레이아웃에

android:onClick="addClick"

를 추가하면 버튼이 눌렸을 때 해당 함수를 수행하도록 한다.

EditText = 텍스트 창

Button = 버튼

Button button = (Button)findViewById(R.id.buttonid)로 레이아웃 상의 개체를 아이디로 가져올 수 있다.

텍스트 창으로부터 숫자 가져오기

Int n = Integer.parseInt(textid.getText().toString());

텍스트 창에 텍스트 설정

Textid.setText(내용)

알림창 띄우기

AlertDialog.Builder 객체

AlertDialog.Builder builder = new AlertDialog.Builder(this/MainActivity.this);

객체 생성 후 알림창 제목 지정

Builder.setTitle(“ ”);

제목 지정 + 내용

Builder.setTitle(“ ”).setText(“ ”);

알림창에 리스트 추가

builder.setItems(items, new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int i) {  
 Toast.*makeText*(getApplicationContext(),items[i], Toast.*LENGTH\_SHORT*).show();  
 }  
});

첫번째 인자로 리스트, 두번째 인자로 리스트 항목 클릭시 처리이다.

Int i부분에 자동으로 몇번째 항목을 선택했는지가 들어온다.

해당 리스트 내용을 Toast메시지로 출력한다.

Toast 메시지는 간단한 알림 메시지라고 볼 수 있다.

첫 인자로 context를 필요로 한다

Context는 어떤 Activity, application인지를 구별하는 정보로 getapplicationcontext() 는 application에 대한 context를 가져온다고 보면 된다.

Items[i]로 해당 리스트의 항목을 선택하고, Toast.LENGTH\_SHORT로 표시할 길이를 설정한다.

.show()로 출력한다

AlertDialog alertDialog = builder.create()로 알림창 구성을 마친 후 알림창 객체를 생성하고

AlertDialog.show(); 로 출력을 표시한다.

알림창에 버튼 추가

Builder에

.setNegativeButton("취소", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
 dialog.cancel();  
 }  
})

를 추가함으로서 알림창에 yes/no 버튼을 만들 수 있다.

Dialog.cancel()은 알림창을 끄는 것이고.

Finish()는 어플을 종료한다.

내비게이션 메뉴바 만들기, 화면 전환

프로젝트 생성시 Navigation Drawer Activity를 사용함.

layout폴더에 레이아웃 두개를 추가한다

first\_layout을 만들고 root를 relativelayout으로 바꾼다.

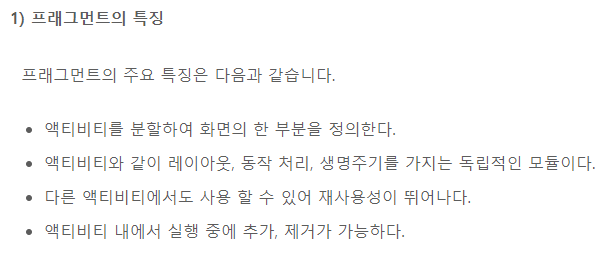
후에 텍스트뷰 하나를 설정하고 first\_layout을 복사하여 second\_layout을 만든다.

Second\_layout의 텍스트 뷰를 바꾸고

Mainactivity가 있는 폴더에 FirstLayout클래스를 추가한다.

해당 클래스에서 Fragment를 상속받는다(extends)

Fragment 는 화면을 구성할 때 많은 view들이 사용되는데 이를 정리하기 위해 화면을 쪼개는 것이다.



Alt + Insert를 통해 override method를 선택한다.

On create view를 선택한다.

View는 화면을 구성하는 모든 객체이다. 버튼,테이블,입력창,레이아웃 등등

눈에 보이는 것은 위젯, 보이지 않는 것을 레이아웃 이라고 한다.

서로 중첩될 수 있다.

view객체를 만들고 다음과 같이 설정한다.

public class FisrtLayout extends Fragment {  
 View v;  
 @Nullable  
 @Override  
 public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {  
 v = inflater.inflate(R.layout.*first\_layout*,container,false);  
 return v;  
 }  
}

inflater 는 xml(사전에 만들어둔 레이아웃)로 정의된 view를 실제 객체화 시킨다.

첫번째 인자로 레이아웃 아이디, 두번째 인자로 부분 화면을 로드할 부모 컨테이너를 지정, 세번째 인자로 두번째 인자에 로드 될지 아닐지를 지정?

해당 클래스를 복사하여 secondlayout으로 붙여넣기한다.(코드수정 필요)

여기까지가 화면을 만들고 등록하는 부분

이후 메뉴바에 추가하는 부분

Menu 폴더에 Activity main drawer 파일로 이동한다.

우리는 2개의 화면만 있으므로 item을 두개만 남기고 이름을 nav\_각 레이아웃의 이름으로 바꾼다.

//nav의 이유는 잘 모르겠음

Content\_main에 레이아웃 속성에

android:id="@+id/content\_main를 추가한다.

mainActivity 클래스에서

implements NavigationView.OnNavigationItemSelectedListener

를 추가한다 (안드로이드 스튜디오 버전에 따라 다름)

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)  
 .setDrawerLayout(drawer)  
 .build();\*/  
ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(  
 this,drawer,toolbar,R.string.*app\_name*,R.string.*app\_name*);  
drawer.setDrawerListener(toggle);  
toggle.syncState();  
/\*NavController navController = Navigation.findNavController(this, R.id.nav\_host\_fragment);  
NavigationUI.setupActionBarWithNavController(this, navController, mAppBarConfiguration);  
NavigationUI.setupWithNavController(navigationView, navController);\*/  
NavigationView navigationView = (NavigationView)findViewById(R.id.*nav\_view*);  
navigationView.setNavigationItemSelectedListener(this);

예제에 따라 다음과 같이 수정한다.

@Override  
public boolean onOptionsItemSelected(MenuItem item) {  
 int id = item.getItemId();  
 if(id == R.id.*action\_settings*){  
 return true;  
 }  
 return super.onOptionsItemSelected(item);  
}  
  
@Override  
public boolean onNavigationItemSelected(MenuItem item) {  
 int id = item.getItemId();  
 FragmentManager manager = getFragmentManager();  
 if (id == R.id.*nav\_first\_layout*){  
 manager.beginTransaction().replace(R.id.*content\_main*,new FirstLayout()).commit();  
 }  
 else if(id == R.id.*nav\_second\_layout*){  
 manager.beginTransaction().replace(R.id.*content\_main*,new SecondLayout()).commit();  
 }  
 DrawerLayout drawerLayout = (DrawerLayout)findViewById(R.id.*drawer\_layout*);  
 drawerLayout.closeDrawer(GravityCompat.*START*);  
 return true;  
}

다음과같이 override한다.

Import시 버전에 따른 차이에 주의한다.

리스트 뷰 꾸미기

리스트 뷰를 추가한다.