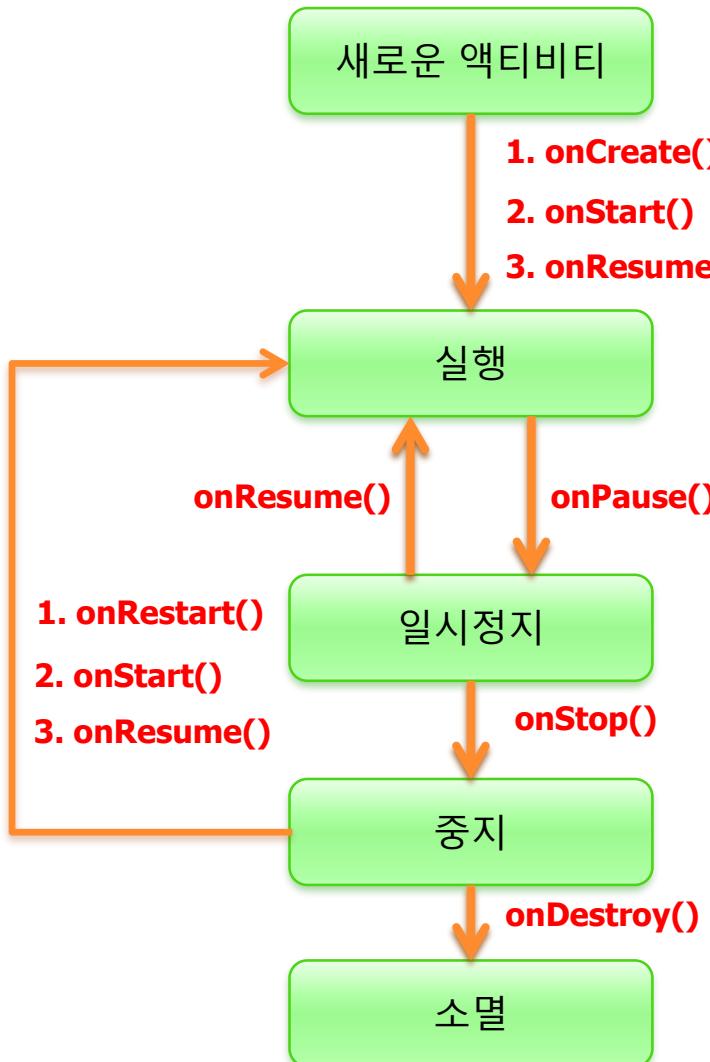




생명주기 (LIFECYCLE)





onCreate()	<ul style="list-style-type: none">액티비티가 처음에 만들어졌을 때 호출화면에 보이는 뷰들의 일반적인 상태를 설정하는 부분이전 상태가 저장되어 있는 경우에는 번들 객체를 참조하여 이전 상태 복원 가능이 메소드 다음에는 항상 onStart() 메소드가 호출됨
onStart()	<ul style="list-style-type: none">액티비티가 화면에 보이기 바로 전에 호출액티비티가 화면 상에 보이면 이 메소드 다음에 onResume() 모소드가 호출액티비티가 화면에서 가려지게 되면 이 메소드 다음에 onStop() 메소드가 호출
onResume()	<ul style="list-style-type: none">액티비티가 사용자와 상호작용하기 바로 전에 호출
onRestart()	<ul style="list-style-type: none">액티비티가 중지된 이후에 호출되는 메소드로 다시 시작되기 바로 전에 호출이 메소드 다음에는 항상 onStart() 메소드가 호출
onPause()	<ul style="list-style-type: none">또 다른 액티비티를 시작하려고 할 때 호출저장되지 않은 데이터를 저장소에 저장하거나 애니메이션 중인 작업을 중지하는 등의 기능을 수행하는 메소드이 메소드가 리턴하기 전에는 다음 액티비티가 시작될 수 없으므로 이 작업은 매우 빨리 수행된 후 리턴되어야 함.액티비티가 이 상태에 들어가면 시스템은 액티비티를 강제 종료할 수 있음
onStop()	<ul style="list-style-type: none">액티비티가 사용자에게 더 이상 보이지 않을 때 호출됨액티비티가 소멸되거나 또 다른 액티비티가 화면을 가릴 때 호출됨액티비티가 이 상태에 들어가면 시스템은 액티비티를 강제 종료할 수 있음
onDestroy()	<ul style="list-style-type: none">액티비티가 소멸되어 없어지기 전에 호출이 메소드는 액티비티가 받는 마지막 호출이 됨액티비티가 애플리케이션에 의해 종료되거나(finish() 메소드 호출) 시스템이 강제로 종료시키는 경우에 호출될 수 있음위의 두 가지 경우를 구분할 때 (isFinishing() 메소드 호출) 시스템액티비티가 이 상태에 들어가면 시스템은 액티비티를 강제 종료할 수 있음

프로젝트 생성



Application name: [LifeCycle](#)

Company Domain: admid.example.com

Project location: [C:\Android\StdWork\lifecycle](#)

Target Android Devices

- Phone and Tablet

- Minimum SDK API 17 : Android 4.0.3

Add an activity to Mobile : [Empty Activity](#)

Customize the Activity

- Activity Name: [MainActivity](#)

- Layout Name: [activity_main](#)



activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="메소드 실행 절차" />

</LinearLayout>
```

The screenshot shows the Android Studio interface with the XML code for the activity_main.xml layout. The code defines a LinearLayout containing a single TextView with the text "메소드 실행 절차". To the right, a preview window displays the resulting application screen. The title bar of the application is labeled "Lifecycle" and the main content area contains the text "메소드 실행 절차". The bottom of the preview window has a white bar.

```
>MainActivity.java x  
1 package com.example.admin.firstmessage;  
2  
3 +import ...  
4  
5 public class MainActivity extends AppCompatActivity {  
6  
7     @Override  
8     protected void onCreate(Bundle savedInstanceState) {  
9         super.onCreate(savedInstanceState);  
10        setContentView(R.layout.activity_main);  
11        Toast.makeText(getApplicationContext(),  
12            "onCreate", Toast.LENGTH_SHORT).show();  
13    }  
14  
15    @Override  
16    protected void onDestroy() {  
17        Toast.makeText(getApplicationContext(),  
18            "onDestroy()", Toast.LENGTH_SHORT).show();  
19        super.onDestroy();  
20    }  
21  
22 }
```

>MainActivity.java

```
23
24     @Override
25     protected void onResume() {
26         Toast.makeText(getApplicationContext(),
27             "onResume()", Toast.LENGTH_SHORT).show();
28         super.onResume();
29     }
30
31     @Override
32     protected void onStart() {
33         Toast.makeText(getApplicationContext(),
34             "onStart()", Toast.LENGTH_SHORT).show();
35         super.onStart();
36     }
37
38     @Override
39     protected void onStop() {
40         Toast.makeText(getApplicationContext(),
41             "onStop()", Toast.LENGTH_SHORT).show();
42         super.onStop();
43     }
44
45     @Override
46     protected void onPause() {
47         Toast.makeText(getApplicationContext(),
48             "onPause()", Toast.LENGTH_SHORT).show();
49         super.onPause();
50     }
51 }
```

