**모바일 프로그래밍**

**Term Project**

**최종 보고서**

모바일프로그래밍 공기석 교수님

제출일자 : 2021.05.26

1팀

2017156042 최성락

2018150039 정주희

2018152028 윤정아

2018256035 정소이

**<목차>**

1. **서론**
   1. **주제………………………………………………….….2**
   2. **주제 선정 이유………………………………….…..2**
   3. **일정………………………………………………….….2**
   4. **역할분담……………………………………………….3**
   5. **회의록…………………………………………………..3**
   6. **구현상의 제약사항………………………………….7**
2. **본론**
   1. **소스코드 및 분석…………………………………..7**
   2. **실행화면……………………………………………..64**
3. **결론**

**3.1 보완할 사항…………………………………………..67**

* 1. **느낀점…………………………………………………68**

1. **서론**
   1. **주제**

“ 쉽고 간편한 MBTI 성격유형 검사 앱”

* 1. **주제 선정 이유**

요즘 MBTI 성격 유형 검사가 유행하며 자신의 MBTI에 대한 관심이 높아지고 자연스레 타인의 MBTI에 대한 관심도 같이 증가하였습니다. MBTI 성격 유형을 통해 자신 및 타인에 대해 쉽게 이해할 수 있고 그로 인한 대인관계 갈등 조정과 조직의 효율적인 운영이 가능합니다. 또한, 검사 결과에 본인 및 타인을 비교해보며 각각의 공통점, 및 차이점을 찾아보는 재미도 있습니다. 그러나, 기존 MBTI 성격 유형 검사는 총 60문항으로 검사하는데 많은 시간이 소요되며 복잡합니다. 이에 쉽고 간편하게 MBTI 검사를 수행할 방법에 대해 강구하게 되었고 12문항만으로도 MBTI 성격 유형을 검사할 수 있는 앱을 만들자는 결론이 도출되었습니다.

* 1. **일정**

****

* 1. **역할분담**
* 최성락 : 발표, 전체 xml, java 틀 구현, 검사화면 xml, java 구현, DB xml, java 구현
* 정주희 : 최종 PPT 제작, 검사 결과 xml, java 구현, 구분기준 xml, java 구현
* 윤정아 : 최종 보고서 제작, 메인 화면 xml, java 구현, 유명인사 xml, java 구현
* 정소이 : 팀장, 전체 레이아웃 및 디자인 통일, 궁합 xml, java 구현
  1. **회의록**

**<첫번째 회의록>**

|  |  |  |  |
| --- | --- | --- | --- |
| 일시 | 2021.05.03 | 장소 | E동 5층 E룸 |
| 작성자 | 윤정아 | 작성일 | 2021.05.03 |
| 참가자 | 최성락, 정주희, 윤정아, 정소이 | | |
| 회의주제 | 제안서 작성을 위한 주제선정 및 액티비티 결정 | | |
| 회의내용 | * 각자 미리 생각해온 주제에 대해 의견을 나누고 투표함. * 주변 주유소 알려주는 앱, 주변 맛집 알려주는 앱, 캘린더에 일별로 사진 넣을 수 있는 일기장 앱, 세탁기의 시작 및 끝 시간을 알려주고 빨래 일정을 알려주는 세탁기 일정 앱, 쉽고 간편한 MBTI 검사 앱이 투표 항목으로 선정됨. * 투표 결과 쉽고 간편한 MBTI 검사 앱이 주제로 결정됨. * 선정된 주제에 맞게 4개 이상의 액티비티를 구상함. * 시작 화면 -> 검사 화면 -> 검사 결과 화면 -> 구분 기준 화면 -> 유명인 확인 화면 -> 궁합 확인 화면 순으로 구현하기로 함. * 시작 화면에서는 검사 액티비티로 넘어가는 시작 버튼과 앱 설명 다이얼로그가 나타나는 설명 버튼을 구현하기로 함. * 검사 화면에서는 검사 항목과 YES, NO 버튼을 만들어 항목에 대한 답을 도출하고 마지막으로 YES, NO 의 개수를 조합하여 사용자의 성격 유형을 출력하는 결과 보기 버튼을 구현하기로 함. * 검사 결과 화면에서는 사용자의 성격 유형 결과를 보여주는 text view를 구현하고 구분 기준 버튼, 유명인 버튼, 궁합버튼, 처음으로 버튼, 종료 버튼을 구현하여 각각의 액티비티로 넘어가게 구현하기로 함. * 구분 기준 화면에서는 성격 유형인 E/I, S/N, T/F, P/J를 각각 탭으로 만들어 누르면 성격 유형을 구분한 기준에 대해 설명을 보여주고 하단에 처음으로, 종료 버튼을 구현하기로 함. * 유명인 확인 화면에서는 MBTI 유형 결과를 보여주고 그 결과에 따른 유명인 2명과 각각의 이름을 출력해주기로 함. 하단에 이전으로, 처음으로, 종료 버튼을 구현하기로 함. * 궁합 확인 화면에서는 각 MBTI 유형 결과에 따른 최고, 최악의 궁합을 보여주기로 함. 이전으로 처음으로 종료 버튼을 구현하기로 함. | | |

**<두번째 회의록>**

|  |  |  |  |
| --- | --- | --- | --- |
| 일시 | 2021.05.10 | 장소 | 제1생활관 쥬씨 |
| 작성자 | 윤정아 | 작성일 | 2021.05.10 |
| 참가자 | 최성락, 정주희, 윤정아, 정소이 | | |
| 회의주제 | MBTI 검사 앱에 필요한 세부적인 자료조사 | | |
| 회의내용 | * MBTI 검사 앱을 만들기 위한 자료조사를 진행함. * MBTI 유형 해석, 궁합 관계 표, 해설, 질문 해석 등에 대한 자료를 공유함. * 각각 E/I, S/N, T/F, P/J 4가지 유형에서 하나씩 담당하여 유형에 대해 YES / NO로 대답할 수 있는 질문 3가지를 작성함. * 사다리 게임을 통해 성격 구분기준 설명, 궁합, 유명인사, 성격 검사 결과 설명 4가지 분야를 각각 하나씩 담당하여 세부적인 자료조사를 진행함. * 성격 구분 기준 설명은 각각의 성격 유형에 대해 5줄 정도로 요약하여 정리하였고 궁합은 최고인 궁합부터 최악인 궁합까지 정도를 나누어 2~5개의 MBTI를 작성함. * 유명인사는 한국인 배우 및 아이돌로 선정하여 각 MBTI당 여자 연예인 1명, 남자 연예인 1명의 사진 및 이름을 작성함. * 성격 검사 결과 설명은 각 MBTI에 해당하는 내용을 한 줄로 나타내고 세부적인 내용을 2~3줄로 요약하여 작성함. * 각자 맡은 분야를 hwp 또는 docx 파일로 작성하여 카카오톡을 활용해 공유함. | | |

**<세번째 회의록>**

|  |  |  |  |
| --- | --- | --- | --- |
| 일시 | 2021.05.17 | 장소 | 제2생활관 그룹활동실 1 |
| 작성자 | 윤정아 | 작성일 | 2021.05.17 |
| 참가자 | 최성락, 정주희, 윤정아, 정소이 | | |
| 회의주제 | 각자 구현한 xml, java 코드에 대한 의견 공유 | | |
| 회의내용 | * 제2생활관 그룹활동실 1을 공간 예약하여 회의를 진행함. * 위 역할분담에서 언급한대로 각자 구현한 xml과 java 코드를 서로 공유하고 의견을 나눔. * 공유한 xml과 java 코드를 한 프로젝트에 통합하여 변수 및 색상 여백 등 세부적인 내용을 통일함. * 구현한 부분 중 막힌 부분이나 어려운 부분에 대해 서로 의견을 주고받으며 코드를 수정함. * 최종 보고서와 최종 PPT에 들어갈 내용 및 정보를 정함. * 최종 보고서에 주제, 주제 선정 이유, 일정, 역할분담, 구현상의 제약사항, 소스코드, 소스코드 분석, 실행화면, 느낀점, 보완할 사항에 대한 내용을 작성하기로 함. * 최종 PPT에는 기존 제안서를 기반으로 교수님께서 지적해주신 부분을 보완하고 개발하며 바뀐 부분에 대한 소스코드 및 설명, 실행화면 등을 추가 작성하기로 함. * PPT제작, 보고서 작성, 디자인 수정, DB 구현 및 발표에 대한 담당자를 나누고 회의를 마침. | | |
| 사진 | 텍스트, 사람, 노트북, 실내이(가) 표시된 사진  자동 생성된 설명 | | |

**<네번째 회의록>**

|  |  |  |  |
| --- | --- | --- | --- |
| 일시 | 2021.05.25 | 장소 | E동 4층 421호 |
| 작성자 | 윤정아 | 작성일 | 2021.05.25 |
| 참가자 | 최성락, 정주희, 윤정아, 정소이 | | |
| 회의주제 | 텀 프로젝트 발표 전 최종점검 | | |
| 회의내용 | * 텀 프로젝트 발표 전 최종적인 점검을 위해 회의를 진행함. * 회의는 빈 강의실(E동 421호)에서 약 2시간 정도 진행함. * 최종적으로 구현이 끝난 프로젝트를 실행시켜 보며 디자인 및 세부사항 들에 대해 의견을 나누며 수정을 거침. * 보다 높은 완성도를 위해 현재 개발되어 있는 내용에서 더 좋은 방향으로 고칠 수 있는지 의견을 공유하고 수정 및 실행함. * 앱을 처음 사용하는 사람들도 쉽게 보고 이해할 수 있도록 가독성 높이기 위해 설명을 정확하고 간결하게 수정함. * 최종 보고서의 내용을 꼼꼼하게 확인하며 서로 더 쓰고 싶은 내용을 공유하고 수정을 마침. * 최종 제작된 PPT를 다 함께 공유하며 추가적인 내용을 수정하고 발표 연습을 진행함. * 발표자에게 각자 개발한 xml 및 java에서 중요하게 생각되는 부분을 정리 및 요약하여 전달함. * 성공적인 발표를 위해 팀원들 간 의지를 다지며 최종 연습을 하고 회의를 마침. | | |
| 사진 | 사람, 의류, 실내, 창문이(가) 표시된 사진  자동 생성된 설명 | | |

* 1. **구현상의 제약사항**
* 4개 이상의 액티비티를 구현해야 한다.
* 이번 학기 학습한 내용을 포함하여 구현해야 한다
* AVD 상에서 구현 가능해야 한다.
* MBTI 검사 결과는 저장 가능해야 한다.

1. **본론**
   1. **소스코드 및 분석**

**<AndroidMenifest.xml 소스코드>**

*<?*xml version="1.0" encoding="utf-8"*?>*<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="kr.ac.kpu.termproject">  
  
 <application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/Theme.TermProject">  
 <activity android:name=".MainActivity">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 <activity android:name=".CheckActivity" android:label="Check Activity"/>  
 <activity android:name=".CheckResultActivity" android:label="Checkresult Activity"/>  
 <activity android:name=".DivisionActivity" android:label="Division Activity"/>  
 <activity android:name=".CelebrityActivity" android:label="Celebrity Activity"/>  
 <activity android:name=".MatchedActivity" android:label="Matched Activity"/>  
 <activity android:name=".DBActivity"/>  
 </application>  
  
</manifest>

**<AndroidMenifest.xml 소스코드 분석>**

<activity android:name=".CheckActivity" android:label="Check Activity"/>  
<activity android:name=".CheckResultActivity" android:label="Checkresult Activity"/>  
<activity android:name=".DivisionActivity" android:label="Division Activity"/>  
<activity android:name=".CelebrityActivity" android:label="Celebrity Activity"/>  
<activity android:name=".MatchedActivity" android:label="Matched Activity"/>  
<activity android:name=".DBActivity"/>

MainActivity 이외의 액티비티 사용을 위해 액티비티 등록

**<activity\_main.xml 소스코드>**

*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity"  
 android:orientation="vertical">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:layout\_marginTop="200dp">  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:text="쉽고 간편한"  
 android:textSize="35sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:gravity="center">  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:text="MBTI "  
 android:textSize="35sp"  
 android:textStyle="bold"  
 android:textColor="#1976D2"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:text="검사"  
 android:textSize="35sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:layout\_marginTop="45dp">  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/startbtn"  
 android:text="시작"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/exbtn"  
 android:text="설명"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <ImageView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="100dp"  
 android:scaleType="fitXY"  
 android:src="@drawable/main">  
 </ImageView>  
</LinearLayout>

**<activity\_main.xml 소스코드 분석>**

* 쉽고 간편한 MBTI 검사를 TextView로 보여주고 MBTI를 강조하기 위해 색을 따로 지정해줌.
* Main 화면의 버튼은 시작버튼과 설명 버튼이 있으며 시작 버튼은 말 그대로 검사의 시작을 의미하며 설명 버튼을 누르면 앱에 대한 설명이 나옴. 이 두 버튼은 orientation을 vertical로 설정하여 수직적으로 화면에 출력됨
* 가장 하단에는 MBTI 검사에 맞는 ImageView를 넣어 산뜻한 느낌을 줌

**<MainActivity.java 소스코드>**

package kr.ac.kpu.termproject;  
  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
import android.content.Intent;  
import android.graphics.Color;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
  
public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 Button startbtn = (Button)findViewById(R.id.*startbtn*);  
 final Button exbtn = (Button)findViewById(R.id.*exbtn*);  
  
 startbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(),CheckActivity.class);  
 startActivity(intent);  
 }  
 });  
  
 exbtn.setOnClickListener(new View.OnClickListener(){  
 public void onClick(View v){  
 AlertDialog.Builder dlg = new AlertDialog.Builder(MainActivity.this);  
 dlg.setIcon(R.drawable.*icon\_explain*);  
 dlg.setTitle("MBTI 검사란?");  
 setTitleColor(Color.*parseColor*("#1976D2"));  
 dlg.setMessage("개인이 쉽게 응답할 수 있는 자기보고식 문항을 통해 각자 선호하는 경향을 찾고," +  
 "\n이러한 선호 경향들이 인간의 행동에 어떠한 영향을 미치는가를 파악하여" +  
 "\n실생활에 응용하도록 제작된 심리 검사이다.");  
 dlg.setPositiveButton("확인",null);  
 dlg.show();  
 }  
 });  
  
 }  
}

**<MainActivity.java 소스코드 분석>**

* setOnClickListener 를 활용하여 시작 버튼을 누르면 CheckActivity로 넘어가 검사를 시작할 수 있게 intent를 할당해줌
* 설명 버튼을 누르면 앱에 대한 설명을 담고 있는 대화상자가 나오게 구현함
* 대화상자의 상단엔 MBTI검사란? 제목이 뜨고 중간에는 앱에 대한 설명을, 오른쪽 하단에는 확인 PositiveButton이 나오게 구현함
* 대화상자의 확인 버튼을 누르면 대화상자가 닫히고 다시 Main 화면으로 돌아감

**<check.xml 소스코드>**

*<?*xml version="1.0" encoding="utf-8"*?>*<RelativeLayout  
 xmlns:android=<http://schemas.android.com/apk/res/android>

xmlns:app="http://schemas.android.com/apk/res-auto"

android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
  
 <ScrollView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="35dp"  
 android:layout\_marginRight="35dp"  
 android:orientation="vertical">  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="47dp"  
 android:text="MBTI 검사"  
 android:textSize="30sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="60dp"  
 android:layout\_marginBottom="12dp"  
 android:textSize="19sp"  
 android:textColor="@color/black"  
 android:text="1. 데이트가 없는 주말에 나는 친구들에게 연락해 약속을 잡는다."/>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:orientation="horizontal">  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginRight="10dp"

app:backgroundTint="#cccccc"

android:id="@+id/E1"  
 android:text="YES"  
 android:textSize="15sp"  
 android:textStyle="bold"

android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/I1"  
 android:text="NO"  
 android:textSize="15sp"  
 android:textStyle="bold"

android:textColor="@color/black"/>  
 </LinearLayout>  
  
----------------------(TextView1개,Button2개)x11개 생략--------------------------

<Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:layout\_marginTop="68dp"  
 android:layout\_marginBottom="40dp"

app:backgroundTint="#cccccc"  
 android:id="@+id/resultbtn"  
 android:text="결과 보기"  
 android:textSize="20sp"  
 android:textStyle="bold"

android:textColor="@color/black"/>  
 </LinearLayout>  
 </ScrollView>  
</RelativeLayout>

**<check.xml 소스코드 분석>**

<RelativeLayout  
 android:orientation="vertical"></RelativeLayout>

RelativeLayout을 이용한 레이아웃 배치를 이용해 깔끔한 UI를 제공함

<ScrollView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"></ ScrollView>

질문의 개수가 많아 화면을 넘어가기 때문에 ScrollView를 이용해 화면 스크롤로 질문들을 확인할 수 있도록 함

**<CheckActivity.java 소스코드>**

package kr.ac.kpu.termproject;  
  
import android.content.Intent;  
import android.graphics.Color;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.Toast;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class CheckActivity extends AppCompatActivity {  
 protected void onCreate(Bundle savedInstanceState){  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*check*);  
  
 int[] e = new int[3];  
 int[] i = new int[3];  
 int[] s = new int[3];  
 int[] n = new int[3];  
 int[] t = new int[3];  
 int[] f = new int[3];  
 int[] j = new int[3];  
 int[] p = new int[3];  
  
 Button resultbtn = (Button)findViewById(R.id.*resultbtn*);  
 Button E1 = (Button)findViewById(R.id.*E1*);  
 Button I1 = (Button)findViewById(R.id.*I1*);  
 Button S1 = (Button)findViewById(R.id.*S1*);  
 Button N1 = (Button)findViewById(R.id.*N1*);  
 Button T1 = (Button)findViewById(R.id.*T1*);  
 Button F1 = (Button)findViewById(R.id.*F1*);  
 Button P1 = (Button)findViewById(R.id.*P1*);  
 Button J1 = (Button)findViewById(R.id.*J1*);  
 Button E2 = (Button)findViewById(R.id.*E2*);  
 Button I2 = (Button)findViewById(R.id.*I2*);  
 Button S2 = (Button)findViewById(R.id.*S2*);  
 Button N2 = (Button)findViewById(R.id.*N2*);  
 Button T2 = (Button)findViewById(R.id.*T2*);  
 Button F2 = (Button)findViewById(R.id.*F2*);  
 Button P2 = (Button)findViewById(R.id.*P2*);  
 Button J2 = (Button)findViewById(R.id.*J2*);  
 Button E3 = (Button)findViewById(R.id.*E3*);  
 Button I3 = (Button)findViewById(R.id.*I3*);  
 Button S3 = (Button)findViewById(R.id.*S3*);  
 Button N3 = (Button)findViewById(R.id.*N3*);  
 Button T3 = (Button)findViewById(R.id.*T3*);  
 Button F3 = (Button)findViewById(R.id.*F3*);  
 Button P3 = (Button)findViewById(R.id.*P3*);  
 Button J3 = (Button)findViewById(R.id.*J3*);  
  
 E1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 e[0] = 1;  
 i[0] = 0;  
 E1.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 I1.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 I1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 e[0] = 0;  
 i[0] = 1;  
 I1.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 E1.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
  
 S1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 s[0] = 1;  
 n[0] = 0;  
 S1.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 N1.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 N1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 s[0] = 0;  
 n[0] = 1;  
 N1.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 S1.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 T1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 t[0] = 1;  
 f[0] = 0;  
 T1.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 F1.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 F1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 t[0] = 0;  
 f[0] = 1;  
 F1.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 T1.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 P1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 p[0] = 1;  
 j[0] = 0;  
 P1.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 J1.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 J1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 p[0] = 0;  
 j[0] = 1;  
 J1.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 P1.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
  
 E2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 e[1] = 1;  
 i[1] = 0;  
 E2.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 I2.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 I2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 e[1] = 0;  
 i[1] = 1;  
 I2.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 E2.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
  
 S2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 s[1] = 1;  
 n[1] = 0;  
 S2.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 N2.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 N2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 s[1] = 0;  
 n[1] = 1;  
 N2.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 S2.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 T2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 t[1] = 1;  
 f[1] = 0;  
 T2.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 F2.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 F2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 t[1] = 0;  
 f[1] = 1;  
 F2.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 T2.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 P2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 p[1] = 1;  
 j[1] = 0;  
 P2.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 J2.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 J2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 p[1] = 0;  
 j[1] = 1;  
 J2.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 P2.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
  
 E3.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 e[2] = 1;  
 i[2] = 0;  
 E3.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 I3.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 I3.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 e[2] = 0;  
 i[2] = 1;  
 I3.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 E3.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
  
 S3.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 s[2] = 1;  
 n[2] = 0;  
 S3.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 N3.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 N3.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 s[2] = 0;  
 n[2] = 1;  
 N3.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 S3.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 T3.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 t[2] = 1;  
 f[2] = 0;  
 T3.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 F3.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 F3.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 t[2] = 0;  
 f[2] = 1;  
 F3.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 T3.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 P3.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 p[2] = 1;  
 j[2] = 0;  
 P3.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 J3.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
 J3.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 p[2] = 0;  
 j[2] = 1;  
 J3.setBackgroundColor(Color.*parseColor*("#77ccff"));  
 P3.setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 });  
  
  
  
 resultbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
  
 int sume = 0, sumi = 0, sums = 0, sumn = 0, sumt = 0, sumf = 0, sumj = 0, sump = 0;  
 String EI, SN, TF, JP, MBTI;  
  
 for (int a = 0; a < i.length; a++) {  
 sume += e[a];  
 sumi += i[a];  
 sums += s[a];  
 sumn += n[a];  
 sumt += t[a];  
 sumf += f[a];  
 sumj += j[a];  
 sump += p[a];  
 }  
 int sumall = sume + sumi + sums + sumn + sumt + sumf + sumj + sump;  
 if (sumall != 12) {  
 Toast.*makeText*(getApplicationContext(), "모든 질문에 답해주세요.", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 if (sume > sumi) {  
 EI = "E";  
 } else {  
 EI = "I";  
 }  
 if (sums > sumn) {  
 SN = "S";  
 } else {  
 SN = "N";  
 }  
 if (sumt > sumf) {  
 TF = "T";  
 } else {  
 TF = "F";  
 }  
 if (sumj > sump) {  
 JP = "J";  
 } else {  
 JP = "P";  
 }  
  
 MBTI = EI + SN + TF + JP;  
  
 Intent intent = new Intent(getApplicationContext(), CheckResultActivity.class);  
 intent.putExtra("MBTI", MBTI);  
 startActivity(intent);  
 }  
 }  
 });  
  
 }  
}

**<CheckActivity.java 소스코드 분석>**

int[] e = new int[3];

각 MBTI 구분 질문에 해당하는 버튼의 체크 횟수 값을 확인하기 위해 선언함

E1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 e[0] = 1;  
 i[0] = 0;  
 E1.setBackgroundColor(Color.*parseColor*("#339966"));  
 I1.setBackgroundColor(Color.*parseColor*("#6600FF"));  
 }  
 });  
 I1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 e[0] = 0;  
 i[0] = 1;  
 I1.setBackgroundColor(Color.*parseColor*("#339966"));  
 E1.setBackgroundColor(Color.*parseColor*("#6600FF"));  
 }  
 });  
MBTI 구분 질문에 대한 답을 버튼 클릭 시 성향에 따른 체크횟수를 카운트하여 확인하기 위해 e[0]=1,i[0]=0 의 알고리즘을 이용해 마지막 결과 버튼 부분에 총합으로 Count할 예정이며, 클릭 시 버튼 색상의 변경으로 사용자가 어느 버튼을 선택했는지 확인이 가능하게 함

for (int a = 0; a < i.length; a++) {  
 sume += e[a];  
 sumi += i[a];  
 sums += s[a];  
 sumn += n[a];  
 sumt += t[a];  
 sumf += f[a];  
 sumj += j[a];  
 sump += p[a];  
 }

각 질문의 성향에 따른 체크횟수를 확인하기 위해 각 성향 별 체크 값들을 더함

int sumall = sume + sumi + sums + sumn + sumt + sumf + sumj + sump;  
 if (sumall != 12) {  
 Toast.*makeText*(getApplicationContext(), "모든 질문에 답해주세요.", Toast.*LENGTH\_SHORT*).show();  
 }

모든 질문이 총 12개 이므로 체크횟수의 총 합이 12가 아니면 진행하지 않도록 예외처리

if (sume > sumi) {  
 EI = "E";  
 } else {  
 EI = "I";  
 }  
 if (sums > sumn) {  
 SN = "S";  
 } else {  
 SN = "N";  
 }  
 if (sumt > sumf) {  
 TF = "T";  
 } else {  
 TF = "F";  
 }  
 if (sumj > sump) {  
 JP = "J";  
 } else {  
 JP = "P";  
 }  
  
 MBTI = EI + SN + TF + JP;

성향 별로 더했던 값들을 이용해 반대되는 성향끼리 값을 비교하여 본인에게 해당하는 성향의 알파벳을 결정한 후 마지막 MBTI에 한 문자열로 합침  
  
 Intent intent = new Intent(getApplicationContext(), checkresultActivity.class);  
 intent.putExtra("MBTI",MBTI);  
 startActivity(intent);  
 }  
 }  
 });  
  
합친 문자열 MBTI를 intent를 이용해 다른 액티비티에 전달함

**<checkresult.xml 소스코드 및 분석(주석)>**

*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:gravity="center"  
 android:layout\_weight="0.2">  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="나의 MBTI 성격 유형은?"  
 android:textSize="30sp"  
 android:layout\_marginTop="100dp"  
 android:gravity="center"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>

*<!--사용자의 MBTI 검사 결과에 대한 설명 출력-->*  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:layout\_weight="0.6">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:id="@+id/checktext"  
 android:layout\_marginTop="40dp"  
 android:layout\_marginBottom="20dp"  
 android:textSize="40sp"  
 android:text="&quot;MBTI 유형&quot;"  
 android:textStyle="bold"  
 android:textColor="#6A1B9A"/>  
  
 <TextView  
 android:id="@+id/exname"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:text="MBTI 한 줄 설명"  
 android:textSize="25sp"  
 android:textStyle="bold"/>  
  
 <TextView  
 android:id="@+id/extext"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="30dp"  
 android:gravity="center"  
 android:text="MBTI 설명"  
 android:textSize="16sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>

*<!--구분기준, 유명인, 궁합, 결과 저장하기, 처음으로,종료 버튼-->*  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:gravity="center"  
 android:layout\_weight="0.5">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:gravity="center">  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/divbtn"  
 android:text="구분 기준"  
 android:layout\_marginRight="10dp"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/celbtn"  
 android:text="유명인"  
 android:layout\_marginRight="10dp"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/matbtn"  
 android:text="궁합"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:gravity="center">  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/savebtn"  
 android:text="결과 저장하기"  
 android:layout\_marginRight="10dp"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/homebtn"  
 android:text="처음으로"  
 android:layout\_marginRight="10dp"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/exitbtn"  
 android:text="종료"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
 </LinearLayout>  
</LinearLayout>

**<CheckResultActivity.java 소스코드>**

package kr.ac.kpu.termproject;  
  
import android.content.Intent;  
import android.graphics.Color;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class CheckResultActivity extends AppCompatActivity {  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*checkresult*);  
  
 Intent intent = getIntent();  
 final String MBTI = intent.getStringExtra("MBTI");  
  
 TextView checktext = (TextView)findViewById(R.id.*checktext*);  
 TextView exname = (TextView)findViewById(R.id.*exname*);  
 TextView extext = (TextView)findViewById(R.id.*extext*);  
  
 Button divbtn = (Button)findViewById(R.id.*divbtn*);  
 Button celbtn = (Button)findViewById(R.id.*celbtn*);  
 Button matbtn = (Button)findViewById(R.id.*matbtn*);  
 Button savebtn = (Button)findViewById(R.id.*savebtn*);  
 Button homebtn = (Button)findViewById(R.id.*homebtn*);  
 Button exitbtn = (Button)findViewById(R.id.*exitbtn*);  
  
  
 checktext.setText("\""+ MBTI+"\"");  
 if(MBTI.equals("INTJ")){  
 exname.setText("용의주도한 전략가");  
 exname.setTextColor(Color.*parseColor*("#6A1B9A"));  
 extext.setText("상상력이 풍부하며 철두철미한 계획을 세우는 전락가형");  
 checktext.setTextColor(Color.*parseColor*("#6A1B9A"));  
 }  
  
 if(MBTI.equals("INTP")){  
 exname.setText("논리적인 사색가");  
 exname.setTextColor(Color.*parseColor*("#6A1B9A"));  
 extext.setText("끊임없이 새로운 지식에 목말라 하는 혁신가형");  
 checktext.setTextColor(Color.*parseColor*("#6A1B9A"));  
 }  
  
 if(MBTI.equals("ENTJ")){  
 exname.setText("대담한 통솔자");  
 exname.setTextColor(Color.*parseColor*("#6A1B9A"));  
 extext.setText("대담하면서도 상상력이 풍부한 소유자로, \n다양한 방법을 모색하거나 여의치 않을 경우\n 새로운 방안을 창출하는 리더");  
 checktext.setTextColor(Color.*parseColor*("#6A1B9A"));  
 }  
  
 if(MBTI.equals("ENTP")){  
 exname.setText("뜨거운 논쟁을 즐기는 변론가");  
 exname.setTextColor(Color.*parseColor*("#6A1B9A"));  
 extext.setText("지적인 도전을 두려워하지 않는 똑똑한 호기심형");  
 checktext.setTextColor(Color.*parseColor*("#6A1B9A"));  
 }  
  
 if(MBTI.equals("INFJ")){  
 exname.setText("선의의 옹호자");  
 exname.setTextColor(Color.*parseColor*("#3CB371"));  
 extext.setText("조용하고 신비로우며 샘솟는 영감으로 \n지칠 줄 모르는 이상주의자");  
 checktext.setTextColor(Color.*parseColor*("#3CB371"));  
 }  
  
 if(MBTI.equals("INFP")){  
 exname.setText("열정적인 중재자");  
 exname.setTextColor(Color.*parseColor*("#3CB371"));  
 extext.setText("상냥한 성격의 이타주의자로 건강하고 \n밝은 사회 건설에 앞장서는 낭만형");  
 checktext.setTextColor(Color.*parseColor*("#3CB371"));  
 }  
  
 if(MBTI.equals("ENFJ")){  
 exname.setText("정의로운 사회운동가");  
 exname.setTextColor(Color.*parseColor*("#3CB371"));  
 extext.setText("넘치는 카리스마와 영향력으로 청중을 압도하는 리더형");  
 checktext.setTextColor(Color.*parseColor*("#3CB371"));  
 }  
  
 if(MBTI.equals("ENFP")){  
 exname.setText("재기발랄한 활동가");  
 exname.setTextColor(Color.*parseColor*("#3CB371"));  
 extext.setText("항상 웃을 거리를 찾아 다니는 활발한 성격으로\n 사람들과 자유롭게 어울리기를 좋아하는 \n넘치는 열정을 가진 소유자");  
 checktext.setTextColor(Color.*parseColor*("#3CB371"));  
 }  
  
 if(MBTI.equals("ISTP")){  
 exname.setText("만능 재주꾼");  
 exname.setTextColor(Color.*parseColor*("#FFCC33"));  
 extext.setText("대담하고 현식적인 성향으로\n 다양한 도구 사용에 능숙한 탐험형");  
 checktext.setTextColor(Color.*parseColor*("#FFCC33"));  
 }  
  
 if(MBTI.equals("ISFP")){  
 exname.setText("호기심 많은 예술가");  
 exname.setTextColor(Color.*parseColor*("#FFCC33"));  
 extext.setText("항시 새로운 것을 찾아 시도하거나 \n도전할 준비가 되어 있는 융통성 있는 성격의\n 매력 넘치는 예술가형");  
 checktext.setTextColor(Color.*parseColor*("#FFCC33"));  
 }  
  
 if(MBTI.equals("ESTP")){  
 exname.setText("모험을 즐기는 사업가");  
 exname.setTextColor(Color.*parseColor*("#FFCC33"));  
 extext.setText("벼랑 끝의 아슬아슬한 삶을 즐길 줄 알고\n 명석한 두뇌와 에너지, \n그리고 뛰어난 직관력을 가지고 있는 유형");  
 checktext.setTextColor(Color.*parseColor*("#FFCC33"));  
 }  
  
 if(MBTI.equals("ESFP")){  
 exname.setText("자유로운 영혼의 연예인");  
 exname.setTextColor(Color.*parseColor*("#FFCC33"));  
 extext.setText("주위에 있으면 인생이 지루하지 않을 정도로\n 즉흥적이며 열정과 에너지가 넘치는 연예인형");  
 checktext.setTextColor(Color.*parseColor*("#FFCC33"));  
 }  
  
 if(MBTI.equals("ISTJ")){  
 exname.setText("청렴결백한 논리주의자");  
 exname.setTextColor(Color.*parseColor*("#3399CC"));  
 extext.setText("사실에 근거하여 사고하며 이들의 행동이나 결정에\n 한 치의 의심을 사지 않는 현실주의자");  
 checktext.setTextColor(Color.*parseColor*("#3399CC"));  
 }  
  
 if(MBTI.equals("ISFJ")){  
 exname.setText("용감한 수호자");  
 exname.setTextColor(Color.*parseColor*("#3399CC"));  
 extext.setText("소중한 이들을 수호하는데 심혈을 기울이는 \n헌식적이며 성실한 방어자형");  
 checktext.setTextColor(Color.*parseColor*("#3399CC"));  
 }  
  
 if(MBTI.equals("ESTJ")){  
 exname.setText("엄격한 관리자");  
 exname.setTextColor(Color.*parseColor*("#3399CC"));  
 extext.setText("사물이나 사람을 관리하는데 독보적인\n 뛰어난 실력을 갖춘 관리자형" );  
 checktext.setTextColor(Color.*parseColor*("#3399CC"));  
 }  
  
 if(MBTI.equals("ESFJ")){  
 exname.setText("사교적인 외교관");  
 exname.setTextColor(Color.*parseColor*("#3399CC"));  
 extext.setText("타인을 향한 세심한 관심과 사교적인 성향으로\n 사람들 내에서 인기가 많으며, \n타인을 돕는데 열정적인 세심형");  
 checktext.setTextColor(Color.*parseColor*("#3399CC"));  
 }  
  
  
 divbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), DivisionActivity.class);  
 startActivity(intent);  
 }  
 });  
  
 celbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), CelebrityActivity.class);  
 intent.putExtra("MBTI",MBTI);  
 startActivity(intent);  
 }  
 });  
  
 matbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), MatchedActivity.class);  
 intent.putExtra("MBTI",MBTI);  
 startActivity(intent);  
 }  
 });  
  
 savebtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), DBActivity.class);  
 intent.putExtra("MBTI",MBTI);  
 startActivity(intent);  
 }  
 });  
  
 exitbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 moveTaskToBack(true);  
 finish();  
 android.os.Process.*killProcess*(android.os.Process.*myPid*());  
 }  
 });  
  
 homebtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), MainActivity.class);  
 startActivity(intent);  
 }  
 });  
 }  
}

**<CheckResultActivity.java 소스코드 분석>**

Intent intent = getIntent();  
final String MBTI = intent.getStringExtra("MBTI");

CheckActivity에서 넘긴 인텐트를 getIntent()로 받은 후, 넘겨받은 MBTI 결과 값을 MBTI 문자열에 저장

checktext.setText("\""+ MBTI+"\"");

사용자의 MBTI 검사 결과 출력

if(MBTI.equals("INTJ")){  
 exname.setText("용의주도한 전략가");  
 exname.setTextColor(Color.*parseColor*("#6A1B9A"));  
 extext.setText("상상력이 풍부하며 철두철미한 계획을 세우는 전락가형");  
 checktext.setTextColor(Color.*parseColor*("#6A1B9A"));  
}

사용자의 MBTI가 INFP일 때 그에 맞는 설명을 출력

→ 위의 사항은 다른 모든 MBTI의 유형에 동일하게 적용

divbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), DivisionActivity.class);  
 startActivity(intent);  
 }  
});

구분기준 버튼을 누르면 구분기준 액티비티로 넘어감

celbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), CelebrityActivity.class);  
 intent.putExtra("MBTI",MBTI);  
 startActivity(intent);  
 }  
});

유명인 버튼을 누르면 유명인 액티비티로 넘어감

matbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), MatchedActivity.class);  
 intent.putExtra("MBTI",MBTI);  
 startActivity(intent);  
 }  
});

궁합 버튼을 누르면 궁합 액티비티로 넘어감

savebtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), DBActivity.class);  
 intent.putExtra("MBTI",MBTI);  
 startActivity(intent);  
 }  
});

결과 저장하기 버튼을 누르면 결과 저장하기 액티비티로 넘어감

exitbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 moveTaskToBack(true);  
 finish();  
 android.os.Process.*killProcess*(android.os.Process.*myPid*());  
 }  
});

종료 버튼을 누르면 앱이 종료됨

homebtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), MainActivity.class);  
 startActivity(intent);  
 }  
});

처음으로 버튼을 누르면 초기 화면인 MainActivity로 넘어감

**<division.xml 소스코드 및 분석(주석)>**

*<?*xml version="1.0" encoding="utf-8"*?>*<TabHost  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:id="@android:id/tabhost">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical">

*<!-- 탭위젯을 상단에 배치 -->*  
 <TabWidget  
 android:id="@android:id/tabs"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="20dp">  
 </TabWidget>

*<!-- 프레임 레이아웃에 구분기준 설명 출력-->*  
 <FrameLayout  
 android:id="@android:id/tabcontent"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <LinearLayout  
 android:id="@+id/divEI"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="35dp"  
 android:layout\_marginTop="40dp"  
 android:orientation="vertical">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text="E"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="#6A1B9A"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text=" (외향형-Extraversion)"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="- 자신의 외부에 주의집중\n- 말로 표현\n- 외부활동에 정열적\n- 다수와 폭넓은 대화\n- 소모에 의한 에너지 충전"  
 android:textSize="17sp"  
 android:textColor="@color/black"/>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="45dp"  
 android:orientation="horizontal">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text="I"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="#6A1B9A"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text=" (내향형-Introversion)"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="- 자신의 내부에 주의집중\n- 글로 표현\n- 내부활동에 집중적\n- 소수와 깊은 대화\n- 비축에 의한 에너지 충전"  
 android:textSize="17sp"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:id="@+id/divSN"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="35dp"  
 android:layout\_marginTop="40dp"  
 android:orientation="vertical">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text="S"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="#3CB371"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text=" (감각형-Sensing)"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="- 지금의 현실성에 초점\n- 실제의 경험에 의한 사실적 사건 묘사\n- 나무를 보려는 경향 \n- 현실 수용에 의한 실태 파악\n- 관례를 따르는 경향"  
 android:textSize="17sp"  
 android:textColor="@color/black"/>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="45dp"  
 android:orientation="horizontal">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text="N"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="#3CB371"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text=" (직관형-iNtuition)"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="- 미래의 가능성에 초점\n- 상상의 아이디어에 의한 비유적 사건 묘사\n- 숲을 보려는 경향 \n- 미래 지향에 의한 가능성과 의미추구\n- 새로운 시도를 도전하는 경향"  
 android:textSize="17sp"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:id="@+id/divTF"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="35dp"  
 android:layout\_marginTop="40dp"  
 android:orientation="vertical">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text="T"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="#3399CC"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text=" (사고형-Thinking)"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="- 사실에 의한 진실에 주된 관심\n- 맞다/틀리다의 판단\n- 원리와 원칙에 따르는 결과 중시\n- 규범 준수 중시\n- 객관적 인과관계에 의한 판단"  
 android:textSize="17sp"  
 android:textColor="@color/black"/>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="45dp"  
 android:orientation="horizontal">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text="F"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="#3399CC"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text=" (감정형-Feeling)"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="- 대인관계에 주된 관심\n- 좋다/나쁘다의 판단\n- 상황을 참작한 결과 중시\n- 우호적 협조 중시\n- 주관적 보편적인 선에 의한 판단"  
 android:textSize="17sp"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:id="@+id/divPJ"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="35dp"  
 android:layout\_marginTop="40dp"  
 android:orientation="vertical">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text="P"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="#FFCC33"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text=" (인식형-Perceiving)"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="- 상황에 맞추는 개방성\n- 재량에 따라 처리될수 있는 포용\n- 유연한 시간관리\n- 상황에 융통과 적응\n- 즉흥적인 뜻밖의 일을 즐김"  
 android:textSize="17sp"  
 android:textColor="@color/black"/>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="45dp"  
 android:orientation="horizontal">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text="J"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="#FFCC33"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="10dp"  
 android:text=" (판단형-Judging)"  
 android:textSize="25sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="- 양식에 맞추는 계획성\n- 분명한 목적의식의 자기의사\n- 철저한 시간관리\n- 상황을 통제와 조정\n- 계획을 결정하는 일을 즐김"  
 android:textSize="17sp"  
 android:textColor="@color/black"/>  
  
 </LinearLayout>  
 </FrameLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:layout\_marginTop="100dp"  
 android:gravity="center">  
  
 <Button  
 android:id="@+id/backbtn"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:text="이전으로"  
 android:textSize="20sp"  
 android:layout\_marginRight="10dp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 <Button  
 android:id="@+id/homebtn"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:text="처음으로"  
 android:textSize="20sp"  
 android:layout\_marginRight="10dp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:id="@+id/exitbtn"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:text="종료"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
 </LinearLayout>  
</TabHost>

**<DivisionActivity.java 소스코드>**

package kr.ac.kpu.termproject;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.app.TabActivity;  
import android.content.Intent;  
import android.graphics.Color;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TabHost;  
  
@SuppressWarnings("deprecation")  
public class DivisionActivity extends TabActivity {  
  
 Button homeBtn, exitBtn, backBtn;  
 String tagName;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*division*);  
  
 TabHost tabHost = getTabHost();  
  
 TabHost.TabSpec tabSpecEI = tabHost.newTabSpec("EI").setIndicator("E / I");  
 tabSpecEI.setContent(R.id.*divEI*);  
 tabHost.addTab(tabSpecEI);  
  
 TabHost.TabSpec tabSpecSN = tabHost.newTabSpec("SN").setIndicator("S / N");  
 tabSpecSN.setContent(R.id.*divSN*);  
 tabHost.addTab(tabSpecSN);  
  
 TabHost.TabSpec tabSpecTF = tabHost.newTabSpec("TF").setIndicator("T / F");  
 tabSpecTF.setContent(R.id.*divTF*);  
 tabHost.addTab(tabSpecTF);  
  
 TabHost.TabSpec tabSpecPJ = tabHost.newTabSpec("PJ").setIndicator("P / J");  
 tabSpecPJ.setContent(R.id.*divPJ*);  
 tabHost.addTab(tabSpecPJ);  
  
  
 tabHost.getTabWidget().setBackgroundColor(Color.*parseColor*("#cccccc"));  
 tabHost.getTabWidget().setCurrentTab(0);  
 tabHost.getTabWidget().getChildAt(0).setBackgroundColor(Color.*parseColor*("#88ccff"));  
  
 tabHost.setOnTabChangedListener(new TabHost.OnTabChangeListener() {  
 @Override  
 public void onTabChanged(String s) {  
 for (int i = 0; i < tabHost.getTabWidget().getChildCount(); i++) {  
 tabHost.getTabWidget().getChildAt(i).setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 tabHost.getTabWidget().getChildAt(tabHost.getCurrentTab()).setBackgroundColor(Color.*parseColor*("#88ccff"));  
 }  
 });  
  
 homeBtn = (Button) findViewById(R.id.*homebtn*);  
 exitBtn = (Button) findViewById(R.id.*exitbtn*);  
 backBtn = (Button) findViewById(R.id.*backbtn*);  
  
 backBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 finish();  
 }  
 });  
  
 homeBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), MainActivity.class);  
 startActivity(intent);  
 }  
 });  
  
 exitBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 moveTaskToBack(true);  
 finish();  
 android.os.Process.*killProcess*(android.os.Process.*myPid*());  
 }  
 });  
  
 }  
}

**<DivisionActivity.java 소스코드 분석>**

TabHost tabHost = getTabHost();  
  
 TabHost.TabSpec tabSpecEI = tabHost.newTabSpec("EI").setIndicator("E / I");  
 tabSpecEI.setContent(R.id.*divEI*);  
 tabHost.addTab(tabSpecEI);  
  
탭호스트 변수를 생성 후 탭스펙과 연결하여 탭위젯의 Indicator을 지정

탭스펙을 탭과 연결 후 탭을 탭호스트에 추가

→ 위의 사항은 다른 모든 탭에 동일하게 적용

tabHost.getTabWidget().setBackgroundColor(Color.*parseColor*("#cccccc"));  
tabHost.getTabWidget().setCurrentTab(0);  
tabHost.getTabWidget().getChildAt(0).setBackgroundColor(Color.*parseColor*("#88ccff"));

탭위젯의 색상을 지정한 후 탭위젯의 처음 위치를 0번째로 선택하고 색은 다른 색으로 지정

tabHost.setOnTabChangedListener(new TabHost.OnTabChangeListener() {  
 @Override  
 public void onTabChanged(String s) {  
 for (int i = 0; i < tabHost.getTabWidget().getChildCount(); i++) {  
 tabHost.getTabWidget().getChildAt(i).setBackgroundColor(Color.*parseColor*("#cccccc"));  
 }  
 tabHost.getTabWidget().getChildAt(tabHost.getCurrentTab()).setBackgroundColor(Color.*parseColor*("#88ccff"));  
 }  
});

선택된 탭위젯의 위치에 따라 색상을 지정

backBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 finish();  
 }  
});

이전으로 버튼을 누르면 현재 액티비티를 종료하고, 이전 액티비티인 CheckResultActivity로 넘어감

homebtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), MainActivity.class);  
 startActivity(intent);  
 }  
});

처음으로 버튼을 누르면 초기 화면인 MainActivity로 넘어감

exitbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 moveTaskToBack(true);  
 finish();  
 android.os.Process.*killProcess*(android.os.Process.*myPid*());  
 }  
});

종료 버튼을 누르면 앱이 종료됨

**<celebrity.xml 소스코드>**

*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:gravity="center"  
 android:layout\_marginTop="100dp">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textSize="30sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"  
 android:text="나와 같은 MBTI를 가진 유명인" />  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="50dp"  
 android:layout\_marginBottom="25dp"  
 android:textSize="35sp"  
 android:id="@+id/checktext"  
 android:text="&quot;INTJ&quot;"  
 android:textStyle="bold"  
 android:textColor="#6A1B9A"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:gravity="center">  
  
 <LinearLayout  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:orientation="vertical">  
  
 <ImageView  
 android:id="@+id/im1"  
 android:layout\_width="150dp"  
 android:layout\_height="150dp"  
 android:layout\_margin="20dp"  
 android:src="@drawable/ic\_launcher\_background" />  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/name1"  
 android:text="인물이름1"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:orientation="vertical">  
  
 <ImageView  
 android:id="@+id/im2"  
 android:layout\_width="150dp"  
 android:layout\_height="150dp"  
 android:layout\_margin="20dp"  
 android:src="@drawable/ic\_launcher\_background" />  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:id="@+id/name2"  
 android:text="인물이름2"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:layout\_weight="0.5"  
 android:gravity="center">  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginRight="10dp"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/backbtn"  
 android:text="이전으로"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginRight="10dp"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/homebtn"  
 android:text="처음으로"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/exitbtn"  
 android:text="종료"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
</LinearLayout>

**<celebrity.xml 소스코드 분석>**

* 전체적인 형식은 LinearLayout임
* ‘나와 같은 MBTI를 가진 유명인’과 결과로 나온 ‘MBTI’는 각각 TextView로 구현했으며 이 2개의 TextView는 orientation을 vertical로 설정하여 수직으로 출력됨
* 두명의 연예인들의 사진과 이름을 출력해주기 위해 각각의 사진과 이름을 1차적으로 LinearLayout으로 묶고 묶여진 LinearLayout 2개를 다시 LinearLayout이 감싸는 형식으로 구현함
* 한 LinearLayout으로 구현하면 사진과 이름이 가운데 정렬이 되지 않고 여백 설정에 어려움이 있어 위와 같은 방식으로 구현함
* 각 사진과 이름을 set로 보고 orientation을 vertical로 주어 수직으로 출력되게 한 다음 그 두 LinearLayout을 다시 orientation이 horizontal인 LinearLayout으로 감싸 수평적으로 출력되게 함
* 이전으로, 처음으로, 종료 버튼을 가장 하단에 만들었음

**<CelebrityActivity.java 소스코드>**

package kr.ac.kpu.termproject;  
  
import android.content.Intent;  
import android.graphics.Color;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.ImageView;  
import android.widget.TextView;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class CelebrityActivity extends AppCompatActivity {  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*celebrity*);  
  
 Intent intent = getIntent();  
 final String MBTI = intent.getStringExtra("MBTI");  
  
 TextView checktext = (TextView)findViewById(R.id.*checktext*);  
 Button backbtn = (Button)findViewById(R.id.*backbtn*);  
 Button homebtn = (Button)findViewById(R.id.*homebtn*);  
 Button exitbtn = (Button)findViewById(R.id.*exitbtn*);  
 ImageView img1 = (ImageView)findViewById(R.id.*im1*);  
 TextView t1 = (TextView)findViewById(R.id.*name1*);  
 ImageView img2 = (ImageView)findViewById(R.id.*im2*);  
 TextView t2 = (TextView)findViewById(R.id.*name2*);  
  
 checktext.setText("\""+ MBTI+"\"");  
  
 if(MBTI.equals("INTJ")){  
 checktext.setTextColor(Color.*parseColor*("#6A1B9A"));  
 img1.setImageResource(R.drawable.*intjf*);  
 img2.setImageResource(R.drawable.*intjm*);  
 t1.setText("김유정");  
 t2.setText("강동원");  
 }  
 if(MBTI.equals("INTP")){  
 checktext.setTextColor(Color.*parseColor*("#6A1B9A"));  
 img1.setImageResource(R.drawable.*intpf*);  
 img2.setImageResource(R.drawable.*intpm*);  
 t1.setText("에이핑크 정은지");  
 t2.setText("BTS 진");  
 }  
 if(MBTI.equals("ENTJ")){  
 checktext.setTextColor(Color.*parseColor*("#6A1B9A"));  
 img1.setImageResource(R.drawable.*entjf*);  
 img2.setImageResource(R.drawable.*entjm*);  
 t1.setText("소녀시대 티파니");  
 t2.setText("이승기");  
 }  
 if(MBTI.equals("ENTP")){  
 checktext.setTextColor(Color.*parseColor*("#6A1B9A"));  
 img1.setImageResource(R.drawable.*entpf*);  
 img2.setImageResource(R.drawable.*entpm*);  
 t1.setText("한예슬");  
 t2.setText("비투비 육성재");  
 }  
 if(MBTI.equals("INFJ")){  
 checktext.setTextColor(Color.*parseColor*("#3CB371"));  
 img1.setImageResource(R.drawable.*infjf*);  
 img2.setImageResource(R.drawable.*infjm*);  
 t1.setText("소녀시대 태연");  
 t2.setText("아스트로 차은우");  
 }  
 if(MBTI.equals("INFP")){  
 checktext.setTextColor(Color.*parseColor*("#3CB371"));  
 img1.setImageResource(R.drawable.*infpf*);  
 img2.setImageResource(R.drawable.*infpm*);  
 t1.setText("아이유");  
 t2.setText("BTS 정국");  
 }  
 if(MBTI.equals("ENFJ")){  
 checktext.setTextColor(Color.*parseColor*("#3CB371"));  
 img1.setImageResource(R.drawable.*enfjf*);  
 img2.setImageResource(R.drawable.*enfjm*);  
 t1.setText("김연경");  
 t2.setText("BTS 지민");  
 }  
 if(MBTI.equals("ENFP")){  
 checktext.setTextColor(Color.*parseColor*("#3CB371"));  
 img1.setImageResource(R.drawable.*enfpf*);  
 img2.setImageResource(R.drawable.*enfpm*);  
 t1.setText("이효리");  
 t2.setText("BTS 뷔");  
 }  
 if(MBTI.equals("ISTP")){  
 checktext.setTextColor(Color.*parseColor*("#FFCC33"));  
 img1.setImageResource(R.drawable.*istpf*);  
 img2.setImageResource(R.drawable.*istpm*);  
 t1.setText("김연아");  
 t2.setText("장성규");  
 }  
 if(MBTI.equals("ISFP")){  
 checktext.setTextColor(Color.*parseColor*("#FFCC33"));  
 img1.setImageResource(R.drawable.*isfpf*);  
 img2.setImageResource(R.drawable.*isfpm*);  
 t1.setText("김다미");  
 t2.setText("유재석");  
 }  
 if(MBTI.equals("ESTP")){  
 checktext.setTextColor(Color.*parseColor*("#FFCC33"));  
 img1.setImageResource(R.drawable.*estpf*);  
 img2.setImageResource(R.drawable.*estpm*);  
 t1.setText("효린");  
 t2.setText("강호동");  
 }  
 if(MBTI.equals("ESFP")){  
 checktext.setTextColor(Color.*parseColor*("#FFCC33"));  
 img1.setImageResource(R.drawable.*esfpf*);  
 img2.setImageResource(R.drawable.*esfpm*);  
 t1.setText("소녀시대 수영");  
 t2.setText("비");  
 }  
 if(MBTI.equals("ISTJ")){  
 checktext.setTextColor(Color.*parseColor*("#3399CC"));  
 img1.setImageResource(R.drawable.*istjf*);  
 img2.setImageResource(R.drawable.*istjm*);  
 t1.setText("안소희");  
 t2.setText("차태현");  
 }  
 if(MBTI.equals("ISFJ")){  
 checktext.setTextColor(Color.*parseColor*("#3399CC"));  
 img1.setImageResource(R.drawable.*infjf*);  
 img2.setImageResource(R.drawable.*infjm*);  
 t1.setText("장도연");  
 t2.setText("최강창민");  
 }  
 if(MBTI.equals("ESTJ")){  
 checktext.setTextColor(Color.*parseColor*("#3399CC"));  
 img1.setImageResource(R.drawable.*estjf*);  
 img2.setImageResource(R.drawable.*estjm*);  
 t1.setText("한채영");  
 t2.setText("김준수");  
 }  
 if(MBTI.equals("ESFJ")){  
 checktext.setTextColor(Color.*parseColor*("#3399CC"));  
 img1.setImageResource(R.drawable.*esfjf*);  
 img2.setImageResource(R.drawable.*esfjm*);  
 t1.setText("혜리");  
 t2.setText("슈퍼주니어 규현");  
 }  
  
 backbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 finish();  
 }  
 });  
  
 homebtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(),MainActivity.class);  
 startActivity(intent);  
 }  
 });  
  
 exitbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 android.os.Process.*killProcess*(android.os.Process.*myPid*());  
 }  
 });  
  
 }  
}

**<CelebrityActivity.java 소스코드 분석>**

Intent intent = getIntent();  
final String MBTI = intent.getStringExtra("MBTI");

: CheckResultActivity에서 넘긴 인텐트를 getIntent()로 받은 후, 넘겨받은 MBTI 결과 값을 MBTI 문자열에 저장함.

checktext.setText("\""+ MBTI+"\"");

: 사용자의 MBTI 검사 결과 출력

if(MBTI.equals("INTJ")){  
 checktext.setTextColor(Color.*parseColor*("#6A1B9A"));  
 img1.setImageResource(R.drawable.*intjf*);  
 img2.setImageResource(R.drawable.*intjm*);  
 t1.setText("김유정");  
 t2.setText("강동원");  
 }  
위와 같이 MBTI 검사 결과가 INTJ일 때, 해당하는 사진과 이름이 출력되게 함

사진은 drawable 폴더에 있는 사진에서 각 MBTI에 해당하는 사진을 할당하였고,

이름은 결과에 맞게 직접 지정해줌

이와 같은 형식을 각 MBTI 16개에 맞게 반복적으로 지정해줌

backbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 finish();  
 }  
 });

이전으로 버튼을 누르면 이전 화면인 검사결과 화면으로 되돌아감

homebtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(),MainActivity.class);  
 startActivity(intent);  
 }  
 });

처음으로 버튼을 누르면 MainActivity 화면으로 돌아감

exitbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 android.os.Process.*killProcess*(android.os.Process.*myPid*());  
 }  
 });

종료 버튼을 누르면 앱이 종료됨

**<matched.xml 소스코드 및 분석(주석)>**

*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="80dp"  
 android:layout\_marginBottom="40dp"  
 android:orientation="horizontal"  
 android:gravity="center">

<!--사용자의 MBTI 검사 결과 출력-->  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/result"  
 android:text="&quot;MBTI 검사 결과&quot;"  
 android:textSize="32sp"  
 android:textStyle="bold"  
 android:textColor="#6A1B9A"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text=" 궁합"  
 android:textSize="30sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>

<!--사용자의 MBTI와 다른 MBTI와의 궁합 출력 (ex. BEST, GOOD, NORMAL, NOT BAD, BAD)-->  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:layout\_weight="1">  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:layout\_weight="0.5"  
 android:layout\_marginLeft="35dp">  
  
 <ImageView  
 android:layout\_width="45dp"  
 android:layout\_height="45dp"  
 android:src="@drawable/best"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="10dp"  
 android:layout\_marginLeft="13dp"  
 android:id="@+id/bestMatched"  
 android:text="BEST 궁합 MBTI"  
 android:textSize="23sp"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/good"  
 android:orientation="horizontal"  
 android:layout\_weight="0.5"  
 android:layout\_marginLeft="35dp">  
  
 <ImageView  
 android:layout\_width="45dp"  
 android:layout\_height="45dp"  
 android:src="@drawable/good"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="10dp"  
 android:layout\_marginLeft="13dp"  
 android:id="@+id/goodMatched"  
 android:text="GOOD 궁합 MBTI"  
 android:textSize="23sp"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/normal"  
 android:orientation="horizontal"  
 android:layout\_weight="0.5"  
 android:layout\_marginLeft="35dp">  
  
 <ImageView  
 android:layout\_width="45dp"  
 android:layout\_height="45dp"  
 android:src="@drawable/normal"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="10dp"  
 android:layout\_marginLeft="13dp"  
 android:id="@+id/normalMatched"  
 android:text="NORMAL 궁합 MBTI"  
 android:textSize="23sp"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/notbad"  
 android:orientation="horizontal"  
 android:layout\_weight="0.5"  
 android:layout\_marginLeft="35dp">  
  
 <ImageView  
 android:layout\_width="45dp"  
 android:layout\_height="45dp"  
 android:src="@drawable/not\_bad"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="10dp"  
 android:layout\_marginLeft="13dp"  
 android:id="@+id/notbadMatched"  
 android:text="NOT BAD 궁합 MBTI"  
 android:textSize="23sp"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/bad"  
 android:orientation="horizontal"  
 android:layout\_marginLeft="35dp">  
  
 <ImageView  
 android:layout\_width="45dp"  
 android:layout\_height="45dp"  
 android:src="@drawable/bad"/>  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="10dp"  
 android:layout\_marginLeft="13dp"  
 android:id="@+id/badMatched"  
 android:text="BAD 궁합 MBTI"  
 android:textSize="23sp"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
 <!--스마일 이미지 각각의 의미(BEST, GOOD, NORMAL, NOT BAD, BAD) 출력-->  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="10dp"  
 android:orientation="horizontal"  
 android:layout\_weight="0.5"  
 android:gravity="center">  
  
 <ImageView  
 android:layout\_width="20dp"  
 android:layout\_height="20dp"  
 android:src="@drawable/best"/>  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text=" BEST"  
 android:textSize="13sp"  
 android:textColor="@color/black"/>  
  
 <ImageView  
 android:layout\_width="20dp"  
 android:layout\_height="20dp"  
 android:layout\_marginLeft="12dp"  
 android:src="@drawable/good"/>  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text=" GOOD"  
 android:textSize="13sp"  
 android:textColor="@color/black"/>  
  
 <ImageView  
 android:layout\_width="20dp"  
 android:layout\_height="20dp"  
 android:layout\_marginLeft="12dp"  
 android:src="@drawable/normal"/>  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text=" NORMAL"  
 android:textSize="13sp"  
 android:textColor="@color/black"/>  
  
 <ImageView  
 android:layout\_width="20dp"  
 android:layout\_height="20dp"  
 android:layout\_marginLeft="12dp"  
 android:src="@drawable/not\_bad"/>  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text=" NOT BAD"  
 android:textSize="13sp"  
 android:textColor="@color/black"/>  
  
 <ImageView  
 android:layout\_width="20dp"  
 android:layout\_height="20dp"  
 android:layout\_marginLeft="12dp"  
 android:src="@drawable/bad"/>  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text=" BAD"  
 android:textSize="13sp"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 </LinearLayout>  
 <!-- 이전으로, 처음으로, 종료 버튼-->  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:layout\_weight="0.5"  
 android:gravity="center">  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginRight="10dp"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/backbtn"  
 android:text="이전으로"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginRight="10dp"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/homebtn"  
 android:text="처음으로"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/exitbtn"  
 android:text="종료"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
</LinearLayout>

**<MatchedActivity.java 소스코드>**

package kr.ac.kpu.termproject;  
  
import android.content.Intent;  
import android.graphics.Color;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.LinearLayout;  
import android.widget.TextView;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MatchedActivity extends AppCompatActivity {  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*matched*);  
  
 Intent intent = getIntent();  
 final String MBTI = intent.getStringExtra("MBTI");  
  
 TextView result = (TextView)findViewById(R.id.*result*);  
  
 LinearLayout good = (LinearLayout)findViewById(R.id.*good*);  
 LinearLayout normal = (LinearLayout)findViewById(R.id.*normal*);  
 LinearLayout notbad = (LinearLayout)findViewById(R.id.*notbad*);  
 LinearLayout bad = (LinearLayout)findViewById(R.id.*bad*);  
  
 TextView bestmatched = (TextView)findViewById(R.id.*bestMatched*);  
 TextView goodmatched = (TextView)findViewById(R.id.*goodMatched*);  
 TextView normalmatched = (TextView)findViewById(R.id.*normalMatched*);  
 TextView notbadmatched = (TextView)findViewById(R.id.*notbadMatched*);  
 TextView badmatched = (TextView)findViewById(R.id.*badMatched*);  
  
 Button backbtn = (Button)findViewById(R.id.*backbtn*);  
 Button homebtn = (Button)findViewById(R.id.*homebtn*);  
 Button exitbtn = (Button)findViewById(R.id.*exitbtn*);  
  
 result.setText("\""+MBTI+"\"");  
  
 if(MBTI.equals("INFP")) {  
 result.setTextColor(Color.*parseColor*("#3CB371"));  
  
 bestmatched.setText("ENFJ, ENTJ");  
 goodmatched.setText("INFP, INFJ, ENTP");  
 badmatched.setText("ISFP, ESTP, ISTJ");  
  
 normal.setVisibility(View.*GONE*);  
 notbad.setVisibility(View.*GONE*);  
 }  
  
 else if(MBTI.equals("ENFP")) {  
 result.setTextColor(Color.*parseColor*("#3CB371"));  
  
 bestmatched.setText("INFJ, INTJ");  
 goodmatched.setText("INFP, ENFJ, ENTP");  
 badmatched.setText("ISFP, ESTP, ISTJ");  
  
 normal.setVisibility(View.*GONE*);  
 notbad.setVisibility(View.*GONE*);  
 }  
  
 else if(MBTI.equals("INFJ")) {  
 result.setTextColor(Color.*parseColor*("#3CB371"));  
  
 bestmatched.setText("ENFP, ENTP");  
 goodmatched.setText("INFP, ENFJ, INTP");  
 badmatched.setText("ISFP, ESTP, ISTJ");  
  
 normal.setVisibility(View.*GONE*);  
 notbad.setVisibility(View.*GONE*);  
 }  
  
 else if(MBTI.equals("ENFJ")) {  
 result.setTextColor(Color.*parseColor*("#3CB371"));  
  
 bestmatched.setText("INFP, ISFP");  
 goodmatched.setText("ENFP, INFJ, ENTP");  
 badmatched.setText("ESFP, ISTP, ESTP");  
  
 normal.setVisibility(View.*GONE*);  
 notbad.setVisibility(View.*GONE*);  
 }  
  
 else if(MBTI.equals("INTJ")) {  
 result.setTextColor(Color.*parseColor*("#6A1B9A"));  
  
 bestmatched.setText("ENFP, ENTP");  
 goodmatched.setText("INFP, ENFJ, INTP");  
 normalmatched.setText("ISFP, ISTP, ESTP");  
 notbadmatched.setText("ISFJ, ESFJ, ISTJ");  
  
 bad.setVisibility(View.*GONE*);  
 }  
  
 else if(MBTI.equals("ENTJ")) {  
 result.setTextColor(Color.*parseColor*("#6A1B9A"));  
  
 bestmatched.setText("INFP, INTP");  
 goodmatched.setText("ENFP, INFJ, INTJ");  
 normalmatched.setText("ISFP, ESTP, ESFJ");  
 notbadmatched.setText("ISFP, ESTP, ISTJ");  
  
 bad.setVisibility(View.*GONE*);  
 }  
  
 else if(MBTI.equals("INTP")) {  
 result.setTextColor(Color.*parseColor*("#6A1B9A"));  
  
 bestmatched.setText("ENTJ, ESTJ");  
 goodmatched.setText("INFP, INFJ, ENTP");  
 normalmatched.setText("ISFP, ISTP, ESTP");  
 notbadmatched.setText("ISFJ, ESFJ, ISTJ");  
  
 bad.setVisibility(View.*GONE*);  
 }  
  
 else if(MBTI.equals("ENTP")) {  
 result.setTextColor(Color.*parseColor*("#6A1B9A"));  
  
 bestmatched.setText("INFJ, INTJ");  
 goodmatched.setText("INFP, ENFJ, INTP");  
 normalmatched.setText("ISFP, ISTP, ESTP");  
 notbadmatched.setText("ISFJ, ESFJ, ISTJ");  
  
 bad.setVisibility(View.*GONE*);  
 }  
  
 else if(MBTI.equals("ISFP")) {  
 result.setTextColor(Color.*parseColor*("#FFCC33"));  
  
 bestmatched.setText("ENFJ, ESFJ, ESTJ");  
 normalmatched.setText("INTJ, INTP, ENTP");  
 notbadmatched.setText("ISFP, ISTP, ESTP");  
 badmatched.setText("INFP, ENFP, INFJ");  
  
 good.setVisibility(View.*GONE*);  
 }  
  
 else if(MBTI.equals("ESFP")) {  
 result.setTextColor(Color.*parseColor*("#FFCC33"));  
  
 bestmatched.setText("ISFJ, ISTJ");  
 normalmatched.setText("INTJ, INTP, ENTP");  
 notbadmatched.setText("ISFP, ISTP, ESTP");  
 badmatched.setText("INFP, ENFP, ENFJ");  
  
 good.setVisibility(View.*GONE*);  
 }  
  
 else if(MBTI.equals("ISTP")) {  
 result.setTextColor(Color.*parseColor*("#FFCC33"));  
  
 bestmatched.setText("ESFJ, ESTJ");  
 normalmatched.setText("INTJ, INTP, ENTP");  
 notbadmatched.setText("ISFP, ISTP, ESTP");  
 badmatched.setText("INFP, ENFP, ENFJ");  
  
 good.setVisibility(View.*GONE*);  
 }  
  
 else if(MBTI.equals("ESTP")) {  
 result.setTextColor(Color.*parseColor*("#FFCC33"));  
  
 bestmatched.setText("ISFJ, ISTJ");  
 normalmatched.setText("INTJ, INTP, ENTP");  
 notbadmatched.setText("ISFP, ISTP, ESTP");  
 badmatched.setText("INFP, ENFP, ENFJ");  
  
 good.setVisibility(View.*GONE*);  
 }  
  
 else if(MBTI.equals("ISFJ")) {  
 result.setTextColor(Color.*parseColor*("#3399CC"));  
  
 bestmatched.setText("ESFP, ESTP");  
 goodmatched.setText("ISFJ, ISTJ, ESTJ");  
 normalmatched.setText("ENTJ, ISFP, ISTP");  
 notbadmatched.setText("INTJ, INTP, ENTP");  
 badmatched.setText("INFP, ENFP, ENFJ");  
 }  
  
 else if(MBTI.equals("ESFJ")) {  
 result.setTextColor(Color.*parseColor*("#3399CC"));  
  
 bestmatched.setText("ISFP, ISTP");  
 goodmatched.setText("ISFJ, ISTJ, ESTJ");  
 normalmatched.setText("ENTJ, ESFP, ESTP");  
 notbadmatched.setText("INTJ, INTP, ENTP");  
 badmatched.setText("INFP, ENFP, ENFJ");  
 }  
  
 else if(MBTI.equals("ISTJ")) {  
 result.setTextColor(Color.*parseColor*("#3399CC"));  
  
 bestmatched.setText("ESFP, ESTP");  
 goodmatched.setText("ISFJ, ISTJ, ESTJ");  
 normalmatched.setText("ENTJ, ISFP, ISTP");  
 notbadmatched.setText("INTJ, INTP, ENTP");  
 badmatched.setText("INFP, ENFP, ENFJ");  
 }  
  
 else if(MBTI.equals("ESTJ")) {  
 result.setTextColor(Color.*parseColor*("#3399CC"));  
  
 bestmatched.setText("INTP, ISFP, ISTP");  
 goodmatched.setText("ISFJ, ISTJ, ESTJ");  
 normalmatched.setText("ENTJ, ESFP, ESTP");  
 notbadmatched.setText("INTJ, ENTP");  
 badmatched.setText("INFP, ENFP, ENFJ");  
 }  
  
 backbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 finish();  
 }  
 });  
  
 exitbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 moveTaskToBack(true);  
 finish();  
 android.os.Process.*killProcess*(android.os.Process.*myPid*());  
 }  
 });  
  
 homebtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), MainActivity.class);  
 startActivity(intent);  
 }  
 });  
 }  
}

**<MatchedActivity.java 소스코드 분석>**

* Intent intent = getIntent();  
  final String MBTI = intent.getStringExtra("MBTI");

CheckResultActivity에서 넘긴 인텐트를 getIntent()로 받은 후, 넘겨받은 MBTI 결과 값을 MBTI 문자열에 저장

* result.setText("\""+MBTI+"\"");

사용자의 MBTI 검사 결과 출력

* if(MBTI.equals("INFP")) {  
   result.setTextColor(Color.*parseColor*("#3CB371"));  
    
   bestmatched.setText("ENFJ, ENTJ");  
   goodmatched.setText("INFP, INFJ, ENTP");  
   badmatched.setText("ISFP, ESTP, ISTJ");  
    
   normal.setVisibility(View.*GONE*);  
   notbad.setVisibility(View.*GONE*);  
  }

사용자의 MBTI가 INFP일 때 다른 MBTI와의 궁합을 출력해주고 NORMAL, NOT BAD와

같이 해당하는 궁합이 없을 경우에는 setVisibility(View.GONE)을 통해 text뿐만 아니라 차

지하는 공간까지 제거

→ 위의 사항은 다른 모든 MBTI도 동일하게 적용

* backbtn.setOnClickListener(new View.OnClickListener() {  
   @Override  
   public void onClick(View view) {  
   finish();  
   }  
  });

이전으로 버튼을 누르면 현재 액티비티를 종료하고, 이전 액티비티인

CheckResultActivity로 넘어감

* homebtn.setOnClickListener(new View.OnClickListener() {  
   @Override  
   public void onClick(View view) {  
   Intent intent = new Intent(getApplicationContext(), MainActivity.class);  
   startActivity(intent);  
   }  
  });

처음으로 버튼을 누르면 초기 화면인 MainActivity로 넘어감

* exitbtn.setOnClickListener(new View.OnClickListener() {  
   @Override  
   public void onClick(View view) {  
   moveTaskToBack(true);  
   finish();  
   android.os.Process.*killProcess*(android.os.Process.*myPid*());  
   }  
  });

종료 버튼을 누르면 앱이 종료됨

**<database.xml 소스코드>**

*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:layout\_marginTop="50dp"  
 android:layout\_marginBottom="10dp"  
 android:orientation="horizontal">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="이름 : "  
 android:textSize="25sp"  
 android:textColor="@color/black"  
 android:textStyle="bold"/>  
 <EditText  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:hint="이름을 입력하세요."  
 android:id="@+id/edtname"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:orientation="horizontal">  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginRight="10dp"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/inputbtn"  
 android:text="입력"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginRight="10dp"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/checkbtn"  
 android:text="조회"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/resetbtn"  
 android:text="초기화"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="30dp"  
 android:layout\_weight="2">  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:id="@+id/nameresult"/>  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:id="@+id/mbtiresult"/>  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:id="@+id/time"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="0.3"  
 android:orientation="horizontal"  
 android:gravity="center">  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginRight="10dp"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/backbtn"  
 android:text="이전으로"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:backgroundTint="#cccccc"  
 android:id="@+id/homebtn"  
 android:text="처음으로"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:textColor="@color/black"/>  
 </LinearLayout>  
</LinearLayout>

**<DBActivity.java 소스코드>**

package kr.ac.kpu.termproject;  
  
import android.content.Context;  
import android.content.Intent;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import java.text.SimpleDateFormat;  
import java.util.Date;  
  
public class DBActivity extends AppCompatActivity {  
 myDB myHelper;  
 EditText edtname;  
 TextView nameresult, mbtiresult, time;  
 Button resetbtn, inputbtn, checkbtn, backBtn, homeBtn;  
 SQLiteDatabase sqlDB;  
  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*database*);  
 Intent intent = getIntent();  
 final String MBTI = intent.getStringExtra("MBTI");  
  
 long now = System.*currentTimeMillis*();  
 Date mDate = new Date(now);  
 SimpleDateFormat simpleDate = new SimpleDateFormat("YYYY-MM-dd HH:mm:ss");  
 String gettime = simpleDate.format(mDate);  
  
 edtname = (EditText) findViewById(R.id.*edtname*);  
 nameresult = (TextView) findViewById(R.id.*nameresult*);  
 mbtiresult = (TextView) findViewById(R.id.*mbtiresult*);  
 time = (TextView) findViewById(R.id.*time*);  
 resetbtn = (Button) findViewById(R.id.*resetbtn*);  
 inputbtn = (Button) findViewById(R.id.*inputbtn*);  
 checkbtn = (Button) findViewById(R.id.*checkbtn*);  
 myHelper = new myDB(this);  
 backBtn = (Button) findViewById(R.id.*backbtn*);  
 homeBtn = (Button) findViewById(R.id.*homebtn*);  
  
 resetbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 sqlDB = myHelper.getWritableDatabase();  
 myHelper.onUpgrade(sqlDB, 1, 2);  
 sqlDB.close();  
 }  
 });  
  
 inputbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 if (edtname.getText().toString().length() <= 0) {  
 Toast.*makeText*(getApplicationContext(), "이름을 입력해 주세요.", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 sqlDB = myHelper.getWritableDatabase();  
 sqlDB.execSQL("INSERT INTO groupTBL(gName,gNumber,gTime) VALUES('" + edtname.getText().toString() + "','" + MBTI + "','" + gettime + "');");  
 sqlDB.close();  
 Toast.*makeText*(getApplicationContext(), "성공적으로 저장되었습니다!", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 });  
  
 checkbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 sqlDB = myHelper.getReadableDatabase();  
 Cursor cursor;  
 cursor = sqlDB.rawQuery("SELECT \* FROM groupTBL;", null);  
  
 String strname = "NAME" + "\r\n" + "----------------------------------" + "\r\n";  
 String strmbti = "MBTI" + "\r\n" + "----------------------------------" + "\r\n";  
 String strtime = "TIME" + "\r\n" + "----------------------------------" + "\r\n";  
  
 while (cursor.moveToNext()) {  
 strname += cursor.getString(0) + "\r\n";  
 strmbti += cursor.getString(1) + "\r\n";  
 strtime += cursor.getString(2) + "\r\n";  
 }  
  
 nameresult.setText(strname);  
 mbtiresult.setText(strmbti);  
 time.setText(strtime);  
  
 cursor.close();  
 sqlDB.close();  
 }  
 });  
  
 backBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 finish();  
 }  
 });  
  
 homeBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Intent intent = new Intent(getApplicationContext(), MainActivity.class);  
 startActivity(intent);  
 }  
 });  
 }  
  
 public class myDB extends SQLiteOpenHelper {  
 public myDB(Context context) {  
 super(context, "groupDB", null, 1);  
 }  
  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 db.execSQL("CREATE TABLE groupTBL(gName TEXT,gNumber TEXT,gTime TEXT);");  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int i, int i1) {  
 db.execSQL("DROP TABLE IF EXISTS groupTBL");  
 onCreate(db);  
 }  
 }  
}

**<DBActivity.java 소스코드 분석>**

public class myDB extends SQLiteOpenHelper{  
 public myDB(Context context){  
 super(context,"groupDB",null,1);  
 }  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 db.execSQL("CREATE TABLE groupTBL(gName TEXT,gNumber TEXT,gTime TEXT);");  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int i, int i1) {  
 db.execSQL("DROP TABLE IF EXISTS groupTBL");  
 onCreate(db);  
 }

public myDB(Context context){  
 super(context,"groupDB",null,1);  
 }  
DB 클래스에 대한 생성자를 만듦

myDB 클래스를 선언해 데이터베이스를 활용하고, onCeate()에서 execSQL로   
("CREATE TABLE groupTBL(gName TEXT,gNumber TEXT,gTime TEXT);를 삽입해   
이름이 groupTBL이고, 인자는 이름(문자열),MBTI(문자열),현재 시간(문자열)  
을 가지는 테이블 생성함  
groupTBL에 데이터의 존재 여부를 확인하고 초기화하는 onUpgrage().

myDB myHelper;

데이터베이스 생성자를 이용해 myHelper선언

resetbtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 sqlDB = myHelper.getWritableDatabase();  
 myHelper.onUpgrade(sqlDB,1,2);  
 sqlDB.close();  
 }  
 });

sqlDB = myHelper.getWritableDatabase();로 데이터베이스를 수정하기 위한 쓰기모드로 데이터베이스 접근  
초기화 버튼을 눌렀을 때 myHelper.onUpgrade(sqlDB,1,2);를 이용해 초기화 진행  
sqlDB.close(); 데이터베이스에 대한 연결 해제

if(edtname.getText().toString().length()<=0){  
 Toast.*makeText*(getApplicationContext(),"이름을 입력해 주세요.",Toast.*LENGTH\_SHORT*).show();

이름(닉네임)을 입력 받는 칸이 비어 있을 경우 Toast메시지를 사용해 예외 처리함

sqlDB = myHelper.getWritableDatabase();  
 sqlDB.execSQL("INSERT INTO groupTBL(gName,gNumber,gTime) VALUES('"+edtname.getText().toString()+"','"+MBTI+"','"+gettime+"');");  
 sqlDB.close();

쓰기 모드로 데이터베이스 접근  
execSQL을 이용해 groupTBL의 이름, MBTI, 시간 데이터를 SQL 삽입으로 저장  
데이터베이스 연결 해제

sqlDB = myHelper.getReadableDatabase();

데이터 베이스를 읽기모드로 접근

Cursor cursor;

데이터베이스 탐색을 위한 커서 선언

cursor = sqlDB.rawQuery("SELECT \* FROM groupTBL;",null);

커서를 데이터베이스에 연결하는 작업  
  
 while (cursor.moveToNext()){  
 strname += cursor.getString(0) + "\r\n";  
 strmbti += cursor.getString(1) + "\r\n";  
 strtime += cursor.getString(2) + "\r\n";  
 }

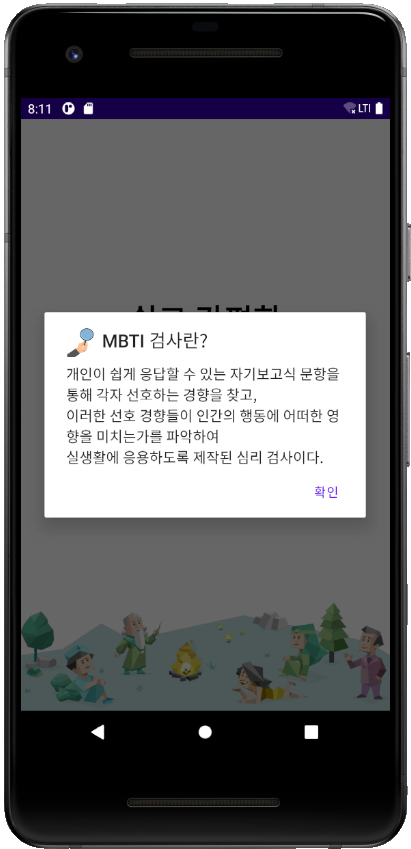
커서가 다음 열의 데이터를 탐색하며 모든 데이터를 한 번씩 탐색, 모든 데이터를 한 열로 출력하기 위한 작업 실행

cursor.close();  
 sqlDB.close();

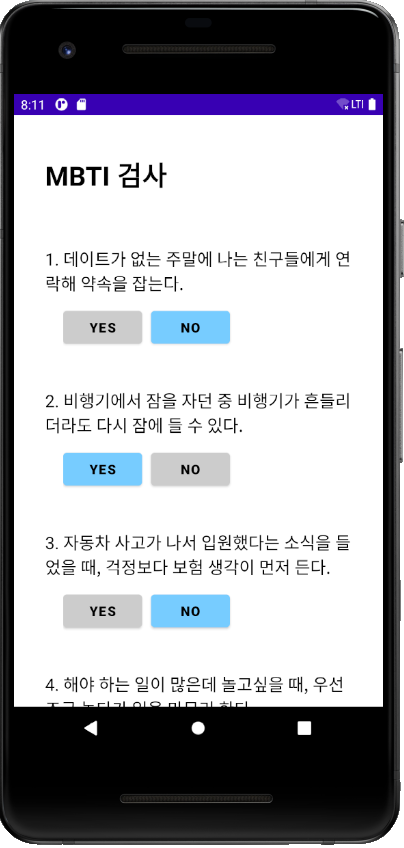
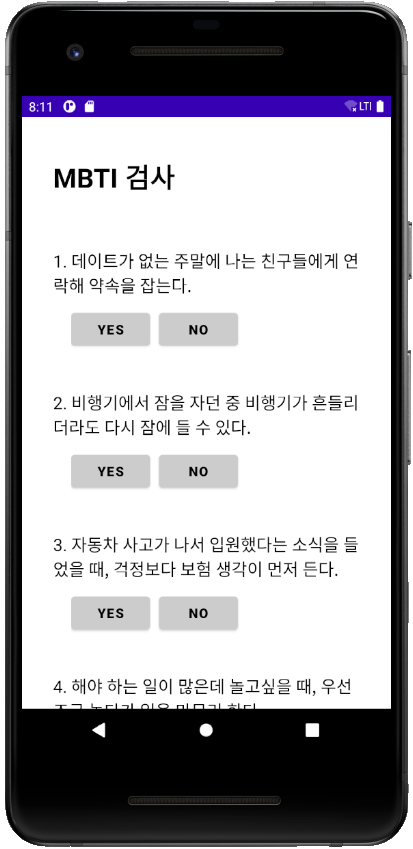
커서 데이터베이스 연결 해제  
데이터베이스 연결 해제

* 1. **실행화면**

**<시작 화면 & 설명버튼 눌렀을 때 화면>**



**<검사 화면 & YES/NO 선택했을 때 화면>**



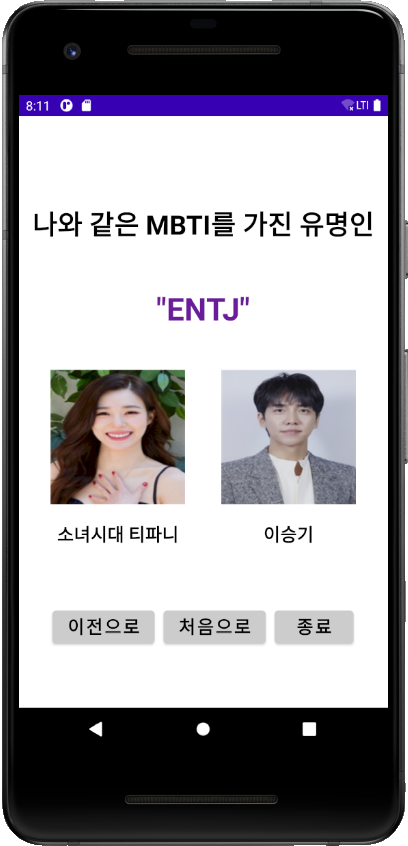
**<검사 결과 화면>**

****

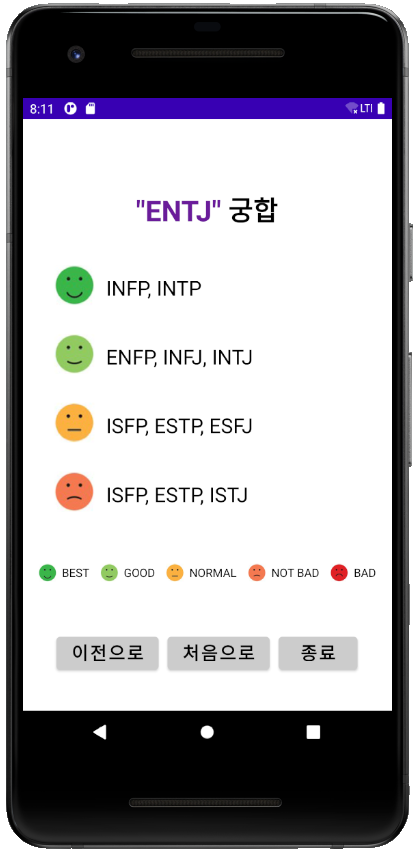
**<구분기준 화면>**

****

**<유명인 화면>**

****

**<궁합 화면>**

****

**<결과 저장하기 화면 & 입력,조회한 화면 & 초기화,조회한 화면>**

****

1. **결론**
   1. **보완할 사항**

* 현재 개발한 ‘쉽고 간편한 MBTI 검사’ 앱은 문항수가 12개로 원래 MBTI 검사의 60문항에 비해 적습니다. 문항수가 적어 검사하기에는 편리하지만 검사보다 정확도가 떨어질 가능성이 높습니다. 이에 다양한 선택지와 지금보다 정확한 문항, 선택지를 추가하여 검사 결과의 정확도를 향상시켜야 합니다.
* ‘쉽고 간편한 MBTI 검사’ 앱에 서버 구축을 통하여 SNS 공유 기능을 추가한다면 앱을 이용하는 사용자들의 흥미를 끄는데 도움이 될 것이라고 생각합니다.
* 현재 구현된 DB 액티비티는 이름을 입력하고 입력 버튼을 클릭하면 이름, MBTI, 시간이 출력되어 나타나게 됩니다. 여기서 입력을 반복해서 누르면 똑같은 내용이 반복적으로 출력된다는 문제점을 발견하였으나 예외처리를 하지 못했습니다. 이에 이 문제점을 해결할 수 있는 방법을 강구하여 보완해야 한다고 생각했습니다.
  1. **느낀점**

**최성락**

아주 작은 규모의 짧은 프로젝트였지만 제안서와 계획일정, 역할 분담, 개발 등을 진행하면서 팀원들 간의 의사소통이 굉장히 중요하다는 것을 느꼈고 실제 개발 현장에서도 활발한 의사소통이 이루어져야 고품질의 소프트웨어가 나올 것 같다는 생각이 들었다. 또한 구현 과정에서 수업시간에 배운 것들을 활용하여 각자 맡은 부분을 개발하고 통합하는 작업과 피드백, 수정을 통해 예외처리, 새로운 알고리즘 사용 등 많은 부분들을 통해 나의 개발 실력이 향상되고 있다는 느낌을 받았고 마지막으로 모바일 프로그래밍 수업을 통해 소프트웨어에 관한 식견이 넓어진 것 같고 앱 개발에 대한 흥미가 생겨 앞으로 더 공부해보고 싶다는 생각이 들었다.

**정주희**

현재 애플리케이션이 없는 일상은 상상할 수 없다. 항상 나의 생활의 일부분으로 생각한 애플리케이션을 직접 만든 다니 내 스스로가 신기했던 것 같다. 요즘 같은 상황으로 인해 사람들 간의 교류가 단절됐었는데, 팀 프로젝트의 계기로 팀원들과 회의하는 시간도 하나의 소통이라 생각하니 즐거웠던 것 같다. 물론 구현하는 과정에 있어 부족함을 많이 느꼈고, 공부의 중요성도 더욱 상기시키게 되었다. 이번 팀 프로젝트를 통해 부족한 부분을 알게 되었고, 공부의 방향성을 잡을 수 있었다. 무엇보다 안드로이드에 대한 흥미가 높아진 것이 가장 큰 즐거움을 얻은 것 같다.

**윤정아**

우선, 마지막까지 한 명도 빠짐없이 각자 맡은 분야에 최선을 다해준 팀원들에게 고마움을 느낀다. 지금까지 해왔던 프로젝트 중 가장 의견 공유가 활발하고 팀워크가 좋았던 것 같다. 아무리 사소한 문제더라도 함께 고민하고 노력하니 좋은 결과를 도출해낼 수 있었다. 처음엔 어렵기만 보였던 프로젝트가 팀원들과 함께하니 즐거운 분위기로 수월하게 진행되었다. 이에 팀원들의 소중함을 느낄 수 있었다. 또한, 평소 주어졌던 과제와 달리 프로젝트 조별과제를 수행하면서 더 심화된 내용을 주도적으로 공부해보니 개발 능력도 조금 향상된 것 같다. 나와는 다른 방법으로 구현하는 팀원들의 결과물을 보며 개발의 방법에는 다양함이 존재함을 알게 되었고 서로 많이 배울 수 있었다. 하루 온종일 안드로이드 스튜디오와 AVD를 실행하며 수정에 수정을 거치는 작업이 고되기도 했지만 책, 강의자료, 검색 등을 활용해 공부하고 팀원들의 협력으로 잘 마무리할 수 있었던 것 같다. 이번 기회를 발판삼아 안드로이드 프로그래밍을 더 열심히 공부하여 앞으로의 졸업작품 및 취업준비에 활용할 것이다.

**정소이**

이번 팀 프로젝트를 통해 평소에 관심이 많았던 앱 개발을 직접 해볼 수 있게 되어 정말 좋았다. 아주 간단한 앱을 만드는 작은 프로젝트였지만 앱 개발 계획, 역할 분담, 구현, 테스트 등 수업 시간에 배운 내용을 통해 직접 전체적인 앱 개발 과정을 하나씩 이루어 나감으로써, 수업 시간에 배운 내용 이외의 것도 알아가고 나의 개발 실력도 조금씩 나아지고 있는 것 같아, 이번 팀 프로젝트가 정말 좋은 경험이었다고 생각한다. 그리고 역시 팀 프로젝트는 팀원들 간의 의사 소통과 협력이 중요하다는 것을 느꼈다. 능력 좋고 서로를 배려해주는 팀원들을 만나 이번 프로젝트가 재미있었고, 잘 마무리된 것 같다. 무엇보다도 앱 개발에 대한 흥미가 더 높아져, 더 심화된 내용을 배우고 싶다는 생각이 들었다.