

[액티비티]

이전 액티비티에서 넘어온 인텐트 받는방법

```
Intent intent = getIntent();
if(intent!=null){
    binding.textViewFromMainActivity.setText(intent.getStringExtra( name: "extra"));
}
```

다이얼로그 형식 적용하기

manifest파일의 activity에 theme으로 다이얼로그형태 적용하기

코드에서 백 버튼을 누른 것과 같은 효과를 내는 액티비티 메소드 : finish();

putExtra("key","value")로 보내면 받는 쪽에선 getXXXExtra("key")으로 값을 찾아서 사용

startActivityForResult vs. 런처 → 결과를 기대하는 애들

▶ putExtra로 requestCode같이 보내고, 이 대답을 받았을 때의 콜백 메소드(onActivityResult)도 오버라이드 해서 이 함수 내에서 requestCode받아서 처리한다.

▶ ActivityResultLauncher객체는 registerForActivityResult() 함수로 만들어짐 여기서 new ActivityResultContracts.StartActivityForResult해서 리스너 단다.

requestCode를 직접 만들 필요가 없다

TextView의 자식이 아닌데 상태 복원해야 할 때

1.

```
<activity
    android:name=".MainActivity"
    android:configChanges="screenSize|orientation"
```

2.

```
@Override
protected void onSaveInstanceState(@NonNull Bundle outState) {
    Log.d( tag: "SSU", msg: "화면돌려도안사라지게저장");
    super.onSaveInstanceState(outState);

    outState.putString("save", binding.sampleEditText.getText().toString());
}

@Override
protected void onRestoreInstanceState(@NonNull Bundle savedInstanceState) {
    Log.d( tag: "SSU", msg: "아까 저장 해 놓은거 불러와");
    super.onRestoreInstanceState(savedInstanceState);

    binding.sampleEditText.setText(savedInstanceState.getString( key: "save"));
}
```

[리사이클러 뷰] <https://59595959.tistory.com/12> -> 최고다.....

이 링크 바탕으로 내가 만든 프로젝트는 " Adapter_practice "

#만드는 순서

1. 리사이클러 뷰 띄울 액티비티의 xml에 리사이클러뷰 적용
2. 한 행의 레이아웃을 내가 만들어야 함 → layout에 item.xml 생성해서 레이아웃 만들기
3. 위에 두 줄 강 외워라

```
binding.recyclerView.setLayoutManager(new LinearLayoutManager(context, this));  
binding.recyclerView.setAdapter(new MyAdapter(list));  
binding.recyclerView.addItemDecoration(new MyItemDecoration());
```

!!<리사이클러뷰>!! 구현순서

[일단 메인에서는..]

1. 무조건 private List<Book> data; 를 필드변수로 설정
2. 뉴 레이아웃 인플레터, 뉴 어댑터 바인딩
3. 리스트 new 해서 여기에 데이터 넣기

[뷰홀더에서는..]

1. 띄울 "뷰"를 필드변수로 선언
2. onCreate 안에서 "itemView"의 아이디 찾아옴.

[어댑터에서는..]

1. 필드변수 설정, 생성자 안에 필드변수 넣어서 세팅! 두 개 세트임
2. create뷰홀더: 인플레이트해서 return하는데 개발자 문서 참고
3. bind뷰홀더: Book형 data 선언해서 위에서 만든 필드변수에서 .get(position)
4. holder. 보이고싶은 뷰홀더에서 찾아온 뷰아이디 .setText(data.getText());

[프래그먼트-바텀네비게이션] 내가 연습한 프로젝트는 " Fragment_Practice "

[바텀шит]

#방법1

그냥 BottomSheet로 동작하게 만든 layout을 코디네이터레이아웃 안에 include. (view="@")
그리고 자바 코드에서 아이디 찾아서 씀

#방법2) 다이얼로그 형태 띄우기

1.안드로이드에서 제공함 (필수)

```
private BottomSheetDialog dialog; //필수요소
```

2.onCreate()함수 안 (필수)

```
dialog=new BottomSheetDialog( context: this);
```

이 액티비티에 바텀шит다이얼로그 생성

3. 액티비티의 다이얼로그 띄우는 용도의 버튼 누르면 ..

- 1) 그 때 dialog로 쓸 애의 View를 inflate 해야함. 그리고 그 View를 아까 만든 dialog에 세팅.
- 2) 다이얼로그 안에서 일어나는 이벤트를 처리해야함 → 함수
- 3) dialog.show(); 띄워야 하니까 빼먹지마!!

```
button.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        View contentView = SecondActivity.this.getLayoutInflater().inflate(R.layout.bottom_sheet, root: null);  
        dialog.setContentView(contentView);  
        DialogListener(contentView); //다이얼로그 안 이벤트 처리 함수  
        dialog.show();  
    }  
});
```

4. 다이얼로그 이벤트 처리 함수 안에서는..

여기서 필요한거 아이디 찾고, 버튼 리스너 달고 리스너 안에서 dialog.dismiss();

[데이터베이스]

1.파일에 쓸 때

쉐얼더프리프런스를 get쉐얼드프리프런스(폴더이름,모드) 해서 얻음

쉐얼더프리프런스.에디터 생성,접근 → edit()실행

editor에 데이터 넣고 → commit(); 실행

```
binding.btn.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
  
        text=binding.content.getText().toString();  
  
        SharedPreferences global = getSharedPreferences( name: "my_folder",MODE_PRIVATE);  
        SharedPreferences.Editor editor=global.edit();  
  
        editor.putString("data1",text);  
        editor.commit();  
        showToast("file save");  
  
        startActivity(new Intent( packageContext: SharedPreferences.this,ReadFileActivity.class));  
    }  
});
```

2.파일에서 읽어 올 때

첫번째줄 똑같이 쓰고.

읽고싶은 데이터 타입 변수 선언.

첫번째줄 변수이름.getStrinf(키,벨류)