

Les fragments :

Ajouter deux layout :

Layout1

Layout2

Ajouter la classe Frag1 dérivée de la classe Fragment

```
class frag1:Fragment
{
    public override View onCreateView(LayoutInflater inflater, ViewGroup
container, Bundle savedInstanceState)
    {
        base.onCreateView(inflater, container, savedInstanceState);
        View v = inflater.inflate(Resource.Layout.layout1, container, false);
        return v;
    }
}
```

De même Ajouter la classe Frag2 dérivée de la classe Fragment

Ajouter le code suivant dans la vue : Activity_Main.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <fragment
        class="App198.frag1"
        android:id="@+id/frag1"
        android:layout_width="match_parent"
```

```

        android:layout_height="wrap_content"
        android:layout_weight="1" />
</LinearLayout>

```

Les fragments et Menu :

- Créer un Menu.

```

<?xml version="1.0" encoding="utf-8" ?>
<menu
xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/search"
        android:icon="@drawable/Cut"
        android:showAsAction="always"
        android:title="action_search"
        />

    <item
        android:id="@+id/share"
        android:icon="@drawable/Cut"
        android:showAsAction="always"
        android:title="action_share"
        />

</menu>

```

-

- Ajouter une layout Layout1 (Vide)
- Ajouter la classe Frag1 :Fragment en relation avec Layoyt1
- Ajouter un fragment Vide à activity_Main.

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"

```

```

android:layout_height="match_parent">
<fragment
    class="App198.frag1"
    android:id="@+id/frag1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1" />
- </LinearLayout>

```

Ajouter layout 2.

Ajouter frag2 :

```

class frag2:Fragment
{
    public override View onCreateView(LayoutInflater inflater, ViewGroup
container, Bundle savedInstanceState)
    {
        base.onCreateView(inflater, container, savedInstanceState);
        View v = inflater.inflate(Resource.Layout.layout2, container, false);
        return v;
    }
}

```

Pour afficher layout2 sur layout1. On utilise le code dans mainactivity.

```

protected override void onCreate(Bundle
savedInstanceState)
{
    base.onCreate(savedInstanceState);

```

```

        // Set our view from the "main" layout
resource    setContentView(Resource.Layout.Main);

    }
    public override bool OnCreateOptionsMenu(IMenu
menu)
    {
        MenuInflater.Inflate(Resource.Menu.menu,
menu);

        return true;
    }
    public override bool
OnOptionsItemSelected(IMenuItem item)
    {
        var id = item.ItemId;
        if (id == Resource.Id.search)
        {
            FragmentTransaction ft =
FragmentManager.BeginTransaction();
            Frag2 f = new Frag2();
            ft.Replace(Resource.Id.frag1,f);
            Toast.MakeText(this, "ok",
ToastLength.Short).Show();
            ft.AddToBackStack(null);
            ft.Commit();
        }
        return base.OnOptionsItemSelected(item);
    }
}

```

Remplacer :

```

public override bool OnOptionsItemSelected(IMenuItem item)
{
    var id = item.ItemId;
    if (id == Resource.Id.search)
    {
        Frag2 f = new Frag2();
        if (f != null)
        {

```

```
        FragmentTransaction ft =  
FragmentManager.beginTransaction();  
        ft.Replace(Resource.Id.frag1, f);  
        ft.addToBackStack(null);  
        ft.commit();  
    }
```

```
        Toast.makeText(this, "ok",  
ToastLength.Short).show();  
    }
```