Android Developer Fundamentals V2

Background Tasks

Lesson 7

7.4 Services

Contents

- Services for long tasks.
- `IntentService`

Services is an advanced topic

- Services are complex.
- Many ways of configuring a service.
- This lesson has introductory information only.
- Explore and learn for yourself if you want to use services.

Services for Long Tasks

What is a service?

A `Service` is an application component that can perform long-running operations in the background and does not provide a user interface.

What are services good for?

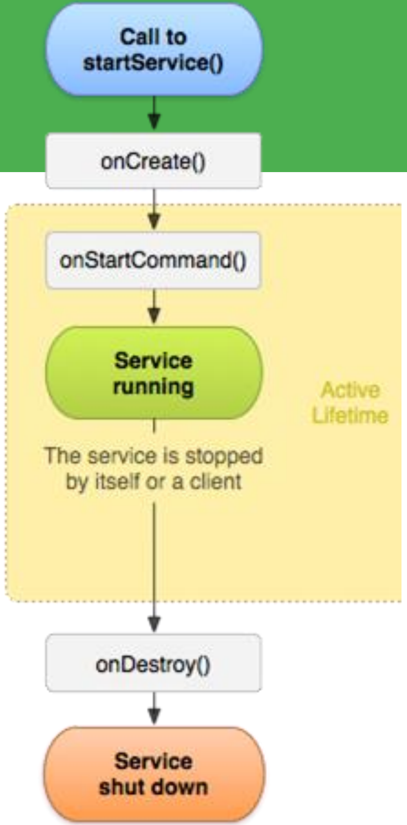
- Network transactions.
- Play music.
- Perform file I/O.
- Interact with a database.

Characteristics of services

- Started with an `Intent`.
- Can stay running when user switches applications.
- Lifecycle—which you must manage.
- Other apps can use the service—manage permissions.
- Runs in the main thread of its hosting process.

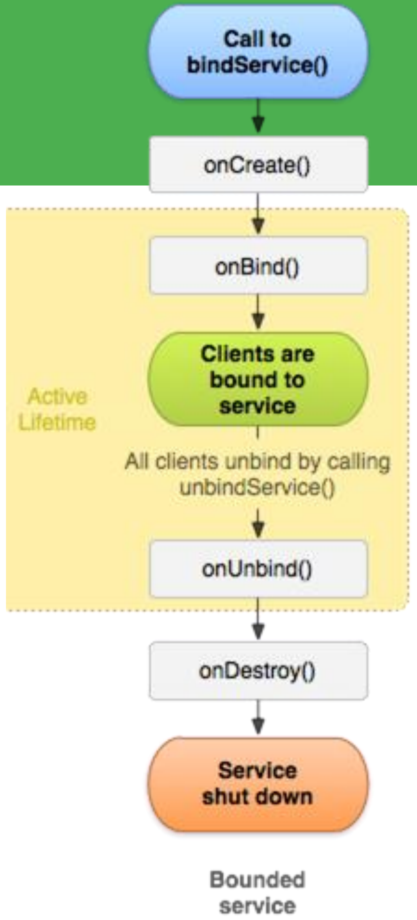
Forms of services: started

- Started with `startService()`
- Runs indefinitely until it stops itself
- Usually does not update the UI



Forms of services: bound

- Offers a client-server interface that allows components to interact with the service
- Clients send requests and get results
- Started with `bindService()`
- Ends when all clients unbind



Services and threads

Although services are separate from the UI, they still run on the main thread by default (except `IntentService`)

Offload CPU-intensive work to a separate thread within the service

Updating the app

If the service can't access the UI, how do you update the app to show the results?

Use a broadcast receiver!

Foreground services

Runs in the background but requires that the user is actively aware it exists—e.g. music player using music service

- Higher priority than background services since user will notice its absence—unlikely to be killed by the system
- Must provide a notification which the user cannot dismiss while the service is running

Background services limitations

- Starting from API 26, background app is not allowed to create a background service.
- A foreground app, can create and run both foreground and background services.
- When an app goes into the background, the system stops the app's background services.
- The `startService()` method now throws an `IllegalStateException` if an app is targeting API 26.
- These limitations don't affect foreground services or bound services.

Creating a service

- `<service android:name=".ExampleService" />`
- Manage permissions.
- Subclass `IntentService` or `Service` class.
- Implement lifecycle methods.
- Start service from `Activity`.
- Make sure service is stoppable.

Stopping a service

- A **started service** must manage its own lifecycle
- If not stopped, will keep running and consuming resources
- The service must stop itself by calling stopSelf()
- Another component can stop it by calling stopService()
- **Bound service** is destroyed when all clients unbound
- **IntentService** is destroyed after onHandleIntent() returns

IntentService

IntentService

- Simple service with simplified lifecycle
- Uses worker threads to fulfill requests
- Stops itself when done

- Ideal for one long task on a single background thread

IntentService Limitations

- Cannot interact with the UI
- Can only run one request at a time
- Cannot be interrupted

IntentService restrictions

- `IntentService` are subjected to the new restrictions on background services.
- For the apps targeting API 26, Android Support Library 26.0.0 introduces a new `JobIntentService`.
- `JobIntentService` provides the same functionality as `IntentService` but uses jobs instead of services.

IntentService Implementation

```
public class HelloIntentService extends IntentService {
    public HelloIntentService() { super("HelloIntentService");}

    @Override
  protected void onHandleIntent(Intent intent) {
      try {
          // Do some work
      } catch (InterruptedException e) {
          Thread.currentThread().interrupt();
      }
  } // When this method returns, IntentService stops the service, as appropriate.
}
```

Learn more

- [Services overview]
- [Background Execution Limits]

What's Next?

- Concept Chapter: 7.4 Services
- No practical

END