



Marlon Silva Carvalho
@marlonscarvalho
marlon.carvalho@gmail.com



<http://developer.android.com/guide/components/services.html>

Serviços

Permite a execução de tarefas...

...longas em background. Tarefas que não necessitam de uma UI para o usuário



<http://developer.android.com/guide/components/services.html>

Serviços

Mesmo que o objeto que iniciou o serviço...
...deixe de existir, o serviço continua executando



<http://developer.android.com/guide/components/services.html>

Serviços

Parece a ideia de Threads...

...mas NÃO É. A Thread morre junto com quem iniciou ela.



<http://developer.android.com/guide/components/services.html>

Serviços

Serviços podem executar constantemente...

...e prover informações para outras aplicações ou para a sua.



<http://developer.android.com/guide/components/services.html>

Serviços
Ou podem realizar...
...uma determinada tarefa e morrer!



<http://developer.android.com/guide/components/services.html>

Serviços

Serviços NÃO SÃO uma Thread...

...ou um processo separado da sua aplicação.



<http://developer.android.com/guide/components/services.html>

Serviços

Rodam na mesma Thread da aplicação...

...portanto, podem bloquear a UI do seu aplicativo!



<http://developer.android.com/guide/components/services.html>

Serviços

Para ter um Serviço, você pode estender de...
android.app.Service ou android.app.IntentService



<http://developer.android.com/guide/components/services.html>

android.app.Service

Uma instância de...

...android.app.Service e que implementa onStartCommand()

```
public class Servico extends Service {
```

```
    @Override
```

```
    public int onStartCommand(Intent intent, int flags, int startId) {  
    }
```

```
}
```

```
<service android:name=".Servico" >  
</service>
```



<http://developer.android.com/guide/components/services.html>

android.app.Service

Para lançar um serviço...

...usa-se uma Intent e startService()

```
public class Servico extends Service {
```

```
    @Override
```

```
    public int onStartCommand(Intent intent, int flags, int startId) {  
    }
```

```
}
```

```
<service android:name=".Servico" >  
</service>
```



<http://developer.android.com/guide/components/services.html>

android.app.Service

Neste formato, você TEM que tratar...

...multithreading. O ideal é sempre lançar Threads senão trava a UI.

```
public class Servico extends Service {
```

```
    @Override
```

```
    public int onStartCommand(Intent intent, int flags, int startId) {  
    }
```

```
}
```

```
<service android:name=".Servico" >  
</service>
```



Prática 1

Service

Criar um Service para executar uma tarefa



<http://developer.android.com/guide/components/services.html>

android.app.IntentService

Já lança uma WorkerThread...

...para cada chamada ao serviço. Você não precisa lançar Threads.

```
public class ServicoIntent extends IntentService {  
  
    public ServicoIntent() {  
        super("Serviço");  
    }  
  
    @Override  
    protected void onHandleIntent(Intent intent) {  
        Log.d("TAG", "FOI!");  
    }  
}
```





<http://developer.android.com/guide/components/services.html>

android.app.IntentService

Não faça um while(true)!

Se precisar disso, use android.app.Service

```
public class ServicoIntent extends IntentService {  
  
    public ServicoIntent() {  
        super("Serviço");  
    }  
  
    @Override  
    protected void onHandleIntent(Intent intent) {  
        Log.d("TAG", "FOI!");  
    }  
}
```





<http://developer.android.com/guide/components/services.html>

android.app.IntentService

Não lance outras Threads...

...a partir de um IntentService!

```
public class ServicoIntent extends IntentService {  
  
    public ServicoIntent() {  
        super("Serviço");  
    }  
  
    @Override  
    protected void onHandleIntent(Intent intent) {  
        Log.d("TAG", "FOI!");  
    }  
}
```





Prática 2

IntentService

Criar um IntentService para executar uma tarefa



<http://developer.android.com/guide/components/services.html>

android.app.IntentService

É muito usado com AlarmManager...

...para agendar uma Tarefa para ser executada em algum momento no futuro



Prática 3

IntentService

Lançar uma IntentService com um AlarmManager



<http://developer.android.com/guide/components/services.html>

android.app.IntentService

Também para fazer Upload...

...de grande quantidade de informações



<http://developer.android.com/guide/components/services.html>

android.app.Service

Podemos criar serviços...

...que se comunicam com sua aplicação através de um "Bind"



<http://developer.android.com/guide/components/services.html>

android.app.Service

Este tipo de serviço vive enquanto...

...enquanto tiver "binds". Morre quando acaba o último bind.



<http://developer.android.com/guide/components/services.html>

android.app.Service

Um serviço pode viver indefinidamente...

...e também receber binds. Se foi iniciando com Bind onStartCommand() não é executado!



Prática 4

Serviço com Bind

Criar um Serviço que se comunica através de um IBinder



<http://developer.android.com/guide/components/services.html>

android.app.Service

A comunicação com o Serviço também...

...pode ser feita usando Messengers. Até mesmo interprocessos!



Prática 5

Serviços com Messenger

Criar um Serviço que se comunica através de um Messenger