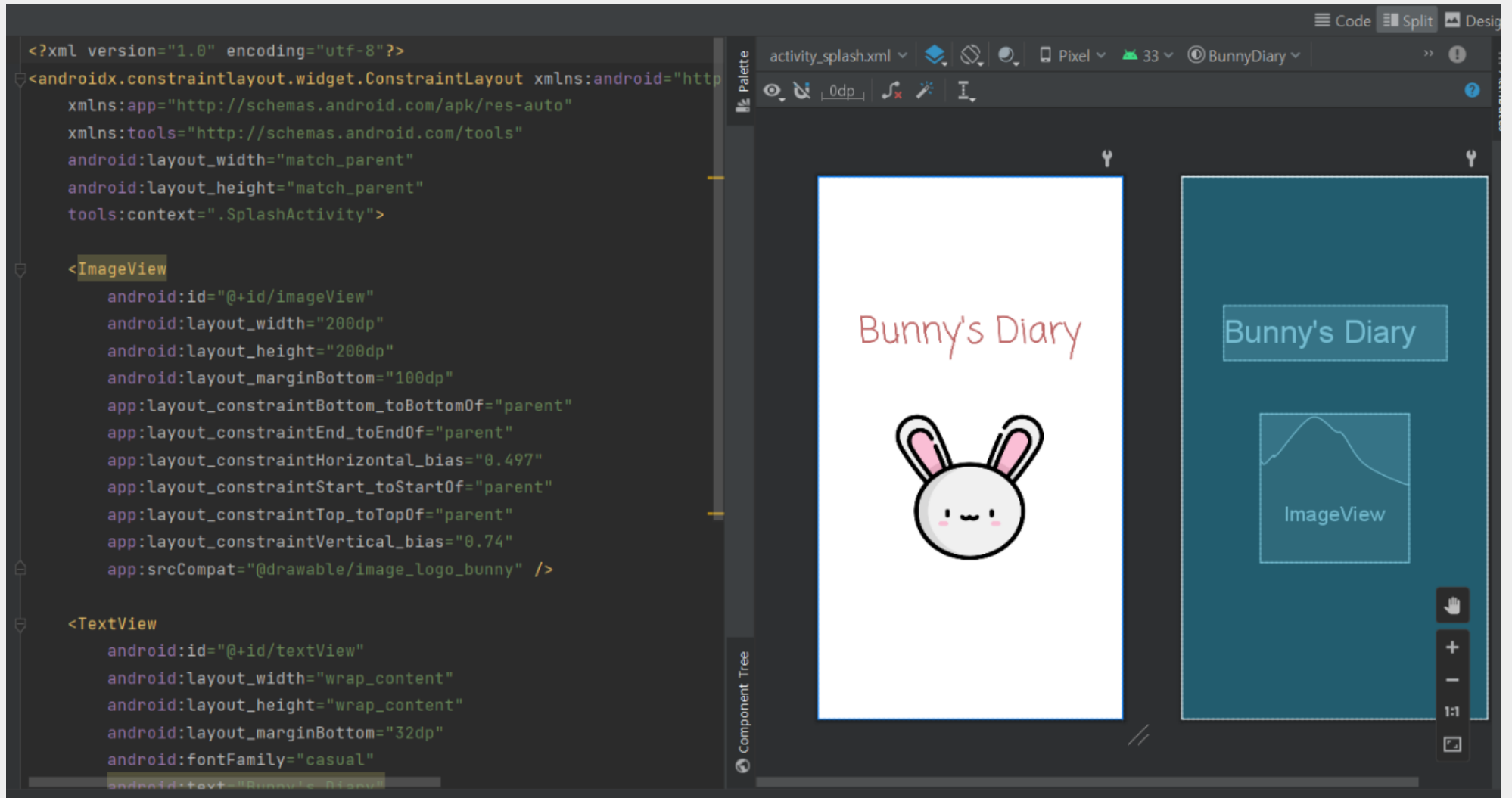


로딩페이지



로딩페이지

```
package com.example.bunnydiary

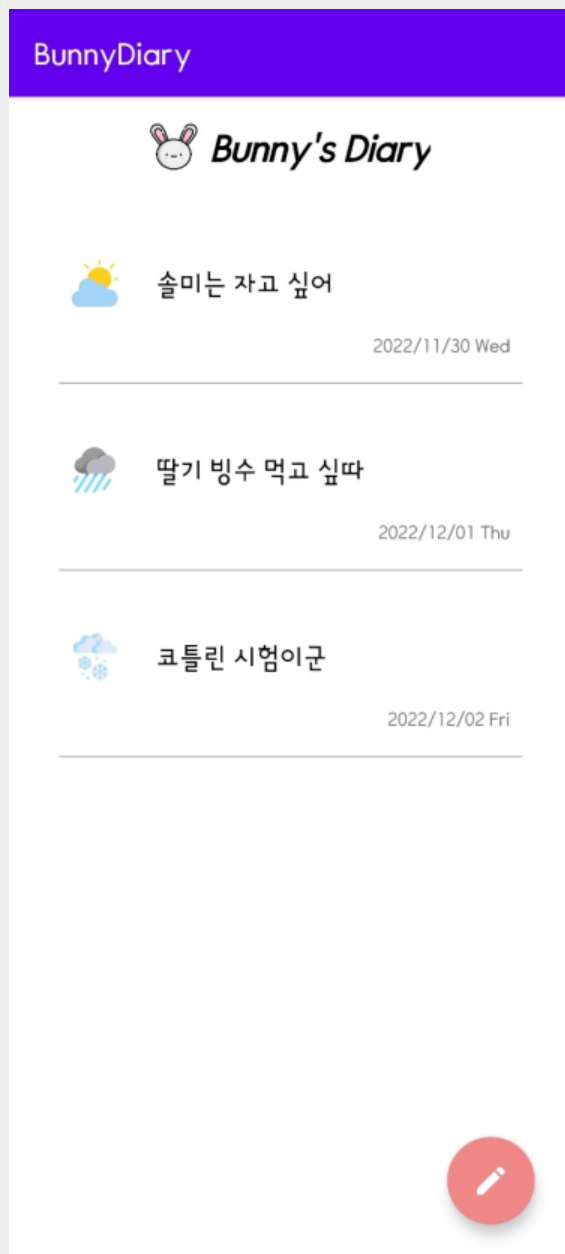
import ...

class SplashActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_splash)
        //딜레이를 발생시켜 그 이후에 메인 액티비티로 이동

        Handler(Looper.getMainLooper()).postDelayed({
            val intent = Intent(this@SplashActivity, MainActivity::class.java)
            startActivity(intent) },3000)
    }
}
```

딜레이(3초)를
발생시킨 후
메인 페이지로 이동

메인페이지



```
context = findViewById<Context>()

<!-- 리니어 레이아웃 : orientation 방향으로 쌓아 나감-->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <!--Header-->
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:layout_margin="16dp"
        android:orientation="horizontal">

        <ImageView
            android:layout_width="30dp"
            android:layout_height="30dp"
            android:src="@drawable/image_logo_bunny"/>

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Bunny's Diary"
            android:layout_marginLeft="8dp"
            android:textColor="@color/black"
            android:textSize="24sp"
            android:textStyle="bolditalic"/>
    </LinearLayout>
</LinearLayout>
```



일기 쓰기, 수정 페이지

```
<!--date-->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="16dp"
    android:orientation="vertical">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="날짜"
        android:textSize="18sp"
        android:textStyle="bold"
        android:textColor="@color/black"/>
    <TextView
        android:id="@+id/tvDate"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="2022/12/01 목요일"
        android:textColor="#039be5"
        android:textStyle="bold"
        android:textSize="12sp"
        android:layout_marginTop="4dp"/>
</LinearLayout>
```



코드

```
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:state_checked="false" android:drawable="@drawable/snow_gray" />
    <item android:state_checked="true" android:drawable="@drawable/snow" />
</selector>
```

날씨 버튼을 체크하면
무색 -> 컬러로 이미지 변환

```
inner class ViewHolder(view: View): RecyclerView.ViewHolder(view) {
    // 하나의 아이템 뷰
    init{
        itemView.setOnClickListener(View.OnClickListener {
            //현재 클릭이 된 위치(배열 개념이어서 첫 시작이 0부터 세는 기준)
            var currentPosition: Int = adapterPosition
            //현재 클릭 된 리스트 아이템 정보를 가지는 변수
            var diaryModel: DiaryModel = mLstDiary[currentPosition]

            //화면 이동 및 다이어리 데이터 다음 액티비티로 전달
            var diaryDetailIntent: Intent = Intent(mContext, DiaryDetailActivity::class.java)
            diaryDetailIntent.putExtra("diaryModel", diaryModel) // 다이어리 데이터 넣기기
            diaryDetailIntent.putExtra("mode", "detail") //상세보기 모드로 설정
            mContext.startActivity(diaryDetailIntent)
        })
    }
}
```



코드

```
// 선택지 옵션 팝업(수정, 삭제)
itemView.setOnLongClickListener(View.OnLongClickListener{
    //현재 클릭이 된 위치(배열 개념이어서 첫 시작이 0부터 세는 기준)
    var currentPosition:Int = adapterPosition
    //현재 클릭 된 리스트 아이템 정보를 가지는 변수
    var diaryModel:DiaryModel = mLstDiary[currentPosition]

    var strChoiceArray = arrayOf<String>("수정 하기", "삭제 하기")
    // 팝업화면 표시
    AlertDialog.Builder(mContext).setTitle("원하시는 동작을 선택하세요.").setItems(strChoiceArray, DialogInterface.OnClickListener{
        dialogInterface, position -> if (position==0){
            //수정하기 버튼 눌렀을 때

            //화면 이동 및 다이어리 데이터 다음 액티비티로 전달
            var diaryDetailIntent:Intent = Intent(mContext, DiaryDetailActivity::class.java)
            diaryDetailIntent.putExtra("diaryModel", diaryModel) // 다이어리 데이터 넘기기
            diaryDetailIntent.putExtra("mode", "modify") //수정하기 모드로 설정
            mContext.startActivity(diaryDetailIntent)
        }else{
            //삭제하기 버튼 눌렀을 때
            mLstDiary.removeAt(currentPosition)
            notifyItemRemoved(currentPosition)
        }
    }).show()
    return@OnLongClickListener false
})
}
```



코드

```
override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): DiaryListAdapter.ViewHolder {
    //아이템 뷰가 최초로 생성이 될 때 호출 되는 곳
    mContext = parent.context
    var holder:View = LayoutInflater.from(mContext).inflate(R.layout.list_item_diary,parent,false)
    return ViewHolder(holder)
}

override fun onBindViewHolder(holder: DiaryListAdapter.ViewHolder, position: Int) {
    // 생성된 아이템 뷰가 실제 연동이 되어지는 곳

    //날씨의 경우의 수 작성
    var weatherType:Int = mLstDiary[position].weatherType
    when(weatherType){
        //gm=
        0 -> holder.ivWeather.setImageResource(R.drawable.cloud);//흐림
        1 -> holder.ivWeather.setImageResource(R.drawable.wind);//바람
        2 -> holder.ivWeather.setImageResource(R.drawable.cloudy);//흐린 뒤 맑음
        3 -> holder.ivWeather.setImageResource(R.drawable.rainy);//비
        4 -> holder.ivWeather.setImageResource(R.drawable.snow);//눈
        5 -> holder.ivWeather.setImageResource(R.drawable.sun);//맑음
    }

    //날씨 제목, 날짜
    var title:String = mLstDiary[position].title
    var userDate:String = mLstDiary[position].userDate
    holder.tvTitle.text = title
    holder.tvUserDate.text = userDate
}

fun setOnItemClickListener(listener: OnDiaryClickListener){
```



코드

```
import ...
//
/*
 *상세보기 화면 or 작성하기 화면
 */

//class DiaryDetailActivity : AppCompatActivity(){}
class DiaryDetailActivity : AppCompatActivity(),View.OnClickListener {

    private lateinit var mTvDate: TextView //일시 설정 텍스트
    private lateinit var mEtTitle: EditText //일기 제목
    private lateinit var mEtContent: EditText //일기 내용
    private lateinit var mRqWeather: RadioGroup

    private var mDetailMode: String = "" // intent로 받아낸 게시글 모드
    private var mBeforeDate: String = "" // intent로 받아낸 게시글 기준 작성일자
    private var mSelectedUserDate: String = "" //선택된 날짜
    private var mSelectedWeatherType = -1 //선택된 날씨 값(0~5)

    private val sharedPreferences by lazy{
        sharedPreferences("record", Context.MODE_PRIVATE)
    }

    private val binding by lazy {
        ActivityDiaryDetailBinding.inflate(layoutInflater)
    }
```




코드

```
override fun onCreate(savedInstanceState: Bundle?) {  
    super.onCreate(savedInstanceState)  
    setContentView(binding.root)  
  
    mTvDate = binding.tvDate //날짜 설정 텍스트  
    mEtTitle = binding.etTitle //제목 입력 필드  
    mEtContent = binding.etContent //내용 입력 필드  
    mRgWeather = binding.rgWeather // 날씨 선택 라디오 그룹  
  
    var ivBack: ImageView = binding.ivBack //뒤로가기 버튼  
    var ivCheck: ImageView = binding.ivCheck //작성완료 버튼  
  
    ivBack.setOnClickListener(this) //클릭 기능 부여  
    ivCheck.setOnClickListener(this) //클릭 기능 부여  
    mTvDate.setOnClickListener(this) //클릭 기능 부여  
  
    //기본으로 설정될 날짜의 값을 지정(디바이스 현재 시간 기준)  
    mSelectedUserDate = SimpleDateFormat("yyyy/MM/dd E요일", Locale.KOREA).format(Date())  
    mTvDate.text = mSelectedUserDate  
}
```



코드

```
override fun onClick(view: View) {  
    //setOnClickListener가 붙어있는 뷰들은 클릭이 발생하면 모두 이곳을 수행하게 됨  
    var bool: Boolean = mEtTitle.text.isEmpty() || mEtContent.text.isEmpty()  
    when (view.id) {  
        //뒤로가기  
        R.id.ivBack -> finish() //현재 액티비티 종료  
  
        //작성 완료  
        //라디오 그룹의 버튼 클릭 현재 상황 가져오기  
        R.id.ivCheck -> {  
            mSelectedWeatherType =  
                mRgWeather.indexOfChild(findViewById(mRgWeather.checkedRadioButtonId))  
            //입력 필드 작성란이 비어있는지 확인  
            if (bool) {  
                //error  
                Toast.makeText(this, "입력되지 않은 필드가 존재합니다.", Toast.LENGTH_SHORT).show()  
                return  
            }  
            //날씨 선택이 되어있는지 체크  
            if (mSelectedWeatherType == -1) {  
                //error  
                Toast.makeText(this, "날씨를 선택해주세요.", Toast.LENGTH_SHORT).show()  
                return  
            }  
            //여기까지 도착했으면 에러 상황이 없으므로 데이터 저장  
  
            with(sharedPreference.edit()){  
                var title: String = mEtTitle.text.toString() // 제목 입력 값
```



```
var writeDate: String =
    SimpleDateFormat("yyyy/MM/dd HH:mm:ss", Locale.KOREA).format(Date())
//데이터베이스에 저장
this.putString("title",title)
this.putString("content",content)
this.putString("date",userDate)
finish()// 현재 액티비티 종료
}

//일시 설정 텍스트
R.id.tvDate -> {var calender:Calendar = Calendar.getInstance()
    val dateSetListener = DatePickerDialog.OnDateSetListener { datePicker, year, month, day ->
        var innerCal:Calendar = Calendar.getInstance()
        innerCal.set(Calendar.YEAR,year)
        innerCal.set(Calendar.MONTH,month)
        innerCal.set(Calendar.DAY_OF_MONTH, day)

        mSelectedUserDate = SimpleDateFormat("yyyy/MM/dd E요일", Locale.KOREA).format(innerCal.time)
        mTvDate.text = mSelectedUserDate
    }
    DatePickerDialog(this, dateSetListener, calender.get(Calendar.YEAR),calender.get(Calendar.MONTH),calender.get(Calendar.DAY_OF_MON
})
```