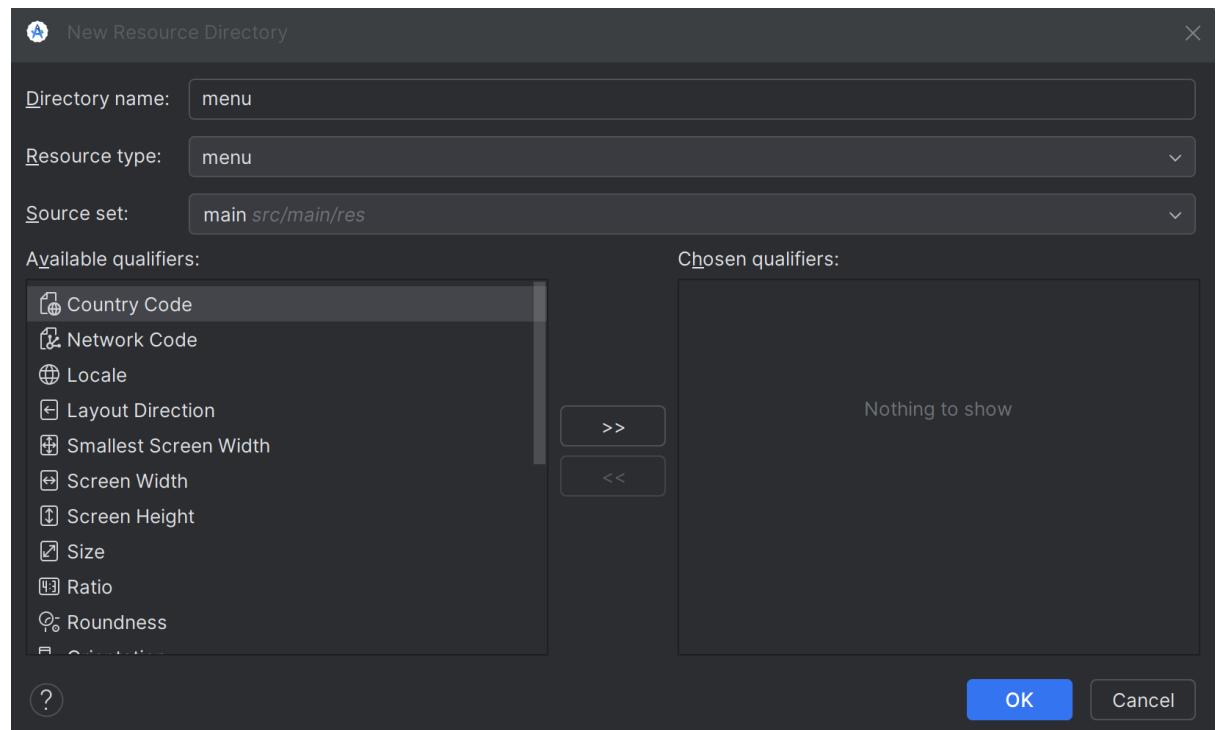
Q: kak beda nya sama coffee_menu tadi apa?

A: beda, coffee_menu itu drawable atau icon sedangkan menu_coffee itu string

Q: terus kenapa harus dibedain gitu kak?

9. Kita akan membuat folder baru bernama "menu" di dalam folder res. Caranya:

- Klik kanan mouse pada folder res -> new -> android resource directory -> Resource type: menu, lalu klik ok

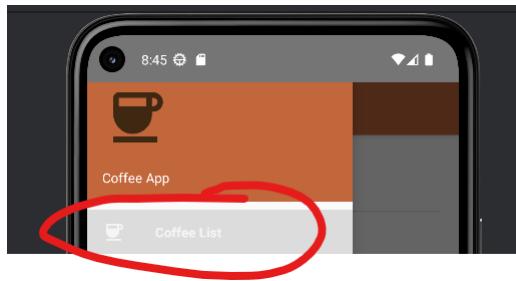


10. Kita ke file activity_main_drawer.xml ini sama saja seperti activity_main pada week 03, bedanya activity_main_drawer itu untuk mendefinisikan tampilan dan struktur **navigation drawer** dalam sebuah activity. Kalian akan masukkan coffee_menu dan menu_coffee yang kalian sudah buat dengan kode dibawah ini:

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <menu xmlns:android="http://schemas.android.com/apk/res/android"
3  xmlns:tools="http://schemas.android.com/tools"
4  tools:showIn="navigation_view">
5  <group
6      android:id="@+id/menu_top"
7      android:checkableBehavior="single">
8          <item
9              android:id="@+id/listFragment"
10             android:icon="@drawable/coffee_menu"
11             android:title="@string/menu_coffee" />
12     </group>
```

Q: kak itu group fungsinya untuk apa?

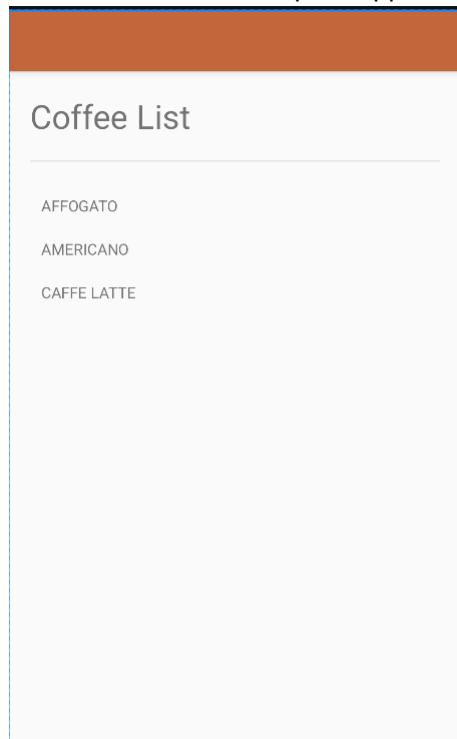
A: itu kan masih satu group. Intinya coffee_menu itu untuk panggil icon, menu_coffee itu untuk panggil coffee list



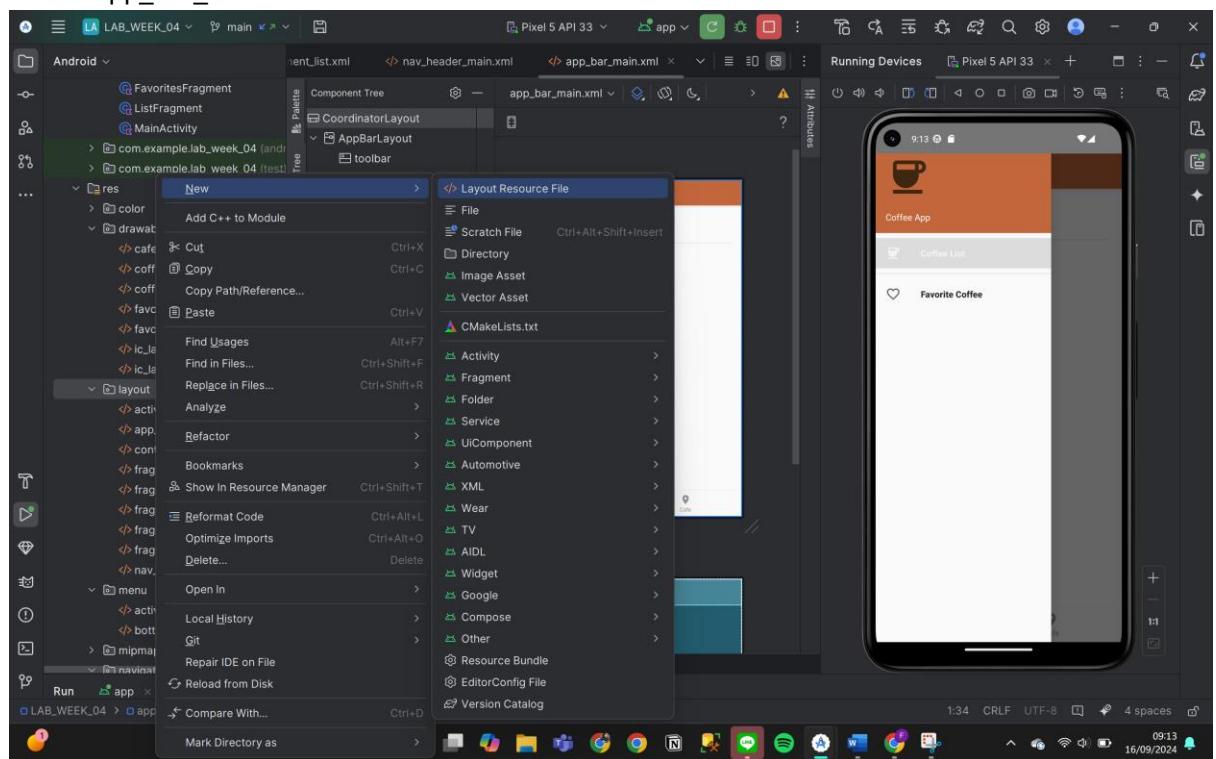
11. Kita ke file activity_main.xml untuk memanggil file nav_header_main dan activity_main_drawer , berikut code nya:

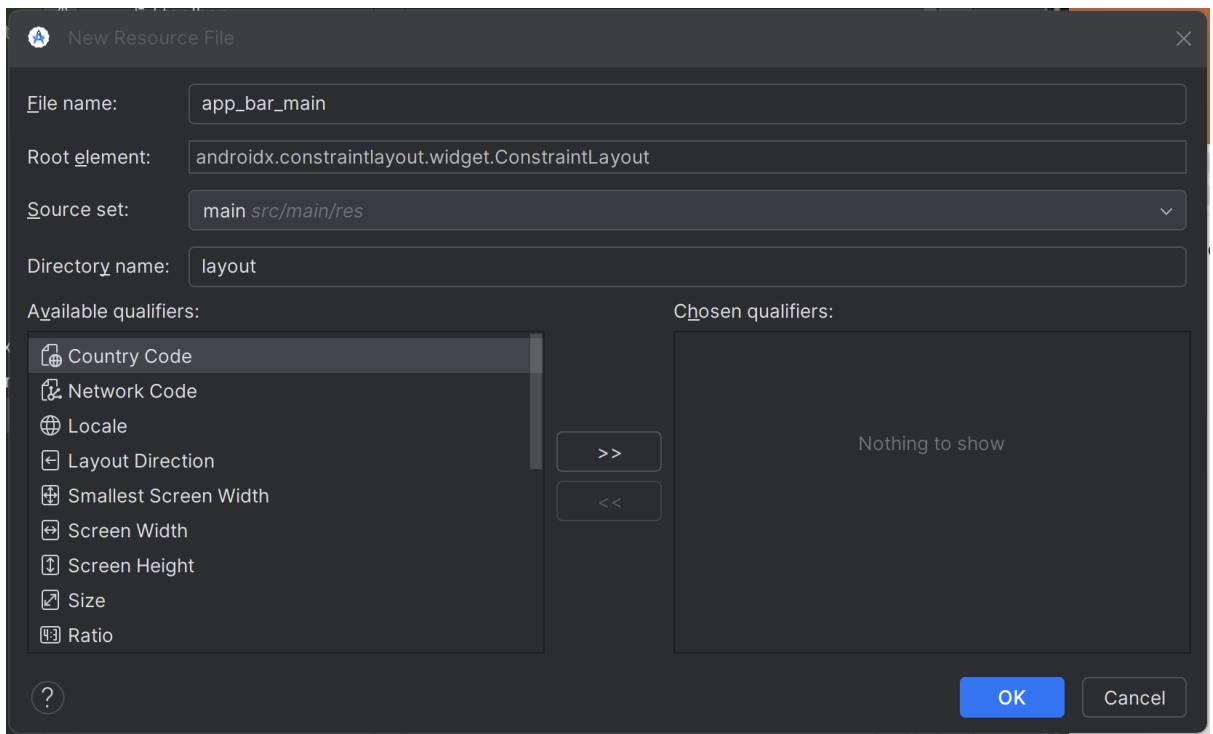
```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.drawerlayout.widget.DrawerLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:tools="http://schemas.android.com/tools"
5      xmlns:app="http://schemas.android.com/apk/res-auto"
6      android:id="@+id/drawer_layout"
7      android:layout_width="match_parent"
8      android:layout_height="match_parent"
9      android:fitsSystemWindows="true"
10     tools:openDrawer="start"
11     tools:context=".MainActivity">
12     <include
13         layout="@layout/app_bar_main"
14         android:layout_width="match_parent"
15         android:layout_height="match_parent" />
16
17     <com.google.android.material.navigation.NavigationView
18         android:id="@+id/nav_view"
19         android:layout_width="wrap_content"
20         android:layout_height="match_parent"
21         android:layout_gravity="start"
22         android:fitsSystemWindows="true"
23         app:headerLayout="@layout/nav_header_main"
24         app:menu="@menu/activity_main_drawer" />
25 </androidx.drawerlayout.widget.DrawerLayout>
```

12. Kita akan membuat tampilan app bar seperti ini



Caranya ke folder layout -> klik kanan mouse -> new -> layout resource file -> masukkan nama "app_bar_main.xml"

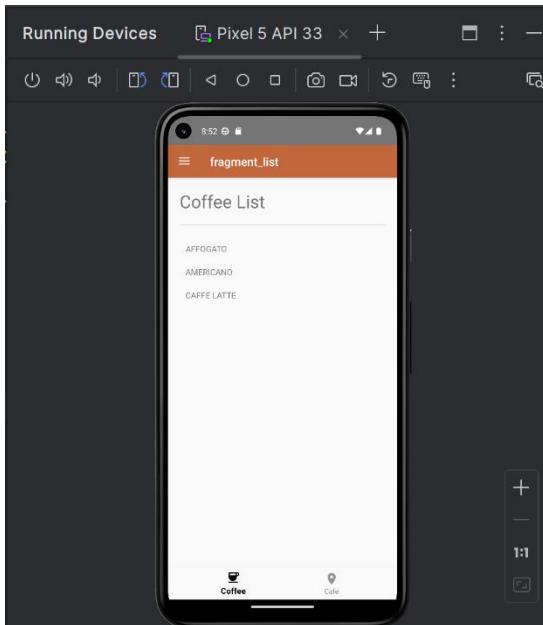




Lalu klik ok, dan pada file tersebut masukkan kode dibawah ini:

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.coordinatorlayout.widget.CoordinatorLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      android:orientation="vertical"
9      tools:context=".MainActivity">
10     <com.google.android.material.appbar.AppBarLayout
11         android:layout_width="match_parent"
12         android:layout_height="wrap_content"
13         android:theme= "@style/Theme.NavigationDrawer.AppBarOverlay">
14             <androidx.appcompat.widget.Toolbar
15                 android:id="@+id/toolbar"
16                 android:layout_width="match_parent"
17                 android:layout_height="?attr/actionBarSize"
18                 android:background="@color/light_brown"
19                 app:popupTheme="@style/Theme.NavigationDrawer.PopupOverlay" />
20         </com.google.android.material.appbar.AppBarLayout>
21         <include
22             android:id="@+id/include"
23             android:layout_width="match_parent"
24             android:layout_height="wrap_content"
25             layout="@layout/content_main"
26             app:layout_behavior="com.google.android.material.appbar.AppBarLayout$Scrolli..."/>
27
28     <com.google.android.material.bottomnavigation.BottomNavigationView
29         android:id="@+id/bottom_nav"
30         android:layout_width="match_parent"
31         android:layout_height="wrap_content"
32
33         android:layout_gravity="bottom"
34         android:background="?android:attr/windowBackground"
35         app:itemTextColor="@color/bottom_nav_color"
36         app:itemIconTint="@color/bottom_nav_color"
37         app:labelVisibilityMode="labeled"
38         app:menu="@menu/bottom_nav_menu" />
39     </androidx.coordinatorlayout.widget.CoordinatorLayout>
```

13. Kita akan masuk ke menu Coffee List, ketika di klik menu Coffee List maka akan muncul Coffee List sama seperti di week 03



Tetap gunakan kode ListFragment.kt di week 03, akan tetapi ada sedikit perbedaan kode di baris akhir kode

Week 03:

```
class ListFragment : Fragment() {  
    private var param1: String? = null  
    private var param2: String? = null
```

Week 04:

Tidak perlu deklarasi variabel lagi

Function onCreate Week 03:

Kita membuat variable param1 dan param2 untuk memanggil function

```
override fun onCreate(savedInstanceState: Bundle?) {  
    super.onCreate(savedInstanceState)  
    arguments?.let {  
        param1 = it.getString(ARG_PARAM1)  
        param2 = it.getString(ARG_PARAM2)  
    }  
}
```

Function onCreate Week 04:

Tidak menggunakan variable

```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
}
```

Companion Object Week 03:

Dikarenakan kita memakai deklarasi variable maka kita harus mengubah nya terlebih dahulu menggunakan java virtual machine (jvm)

```
companion object {
    const val COFFEE_ID = "COFFEE_ID"
    @JvmStatic
    fun newInstance(param1: String, param2: String) =
        ListFragment().apply {
            arguments = Bundle().apply {
                putString(ARG_PARAM1, param1)
                putString(ARG_PARAM2, param2)
            }
        }
}
```

Companion Object Week 04:

Kita hanya perlu memanggil COFFEE_ID

```
companion object {
    const val COFFEE_ID = "COFFEE_ID"
}
```

14. Lalu kita buat fragment_list.xml, kode nya sama persis tidak ada yang berbeda satu pun dari kode week 03.
15. Kita buat file fragment_detail.xml lalu masukkan kode dibawah ini:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:orientation="vertical"
8     android:padding="20dp"
9     tools:context=".DetailFragment">
10    <TextView
11        android:id="@+id/coffee_title"
12        android:textStyle="bold"
13        android:gravity="center"
14        android:layout_width="match_parent"
15        android:layout_height="wrap_content"
16        tools:text="AFFOGATO"/>
17    <View
18        android:layout_width="match_parent"
19        android:layout_height="1dp"
20        android:layout_marginVertical="20dp"
21        android:background="?android:attr/dividerVertical" />
22    <TextView
23        android:id="@+id/coffee_desc"
24        android:layout_width="match_parent"
25        android:layout_height="wrap_content"
26        android:gravity="center"
27        tools:text="Espresso poured on a vanilla ice cream. Served in a cap..."/>
28 </LinearLayout>
```

Tujuan dari kode tersebut untuk membuat:

AFFOGATO

Espresso poured on a vanilla ice cream. Served in a cappuccino cup.

16. Selanjutnya kita ke DetailFragment.kt untuk membuat logic dari fragment_detail.xml, sama seperti ListFragment.kt di DetailFragment.kt kita tidak memerlukan variabel param1 dan param2 lagi

Week 03:

```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    arguments?.let {
        param1 = it.getString(ARG_PARAM1)
        param2 = it.getString(ARG_PARAM2)
    }
}
```

Week 04:

Hapus kode

Week 03:

Pada kode companion object kita memerlukan argument untuk meletakkan coffee id

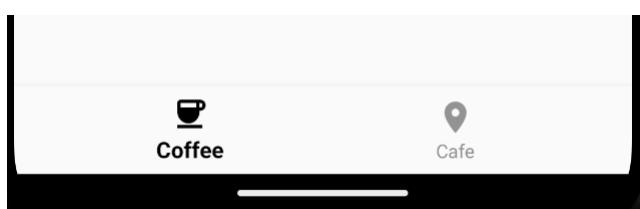
```
companion object {
    private const val COFFEE_ID = "COFFEE_ID"
    fun newInstance(coffeeId: Int) =
        DetailFragment().apply {
            arguments = Bundle().apply {
                putInt(COFFEE_ID, coffeeId)
            }
        }
}
```

Week 04:

Tidak pakai arguments

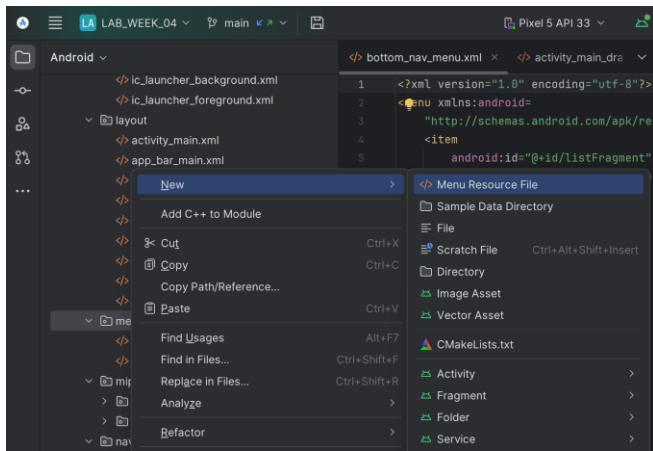
```
companion object {
    private const val COFFEE_ID = "COFFEE_ID"
}
```

17. Selanjutnya kita akan membuat menu bottom navigation:



Caranya:

Klik folder menu, lalu klik kanan, pilih Menu Resource File, lalu ketik namanya "bottom_nav_menu.xml"



Lalu masukkan kode dibawah ini pada file bottom_nav_menu.xml:

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <menu xmlns:android=
3     "http://schemas.android.com/apk/res/android">
4     <item
5         android:id="@+id/listFragment"
6         android:icon="@drawable/coffee"
7         android:title="Coffee" />
8     <item
9         android:id="@+id/cafeFragment"
10        android:icon="@drawable/cafe"
11        android:title="Cafe"/>
12 </menu>

```

Di file bottom_nav_menu.xml kita akan membuat menu, di menu pertama kita akan memanggil kembali icon coffee sebelumnya dan ListFragment dikarenakan ketika kita tekan tombol coffee kita mau mengarahkan ke ListFragment.

Selanjutnya kita juga memanggil icon cafe dan cafeFragment untuk membuat menu café dan ketika di klik menu tersebut akan mengarahkan ke fragment_cafe.

18. Kita ke file app_bar_main.xml lagi dan memanggil kode pada file bottom_nav_menu dengan masukkan kode dibawah ini:

```
28     <com.google.android.material.bottomnavigation.BottomNavigationView
29         android:id="@+id/bottom_nav"
30         android:layout_width="match_parent"
31         android:layout_height="wrap_content"
32         android:layout_gravity="bottom"
33         android:background="?android:attr/windowBackground"
34         app:itemTextColor="@color/bottom_nav_color"
35         app:itemIconTint="@color/bottom_nav_color"
36         app:labelVisibilityMode="labeled"
37         app:menu="@menu/bottom_nav_menu" />
38     </androidx.coordinatorlayout.widget.CoordinatorLayout>
```

Full code app_bar_main.xml:

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.coordinatorlayout.widget.CoordinatorLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      android:orientation="vertical"
9      tools:context=".MainActivity">
10     <com.google.android.material.appbar.AppBarLayout
11         android:layout_width="match_parent"
12         android:layout_height="wrap_content"
13         android:theme= "@style/Theme.NavigationDrawer.AppBarOverlay">
14         <androidx.appcompat.widget.Toolbar
15             android:id="@+id/toolbar"
16             android:layout_width="match_parent"
17             android:layout_height="?attr actionBarSize"
18             android:background="@color/light_brown"
19             app:popupTheme="@style/Theme.NavigationDrawer.PopupOverlay" />
20     </com.google.android.material.appbar.AppBarLayout>
21     <include
22         android:id="@+id/include"
23         android:layout_width="match_parent"
24         android:layout_height="wrap_content"
25         layout="@layout/content_main"
26         app:layout_behavior="com.google.android.material.appbar.AppBarLayout$Scrolli..."/>
27
28     <com.google.android.material.bottomnavigation.BottomNavigationView
29         android:id="@+id/bottom_nav"
30         android:layout_width="match_parent"
31         android:layout_height="wrap_content"
32         android:layout_gravity="bottom"
33         android:background="?android:attr/windowBackground"
34         app:itemTextColor="@color/bottom_nav_color"
35         app:itemIconTint="@color/bottom_nav_color"
36         app:labelVisibilityMode="labeled"
37         app:menu="@menu/bottom_nav_menu" />
38     </androidx.coordinatorlayout.widget.CoordinatorLayout>
```

19. Selanjutnya kita akan membuat menu Favorite Coffee, sama seperti menu Coffee List kita buat icon nya terlebih dahulu, caranya:

- Ke folder drawable, klik kanan
- pilih vector asset
- lalu search "favorite"
- lalu klik ok

20. Selanjutnya kita buat string menu_favorite untuk mendeklarasikan menu Favorite Coffee.

```
1 <resources>
4     <string name="app_title">Coffee App</string>
5
6     <string name="nav_header_desc">Navigation header</string>
7
8     <string name="menu_coffee">Coffee List</string>
9     <string name="menu_favorites">Favorite Coffee</string>
```

21. Kita ke activity_main_drawer.xml untuk menambahkan kode untuk memanggil menu Favorite Coffee

```
13     <group
14         android:id="@+id/menu_bottom"
15         android:checkableBehavior="single">
16         <item
17             android:id="@+id/favoritesFragment"
18             android:icon="@drawable/favorite_menu"
19             android:title="Favorite Coffee" />
20     </group>
```

Full code activity_main_drawer.xml:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <menu xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools"
4     tools:showIn="navigation_view">
5     <group
6         android:id="@+id/menu_top"
7         android:checkableBehavior="single">
8             <item
9                 android:id="@+id/listFragment"
10                android:icon="@drawable/coffee_menu"
11                android:title="Coffee List" />
12         </group>
13         <group
14             android:id="@+id/menu_bottom"
15             android:checkableBehavior="single">
16                 <item
17                     android:id="@+id/favoritesFragment"
18                     android:icon="@drawable/favorite_menu"
19                     android:title="Favorite Coffee" />
20             </group>
21         </menu>
```

22. Kita buat class FavoritesFragment.kt dengan kode dibawah ini:

```
1 package com.example.lab_week_04
2
3 import android.os.Bundle
4 import androidx.fragment.app.Fragment
5 import android.view.LayoutInflater
6 import android.view.View
7 import android.view.ViewGroup
8
9 // TODO: Rename parameter arguments, choose names that match
10 // the fragment initialization parameters, e.g. ARG_ITEM_NUMBER
11 private const val ARG_PARAM1 = "param1"
12 private const val ARG_PARAM2 = "param2"
13
14 /**
15 * A simple [Fragment] subclass.
16 * Use the [FavoritesFragment.newInstance] factory method to
17 * create an instance of this fragment.
18 */
19 </> class FavoritesFragment : Fragment() {
20     // TODO: Rename and change types of parameters
21     private var param1: String? = null
22     private var param2: String? = null
23
24     override fun onCreate(savedInstanceState: Bundle?) {
25         super.onCreate(savedInstanceState)
26         arguments?.let {
27             param1 = it.getString(ARG_PARAM1)
28             param2 = it.getString(ARG_PARAM2)
29         }
30     }
31 }
```

Kita akan menambahkan kode dibawah ini untuk membuat ketika menu navigation favorite coffee di click maka akan ke fragment_favorites.xml

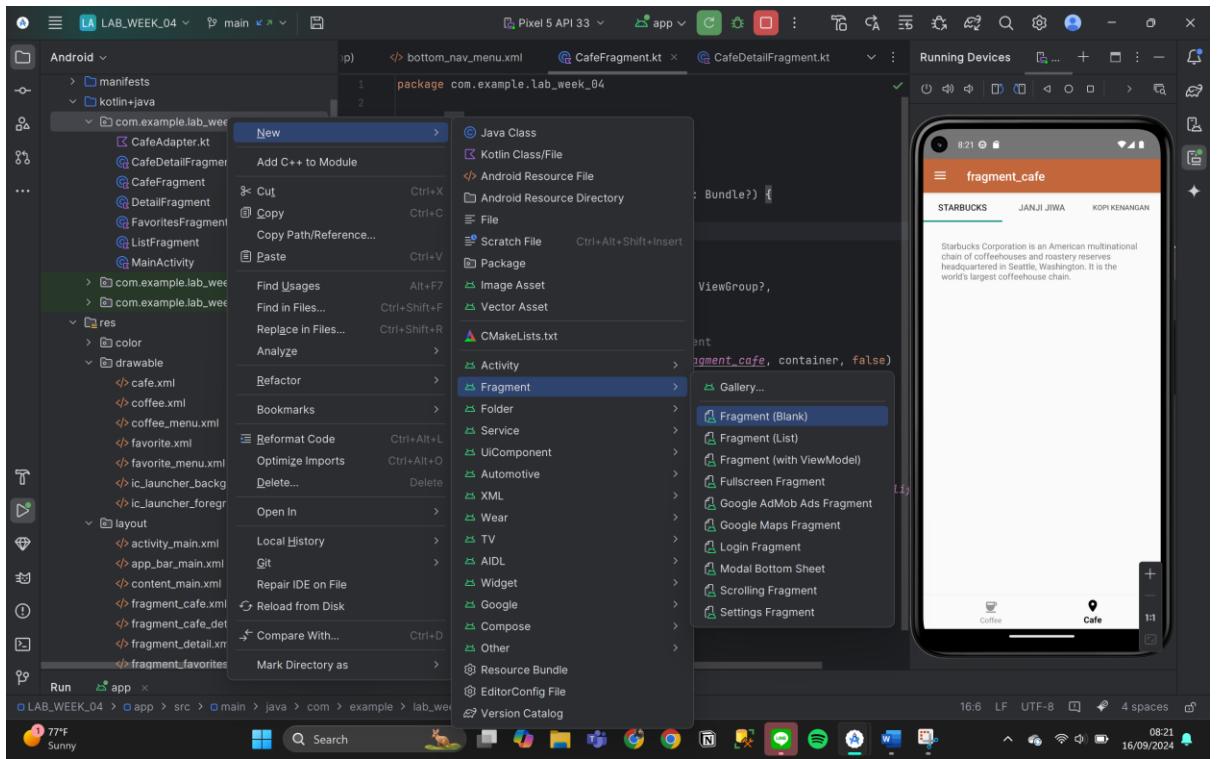
```
32 @Override
33     override fun onCreateView(
34         inflater: LayoutInflater, container: ViewGroup?,
35         savedInstanceState: Bundle?
36     ): View? {
37         // Inflate the layout for this fragment
38         return inflater.inflate(R.layout.fragment_favorites, container, false)
39     }
```

Lalu, kita lanjutkan lagi kode FavoriteFragment.kt dengan kode dibawah ini:

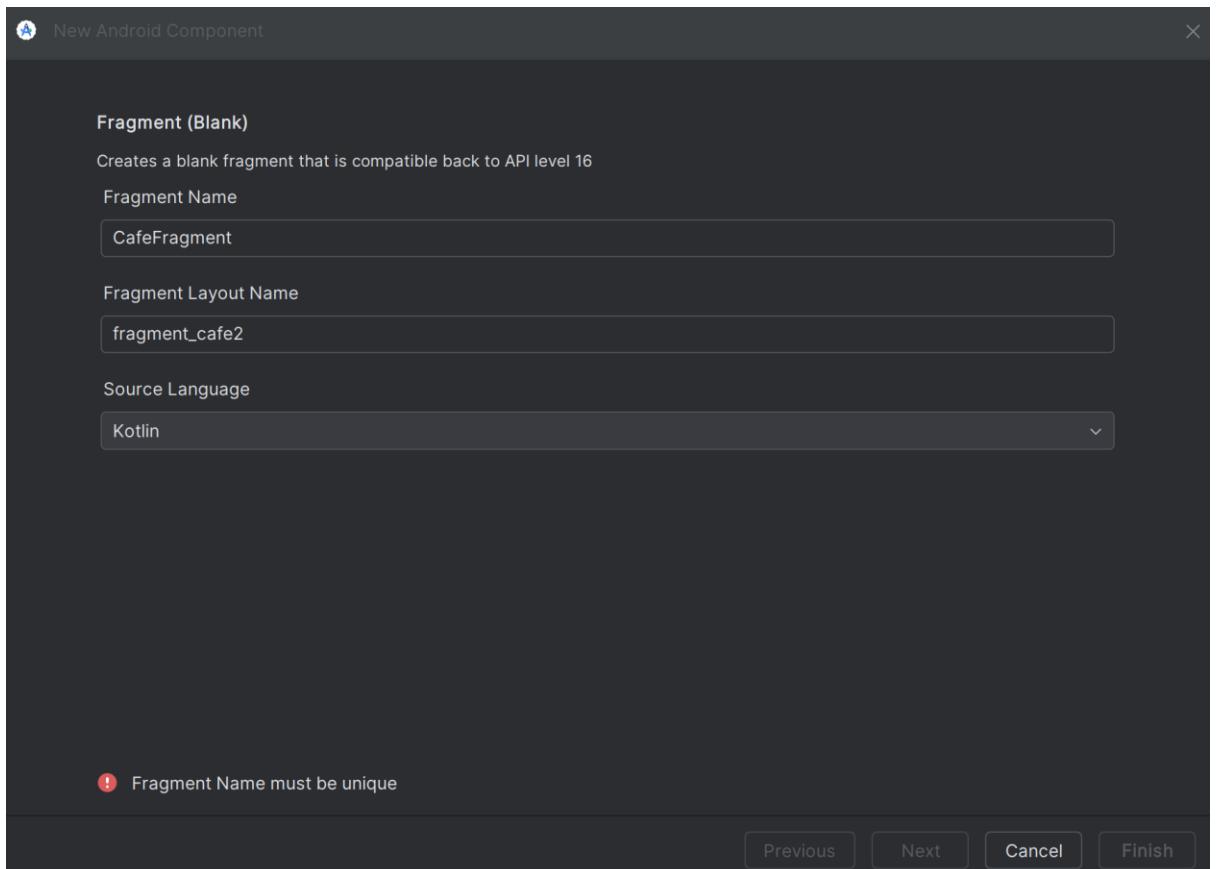
```
40     companion object {
41         /**
42          * Use this factory method to create a new instance of
43          * this fragment using the provided parameters.
44          *
45          * @param param1 Parameter 1.
46          * @param param2 Parameter 2.
47          * @return A new instance of fragment FavoritesFragment.
48          */
49         // TODO: Rename and change types and number of parameters
50         @JvmStatic
51         fun newInstance(param1: String, param2: String) =
52             FavoritesFragment().apply {
53                 arguments = Bundle().apply {
54                     putString(ARG_PARAM1, param1)
55                     putString(ARG_PARAM2, param2)
56                 }
57             }
58         }
59     }
```

23. Kita akan membuat logic frontend fragment_cafe yaitu dengan cara membuat file .kt baru yang Bernama “CafeFragment.kt” , caranya:

Ke folder dibawah folder kotlin+java , lalu klik kanan mouse, pilih new, pilih fragment, pilih blank fragment



Lalu masukkan nama fragment "CafeFragment" seperti dibawah ini:



Lalu masukkan kode dibawah ini di file CafeFragment.kt

```

1  package com.example.lab_week_04
2
3  import android.os.Bundle
4  import androidx.fragment.app.Fragment
5  import android.view.LayoutInflater
6  import android.view.View
7  import android.view.ViewGroup
8  import androidx.viewpager2.widget.ViewPager2
9  import com.google.android.material.tabs.TabLayout
10 import com.google.android.material.tabs.TabLayoutMediator
11
12 </> class CafeFragment : Fragment() {
13
14     @Override fun onCreate(savedInstanceState: Bundle?) {
15         super.onCreate(savedInstanceState)
16     }
17
18     @Override fun onCreateView(
19         inflater: LayoutInflater, container: ViewGroup?,
20         savedInstanceState: Bundle?
21     ): View? {
22         // Inflate the layout for this fragment
23         return inflater.inflate(R.layout.fragment_cafe, container, false)
24     }
25
26     @Override fun onViewCreated(view: View, savedInstanceState: Bundle?) {
27         super.onViewCreated(view, savedInstanceState)
28         val viewPager = view.findViewById<ViewPager2>(R.id.view_pager)
29         val tabLayout = view.findViewById<TabLayout>(R.id.tab_layout)
30         val adapter = CafeAdapter(requireContext(), childFragmentManager, lifecycle)
31         viewPager.adapter = adapter
32         TabLayoutMediator(tabLayout, viewPager) { tab, position ->
33             tab.text = resources.getString(TABS_FIXED[position])
34         }.attach()
35     }
36 }

```

Pada kode di file fragment_cafe.xml dan CafeFragment.kt kita hanya membuat tulisan STARBUCKS dll nya belum ada

fragment_cafe

24. Selanjutnya kita akan membuat tulisan STARBUCKS dll nya muncul, caranya:
- Ke file string.xml lalu masukkan kode dibawah ini

```
25     <string name="starbucks_title">STARBUCKS</string>
26     <string name="starbucks_desc">Starbucks Corporation is an American multinational chain of coff
27     <string name="janjijiwa_title">JANJI JIWA</string>
28     <string name="janjijiwa_desc">It is undeniable that Janji Jiwa outlets have spread to various
29     <string name="kopikenangan_title">KOPI KENANGAN</string>
30     <string name="kopikenangan_desc">At Kopi Kenangan, their dream is to serve high quality coffee
```

- Lalu kita buat file xml baru di folder layout yang Bernama “fragment_cafe_detail” lalu masukkan kode dibawah ini:

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent"
6      tools:context=".CafeDetailFragment"
7      android:padding="30dp">
8      <TextView
9          android:id="@+id/content_description"
10         android:layout_width="match_parent"
11         android:layout_height="match_parent"
12         android:text="" />
13  </FrameLayout>
```

Tujuan dari kode tersebut untuk membuat dibawah ini:

Starbucks Corporation is an American multinational
chain of coffeehouses and roastery reserves
headquartered in Seattle, Washington. It is the
world's largest coffeehouse chain.

Akan tetapi tulisan nya belum ada.

Untuk menampilkan tulisan tersebut, caranya:

- Kita buat file baru yang bernama CafeDetailFragment.kt , lalu masukkan kode dibawah ini:

```

1 package com.example.lab_week_04
2
3 import android.os.Bundle
4 import androidx.fragment.app.Fragment
5 import android.view.LayoutInflater
6 import android.view.View
7 import android.view.ViewGroup
8 import android.widget.TextView
9
10 private const val TAB_CONTENT = "TAB_CONTENT"
11
12 </> class CafeDetailFragment : Fragment() {
13     private var content: String? = null
14
15     override fun onCreate(savedInstanceState: Bundle?) {
16         super.onCreate(savedInstanceState)
17         arguments?.let {
18             content = it.getString(TAB_CONTENT)
19         }
20     }
21
22     override fun onCreateView(
23         inflater: LayoutInflater, container: ViewGroup?,
24         savedInstanceState: Bundle?
25     ): View? {
26         // Inflate the layout for this fragment
27         return inflater.inflate(R.layout.fragment_cafe_detail, container, false)
28     }
29
30     override fun onViewCreated(view: View, savedInstanceState: Bundle?) {
31         super.onViewCreated(view, savedInstanceState)
32         view.findViewById<TextView>(R.id.content_description)

```

```

33             ?.text = content
34         }
35
36         companion object {
37             fun newInstance(content: String) =
38                 CafeDetailFragment().apply {
39                     arguments = Bundle().apply {
40                         putString(TAB_CONTENT, content)
41                     }
42                 }
43             }
44         }

```

Tujuan dari kode tersebut untuk memberikan "otak" dari tampilan UI yang sudah kita buat di fragment_cafe_detail.xml

- Selanjutnya kita buat file baru lagi di bawah folder kotlin+java lagi yang bernama CafeAdapter.kt lalu masukkan kode dibawah ini:

```

1 package com.example.lab_week_04
2
3 import android.content.ContentProvider
4 import android.content.Context
5 import androidx.core.content.ContentProviderCompat.requireContext
6 import androidx.fragment.app.Fragment
7 import androidx.fragment.app.FragmentManager
8 import androidx.lifecycle.Lifecycle
9 import androidx.viewpager2.adapter.FragmentStateAdapter
10 import kotlinx.coroutines.withContext
11
12 val TABS_FIXED = listOf(
13     "STARBUCKS",
14     "JANJI JIWA",
15     "KOPI KENANGAN",
16 )
17
18 val TABS_CONTENT_FIXED = listOf(
19     "Starbucks Corporation is an American multinational chai...",
20     "It is undeniable that Janji Jiwa outlets have spread to...",
21     "At Kopi Kenangan, their dream is to serve high quality ...",
22 )
23
24 class CafeAdapter(private val context: Context, fragmentManager: FragmentManager, lifecycle:
25
26     override fun getItemCount(): Int {
27         return TABS_FIXED.size
28     }
29     override fun createFragment(position: Int): Fragment {
30         return CafeDetailFragment.newInstance(context.getString(TABS_CONTENT_FIXED[position])
31     }
32 }
```

lanjutan kode line 24 bisa diliat di github

Di file CafeAdapter.kt kita akan memanggil judul kopi tadi yang disimpan ke dalam TABS_FIXED

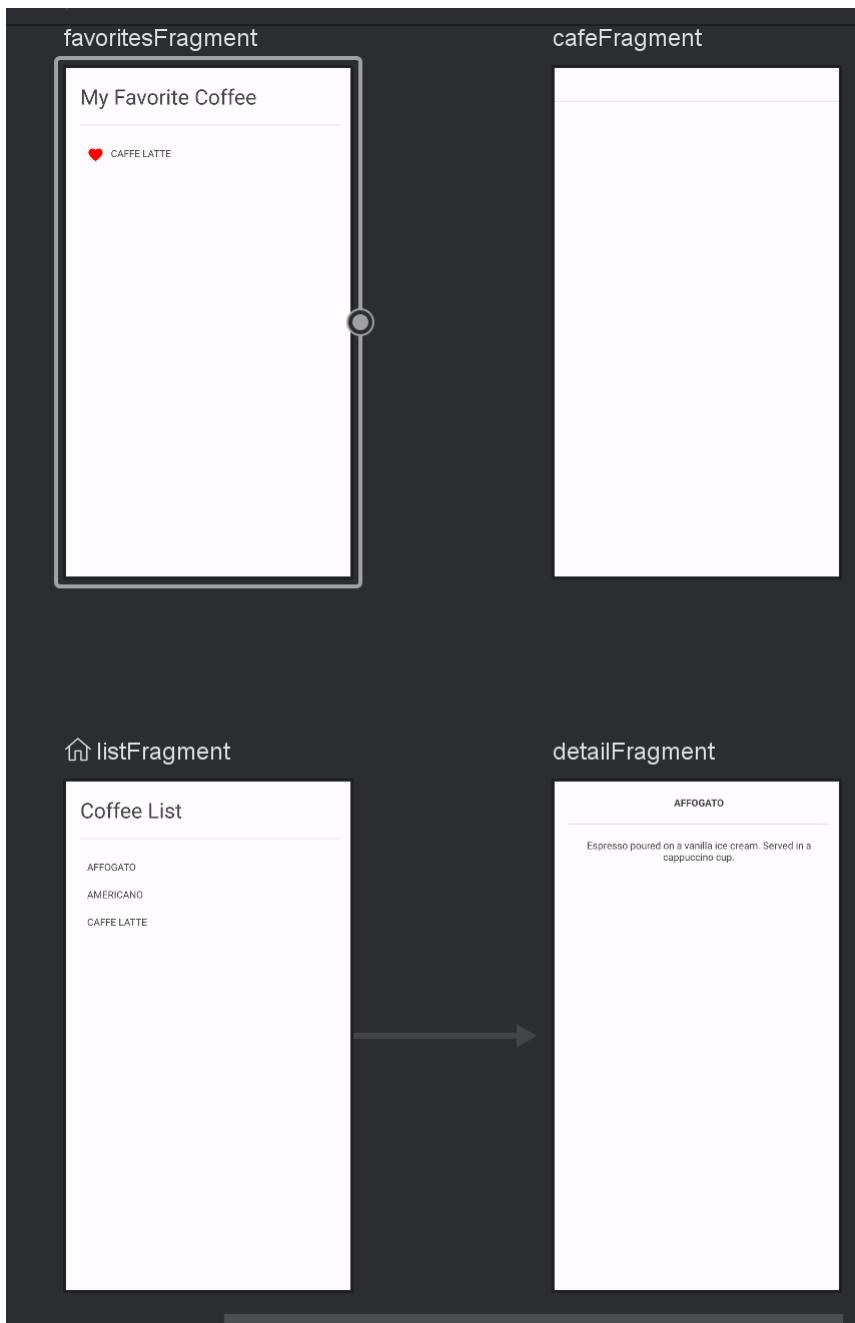
Kita juga memanggil deskripsi kopi yang disimpan ke dalam TABS_CONTENT_FIXED. Terakhir kita akan memanggil CafeDetailFragment dengan cara menggunakan return, dikarenakan kita akan menampilkan semua text yang sudah kita simpan di TABS_FIXED dan TABS_CONTENT_FIXED berada di badan CafeDetailFragment

25. Terakhir, pada nav_graph masukkan kode dibawah ini:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <navigation xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:id="@+id/nav_graph.xml"
6     app:startDestination="@+id/listFragment">
7     <fragment
8         android:id="@+id/listFragment"
9         android:name="com.example.lab_week_04.ListFragment"
10        android:label="fragment_list"
11        tools:layout="@layout/fragment_list" >
12        <action
13            android:id="@+id/action_listFragment_to_detailFragment"
14            app:destination="@+id/detailFragment" />
15    </fragment>
16    <fragment
17        android:id="@+id/detailFragment"
18        android:name="com.example.lab_week_04.DetailFragment"
19        android:label="fragment_detail"
20        tools:layout="@layout/fragment_detail" />
21    <fragment
22        android:id="@+id/favoritesFragment"
23        android:name="com.example.lab_week_04.FavoritesFragment"
24        android:label="fragment_favorites"
25        tools:layout="@layout/fragment_favorites" />
26    <fragment
27        android:id="@+id/cafeFragment"
28        android:name="com.example.lab_week_04.CafeFragment"
29        android:label="fragment_cafe"
30        tools:layout="@layout/fragment_cafe" />
31    </navigation>
```

Pada kode tersebut kita akan memanggil semua fragment sebelumnya. Karena nav_graph itu bertujuan untuk **Menghubungkan Fragment atau Activity** dimana setiap fragment atau activity didefinisikan sebagai sebuah **destination**. File ini menghubungkan setiap destination melalui **action** yang memungkinkan perpindahan dari satu tempat ke tempat lain di aplikasi.

Disini kita terdapat 4 fragment utama yaitu favoriteFragment, cafeFragment, listFragment dan detailFragment:



Q: kak kenapa favoriteFragment sama cafeFragment tidak ada tanda panahnya?

A: ya karena tidak ada hubungan alurnya, beda navigasi

Q: kak kenapa cafeFragment nya kosong yang lainnya ada isi?

A: karena kelas CafeFragment.kt nya kosong dan baru ada isi ketika kita menarik file kode dari fragment_cafe.xml di file CafeFragment.kt nya