# Um pipeline de fluxo de dados simples (Python)

## visão global

Neste laboratório, você abrirá um projeto do Dataflow, usará a filtragem de pipeline e executará o pipeline localmente e na nuvem.

* Projeto Open Dataflow
* Filtragem de pipeline
* Execute o pipeline localmente e na nuvem

## Objetivo

Neste laboratório, você aprende como escrever um pipeline simples do Dataflow e executá-lo localmente e na nuvem.

* Configurar um projeto do Python Dataflow usando o Apache Beam
* Escreva um pipeline simples em Python
* Execute a consulta na máquina local
* Execute a consulta na nuvem

## Configuração

#### O que você precisará

Para concluir este laboratório, você precisará de:

* Acesso a um navegador de internet padrão (navegador Chrome recomendado).
* Tempo. Observe o tempo de **conclusão** do laboratório no Qwiklabs. Essa é uma estimativa do tempo necessário para concluir todas as etapas. Planeje sua programação para que você tenha tempo de concluir o laboratório. Depois de iniciar o laboratório, você não poderá pausar e retornar mais tarde (começará na etapa 1 toda vez que iniciar um laboratório).
* O tempo de **acesso** do laboratório é o tempo que seus recursos de laboratório estarão disponíveis. Se você concluir seu laboratório com o tempo de acesso ainda disponível, poderá explorar o Google Cloud Platform ou trabalhar em qualquer seção do laboratório marcada "se tiver tempo". Quando o tempo de acesso acabar, seu laboratório terminará e todos os recursos serão encerrados.
* Você **NÃO** precisa de uma conta ou projeto do Google Cloud Platform. Uma conta, projeto e recursos associados são fornecidos a você como parte deste laboratório.
* Se você já tiver sua própria conta do GCP, não a use para este laboratório.
* Se o seu laboratório solicitar que você faça login no console, **use apenas a conta do aluno fornecida pelo laboratório** . Isso impede que você incorra em cobranças por atividades de laboratório em sua conta pessoal do GCP.

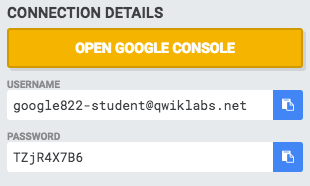
#### Comece seu laboratório

Quando estiver pronto, clique em **Iniciar Laboratório** . Você pode acompanhar o progresso do seu laboratório com a barra de status na parte superior da tela.

**Importante** O que está acontecendo durante esse tempo? Seu laboratório está gerando recursos do GCP para você nos bastidores, incluindo uma conta, um projeto, recursos dentro do projeto e permissão para você controlar os recursos necessários para executar o laboratório. Isso significa que, em vez de gastar tempo manualmente configurando um projeto e construindo recursos do zero como parte de seu laboratório, você pode começar a aprender mais rapidamente.

#### Encontre o nome de usuário e a senha do GCP do seu laboratório

Para acessar os recursos e o console deste laboratório, localize o painel Detalhes da Conexão no Qwiklabs. Aqui você encontrará o ID da conta e a senha da conta que você usará para fazer login no Google Cloud Platform:



Se o seu laboratório fornecer outros identificadores de recursos ou informações relacionadas à conexão, ele também aparecerá nesse painel.

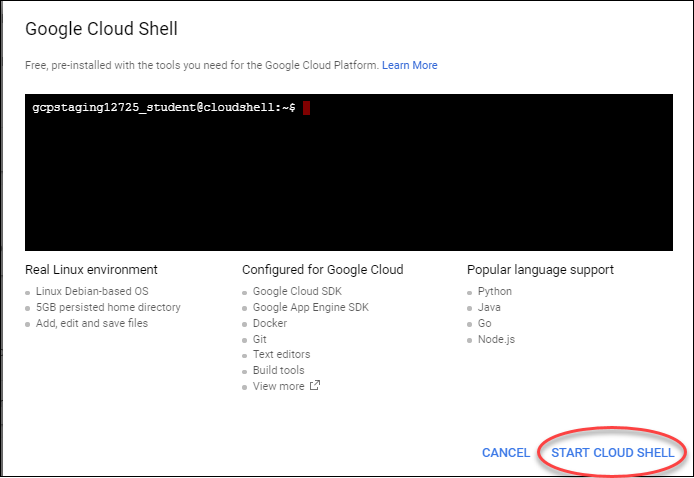
### Ativar o Google Cloud Shell

O Google Cloud Shell fornece acesso via linha de comando aos seus recursos do GCP.

No console do GCP, clique no ícone do **Cloud Shell** na barra de ferramentas superior direita:

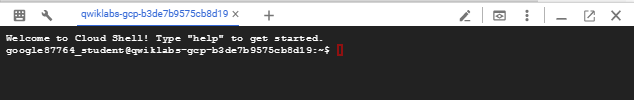


Em seguida, clique em **START CLOUD SHELL** :



Você pode clicar em **START CLOUD SHELL** imediatamente quando a caixa de diálogo aparecer, em vez de esperar na caixa de diálogo até as disposições do Cloud Shell.

Demora alguns momentos para provisionar e se conecta ao ambiente:



O Cloud Shell é uma máquina virtual carregada com todas as ferramentas de desenvolvimento de que você precisa. Ele oferece um diretório pessoal de 5 GB persistente e é executado no Google Cloud, aprimorando muito o desempenho e a autenticação da rede.

Uma vez conectado ao shell da nuvem, você verá que já está autenticado e o projeto está definido como seu PROJECT\_ID :

gcloud auth list

Saída:

Credentialed accounts:

- <myaccount>@<mydomain>.com (active)

**Observação:** gcloud é a ferramenta de linha de comando poderosa e unificada do Google Cloud Platform. A documentação completa está disponível no [Google Cloud gcloud Overview](https://cloud.google.com/sdk/gcloud) . Ele vem pré-instalado no Cloud Shell e suporta a conclusão de guias.

gcloud config list project

Saída:

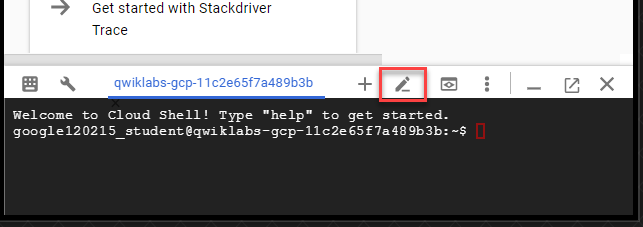
[core]

project = <PROJECT\_ID>

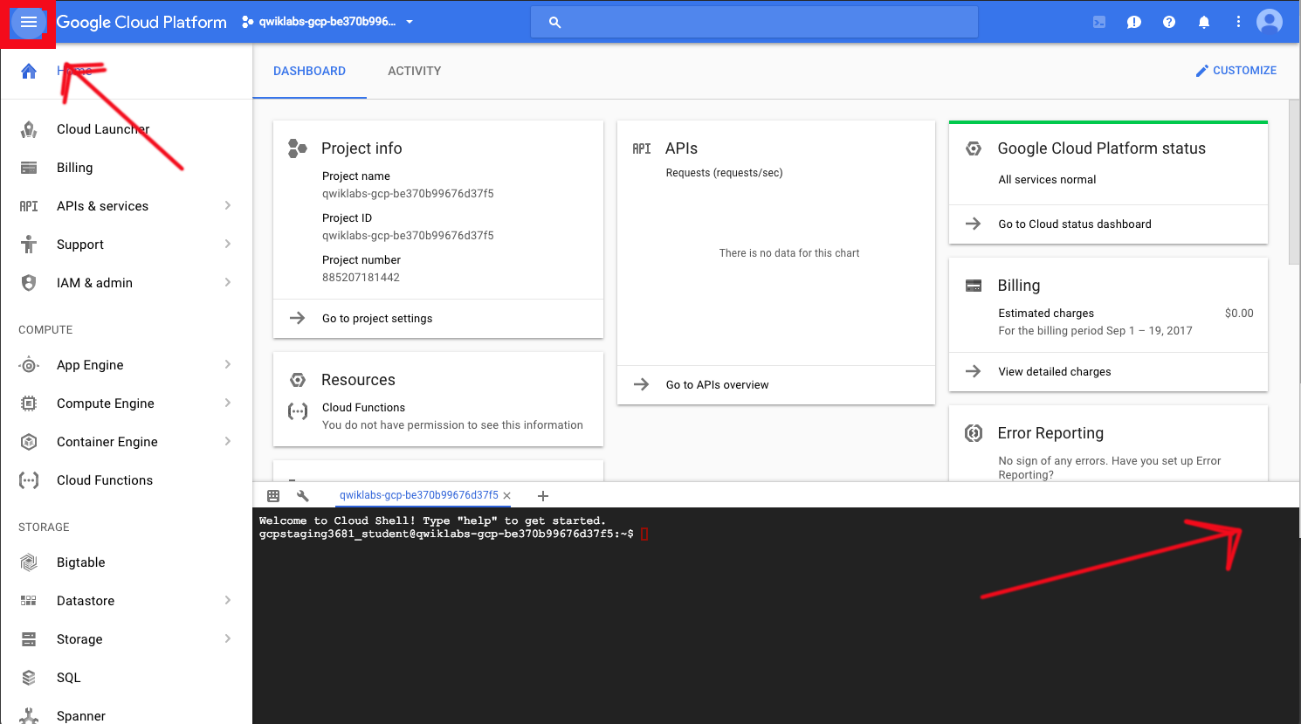
### Inicie o Editor de códigos do Google Cloud Shell

Use o Editor de códigos do Google Cloud Shell para criar e editar facilmente diretórios e arquivos na instância do Cloud Shell.

Depois de ativar o Google Cloud Shell, clique no botão **Iniciar editor de códigos**(parece um lápis) para abrir o Cloud Shell Code Editor.

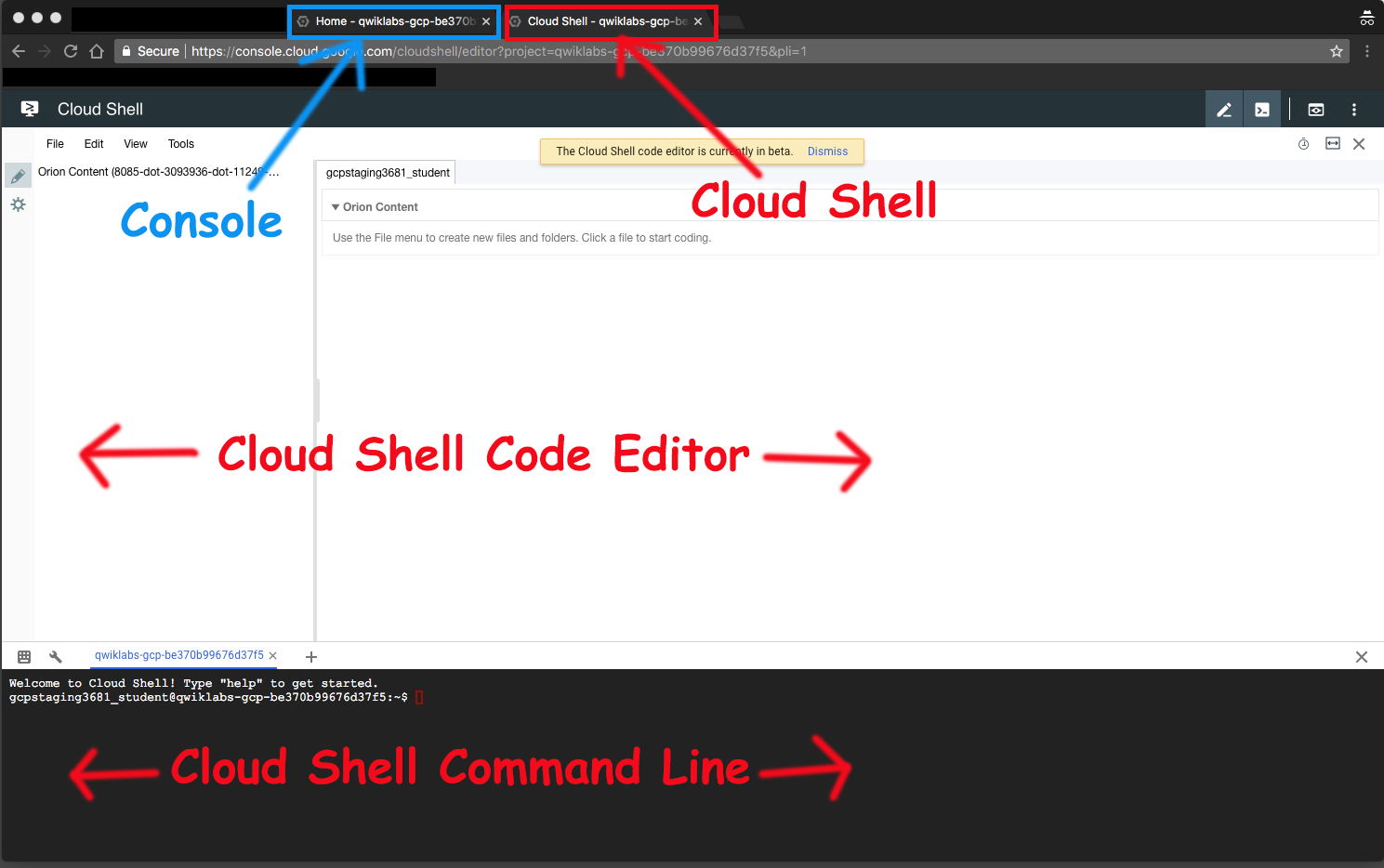


**Nota** : O botão **Launch code editor** pode estar fora da tela à direita. Pode ser necessário clicar no botão do **menu Navegação** para fechar o menu para ver os botões.



Agora você tem três interfaces disponíveis:

* O editor de código do Cloud Shell
* A linha de comando do Cloud Shell
* Console (clicando na guia). Você pode alternar entre o console e o Cloud Shell clicando na guia.



## Tarefa 1. Preparação

Para este laboratório, você precisará dos arquivos de análise de dados de treinamento e de um intervalo do Cloud Storage.

### Verifique se os arquivos do repositório estão no Cloud Shell Editor

1. Clone o repositório na linha de comando do Cloud Shell:

git clone https://github.com/GoogleCloudPlatform/training-data-analyst

1. Clique em **Arquivo> Atualizar** no painel esquerdo do navegador. Você deve ver o diretório de **análise de dados de treinamento** .

### Verifique se você tem um intervalo do Cloud Storage

Se você não tiver um intervalo, siga estas instruções para criar um intervalo.

1. No console, no **menu Navegação** ( 7a91d354499ac9f1.png), clique em **Início** .
2. **Selecione e copie** o ID do projeto. Para simplificar, você usará o ID do projeto Qwiklabs, que já é globalmente exclusivo, como o nome do intervalo.
3. No console, no **menu Navegação** ( 7a91d354499ac9f1.png), clique em **Armazenamento** > **Navegador** .
4. Clique em **Criar Balde** .
5. Especifique o seguinte e deixe as configurações restantes como seus padrões:

|  |  |
| --- | --- |
| **Propriedade** | **Valor**  (digite o valor ou selecione a opção conforme especificado) |
| **Nome** | <your unique bucket name (Project ID)> |
| **Classe de armazenamento padrão** | Multi-Regional |
| **Localização** | <Your location> |

1. Clique em **Criar** .
2. Registre o nome do seu balde. Você precisará disso nas tarefas subseqüentes.
3. No Cloud Shell, digite o seguinte para criar uma variável de ambiente chamada "BUCKET" e verifique se existe com o comando echo.

BUCKET="<your unique bucket name (Project ID)>"

echo $BUCKET

Você pode usar o $ BUCKET nos comandos do Cloud Shell. E se você precisar inserir o nome do intervalo <your-bucket> em um campo de texto no Console, poderá recuperar rapidamente o nome com "echo $ BUCKET".

### Verifique se a API do Dataflow está ativada para este projeto

1. Volte para a guia do navegador para o Console. Na barra de pesquisa superior, insira a **API do Dataflow** . Isso levará você à página, **menu Navegação> APIs e serviços> Painel de controle> API de fluxo de dados** . Ele mostrará uma informação de status ou lhe dará a opção de **ativar** a API.
2. Se necessário, **ative** a API.

## Tarefa 2. Projeto Open Dataflow

O objetivo deste laboratório é familiarizar-se com a estrutura de um projeto do Dataflow e aprender como executar um pipeline do Dataflow. Você precisará atualizar alguns arquivos para instalar o Apache Beam. O Apache Beam é uma plataforma de código aberto para executar fluxos de trabalho de processamento de dados.

1. Volte para a guia do navegador que contém o Cloud Shell. No Cloud Shell, navegue até o diretório deste laboratório:

cd ~/training-data-analyst/courses/data\_analysis/lab2/python

1. Instale as dependências necessárias para o fluxo de dados do Python:

sudo ./install\_packages.sh

1. Verifique se você tem a versão correta do pip. (Deve ser> 8.0):

pip -V

Caso contrário, abra uma nova guia do Cloud Shell e ela deverá selecionar a versão atualizada do pip.

1. Use **File> Refresh** no editor do Cloud Shell para visualizar a cópia local do repositório.

Se a qualquer momento durante os laboratórios do DataFlow você estiver desconectado do Cloud Shell devido à inatividade, quando você fizer o login, os elementos de memória do Apache Beam serão perdidos. Portanto, você precisará reemitir esses comandos antes de prosseguir:

cd ~/training-data-analyst/courses/data\_analysis/lab2/python

sudo ./install\_packages.sh

## Tarefa 3. Filtragem de Pipeline

1. No editor de códigos do Cloud Shell, navegue até o diretório /training-data-analyst/courses/data\_analysis/lab2/pythone visualize o arquivo grep.py. **Não faça alterações no código.**

Alternativamente, você poderia ver o arquivo com nano. **Não faça alterações no código.**

cd ~/training-data-analyst/courses/data\_analysis/lab2/python

nano grep.py

Você pode responder a estas perguntas sobre o arquivo grep.py?

* Quais arquivos estão sendo lidos?
* Qual é o termo de pesquisa?
* Para onde vai a saída?

Existem três transformações no pipeline:

* O que a transformação faz?
* O que a segunda transformação faz?
* De onde vem a sua entrada?
* O que isso faz com essa entrada?
* O que ele escreve em sua saída?
* Para onde vai a saída?
* O que a terceira transformação faz?

## Tarefa 4. Executar o pipeline localmente

1. Na linha de comando do Cloud Shell, execute localmente grep.py.

cd ~/training-data-analyst/courses/data\_analysis/lab2/python

python grep.py

Nota: se você vir um erro que diz "No handlers could be found for logger "oauth2client.contrib.multistore\_file", você pode ignorá-lo. O erro é simplesmente dizer que o log da biblioteca oauth2 irá para stderr.

1. O arquivo de saída será output.txt. Se a saída é grande o suficiente, será sharded em partes separadas com nomes como: output-00000-of-00001. Se necessário, você pode localizar o arquivo correto examinando a hora do arquivo.

ls -al /tmp

1. Examine o arquivo de saída. Substitua "- \*" abaixo pelo sufixo apropriado.

cat /tmp/output-\*

A saída parece lógica?

## Tarefa 5. Executar o pipeline na nuvem

1. Copie alguns arquivos Java para a nuvem.

gsutil cp ../javahelp/src/main/java/com/google/cloud/training/dataanalyst/javahelp/\*.java gs://$BUCKET/javahelp

1. Edite o pipeline do Dataflow em grepc.py. No editor de códigos do Cloud Shell, navegue até o diretório /training-data-analyst/courses/data\_analysis/lab2/pythone edite o arquivo grepc.py.
2. Substitua PROJECT e BUCKET pelo nome do seu projeto e do bucket. Aqui estão maneiras fáceis de recuperar os valores:

echo $DEVSHELL\_PROJECT\_ID

echo $BUCKET

Exemplo de strings antes:

PROJECT='cloud-training-demos'

BUCKET='cloud-training-demos'

Exemplo de strings após edição (use seus valores):

PROJECT='qwiklabs-gcp-your-value'

BUCKET='qwiklabs-gcp-your-value'

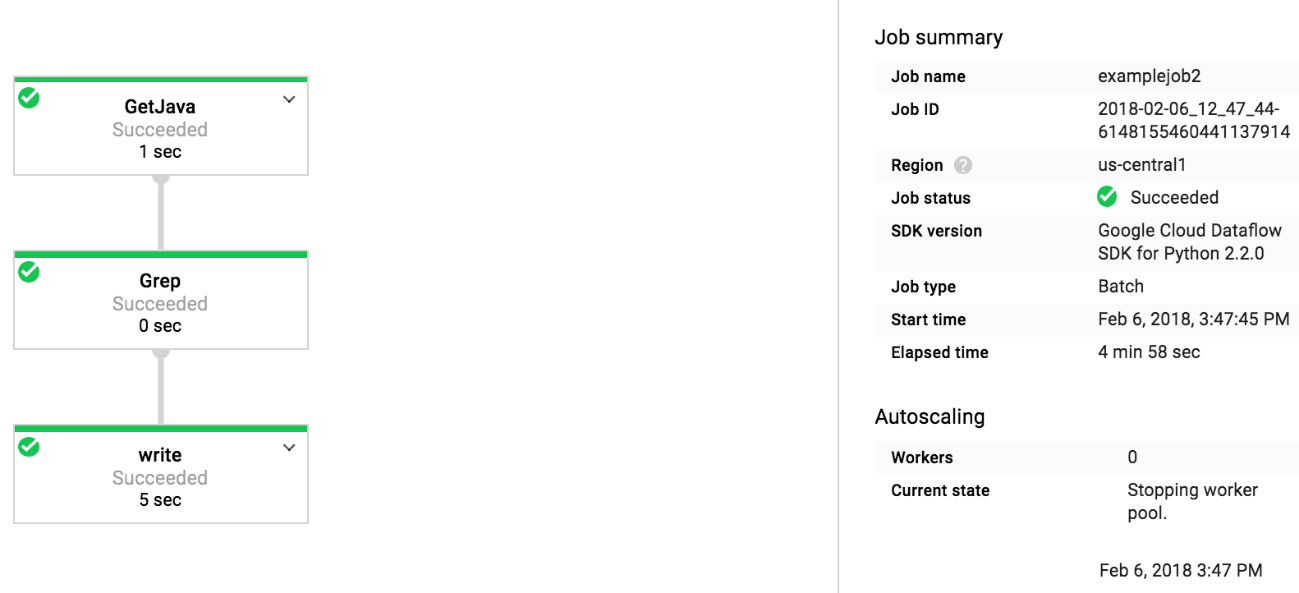
1. Envie o trabalho do Dataflow para a nuvem:

python grepc.py

Como esse é um trabalho pequeno, a execução na nuvem levará muito mais tempo do que executá-lo localmente (da ordem de 2 a 3 minutos).

1. Volte para a guia do navegador para o Console. No **menu Navegação** ( 7a91d354499ac9f1.png), clique em **Fluxo de dados** e clique em seu trabalho para monitorar o progresso.

Exemplo:



1. Aguarde o status do trabalho para transformar em êxito. Nesse momento, seu Cloud Shell exibirá um prompt de linha de comando.
2. Examine a saída no intervalo do Cloud Storage. No **menu Navegação** ( 7a91d354499ac9f1.png), clique em **Armazenamento> Navegador** e clique no seu balde. Clique no diretório **javahelp** . Este trabalho irá gerar o arquivo output.txt. Se o arquivo é grande o suficiente que será sharded em várias partes com nomes como: output-0000x-of-000y. Você pode identificar o arquivo mais recente por nome ou pelo campo **Última modificação** . Clique no arquivo para visualizá-lo.

Como alternativa, você pode fazer o download do arquivo no Cloud Shell e visualizá-lo:

gsutil cp gs://$BUCKET/javahelp/output\* .

cat output\*

## Termine seu laboratório

Quando você tiver concluído seu laboratório, clique em **Finalizar Laboratório** . O Qwiklabs remove os recursos que você usou e limpa a conta para você.

Você terá a oportunidade de avaliar a experiência do laboratório. Selecione o número de estrelas aplicável, digite um comentário e clique em **Enviar** .

O número de estrelas indica o seguinte:

* 1 estrela = muito insatisfeito
* 2 estrelas = insatisfeito
* 3 estrelas = neutra
* 4 estrelas = Satisfeito
* 5 estrelas = muito satisfeito

Você pode fechar a caixa de diálogo se não quiser fornecer feedback.

Para comentários, sugestões ou correções, use a guia **Suporte** .

Data da última atualização: 2018-11-21

Data do último teste: 2018-11-21

© 2018 Google LLC Todos os direitos reservados. Google e o logotipo do Google são marcas registradas do Google LLC. Todos os outros nomes de empresas e produtos podem ser marcas registradas das respectivas empresas com as quais estão associados.