

Ch10. 액티비티와 인텐트2

모바일앱프로그래밍 김지심교수

학습목표

액티비티와 인텐트를 이해하고 활용할 수 있다. 액티비티 생명주기를 이해하고 활용할 수 있다.

목차

- 1. 액티비티와 인텐트 기본 2. 명시적 인텐트 3. 암시적 인텐트 4. 액티비티 생명주기

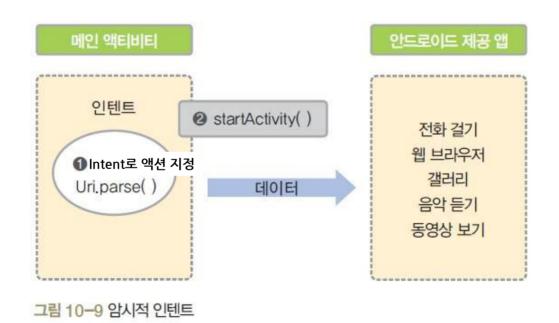
03

암시적 인텐트

암시적 인텐트(Implicit Intent)

약속된 액션(Action)을 지정하여 안드로이드 OS에 넘기고 OS가 액티비티를 찿아 시작시킴

ex. 전화를 걸어주기를 안드로이드 OS에 요청하면 OS가 전화 걸기 앱에 넘겨 전화 걸기 실행



암시적 인텐트의 기본 구성요소 Action(수행할 기능), Data(수행될 대상 데이터)

속성 설명

ACTION_DIAL tel:01077881234 주어진 전화번호를 이용해 전화걸기 화면을 보여줌

ACTION VIEW 웹주소 웹브라우저에 해당 페이지를 보여줌

ACTION_EDIT content://contacts/people/2 전화번호부 데이터베이스에 있는 정보 중에서 ID 값이 2인

정보를 편집하기 위한 화면을 보여줌

ACTION VIEW content://contacts/people 전화번호부 데이터베이스의 내용을 보여줌

ex. 119에 응급전화를 거는 형식

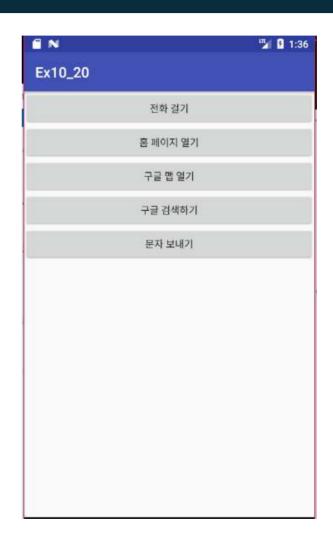
```
Intent intent = new Intent(Intent.ACTION_DIAL, Uri.parse("tel:/119"));
startActivity(intent);
```

전화 걸기를 하려면 AndroidManifest.xml에 다음과 같이 권한을 추가해야 함(optional)

(uses-permission android:name="android.permission.CALL_PHONE"/>

실습: Button을 클릭하면 전화걸기 앱을 실행하여 119로 전화하는 화면을 띄워보자.

예제10-20: Ex10_20 암시적 인텐트 예제



activity_main.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"</pre>
            android: layout width="match parent"
            android: layout height="wrap content"
            android:orientation="vertical" >
            <Button
                android:id="@+id/btnDial"
                android: layout width="match parent"
 9
                android: layout height="wrap content"
                android:text="전화 걸기" />
19
11
12
            <Button
                android:id="@+id/btnWeb"
13
14
                android: layout width="match parent"
15
                android:layout_height="wrap_content"
                android:text="홈페이지 열기" />
16
17
18
            <Button
19
                android:id="@+id/btnGoogle"
20
                android: layout width="match parent"
                android:layout_height="wrap_content"
21
                android:text="구글 맵 열기" />
22
23
24
            <Button
                android:id="@+id/btnSearch"
25
26
                android:layout_width="match_parent"
27
                android:layout_height="wrap_content"
                android:text="구글 검색하기" />
28
29
30
            <Button
31
                android:id="@+id/btnSms"
                android:layout_width="match_parent"
32
                android:layout_height="wrap_content"
33
                android:text="문자 보내기" />
34
35
36
       </LinearLayout>
```

```
public class MainActivity extends AppCompatActivity {
13
14
             @Override
             protected void onCreate(Bundle savedInstanceState) {
15 0
                  super.onCreate(savedInstanceState);
16
                  setContentView(R.layout.activity main);
17
18
                  Button btnDial = (Button) findViewById(R.id.btnDial);
19
                  Button btnWeb = (Button) findViewById(R.id.btnWeb);
28
                  Button btnGoogle = (Button) findViewById(R.id.btnGoogle);
21
                  Button btnSearch = (Button) findViewById(R.id.btnSearch);
22
23
                  Button btnSms = (Button) findViewById(R.id.btnSms);
24
                  btnDial.setOnClickListener(new View.OnClickListener() {
25
                      public void onClick(View v) {
26 6
27
                          Uri uri = Uri.parse("tel:010-1234-5678");
28
                         Intent intent = new Intent(Intent.ACTION_DIAL, uri);
29
                          startActivity(intent);
30
31
32
                 });
33
                  btnWeb.setOnClickListener(new View.OnClickListener() {
34
                      public void onClick(View v) {
35 6
36
                          Uri uri = Uri.parse("http://www.mjc.ac.kr");
37
                          Intent intent = new Intent(Intent.ACTION_VIEW, uri);
                          startActivity(intent);
38
39
                 });
40
```

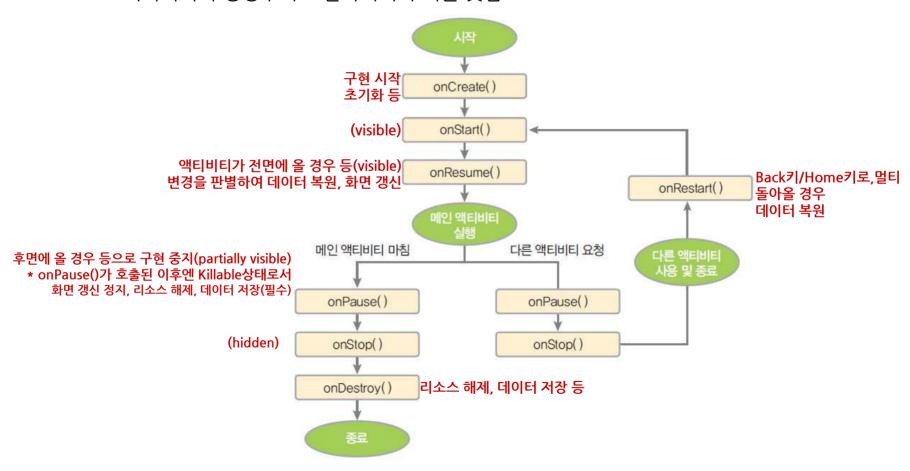
```
btnGoogle.setOnClickListener(new View.OnClickListener() {
42
43 0
                   public void onClick(View v) {
                       Uri uri = Uri.parse("http://maps.google.com/maps?q=" + 37.584402 + "," + 126.924508);
44
                       Intent intent = new Intent(Intent.ACTION VIEW, uri);
45
                       startActivity(intent);
46
47
               });
48
49
               btnSearch.setOnClickListener(new View.OnClickListener() {
50
                   public void onClick(View v) {
51 0
                       Intent intent = new Intent(Intent.ACTION_WEB_SEARCH);
52
                       intent.putExtra(SearchManager.QUERY, "안드로이드");
53
                       startActivity(intent);
54
55
               });
56
57
58 0
               btnSms.setOnClickListener(onClick(v) → {
                       Uri smsUri = Uri.parse("sms:010-1234-1234");
60
                       Intent intent = new Intent(Intent.ACTION_SENDTO, smsUri);
61
                       intent.putExtra("sms_body", "안녕하세요?");
62
                       startActivity(intent);
63
               });
64
66
67
68
```

04

액티비티 생명주기

액티비티 생명주기

액티비티의 생성부터 소멸까지의 주기를 뜻함



액티비티의 상태 메소드 상세

상태 메소드	설 명
onCreate()	- 액티비티가 처음에 만들어졌을 때 호출됨 - 화면에 보이는 뷰들의 일반적인 상태를 설정하는 부분 - 이전 상태가 저장되어 있는 경우에는 번들 객체를 참조하여 이전 상태 복원 가능 - 이 메소드 다음에는 항상 onStart() 메소드가 호출됨
onStart()	- 액티비티가 화면에 보이기 바로 전에 호출됨 - 액티비티가 화면 상에 보이면 이 메소드 다음에 onResume() 메소드가 호출됨 - 액티비티가 화면에서 가려지게 되면 이 메소드 다음에 onStop() 메소드가 호출됨
onResume()	- 액티비티가 사용자와 상호작용하기 바로 전에 호출됨
onRestart()	- 액티비티가 중지된 이후에 호출되는 메소드로 다시 시작되기 바로 전에 호출됨 - 이 메소드 다음에는 항상 onStart() 메소드가 호출됨
onPause()	- 또 다른 액티비티를 시작하려고 할 때 호출됨 - 저장되지 않은 데이터를 저장소에 저장하거나 애니메이션 중인 작업을 중지하는 등의 기능을 수행하는 메소드임 - 액티비티가 이 상태에 들어가면 시스템은 액티비티를 강제 종료할 수 있음
onStop()	- 액티비티가 사용자에게 더 이상 보이지 않을 때 호출됨 - 액티비티가 소멸되거나 또 다른 액티비티가 화면을 가릴 때 호출됨 - 액티비티가 이 상태에 들어가면 시스템은 액티비티를 강제 종료할 수 있음
onDestroy()	- 액티비티가 소멸되어 없어지기 전에 호출됨 - 이 메소드는 액티비티가 받는 마지막 호출이 됨 - 액티비티가 애플리케이션에 의해 종료되거나(finish() 호출) 시스템이 강제로 종료시키는 경우에 호출될 수 있음 - 위의 두 가지 경우를 구분할 때 isFinishing() 메소드를 이용함 - 액티비티가 이 상태에 들어가면 시스템은 액티비티를 강제 종료할 수 있음



한마디 - 로그캣(LogCat)

작성 중인 프로그램에 예기치 못한 오류가 발생했을 때 원인을 파악하는 방법 중 하나가 로그(Log)를 남기는 것 이다. 안드로이드는 android.util,Log 클래스를 제공해서 로그를 남기고, 로그캣(LogCat)이라는 화면을 제공해 서 로그를 확인한다. 프로그래머가 로그를 남기기 위해 사용하는 메소드는 다음 표와 같다. 하지만 절대적인 기 준은 아니며, 프로그래머가 적절한 메소드를 골라 사용해야 한다.

메소드	설명
android.util.Log.d("태그", "메시지")	Debugging : 디버깅 용도로 남기는 로그
android.util.Log.e("태그", "메시지")	Error : 가장 심각한 오류 발생시 남기는 로그
android.util.Log.i("태그", "메시지")	Information : 정보를 남기기 위한 로그
android.util.Log.v("태그", "메시지")	Verbose : 상세한 기록을 남기기 위한 로그
android.util.Log.w("태그", "메시지")	Warning : 경고 수준을 남기기 위한 로그

실습 10-3: Project10_03 MainActivity에서 전화걸기 앱을 실행하여 뒤로 버튼을 눌러 돌아오자.

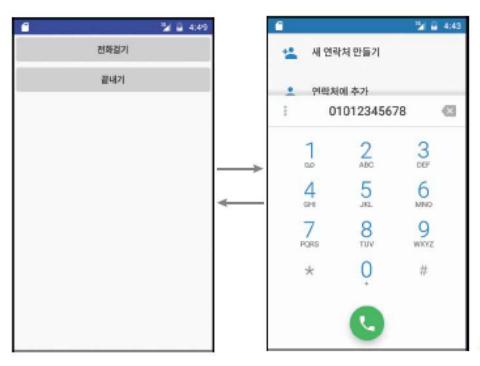


그림 10-11 액티비티 생명주기 테스트 실행 화면

activity_main.xml LogCat으로 메소드 실행 순서 확인

```
에제 10-22 activity_main.xml

1 〈LinearLayout〉
2 〈Button
3 android:id="@+id/btnDial"
4 android:text="전화걸기" /〉
5 〈Button
6 android:id="@+id/btnFinish"
7 android:text="끝내기" /〉
8 〈/LinearLayout〉
```

```
11 8
        public class MainActivity extends AppCompatActivity {
            @Override
            public void onCreate(Bundle savedInstanceState) {
14
                super.onCreate(savedInstanceState);
15
                setContentView(R.layout.activity main);
                setTitle("H 스트 明月");
16
                android.util.Log.i("日 스트", "onCreate()");
17
18
19
                Button btnDial = (Button) findViewById(R.id.btnDial);
26
                btnDial.setOnClickListener(new View.OnClickListener() {
21 0
                    public void onClick(View v) {
22
                        Uri uri = Uri.parse("tel:010-1234-56780"):
23
                        Intent intent = new Intent(Intent.ACTION DIAL, uri);
                        startActivity(intent);
24
2.5
                });
26
27
28
                Button btnFinish = (Button) findViewById(R.id.btnFinish);
29
                btnFinish.setOnClickListener(new View.OnClickListener() {
                    public void onClick(View v) {
30 €
31
                        finish();
32
33
                });
34
35
36
            @Override
37 0
            protected void onDestroy() {
38
                super.onDestroy():
                Log.i("테스트", "onDestroy()");
39
46
41
42
            @Override
43 0
            protected void onPause() {
44
                super.onPause();
                Log. i("H 스트", "onPause()");
45
46
47
48
            Goverride
49 0
            protected void onRestart() {
50
                super.onRestart();
                Log.i("H스트", "onRestart()");
51
52
```

```
@Override
54
55 0
            protected void onResume() {
                super.onResume();
56
               Log.i("테스트", "onResume()");
57
58
59
           @Override
60
            protected void onStart() {
61 0
                super.onStart();
62
               Log.i("테스트", "onStart()");
63
64
65
            @Override
66
            protected void onStop() {
67 of -
                super.onStop();
68
               Log.i("터스트", "onStop()");
69
70
71
72
```

프로젝트 실행 및 결과 확인

Android Studio 하단의 패널 중, [Android Monitor] 클릭 〉 [logcat] 탭을 선택 결과를 실행하여 각 단계마다 로그캣에서 생명주기 확인

