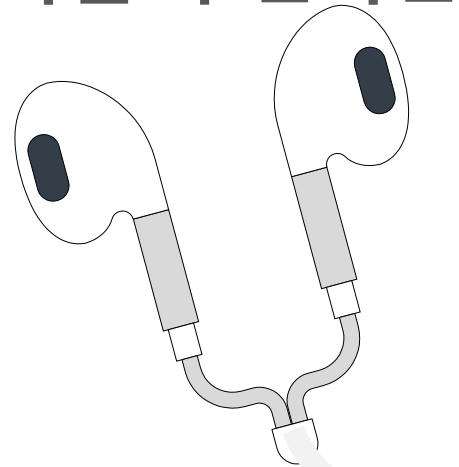
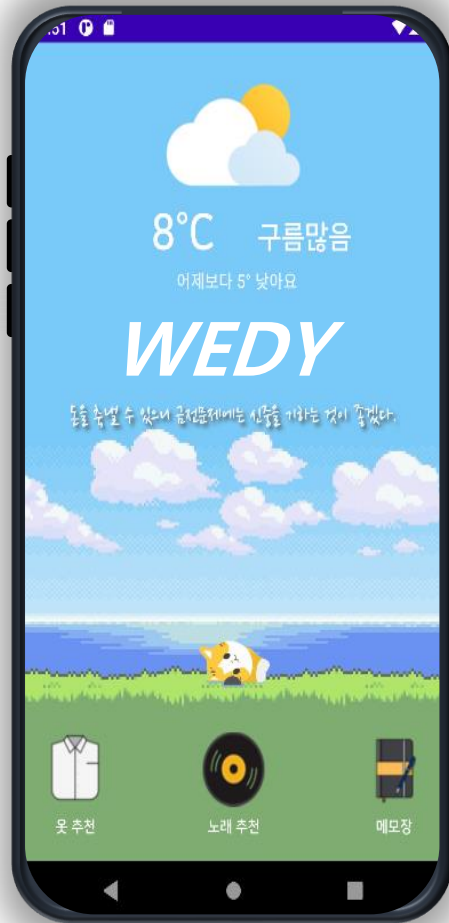


# 날씨별 옷차림/ 음악을 추천하는 Wedy (웨더+코디)



☒ 오늘의 날씨에 따라

☐ 옷을 추천하는 기능

☐ 노래를 추천하는 기능

☐ 힘이 되는 말

코로나 사태로 외출 빈도 낮아짐 – 오랜만에 외출할 때 날씨에 맞지 않는 옷을 입은 적이 많은 경험에서 아이디어 떠올림



# 타 어플과의 차별점



구름냥 - 날씨, 미세먼지, 기온별 옷차림, 날씨 위젯

Guroom Cat 날씨

전체이용가

★★★★★ 811

내 기기에서 사용할 수 있는 앱입니다.

위시리스트에 추가

설치



오늘 뭐 입지?

M2\_Bridge 날씨

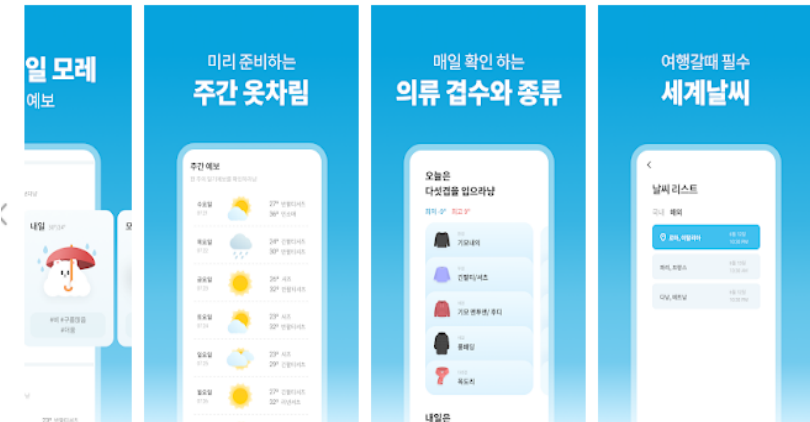
★★★★★ 26

내 기기에서 사용할 수 있는 앱입니다.

내 기기에서 사용할 수 있는 앱입니다.

위시리스트에 추가

설치

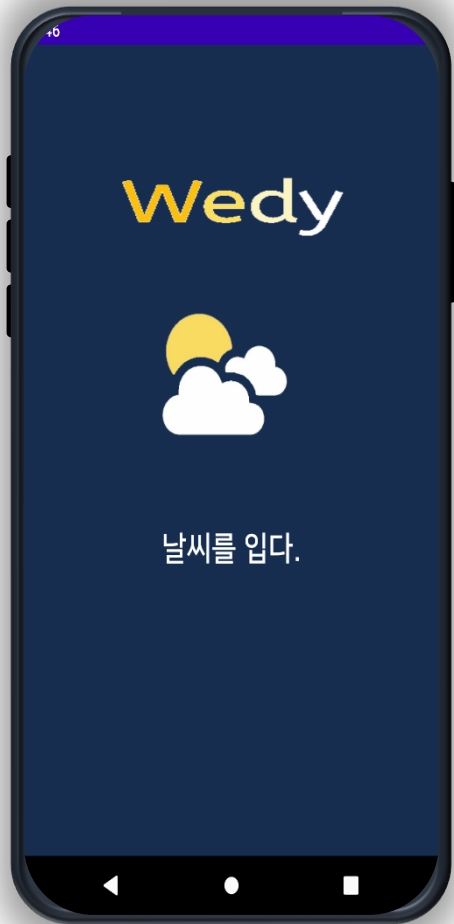


○ 날씨정보와 그에따른 옷의 종류만을 추천하는 타 어플들

- 기온별 옷 추천 뿐만 아니라 실제 코디(조합)까지 추천하는 WEDY
- +쇼핑기능까지 제공하는 WEDY
- 날씨별 음악추천 등 날씨에 필요한 다양한 기능을 가진 WEDY

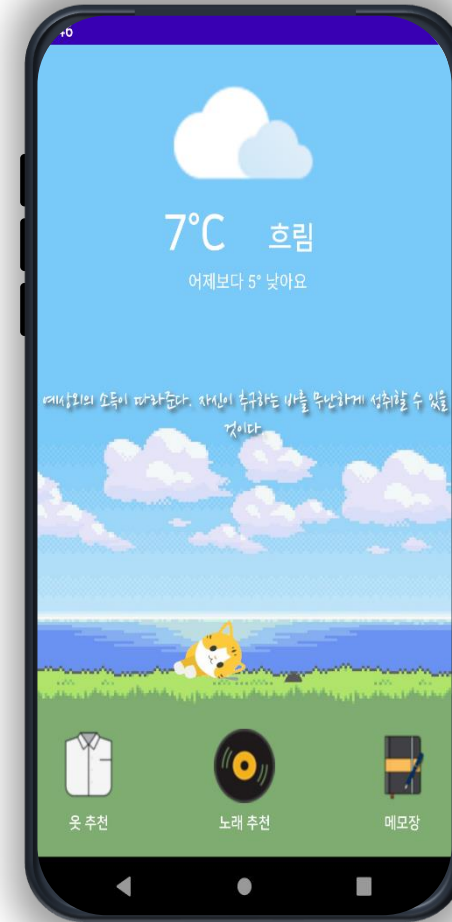


# 1. 로딩화면과 메인화면



로딩화면

앱 실행시 움직이는 이미지와 함께 로딩화면 출력



메인화면

실시간 날씨(서울 기준)

매일 업데이트 되는 오늘의 문구



## 로딩화면: Splash Activity를 통해 구현

```
<activity
    android:name=".SplashActivity"

    android:exported="true">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
<activity
    android:name=".MainActivity"
/>
<activity
    android:name=".Recommend_Music"
/>
<activity
    android:name=".Recommend_DailyLook"
/>
```

← 어플이 시작할 때  
Main Activity 에서 시작하는 것이 아닌  
SplashActivity 에서 시작하도록  
AndroidManifest.xml 수정

SplashActivity.java

```
public class SplashActivity extends AppCompatActivity { //로딩화면

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_splash);

        ImageView gif_image = (ImageView) findViewById(R.id.gif_image);
        Glide.with( activity: this).load(R.drawable.loadingicon).into(gif_image);

        ImageView gif_logo = (ImageView) findViewById(R.id.gif_logo);
        Glide.with( activity: this).load(R.drawable.logo).into(gif_logo);

        Handler handler = new Handler();
        handler.postDelayed(new Runnable() {
            @Override
            public void run() {
                Intent main = new Intent( packageContext: SplashActivity.this, MainActivity.class);
                startActivity(main);
                finish();
            }
        }, delayMillis: 3000); // 로딩시간을 4초로 설정

    }
}
```



## 메인화면: Jsoup 라이브러리를 이용한 웹 크롤링을 통해 실시간 날씨정보와 오늘의 조언 데이터를 가져옴

```
(Thread) run() → {  
  
    try { //현재 기온 가져오기  
        Document doc = Jsoup.connect( url: "https://search.naver.com/search.naver?where=nexearch&sm=top_hyt&fbm=1&ie=utf8&query=%EC%84%9C%EC%9A%B8+%EB%82%A0%EC%94%A8").get();  
        Elements temperature = doc.select( cssQuery: ".temperature_text"); //클어온 html에서 클래스네임이 "temperature_text" 인 값만 선택해서 빼오기  
        temp = temperature.get(0).text().substring(4,7); //temp = contents.get(0).text() 은 contents에서 선별한 클래스중 첫째값을 가져오는 것 그제 현재온도:0'C 이런식으로 나오기때문에 온  
        //하지만 영하일때 -가 4번째 글자이기때문에 -를 가져오기위해선 4번째글자부터 가져와야함  
        temp = temp.replace( target: "도", replacement: ""); //영하가 아닐땐 4번째글자 도가 떨어져나오기에 이를 제거  
        temp = temp.replace( target: "-", replacement: ""); //한자릿수의 기온일시 * 가 떨어져나오기에 이를 제거해줌  
  
        bundle.putString("temperature", temp); //bundle에 temperature라는 키로 temp값 전송  
        Message msg = handler.obtainMessage();  
        msg.setData(bundle);  
        handler.sendMessage(msg);  
  
    } catch (IOException e) {  
        e.printStackTrace();  
    }  
}.start(); //스레드 끝  
  
(Thread) run() → {
```

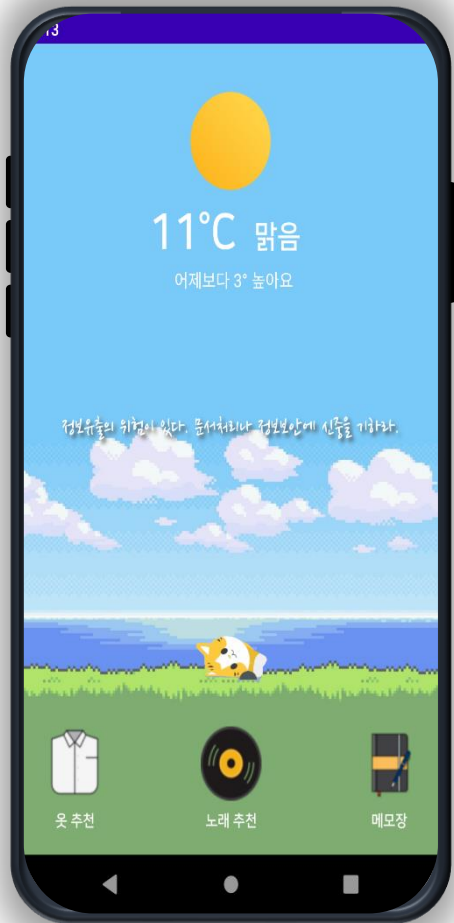
스레드 생성 후 Jsoup을 이용해  
웹페이지의 데이터를 가져온 후  
bundle에 담아 전달

```
Handler handler = handleMessage(msg) → { //핸들러는 메인스레드와 서브스레드를 연결해주는 역할  
    Bundle bundle = msg.getData(); //new Thread에서 작업한 결과를 받기  
    textView.setText(bundle.getString( key: "temperature")+ "°C"); //암호키에 맞는 받아온 데이터 textView에 출력  
};  
  
Handler handler2 = handleMessage(msg) → {  
    Bundle bundle2 = msg.getData();  
    String weatherinfo = bundle2.getString( key: "weatherstatus"); //"어제보다 x도 높아요 맑음  
    weather = weatherinfo.substring(12); //맑음등 날씨정보만 나오도록 13번째 글자부터 끝까지 자르기  
    yesterday = weatherinfo.substring(0,12); //"어제보다 x도 높아요만 떼어내기  
    textView2.setText(weather);  
    textView3.setText(yesterday);  
  
    setWeather();  
};  
  
Handler handler3 = handleMessage(msg) → {  
    Bundle bundle3 = msg.getData();  
    textView4.setText(bundle3.getString( key: "todayMsg"));
```

Bundle에 담겨진 데이터를  
Handler를 통해 받아줌



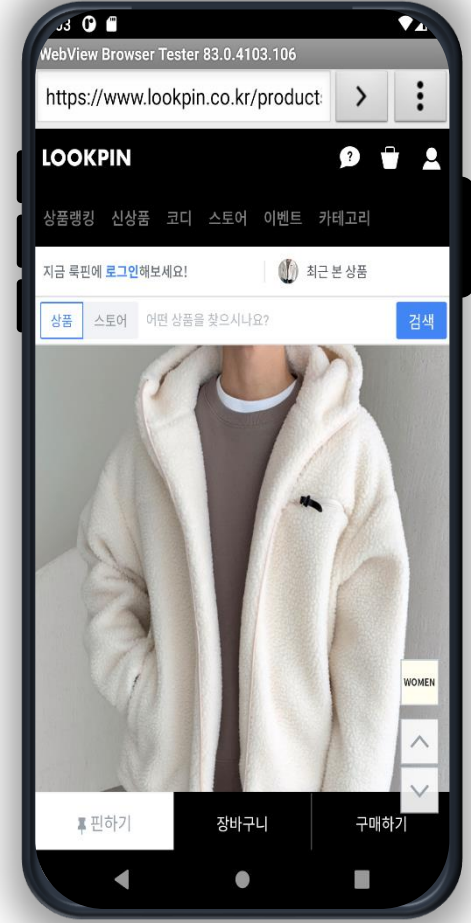
## 2. 옷 추천 기능



메인화면



오늘 날씨에 맞는 코디 추천



쇼핑하기 누르면 해당 코디 사진 속 옷 쇼핑 가능





## 옷 추천 : 이미지 배열 설정 후 랜덤 출력

```
public class Recommend_DailyLook extends AppCompatActivity {
    public int temp;
    public String weatherinfo;

    TextView textWeatherInfo, textMain;
    ImageView imgCloth, imgCharacterCloth;
    ImageButton imgbtnRecall;
    ImageButton imgbtnOuter, imgbtnTop, imgbtnBottom;

    private final static int[] cody1 = new int[] { //랜덤하게 옷 이미지를 뽑기위한 이미지배열
        R.drawable.codycoat1,
        R.drawable.codycoat2,
        R.drawable.codycoat3,
        R.drawable.codyzibup1,
        R.drawable.codyzibup2,
        R.drawable.codyjacket1,
        R.drawable.codyjacket2,
        R.drawable.codyjacket3,
        R.drawable.codyjacket4,
        R.drawable.codyjacket5;
    };
    private final static int[] cody2 = new int[]{
        R.drawable.codypadding1, R.drawable.codypadding2, R.drawable.codypadding3, R.drawable.codypadding4, R.drawable.codypadding5
    };
};
```

```
imgbtnOuter.setOnClickListener(new View.OnClickListener() { //이미지뷰에 온 사진의 아무터 쇼핑
    @Override
    public void onClick(View v) {
        if(imgCloth.getTag().equals(R.drawable.codycoat1)){ //만약 현재 이미지뷰의 태그가 hoodzip의 태그가 같다면 해당 옷정보가 나오도록
            Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("https://www.lookpin.co.kr/products/2229637"));
            startActivity(intent);
        }
        else if(imgCloth.getTag().equals(R.drawable.codycoat2)){
            Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("https://www.lookpin.co.kr/products/2267498"));
            startActivity(intent);
        }
        else if(imgCloth.getTag().equals(R.drawable.codycoat3)){
            Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("https://www.lookpin.co.kr/products/2231482"));
            startActivity(intent);
        }
        else if(imgCloth.getTag().equals(R.drawable.codyzibup1)){
            Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("https://www.lookpin.co.kr/products/2887322"));
            startActivity(intent);
        }
        else if(imgCloth.getTag().equals(R.drawable.codyzibup2)){
            Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("https://www.lookpin.co.kr/products/2273431"));
            startActivity(intent);
        }
        else if(imgCloth.getTag().equals(R.drawable.codyjacket1)){
            Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("https://www.lookpin.co.kr/products/2257537"));
            startActivity(intent);
        }
        else if(imgCloth.getTag().equals(R.drawable.codyjacket2)){
            Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("https://www.lookpin.co.kr/products/2228359"));
            startActivity(intent);
        }
    }
});
```

이미지 배열에 기온별로 코디 사진 넣기  
(서늘한 날씨용 배열엔 코트와 자켓,  
추운 날씨용 배열엔 패딩을 착용한 코디 이미지 등)

```
public void pickCloth(){
    if(temp<=12&&temp>=-2){ //메인액티비티에서 가져온 온도값으로 조건문을통해 옷 배열 출력
        int i = random.nextInt(cody1.length); //이미지 갯수만큼 랜덤 정수로 두고

        imgCloth.setImageResource(cody1[i]); //랜덤으로 이미지배열에서 사진 출력하고
        imgCloth.setTag(cody1[i]); //그 사진의 tag값을 현재 이미지뷰에 저장
    }
    else if(temp<=-2&&temp>=-10){
        int i = random.nextInt(cody2.length);

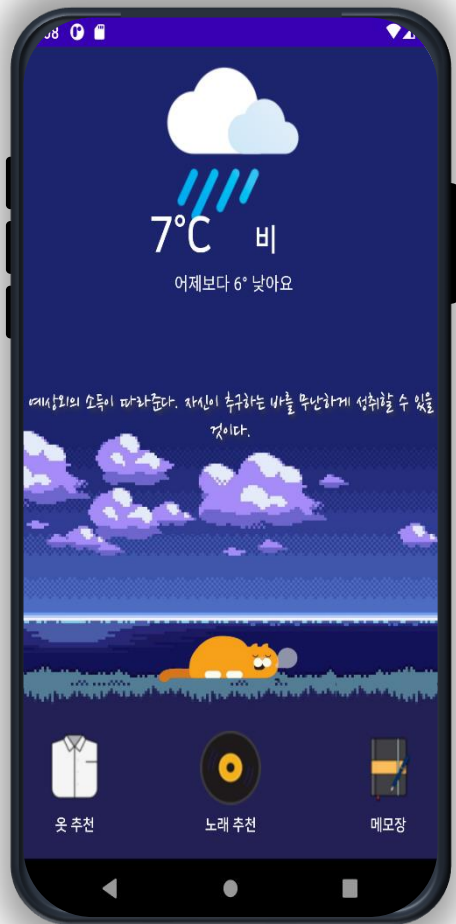
        imgCloth.setImageResource(cody2[i]);
        imgCloth.setTag(cody2[i]);
    }
}
```

MainActivity에서 구한 현재 기온 정보  
를 전달받아 조건문에 이용  
→기온에 따른 이미지 배열을 선택한 후  
배열 속 이미지를 랜덤 출력, 출력 했다면  
출력된 해당 이미지에 태그값을 부여

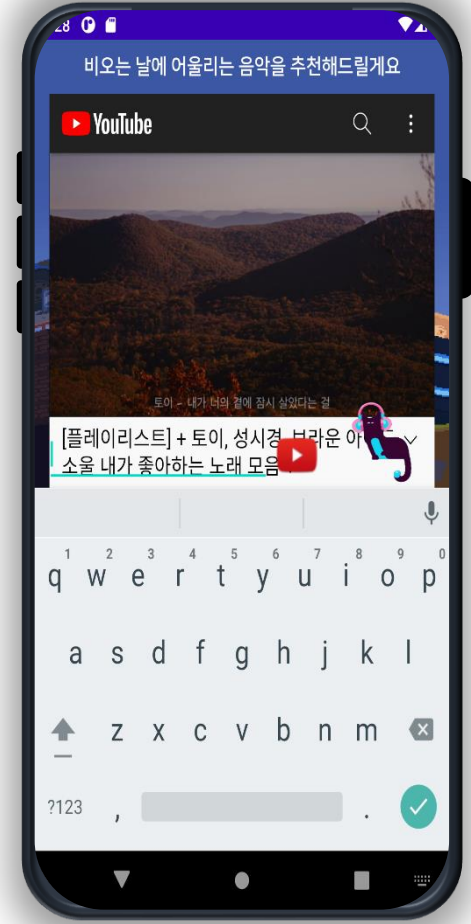
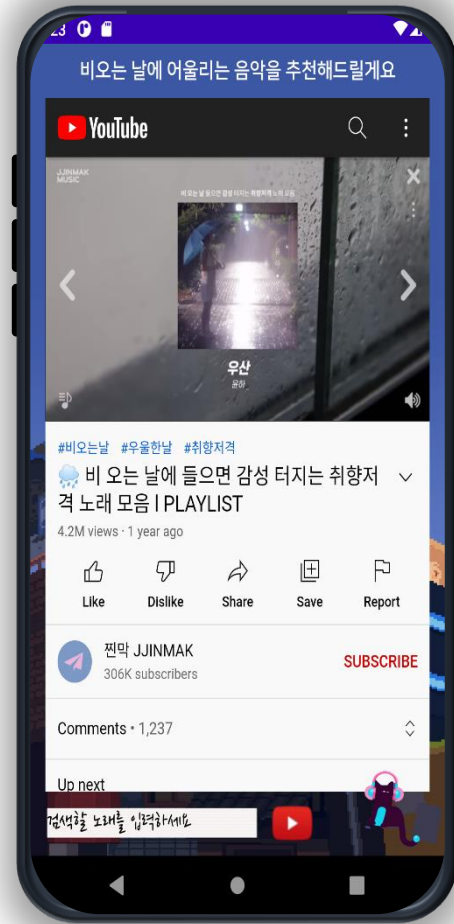
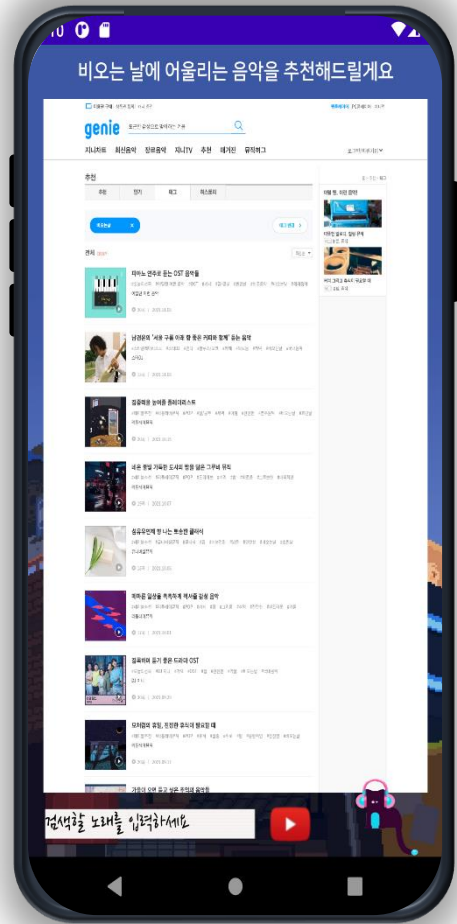
현재 이미지 태그값에 맞는 상  
품을 쇼핑할수 있도록 현재 떠  
있는 코디 이미지의 태그값을  
비교하고, 태그가 같다면 해당  
상품의 쇼핑 링크로 연결



### 3. 노래추천 기능



메인화면



오늘 날씨에 맞는 노래를 추천(지니뮤직, 유튜브) 원하는 노래가 있을 경우 검색 사용





## 노래 추천: 메인 액티비티에서 날씨 정보(맑음, 흐림 등) 받아와 현재 날씨 정보에 따른 플레이리스트 웹뷰로 출력

```
public void Searchmusic(){ //하단에 edittext 창에서 입력한 음악이름을 검색하는 메소드
    String str = musicname.getText().toString();
    if(str.equals("검색할 노래를 입력하세요"))||str.equals("")){ // 만약 검색어를 따로 넣지않고 버튼 누르면 현재 날씨에 맞는 음악 알아서 검색하도록.
        if(weather.contains("맑음")){
            url = "https://www.youtube.com/results?search_query=%EB%A7%91%EC%9D%80%FB%82%A0+%ED%94%8C%EB%A0%88%EC%9D%84%FB%A6%AC%EC%8A%A4%ED%8A%B8";
            webView.loadUrl(url);
        }

        else if(weather.contains("흐림")){
            url = "https://www.youtube.com/results?search_query=%EC%9A%80%EC%9A%B8+%ED%94%8C%EB%A0%88%EC%9D%84%FB%A6%AC%EC%8A%A4%ED%8A%B8";
            webView.loadUrl(url);
        }

        else if(weather.contains("구름")){
            url = "https://www.youtube.com/results?search_query=%EA%B5%AC%EB%A6%84+%ED%94%8C%EB%A0%88%EC%9D%84%FB%A6%AC%EC%8A%A4%ED%8A%B8";
            webView.loadUrl(url);
        }

        else if(weather.contains("비")){
            url = "https://www.youtube.com/results?search_query=%EB%89%84%EC%98%A4%EB%8A%94+%EB%82%A0+%ED%94%8C%EB%A0%88%EC%9D%84%FB%A6%AC%EC%8A%A4%ED%8A%B8";
            webView.loadUrl(url);
        }

        else if(weather.contains("눈")){
            url = "https://www.youtube.com/results?search_query=%EB%88%88%EC%98%A4%EB%8A%94+%EB%82%A0+%ED%94%8C%EB%A0%88%EC%9D%84%FB%A6%AC%EC%8A%A4%ED%8A%B8";
            webView.loadUrl(url);
        }
    }
    else { //검색어를 따로 입력한경우
        str = str.replace( " ", replacement: "+"); //유튜브 주소에선 띄어쓰기를 +로 입력해야함

        url = "https://www.youtube.com/results?search_query=" + str;
        webView.loadUrl(url);
    }
}
```

MainActivity에서 구한 날씨 정보를 받아와 조건문을 통해 해당 날씨 정보에 맞는 플레이리스트를 웹뷰로 출력

```
public boolean onKeyDown(int keyCode, KeyEvent event) { //웹뷰상태에서 뒤로가기 눌르면 원래는 앱이 꺼짐 -> 전화면으로돌아가기위해 설정
    if((keyCode == KeyEvent.KEYCODE_BACK) && webView.canGoBack()){
        webView.goBack();
        return true;
    }

    return super.onKeyDown(keyCode, event);
}
```

원래는 웹뷰를 보고 있는 상황에서 뒤로가기 버튼을 누르면 앱이 나가짐  
→ 이 코드를 통해 해결



# 향후 개선 방향: 개인화

## 개선점

1. 다양한 연령대/성별에서 더 다양한 옷과 음악을 추천
2. 다른 기능은?

현재는 20대 남성 위주의 옷차림으로 제한되어 있음  
→회원가입 & 로그인 기능을 만들어 해당 회원의 성별, 연령, 취향 등에 맞춘 서비스를 제공하는 방향으로 발전시킬 계획

