

MATERIAL DE APOIO

Google Cloud Skill Boost

**Laboratório 01 do Intermediate:
Desenvolver um app com o Gemini**

Sumário

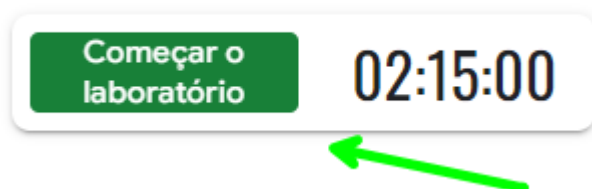
1.	Login no Console do Google Cloud	3
2.	Tarefa 1	5
3.	Tarefa 2	6
4.	Tarefa 3	10

1. Login no Console do Google Cloud

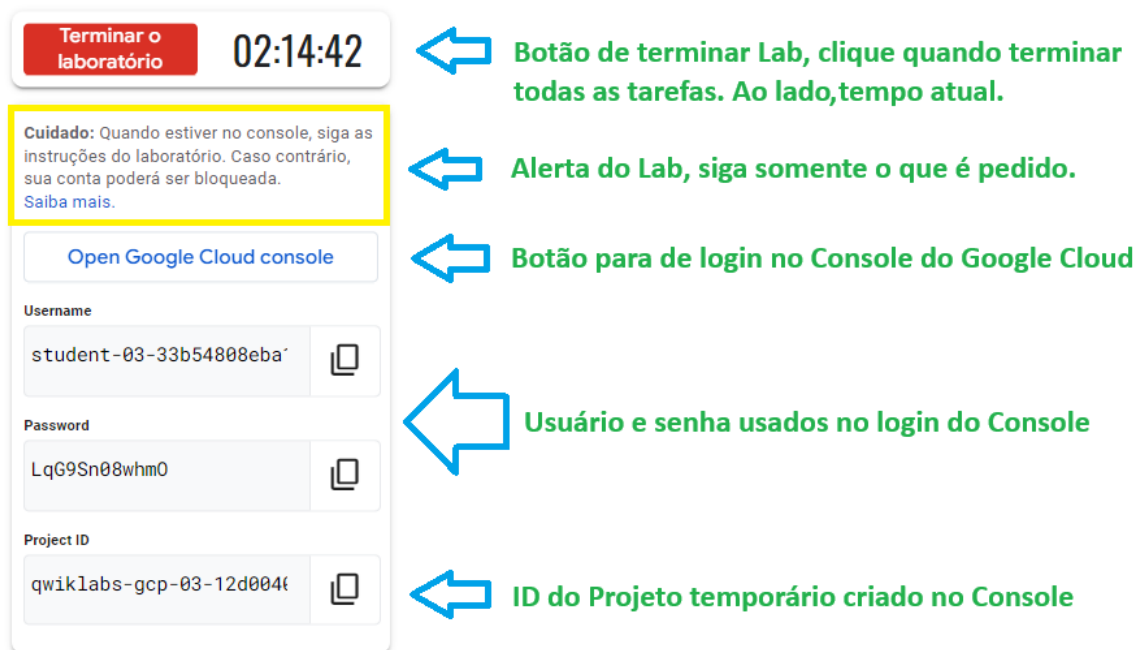
Link para o laboratório:

[https://www.cloudskillsboost.google/paths/236/course_templates/978/labs/488168?locale=pt_B
R](https://www.cloudskillsboost.google/paths/236/course_templates/978/labs/488168?locale=pt_BR)

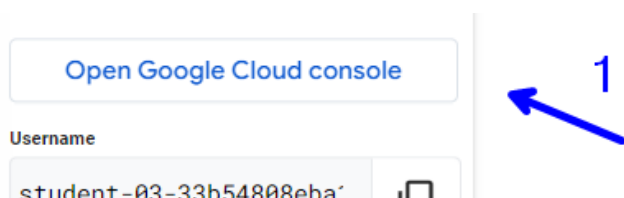
- 1) Primeiro passo é fazer login no Console do Google Cloud, clique no botão verde “Começar o laboratório”:

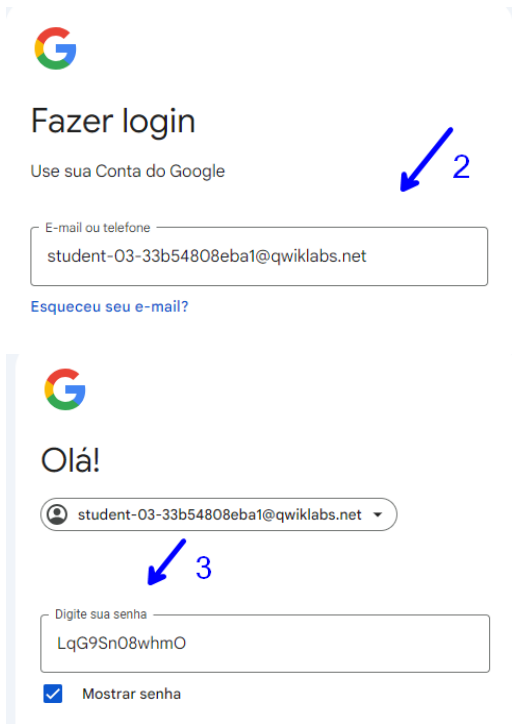


- 2) Após, irá abrir o Painel de login no Console:



- 3) Clique no botão “Open Google Cloud Console” (1) com o botão direito do mouse e escolha para abrir em uma aba Anônima/InPrivate, depois preencha o usuário (2) e senha (3) na página de login:





Fazer login

Use sua Conta do Google

E-mail ou telefone

student-03-33b54808eba1@qwiklabs.net

[Esqueceu seu e-mail?](#)

Olá!

student-03-33b54808eba1@qwiklabs.net

Digite sua senha

LqG9Sn08whmO

☒ Mostrar senha

4) Aceite todos os termos e condições do Google Cloud (1-2):

em myaccount.google.com.

Seu uso dos Serviços do Google com esta conta também é regido por políticas internas da sua organização.

Entendi

Google Cloud

Welcome student fe2a879d!

Create and manage your Google Cloud instances, disks, networks, and other resources in one place.



student fe2a879d

student-03-33b54808eba1@qwiklabs.net

[SWITCH ACCOUNT](#)

Country

Brazil

Terms of Service


☒ I agree to the [Google Cloud Platform Terms of Service](#), and the terms of service of [any applicable services and APIs](#).

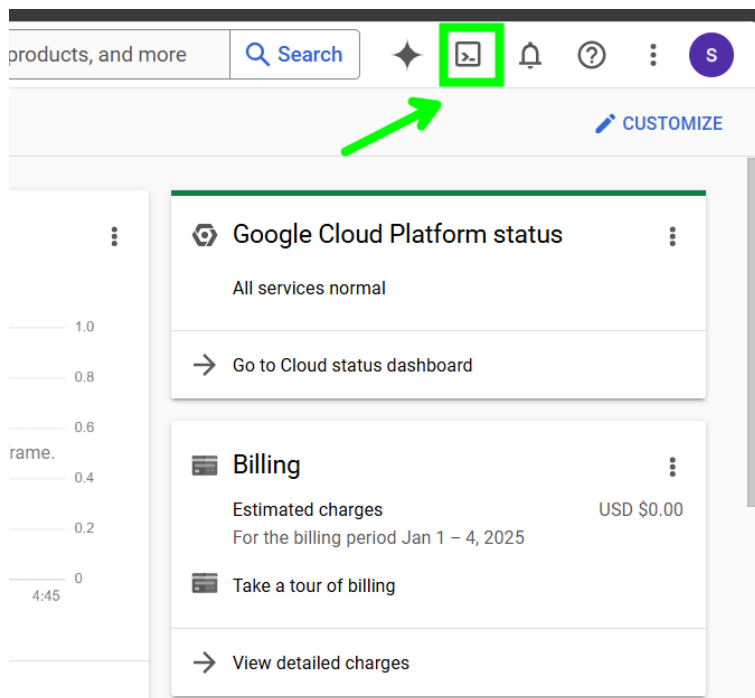
Email updates

☐ I would like to receive periodic emails on news, product updates and special offers from Google Cloud and Google Cloud Partners.

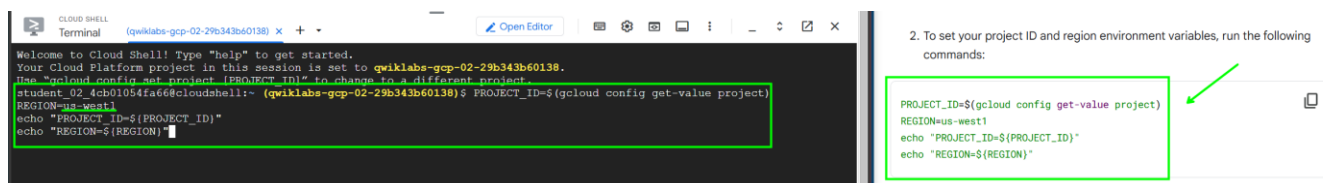
AGREE AND CONTINUE

2. Tarefa 1

1) Habilite o **Cloud Shell** no ícone do canto superior direito :



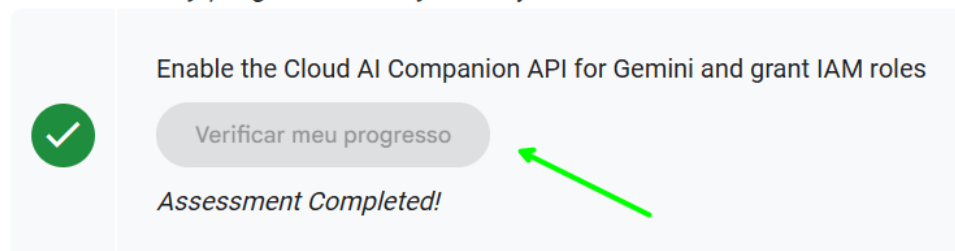
2) Quando terminal abrir, execute **todos os comandos** (Um por vez), indicados na **TASK 1**, do laboratório. Como no exemplo a seguir:



OBS: Para as próximas Tasks, utilize sempre a região que aparece no primeiro comando.

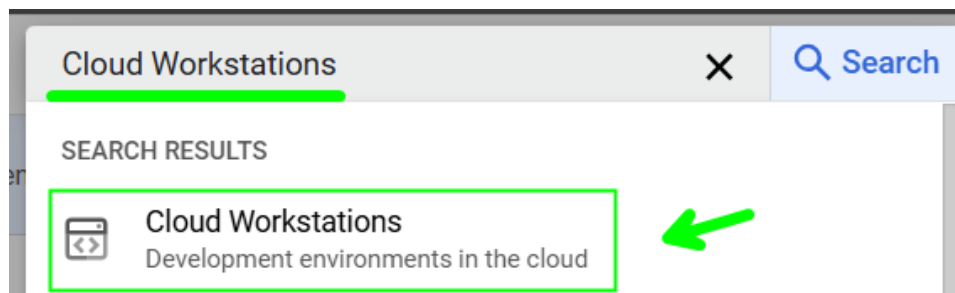
3) Depois, clique no botão do progresso:

Click *Check my progress* to verify the objective.

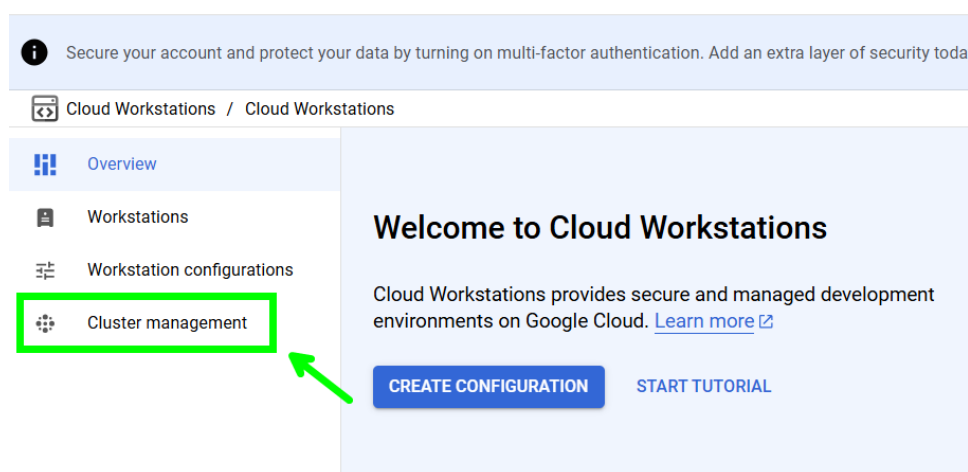


3. Tarefa 2

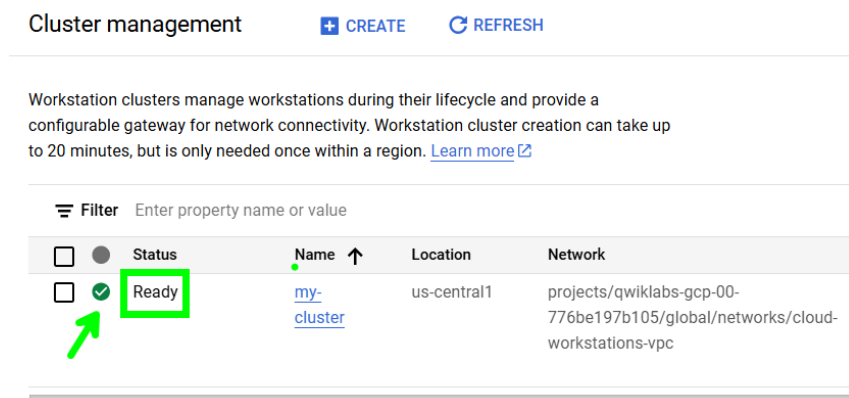
- 1) Pesquise no campo de pesquisa acima da página, por **Cloud Workstations**:



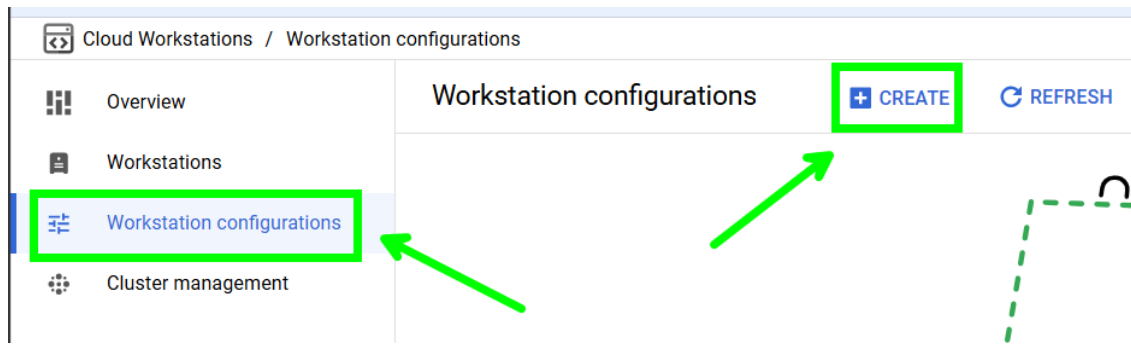
- 2) A seguir, clique em **Cluster Management**:



- 3) Quando Verifique o status do cluster. Se o status do cluster for Reconciling ou Updating, atualize periodicamente e aguarde até que ele mude para Ready antes de seguir para a próxima etapa:



- 4) Após clique em **Workstation configurations**, e depois no botão **CREATE**:



- 5) Na tela seguinte, preencha os dados com as informações com os dados que seu laboratório indica, deixe todos o restante como padrão, e por fim clique em **CREATE** no pé da página:

Propriedade	Valor (digite ou selecione)
Nome	my-config
Cluster de estações de trabalho	my-cluster
Estações de trabalho de início rápido	Desativada

Basic information

i Get AI-powered collaboration with Gemini CodeAssist in Cloud Workstations. [Learn more](#).

Name *
my-config

Workstation cluster *
my-cluster (us-central1)

Quick start workstations ?

☐ Enabled (faster workstation startup)

☒ Disabled (lower cost)

i Disabling this feature will result in lower costs, but workstations can take significantly (several minutes) longer to start.

CREATE CANCEL

- 6) Clique em **REFRESH**, e verifique o Status da configuração que está sendo criada. Se o status for Reconciling ou Updating, atualize periodicamente e aguarde até que ele mude para Ready antes de seguir para a próxima etapa:

Workstation configurations [+ CREATE](#) [REFRESH](#)

Workstation configurations act as templates for workstations. Create a workstation configuration first, and then create workstations. Add creators to a configuration to allow them to create their own workstation. [Learn more](#)

Filter Enter property name or value

<input type="checkbox"/>	Status	Name ↑		Location	Workstation cluster
<input checked="" type="checkbox"/>	Ready	my-config	ADD USERS	us-central1	my-cluster

7) Em seguida, clique em **Workstations**, e depois em **CREATE WORKSTATION**:

Cloud Workstations / My workstations

Overview **Workstations** Workstation configurations Cluster management

Workstations [+ CREATE WORKSTATION](#) [REFRESH](#)

Cloud Workstations provides managed, on-demand, development environments in the cloud. They can be accessed through a browser UI, an SSH terminal, or from your local IDE through an SSH bridge. [Learn more](#)

8) Na tela seguinte, preencha os as informações com os dados que seu laboratório indica, deixe todos o restante como padrão, e por fim clique em **CREATE** no pé da página (Como indicado nas figuras abaixo):

Propriedade	Valor (digite ou selecione)
Nome	my-workstation
Nome de exibição	my-workstation
Configuração	my-config

Cloud Workstations / My workstations / Create workstation

Create workstation

Name *
my-workstation

Display Name
my-workstation

Configuration *
my-config

Environment variables

These environment variables will be passed to the container.

[+ ADD VARIABLE](#)

Labels

Labels are applied to the resource and propagated to the underlying Compute Engine


CREATE


CANCEL


9) Na tela seguinte, clique em **START** para iniciar o workstation:


Workstations [+ CREATE WORKSTATION](#) [REFRESH](#)

My workstations


 **my-workstation**

Status  Stopped


ID my-workstation 




Configuration my-config 

[START](#)

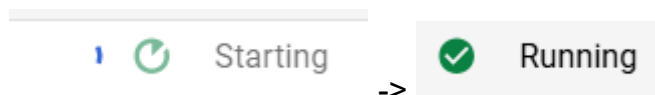


All workstations

 **Filter** Enter property name or value

<input type="checkbox"/>		Status	Name 	Quick action	Location
<input type="checkbox"/>		Stopped	my-workstation	START	us-central1

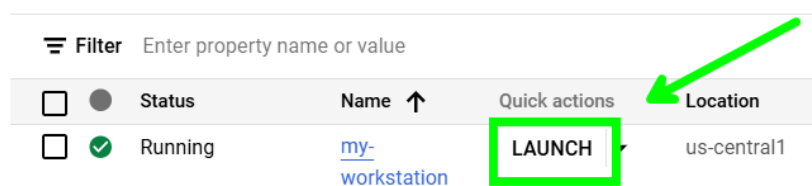
OBS: Durante a inicialização, o status muda para Starting. Espere até que ele mude para Running, o que indica que a estação está pronta para ser usada. Ela pode levar vários minutos para ser inicializada:



10) Siga as orientações do laboratório em **“Iniciar o ambiente de desenvolvimento integrado”**

11) Após clique em **LAUNCH**:

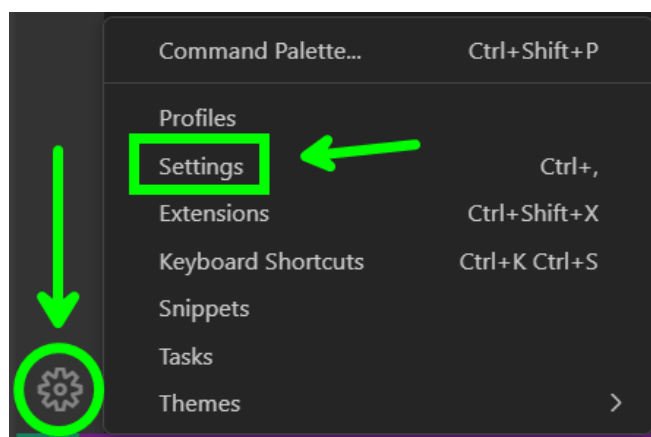
All workstations



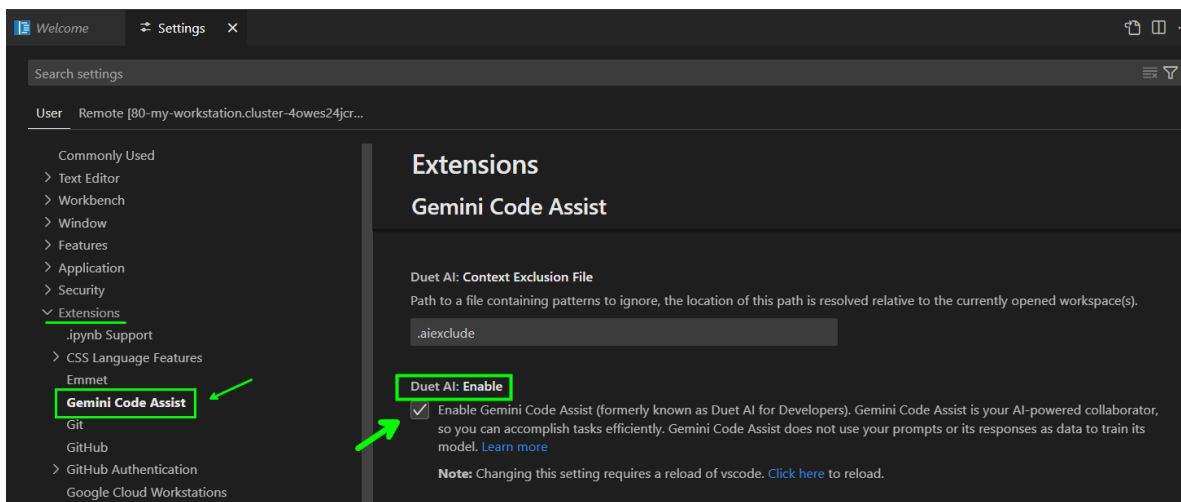
12) Após aberto, clique para ticar a **Task 2** do laboratório.

4. Tarefa 3

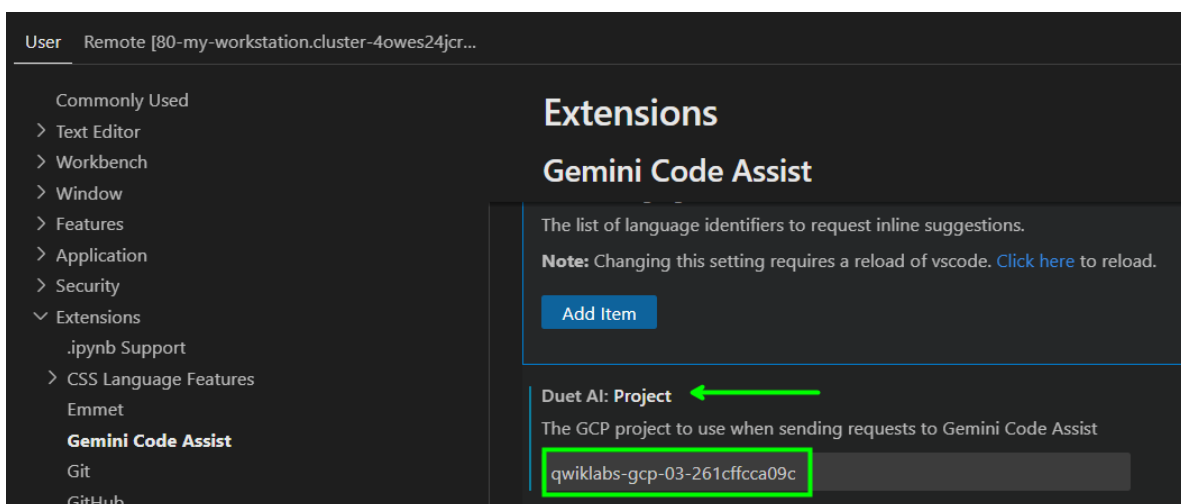
1) Dentro do Workstation, clique na engrenagem que aparece no canto inferior esquerdo, em seguida em **Settings**:



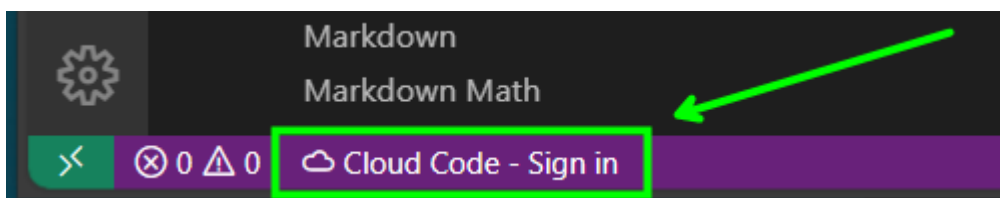
2) Vá em **Extensions > Gemini Code Assist**, e garanta que o **Duet AI** esteja habilitado, através de ticado em “Enable”:



3) Coloque seu **Project ID** em **Duet AI: Project**:



4) No pé da página, no canto inferior esquerdo, embaixo da engrenagem, clique em **Cloud Code – Sign in**:



5) Para iniciar o fluxo de login do Google Cloud, pressione **CONTROL** (para Windows e Linux) ou **COMMAND** (para macOS) e clique no URL no console:

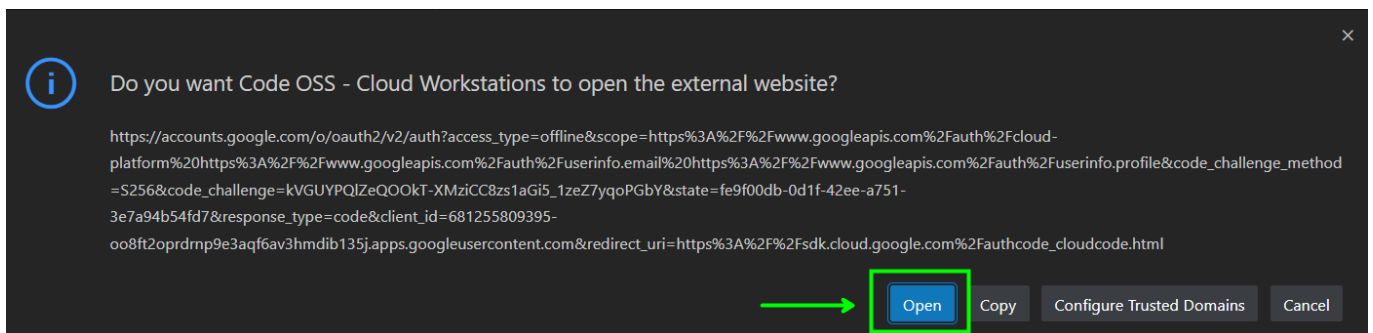
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Go to the following link in your browser:
Follow link (ctrl + click)
https://accounts.google.com/o/oauth2/v2/auth?access_type=offline&scope=https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloud-platform%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fuserinfo.email%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fuserinfo.profile&code_challenge_method=S256&code_challenge=kVGUYPQlZeQ00kT-XMziCC8zs1aGi5_1zeZ7yqoPGbY&state=fe9f00db-0d1f-42ee-a751-3e7a94b54fd7&response_type=code&client_id=681255809395-oo8ft2oprdrnp9e3aqf6av3hmdib135j.apps.googleusercontent.com&redirect_uri=https%3A%2F%2Fsdk.cloud.google.com%2Fauthcode_cloudcode.html

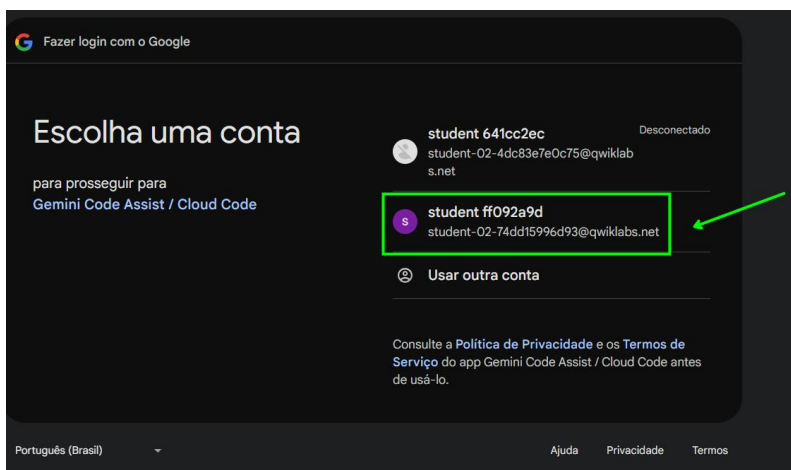
Enter authorization code:

```

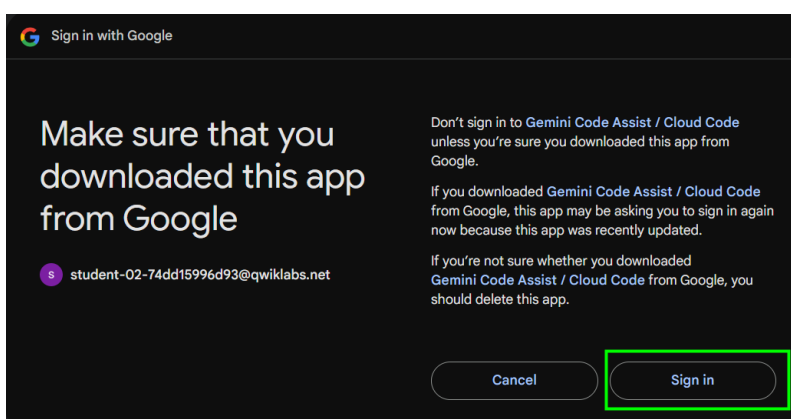
6) Clique em **OPEN** para confirmar a abertura do site externo:



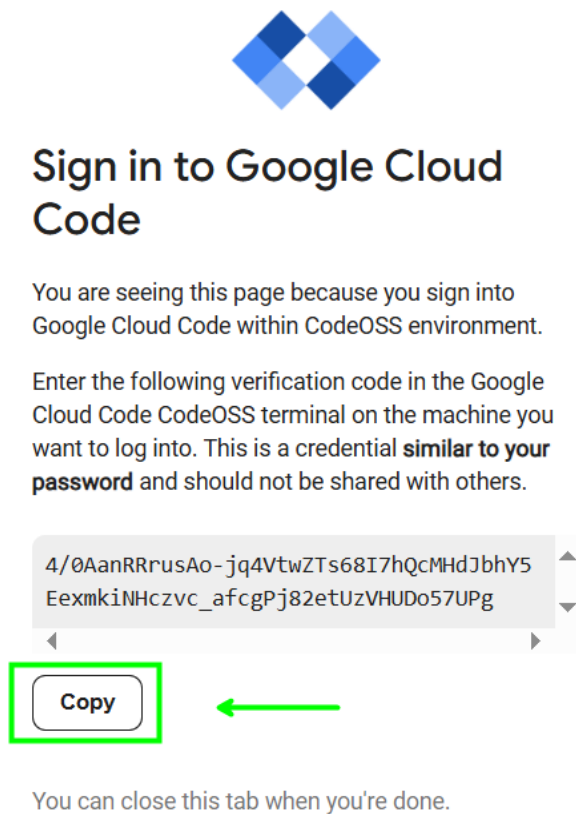
7) Clique no endereço de e-mail de estudante:



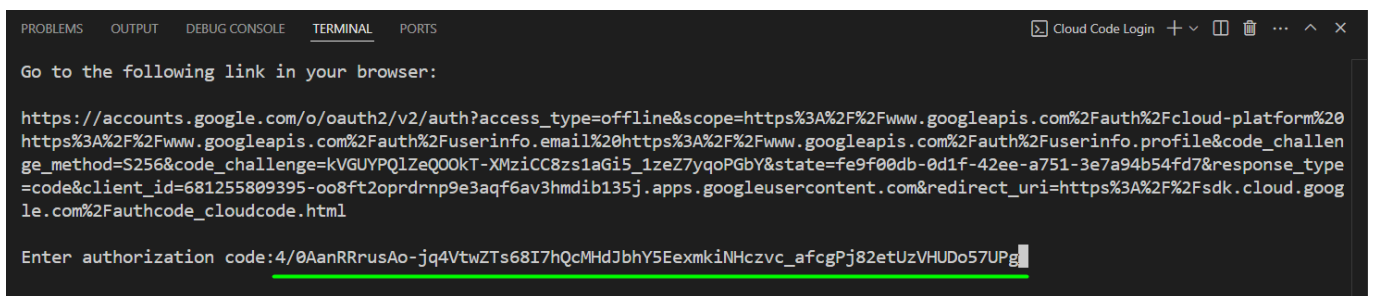
8) Na tela seguinte clique em **SIGN IN**:



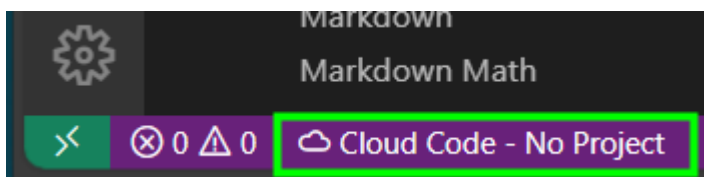
9) Copie o código de autenticação:



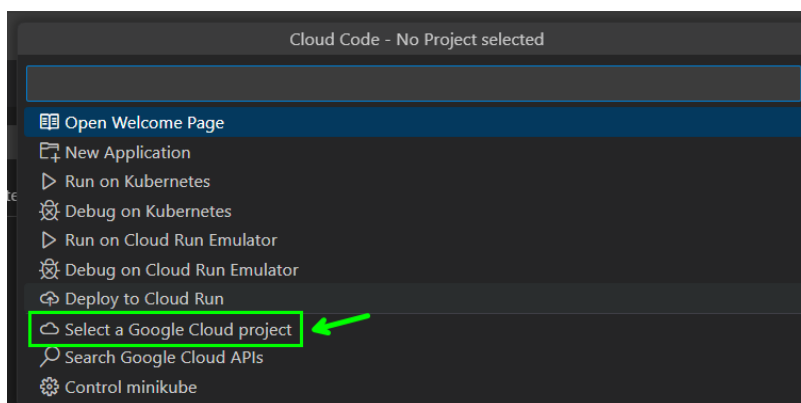
10) E cole no terminal, como indicado na imagem, e em seguida aperte **ENTER**:



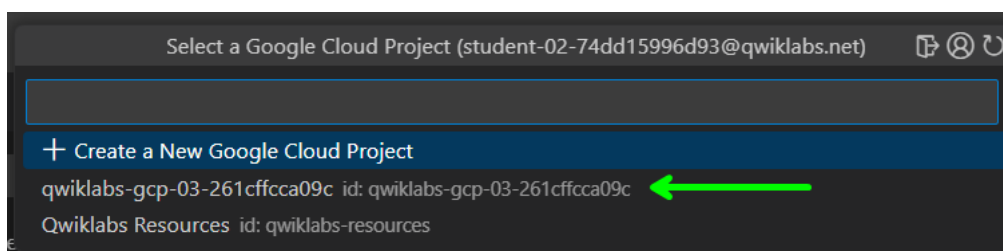
11) Clique novamente no **Cloud Code**, porém agora verá que o botão mudou para **No Project**:



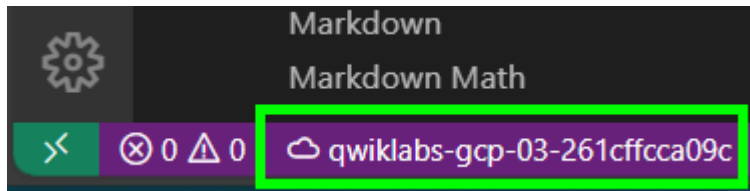
12) Clique em **Select a Google Cloud Project**, como mostra na imagem:



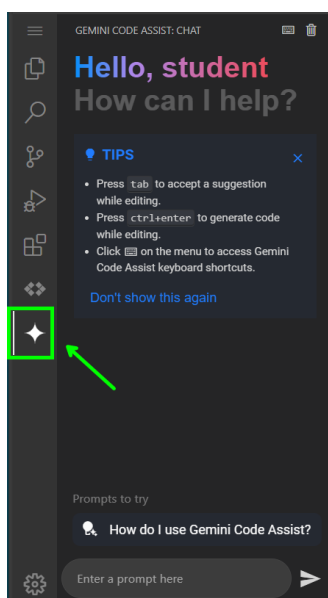
13) Selecione seu Project:



14) Pronto, agora verá uma indicação no pé da página que seu Project foi selecionado, e que você está autenticado:

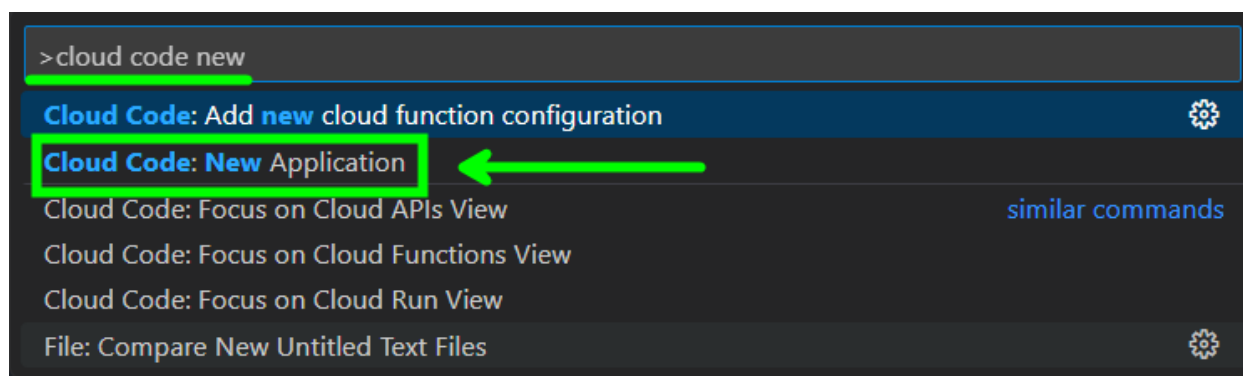


15) Clique no símbolo do Gemini no menu lateral esquerdo, e verifique que está tudo ok antes de continuar:

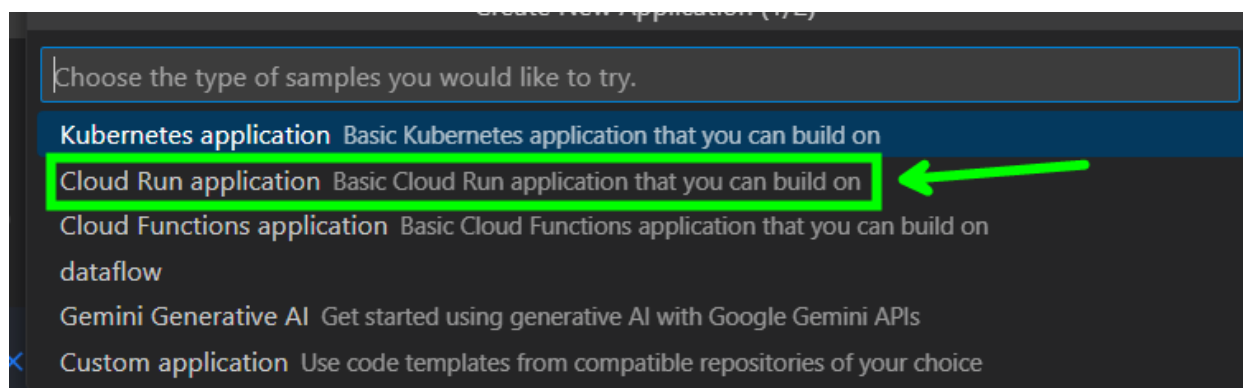


OBS: Se algo estiver diferente, pare imediatamente e siga as alternativas sugeridas pela página do laboratório.

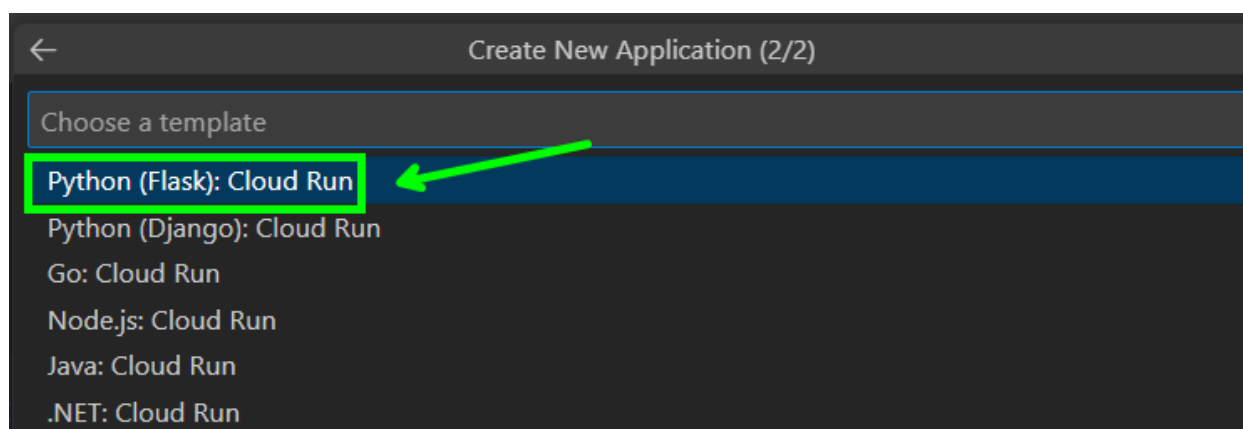
16) Vá para **File > Command Palette**, e comece a digitar **Cloud Code New**, e selecione **Cloud Code: New Application**:



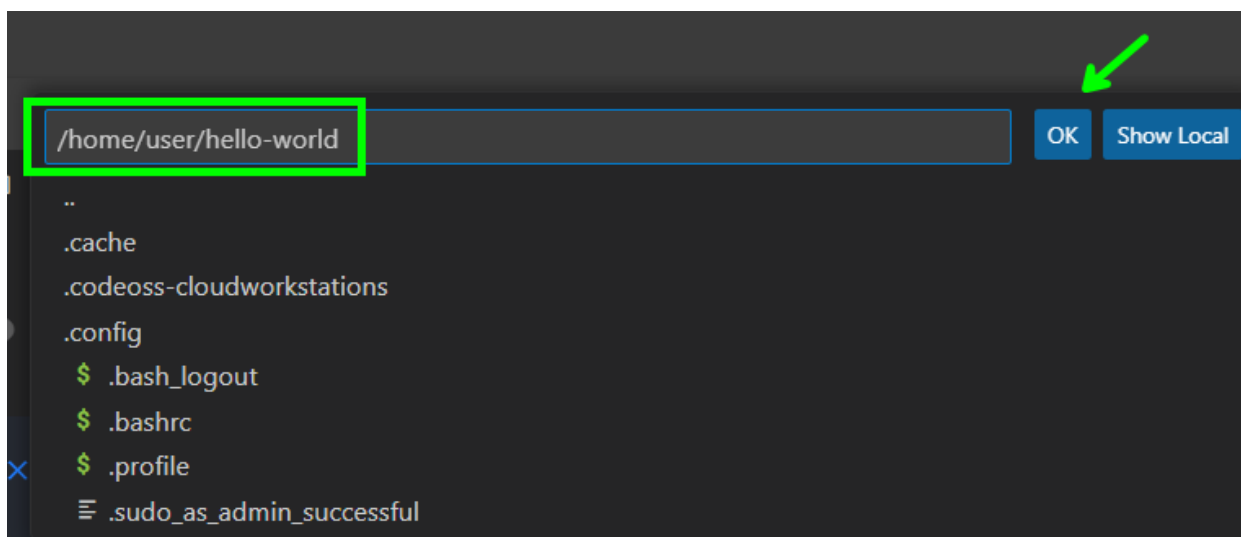
17) Em seguida, seleciona **Cloud Run Application**:



18) E em seguida **Python (Flask): Cloud Run**:

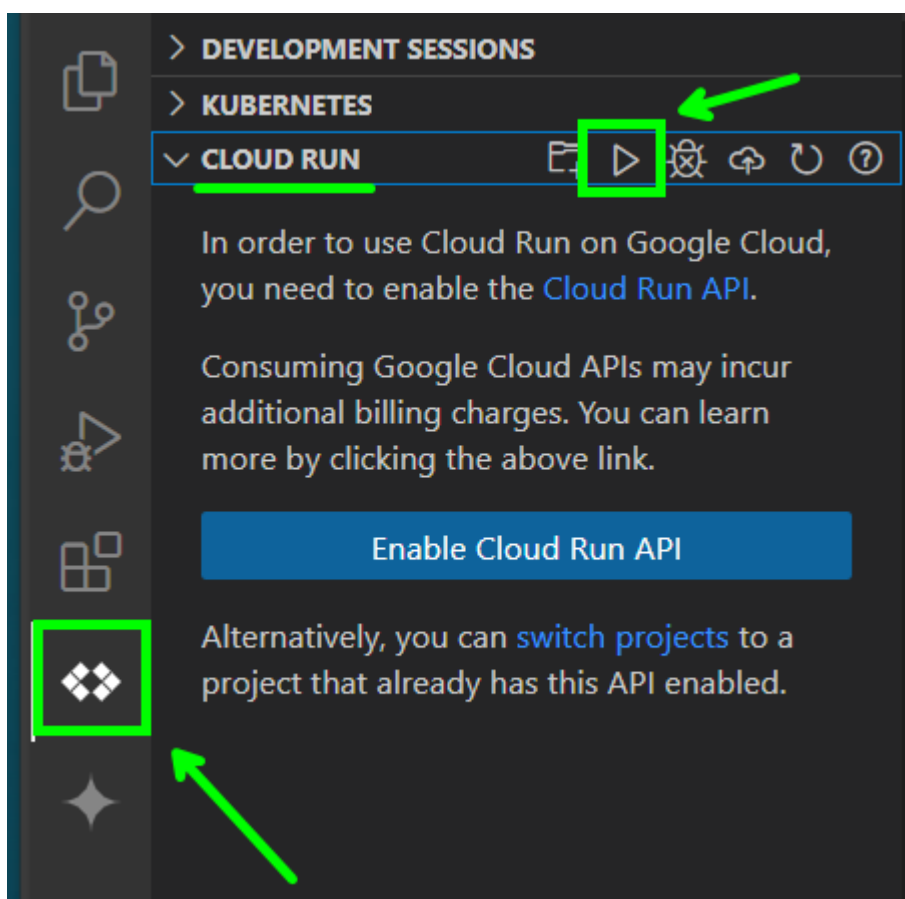


19) Atualize o nome do aplicativo e a pasta de nível superior para `/home/user/hello-world` e, em seguida, clique em Ok.

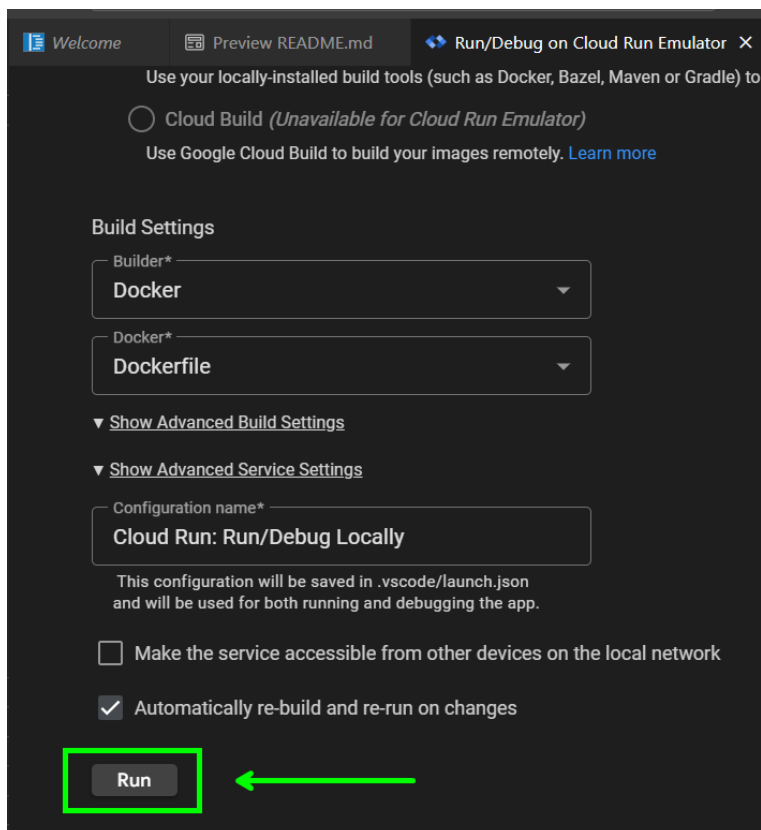


20) Interaja com o Gemini segundo as orientações do laboratório.

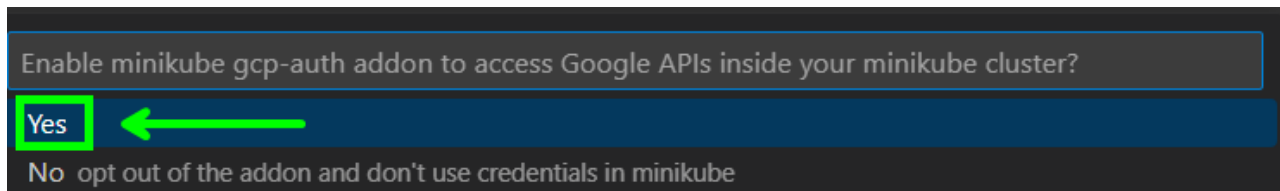
21) No Menu lateral esquerdo, clique no símbolo de **Cloud Code**, e na opção do **Cloud Run**, clique no símbolo ▶ **Run App on Local Cloud Run Emulator**:



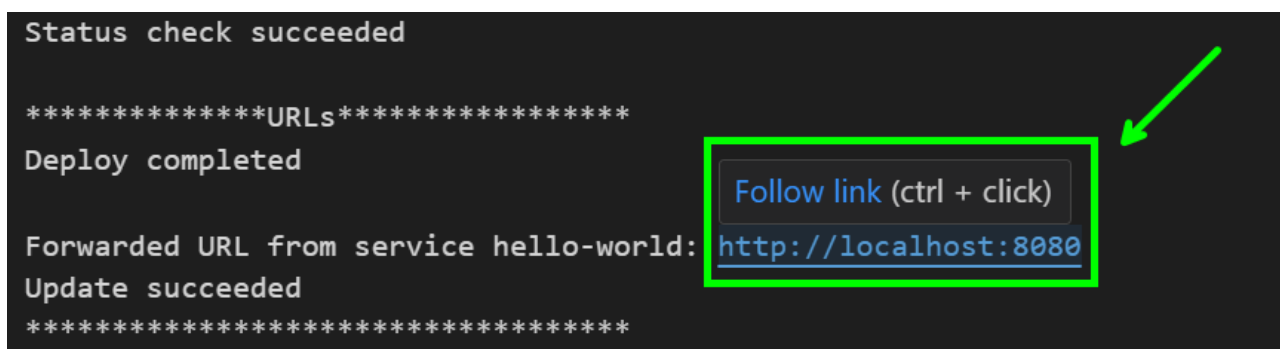
22) Na nova tela que se abre, clique em **RUN**:



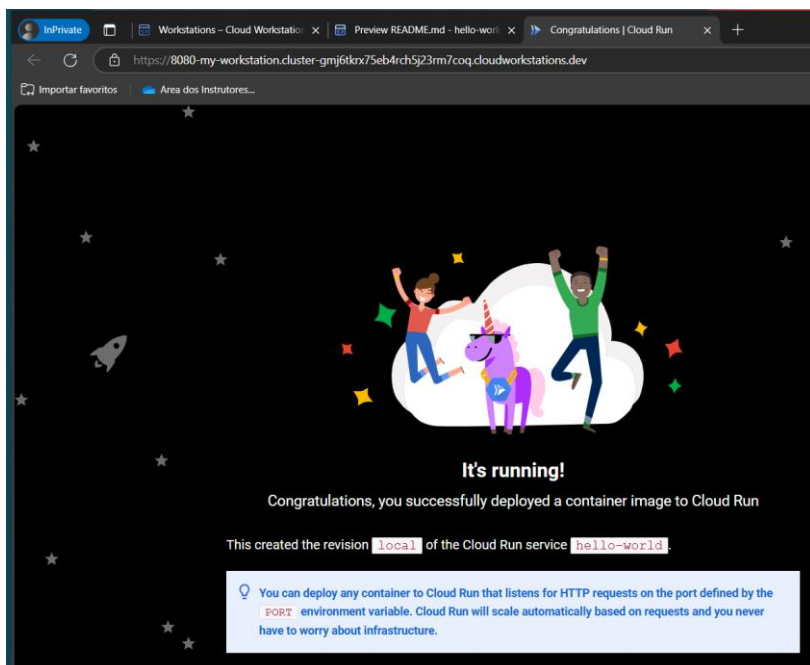
23) Em seguida, clique em **YES** para habilitar o Minikube:



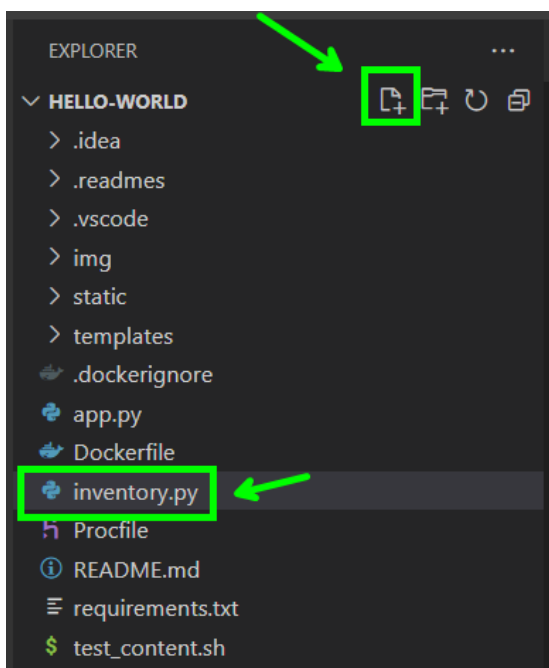
24) Talvez seja necessário se logar novamente, repita o procedimento de login como anteriormente. Após, espere a finalização da implantação local. Clique com **CTRL+Botão Esquerdo do Mouse** para abrir o link da URL:



25) Verifique que o serviço esteja rodando corretamente:



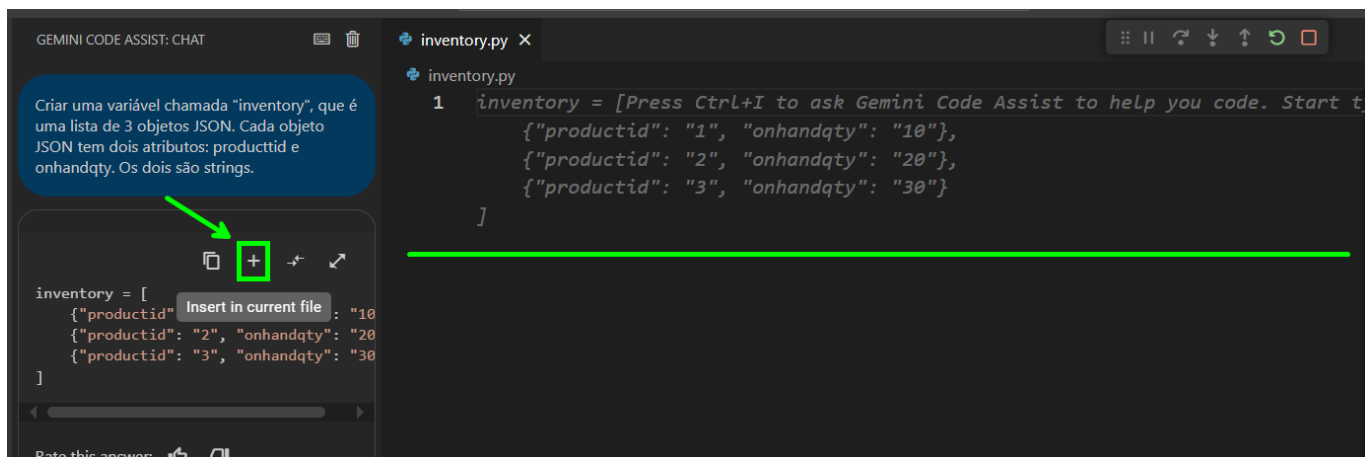
26) Novamente no **Explorer**, clique em **New File**, e crie um arquivo chamado **inventory.py**:



27) Pergunte ao Gemini o seguinte prompt:

Criar uma variável chamada "inventory", que é uma lista de 3 objetos JSON. Cada objeto JSON tem dois atributos: productid e onhandqty. Os dois são strings.

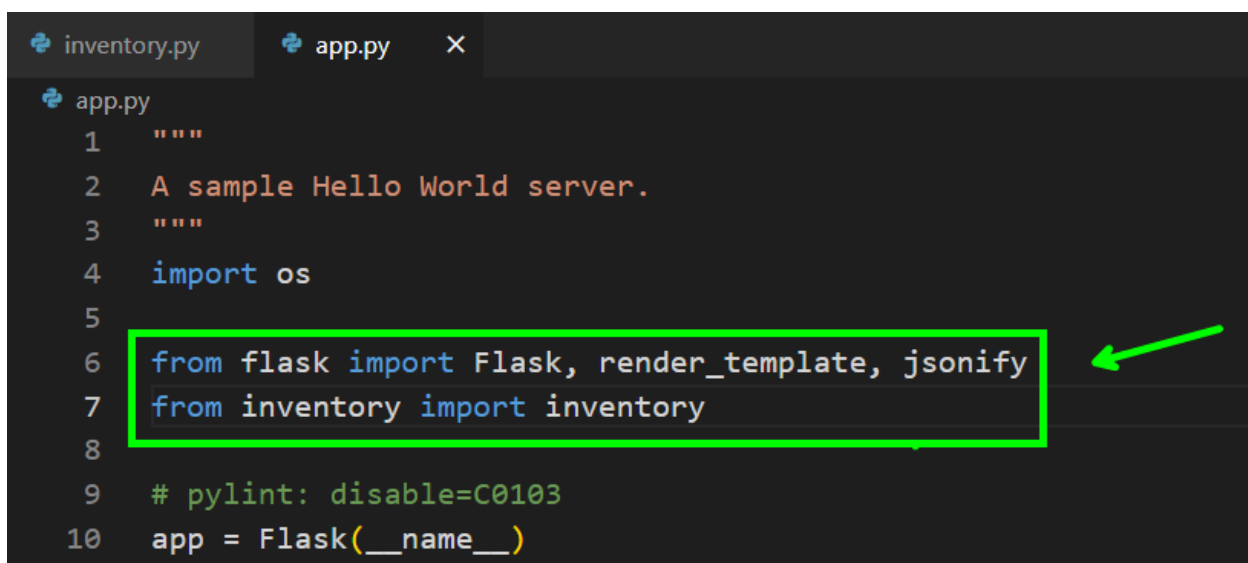
28) Na resposta do Gemini, clique no símbolo de + para incluir o código no arquivo que você criou:



29) Para salvar o arquivo `inventory.py` na pasta `home/user/hello-world`, no menu do ambiente de desenvolvimento integrado (Menu principal do Code OSS), clique em **Arquivo > Salvar**, ou **CRTL+S**.

30) Abra o arquivo `app.py`, altere a importação adicionando as bibliotecas propostas pelo laboratório, ficando como na figura:

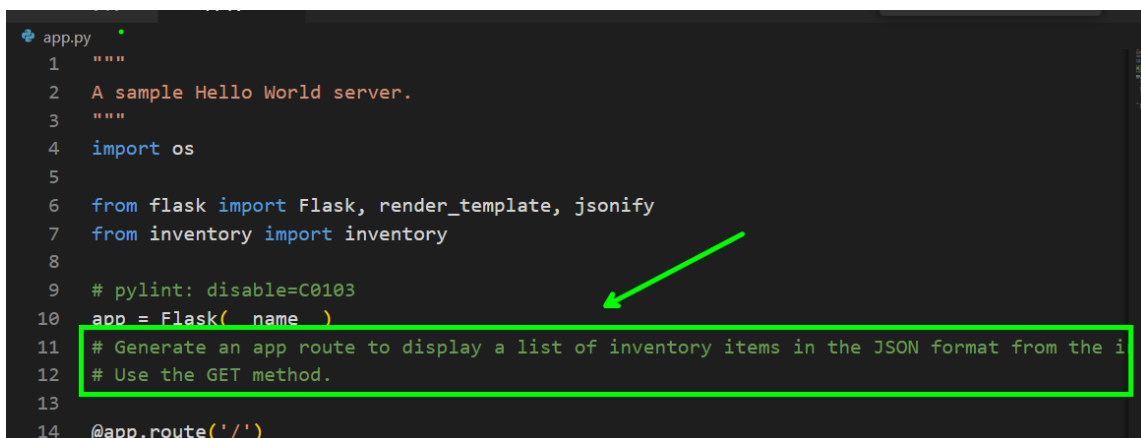
```
from flask import Flask, render_template, jsonify
from inventory import inventory
```



31) Embaixo de `app = Flask(__name__)`, coloque o comentário proposto pelo laboratório, ficando como na figura:

Generate an app route to display a list of inventory items in the JSON format from the inventory.py file.

Use the GET method.

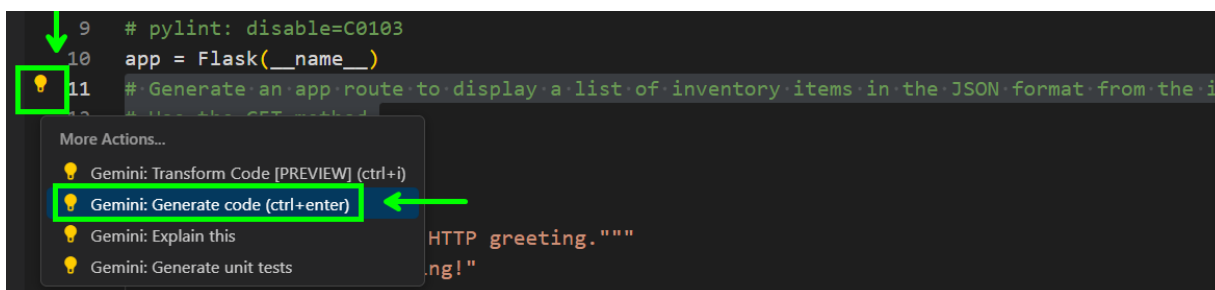


```

1  """
2  A sample Hello World server.
3  """
4  import os
5
6  from flask import Flask, render_template, jsonify
7  from inventory import inventory
8
9  # pylint: disable=C0103
10 app = Flask(__name__)
11 # Generate an app route to display a list of inventory items in the JSON format from the i
12 # Use the GET method.
13
14 @app.route('/')

```

32) Selecione as duas linhas novas. Clique na lâmpada (💡) e, no menu **More Actions**, selecione **Generate Code**:



```

9  # pylint: disable=C0103
10 app = Flask(__name__)
11 # Generate an app route to display a list of inventory items in the JSON format from the i
12 # Use the GET method.

```

More Actions...

- 💡 Gemini: Transform Code [PREVIEW] (ctrl+i)
- 💡 Gemini: Generate code (ctrl+enter)
- 💡 Gemini: Explain this
- 💡 Gemini: Generate unit tests

33) Clique em **Tab** para aceitar o novo código, ou mantenha o ponteiro sobre uma parte da resposta e clique em **Accept**..:



```

8
9  # pylint: disable=C0103
10 app = Flask(__name__)
11 # Generate an app route to display a list of inventory items in the JSON format from the
12 # Use the GET method.
13 @app.route('/inventory', methods=['GET'])
14 def get_inventory():
15     return jsonify(inventory)
16
17

```

34) Se as instruções app.route e return no código gerado for diferente do código mostrado acima, substitua o snippet de código gerado pelo laboratório. Isso garante que o laboratório funcione como planejado.

35) Adicione algumas linhas abaixo, realize o mesmo processo para a próxima função proposta pelo laboratório:

```
# Generate an App route to get a product from the list of inventory items given the productID.
# Use the GET method.
# If there is an invalid productID, return a 404 error with an error message in the JSON.
```

```
# Generate an App route to get a product from the list of inventory items given the productID.
# Use the GET method.
# If there is an invalid productID, return a 404 error with an error message in the JSON.
@app.route('/inventory/<productid>', methods=['GET'])
def get_product(productid):
    product = [product for product in inventory if product['productid'] == productid]
    if product:
        return jsonify(product[0])
    else:
        return jsonify({'error': 'Product not found'}), 404
```

36) Mantenha o ponteiro sobre uma parte da resposta do código gerado. Para aceitar o código, na barra de ferramentas, clique em **Accept**.

37) Se ele for diferente do código mostrado, substitua o snippet de código gerado pelo snippet do laboratório.

38) Salve o arquivo.

39) Realize o procedimento novamente para implantar o aplicativo localmente. Vá em **Cloud Code > Cloud Run > Run App on Local Cloud Run Emulator**

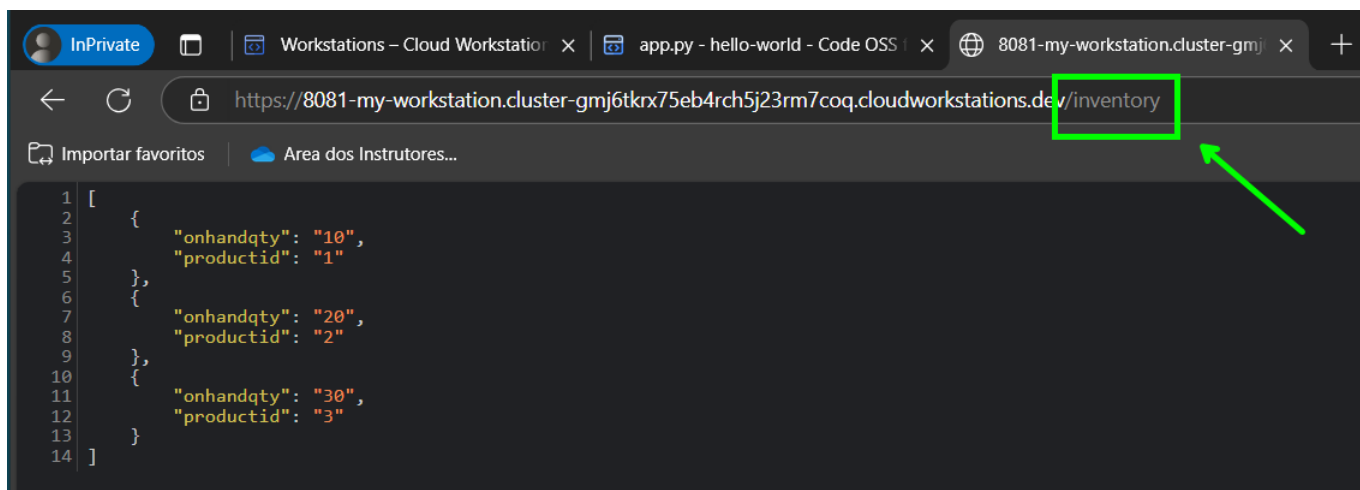
40) Responda **Yes** para habilitar o Minikube.

41) Clique novamente no link:

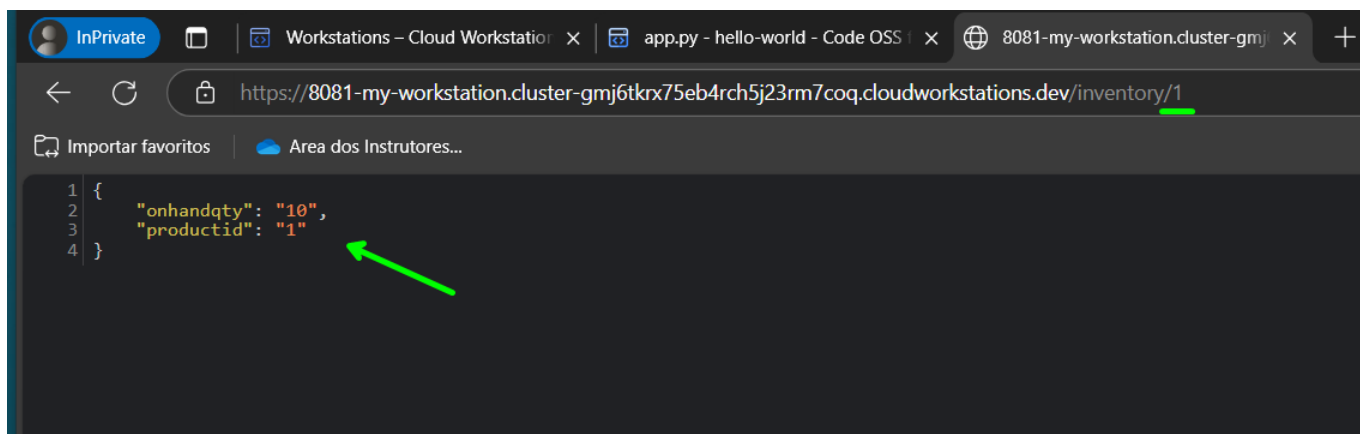
```
*****URLs*****
Deploy completed
Forwarded URL from service hello-world: http://localhost:8081
Update succeeded
*****
```

Follow link (ctrl + click)

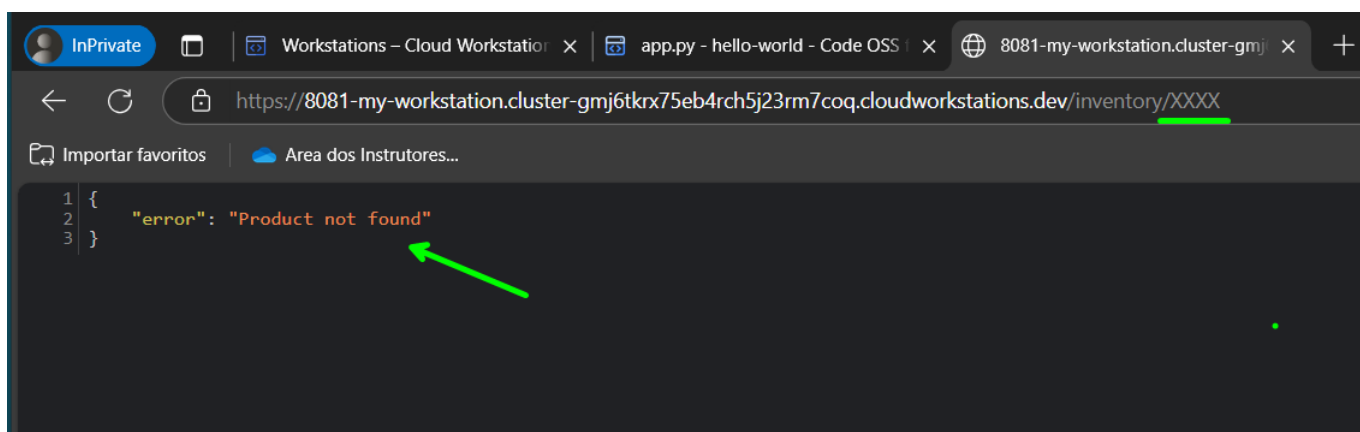
42) Na página seguinte, adicione `/inventory` ao final da URL, agora será possível ver todo seu inventário em formato json:



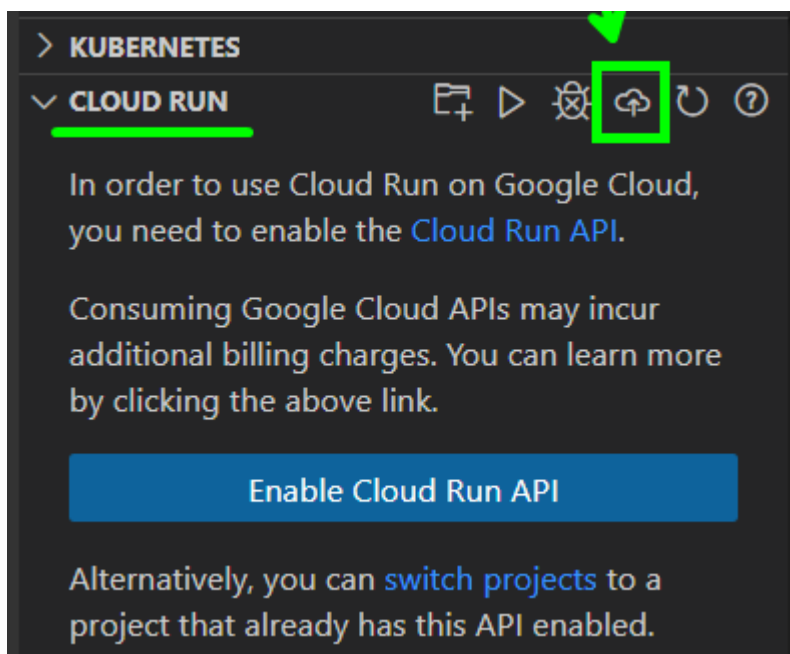
43) Adicione um ID de Produto (Definido no arquivo `inventory.py`) depois de `/inventory`, por exemplo `/inventory/1`. Agora será possível ver somente o produto escolhido:



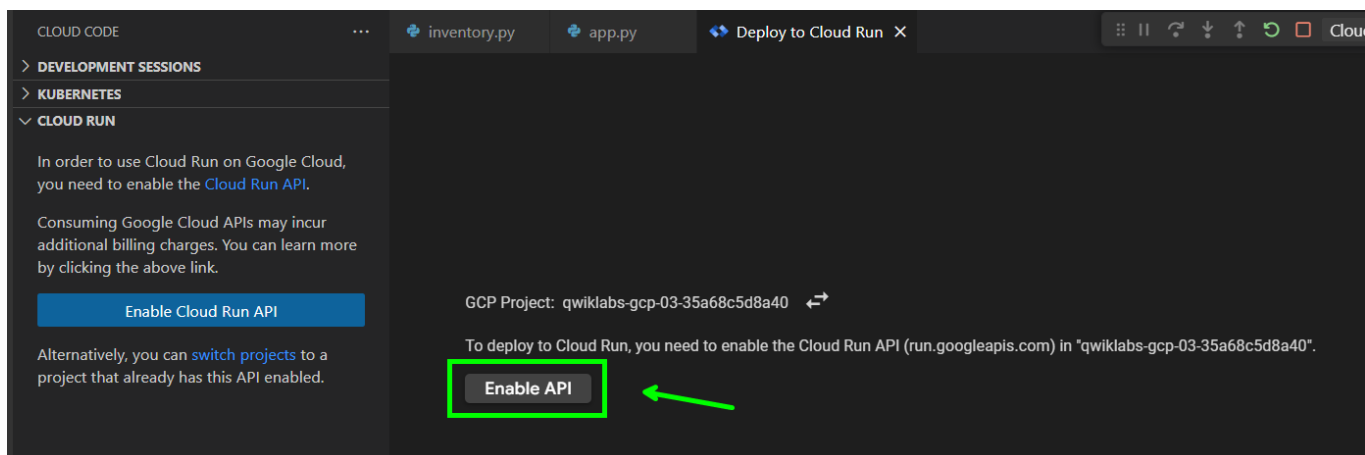
44) Se digitar um produto não válido ao final, por exemplo, `/inventory/XXXX`, agora será possível ver a tratativa de erro que também foi adicionada nessa versão do aplicativo:



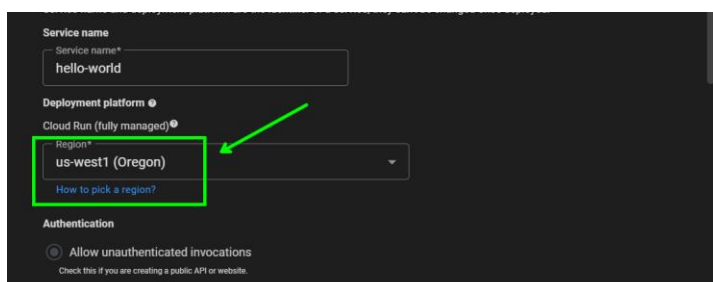
45) Com tudo correto e validado, é hora de implantar efetivamente no **Cloud Run**. Volte no menu do **Cloud Code > Cloud Run**. E selecione o símbolo de uma nuvem com uma seta **Deploy to Cloud Run**:



46) Na tela seguinte, clique em **Enable API** (Se tiver algum erro nessa etapa, clique no botão de Deploy to Cloud Run novamente e tente habilitar a API mais uma vez):

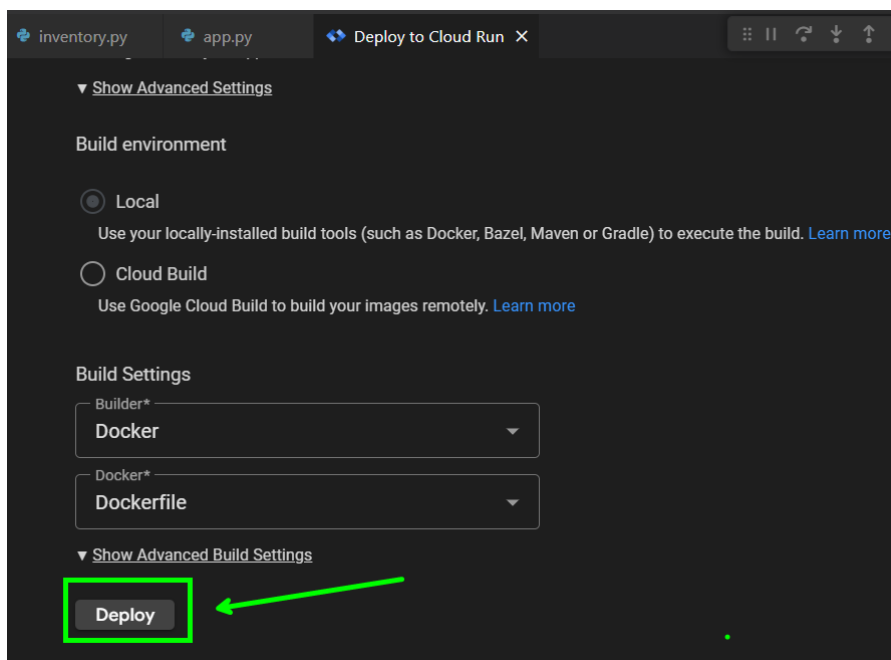


47) Na tela seguinte, troque a região com informação que é dada para você no laboratório:

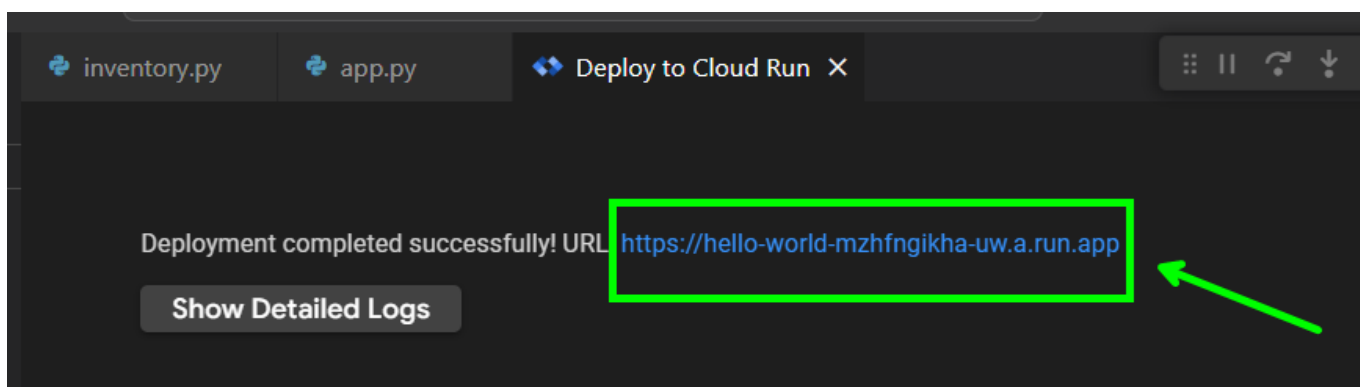


1. No menu principal da barra de atividades (≡), clique em **Exibir > Paleta de comandos**.
2. No campo "Paleta de comandos", digite **Cloud Code Deploy** e, em seguida, selecione **Cloud Code: implantar no Cloud Run** na lista.
3. Para ativar a API Cloud Run no projeto, clique em **Ativar API**.
4. Na página **Configurações de serviço**, para **Região**, selecione **us-west1**.
5. Mantenha os valores padrão para as demais configurações e clique em **Implantar**.

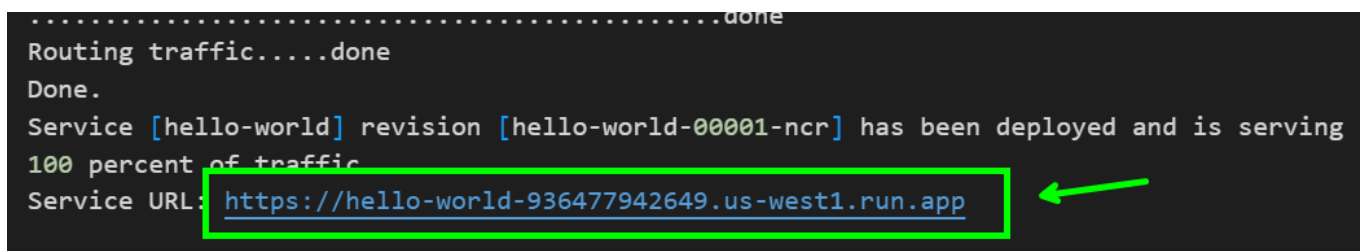
48) Mantenha os valores padrão para as demais configurações e clique em **Deploy** abaixo:



49) Para visualizar o serviço em execução, abra o URL exibido na caixa de diálogo Implantar no Cloud Run.



Ou



50) Teste o serviço anexando os caminhos **/inventory** e **/inventory/{PRODUCTID}** ao URL e verifique a resposta.

OBS: É preciso testar a aplicação conforme descrito para validar a tarefa.



- 51) Tique a última Task do laboratório, garanta que todas as tarefas foram ticadas no quadrado amarelo flutuante localizado ao lado esquerdo da página:

Shell, execute o seguinte comando:

```
export SVC_UF
--region us
--platform
--format='
echo ${SVC_UF
```

Enable relevant APIs and set IAM roles.	Verificar meu progresso	20 / 20
Create and start a Cloud Workstation.	Verificar meu progresso	20 / 20
Deploy your app to Cloud Run.	Verificar meu progresso	60 / 60

Para verificar o objetivo, clique em **Verificar meu progresso**.

Implantar o aplicativo no Cloud Run.

Verificar meu progresso

Assessment Completed!

- 52) Finalize o laboratório no botão vermelho do painel:

Terminar o laboratório
01:00

Cuidado: Quando estiver no console, siga as instruções do laboratório. Caso contrário, sua conta poderá ser bloqueada. [Saiba mais.](#)

[Open Google Cloud Console](#)

Username

student-02-4cb01054fa66

Password

46eMzbAsoBPv

Project ID

qwiklabs-gcp-02-29b343b