

포그라운드 작업과 백그라운드 작업

사용자가 터미널에서 작업할 때 보통의 경우 한번에 하나의 프로세스만 실행할 수 있다. 즉, 어떤 명령을 실행하면 명령이 끝날 때까지 기다려야 다시 터미널을 사용할 수 있는 것이다. 그런데 사용자가 어떤 프로세스를 실행하면서 동시에 다른 프로세스를 실행해야 할 때가 있다. 작업제어 도구는 이렇게 한 터미널에서 동시에 여러 프로세스를 실행하고 관리하기 위해 개발되었다.

포그라운드 작업

터미널에서 작업할 때 일반적으로 사용자가 명령을 입력하면 셸은 사용자가 입력한 명령을 해석하여 실행하고 그 결과를 화면에 출력한다. 그러면 사용자는 화면에 출력된 결과를 보고 다시 명령을 입력하는 대화식으로 작업을 수행한다. 이렇게 사용자가 입력한 명령이 실행되어 결과가 출력될때 까지 기다려야 하는 포그라운드 방식으로 처리되는 프로세스를 포그라운드 프로세스라고 하며, 이를 작업 제어에서는 포그라운드 작업이라고 한다.

포그라운드 작업은 일반적으로 사용자가 명령을 실행하는 방식으로, 안드로이드 앱을 실행시켜 화면에 띄워놓았을 때의 작업이다. 가시적 수명이라고 일컫어지는 가시적 수명 onStart() ~ onStop()까지의 프로세스를 말하며, 사용자가 액티비티를 화면에서 보고 이와 상호작용 할 수 있다.

포그라운드 서비스는 서비스의 동작을 사용자가 명시적으로 인지하는 대신 메모리 부족시 시스템에 의한 종료 대상에서 제외된다.

백그라운드 작업

포그라운드 작업은 명령을 한 번에 하나씩 실행하므로 동시에 여러 개의 프로세스를 실행할 수 없다. 그러나 작업 제어가 제공하는 백그라운드 기능을 사용하면 앞에서 프로세스가 실행되는 동안 뒤에서 다른 프로세스가 실행될 수 있으므로 한 터미널에서 여러 개의 프로세스를 동시에 실행할 수 있다.

백그라운드 방식으로 명령을 실행하면 명령의 처리가 끝나는 것과 관계없이 백그라운드에서 다른 작업을 계속 할 수 있다. 즉 필요한 여러 작업을 백그라운드로 실행한 후 터미널에서는 포그라운드 작업을 계속 진행할 수 있다. 이처럼 백그라운드 방식으로 처리되는 프로세스를 백그라운드 프로세스라고 하며, 작업 제어에서는 백그라운드 작업이라고 한다. 백그라운드 작업은 명령의 실행 시간이 많이 걸릴 것으로 예상되거나 명령을 실행한 후 다른 작업을 할 필요가 있을 때 많이 사용한다.

현재 Activity는 더 이상 화면에 보이지 않게 될 때 호출되는 메소드인 onStop() 이후, onDestroy()이전까지 백그라운드로 실행된다.

백그라운드 서비스는 시스템이 리소스가 부족한 경우 임의로 중단할 수 있다.

onStartCommand()에서 종료시 다음 동작을 정의

START_NOT_STICKY : 서비스 재 실행하지 않음

START_STICKY : 재생성과 onStartCommand() 호출 (null intent)

START_REDELIVER_INTENT : 재생성과 onStartCommand() 호출 (same intent)

Oreo background 실행제한

백그라운드 실행 제한

백그라운드에서 실행될 때마다 앱은 기기의 제한된 리소스를 사용하는데 리소스를 과도하게 사용할 경우 사용자 환경이 손상될 수 있다. 때문에 Android 8.0 오레오 버전부터 백그라운드에서 실행되면서 앱의 동작을 제한하여 한정된 자원을 보다 효율적으로 사용하기 시작했다.

보통 많은 수의 Android 앱과 서비스를 동시에 실행할 수 있지만 동시에 실행되는 앱이 더 많을수록 시스템에 더 많은 부하가 걸리기 때문에 사용자가 앱과 직접 상호작용하지 않을 때 이 앱이 수행할 수 있는 작업을 제한합니다. 앱은 두 가지 방식으로 제한됩니다.

백그라운드 서비스 제한: 앱이 유훈 상태인 경우 백그라운드 서비스의 사용이 제한됩니다. 이 기능은 사용자에게 잘 보이는 포그라운드 서비스에는 적용되지 않습니다.

브로드캐스트 제한: 제한된 예외의 경우, 앱이 암시적 브로드캐스트에 등록하기 위해 자체 매니페스트를 사용할 수 없습니다. 그렇지만 여전히 앱이 런타임에 브로드캐스트에 등록할 수 있으며, 특정 앱을 대상으로 하는 명시적 브로드캐스트에 등록하기 위해 매니페스트를 사용할 수 있습니다.

(? O를 대상으로 하는 앱, O가 뭐지)

포그라운드와 백그라운드의 구분

앱이 포그라운드에 있는 경우

1. 액티비티가 시작되거나 일시 중지되거나 상관없이 보이는 액티비티가 있는 경우
2. 포그라운드 서비스가 있는 경우
3. 앱의 서비스 중 하나에 바인드하거나 앱의 콘텐츠 제공자 중 하나를 사용하여 앱에 또 다른 포그라운드 앱이 연결된 경우
(예를 들어 다른 앱이 다음 항목에 바인드되어 있다면 포그라운드에 있는 것)

- IME
- 배경화면 서비스
- 알람 리스너
- 음성 또는 텍스트 서비스

백그라운드 서비스 제한

위의 어떤 조건에도 해당하지 않는 경우 백그라운드에 있는 것으로 간주합니다.

단, 아래의 작업을 처리하는 경우에는 제한없이 서비스를 시작할 수 있으며, 백그라운드 서비스도 실행이 허용됩니다.

- 우선순위가 높은 Firebase 클라우드 메시징(FCM) 메시지 처리
- SMS/MMS 메시지와 같은 브로드캐스트 수신
- 알림에서 PendingIntent 실행

백그라운드에서 Service를 주기적으로 실행하기 위해서는 서비스를 포그라운드에서 시작하는 새로운 메서드 Context.startForegroundService()를 사용해야 합니다. 시스템이 서비스를 생성한 후, 앱은 5초 이내에 해당 서비스의 startForeground() 메서드를 호출하여 새 서비스의 알림을 사용자에게 표시해야 하며, 앱이 이 시간 한도 내에 startForeground()를 호출하지 않으면 시스템이 서비스를 중단하고 이 앱을 ANR로 선언합니다.

브로드캐스트 제한

앱이 브로드캐스트를 수신하도록 등록한 경우, 앱의 수신기는 브로드캐스트가 전송될 때마다 리소스를 소비합니다. 이 경우 너무 많은 앱이 시스템 이벤트 기반의 브로드캐스트를 수신하도록 등록하면 문제가 될 수 있습니다. 브로드캐스트를 트리거하는 시스템 이벤트로 인해 모든 앱들이 급속하게 리소스를 소비할 수 있으며 이로 인해 사용자 환경이 손상될 수 있습니다. 이러한 문제를 줄이기 위해 Android 7.0부터 브로드캐스트에 제한을 두었으며, Android 8.0는 이러한 제한을 더욱 엄격하였습니다.

참고 URL : <https://developer.android.com/about/versions/oreo/background>