

ca#231;a n#237;quel da sorte

<p>Desenvolvendo um aplicativo Android com servi#231;os de plano de fundo
</p>
<p>Neste artigo, voc#234; aprender#225; a criar um aplicativo Android co
m servi#231;os 2 , É de plano de fundo, fornecendo assim funcionalidades avan#2

31;adas e manuten#231;#227;o de energia eficiente.</p>
<p>1. Crie um novo projeto Android</p>
<p>Abra o Android 2 , É Studio e crie um novo projeto. Nomeie-o e selecione
uma localiza#231;#227;o para salvar o projeto.</p>
<p>2. Adicione as depend#234;ncias necess#225;rias</p>
<p>Em seu 2 , É arquivo</p>
<p>build.gradle</p>
<p>, adicione:</p>
<p>dependencies {</p>
<p> implementation '#39;com.birbit:android-background-task:0.5.0'#39;</p>

</p>
<p> }</p>
<p>3. Crie uma classe de trabalhador</p>
<p>Crie uma nova classe e estenda-a #224; classe</p>
<p>android.app.job.JobService</p>
<p>. 2 , É Implemente o m#233;todo</p>
<p>onStartJob()</p>
<p>:</p>
<p>public boolean onStartJob(JobParameters params) {</p>
<p> // Implementar l#243;gica que executa em plano de fundo</p>
<p> return false;</p>
<p> }</p>
<p>4. 2 , É Scheduling trabalho de fundo</p>
<p>Acione a execu#231;#227;o do trabalho usando o c#243;digo a seguir:</p>

</p>
<p>JobScheduler scheduler = (JobScheduler) getSystemService(Context.JOB_SC) Tj T* B

<p>scheduler.schedule(new JobInfo.Builder(YOUR_JOB_ID, new ComponentName(t) Tj T*

<p>5. Teste o aplicativo</p>
<p>Insira seu dispositivo Android no "Modo 2 , É de desenvolvedor"
</p>
<p>e conecte-o ao computador.</p>
<p>Execute o aplicativo e as tarefas ser#227;o realizadas em segundo plan

o. Observe o status e 2 , É a sa#237;da na janela</p>
<p>Run</p>
<p>.</p>
<p>Configurando o administrador do dispositivo Android no seu telefone</p>

</p>
<p>Localize e habilite c#243;digos de bloqueio da tela em 2 , É seu dispos

itivo e siga os passos abaixo:</p>
<p>Abra</p>
<p>Configura#231;#245;es</p>
<p>. Navegue at#233; a guia</p>
<p>Privacidade ou Seguran#231;a</p>