# **Informações Iniciais**

Passo a passo do Curso:

Lucas Pavanelli – encaminhou o PDF com instruções iniciais :

F:\!\_Dados Gerais\!!!!!\_2020\Gerais\_2020\Python\!!!Curso-Dev\B-Verifying-your-Lab-Environment-MERAKI-track (2)[5542].pdf

**Usuários:**

**\* Git : https://github.com/sessions/verified-device**

- Usuário : MaxwellMGomes

- Senha: Chara@1973

- Email: Maxwell.gomes@telefonica.com

**Webex Teams (Instalamos o aplicativo)**

- usuário : [Maxwell.gomes@telefonica.com](mailto:Maxwell.gomes@telefonica.com)

- Senha: Chara@1973

- Email: [Maxwell.gomes@telefonica.com](mailto:Maxwell.gomes@telefonica.com)

Foi criado uma sala para o treinamento e estou incluso

-----------------------------------------------------------------------------------------------

Ele indicou a instalação dos aplicativos abaixo

Para facilitar abaixo temos os principais requisitos (mas não deixem de dar uma olhada no passo-a-passo anexado!)

1. Git - <https://git-scm.com/downloads>
2. Clone da pasta do DNE-MERAKI - <https://github.com/CiscoDevNet/meraki-code>
3. Postman - <https://www.postman.com/downloads/>
4. Python 3 - <https://www.python.org/downloads/>
5. Webex teams - <https://www.webex.com/downloads.html>

------------------------------------------------------------------------------------------------------------------------------

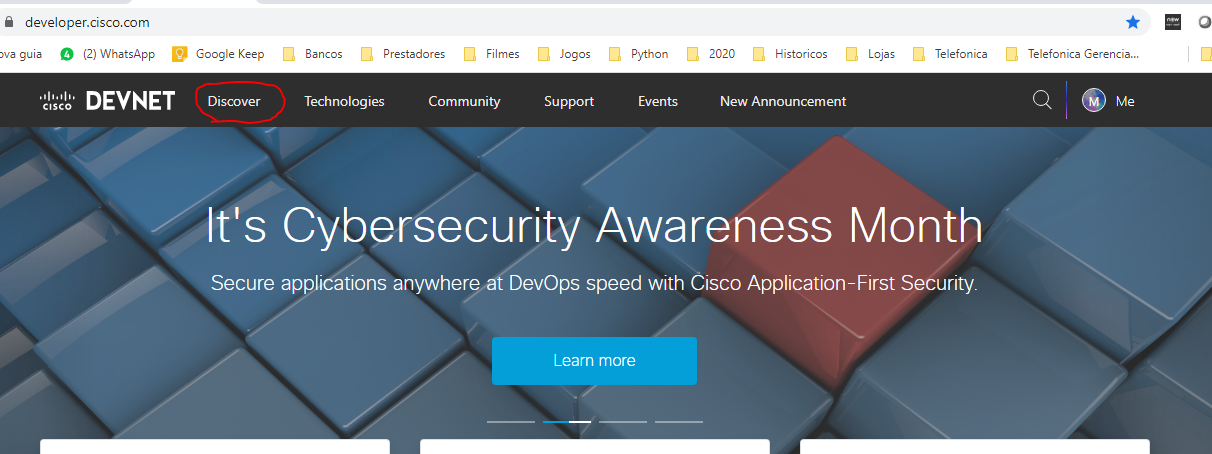
Entrando no ambiente de Labs

https://developer.cisco.com/learning/

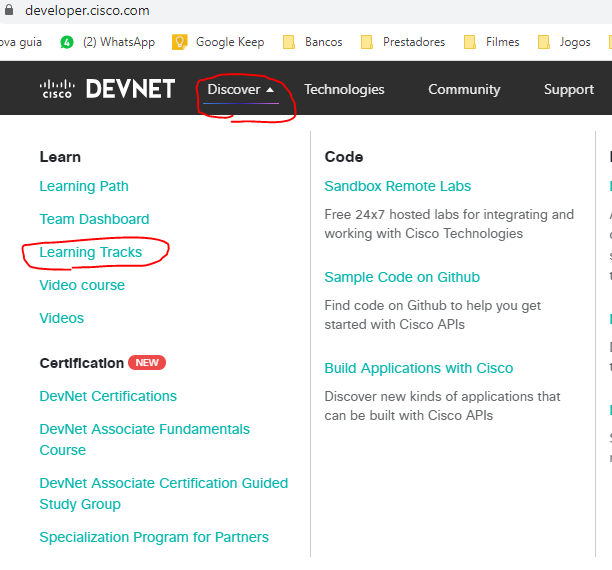
<https://developer.cisco.com/learning/devnet-express/devnet-express-meraki>

Navegando nas páginas do Cisco DevNet pra chegar no curso

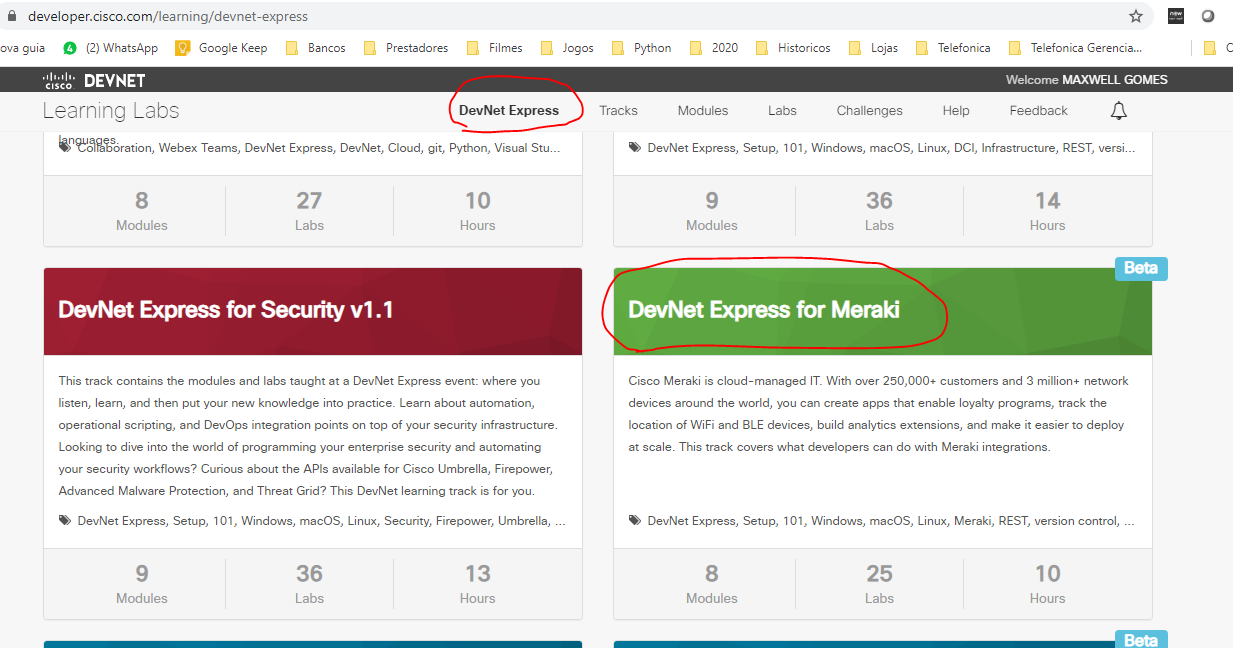
<https://developer.cisco.com/> --- > Logar



Discover 🡪 Learning Tracks

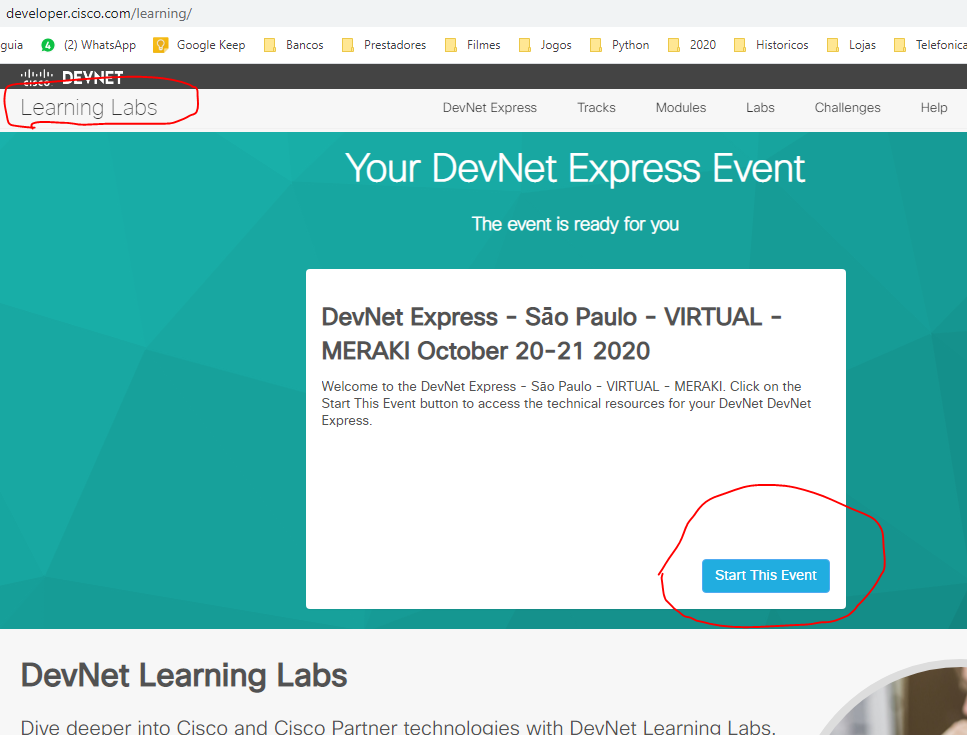


Dentro do Learning Tracks 🡪 Clicar em DevNet Express e escolher o curso 🡪 DevNet Express for Meraki

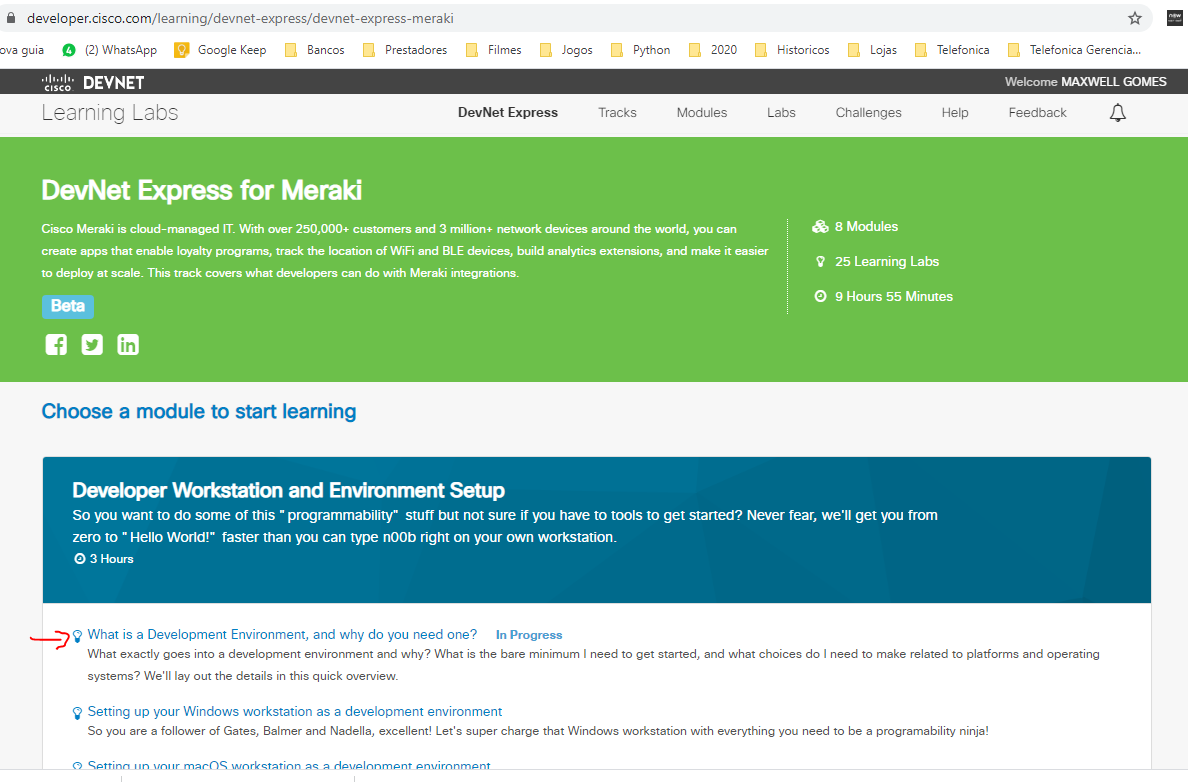


Outro caminho 🡪

Depois clicar no Learnig Labs 🡪 Aparece o evento do curso ativo clicar em “Start This Event”



Para navegar nas Lab do Curso ir em :  [What is a Development Environment, and why do you need one?](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/dev-setup/dev-what/step/1) **In Progress**



# **Módulos do Treinamento :**

* [1 - Estação de trabalho do desenvolvedor e configuração do ambiente](#_1_-_Estação)
* [2 - Verifique o seu ambiente de laboratório](#_2_-_Verifique)
* [3- Introdução aos fundamentos de codificação](#_3_-_Introdução)
* [4 - Fundamentos da API REST](#_4_Fundamentos_da)
* [5 - Introdução às Integrações Meraki](#_5_-_Introdução)
* [6 - API Meraki Dashboard](#_6_-_API)
* [7 Monitoramento Meraki e Integrações Analíticas](#_7_-_Monitoramento)
* [8 Integrações Meraki Captive Portal](#_8_-_Integrações)

## 1 - Estação de trabalho do desenvolvedor e configuração do ambiente

### [1.1 - O que é um ambiente de desenvolvimento e por que você precisa de um?](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/dev-setup/dev-what/step/1)

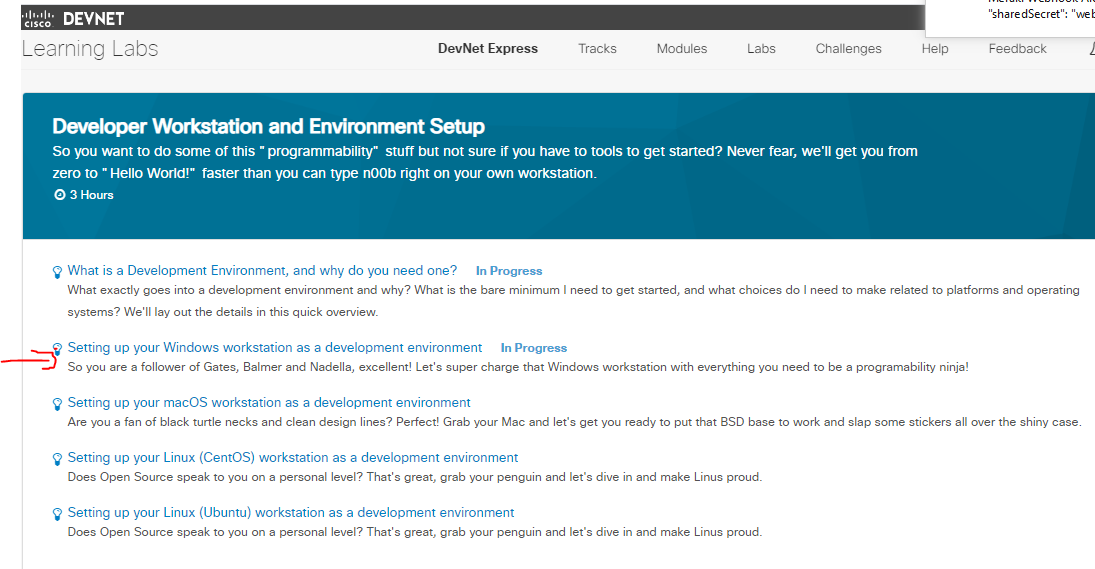
### [1.2 - Configurando sua estação de trabalho Windows como um ambiente de desenvolvimento](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/dev-setup/dev-win/step/1)

### 1.3 - [Configurando sua estação de trabalho macOS como um ambiente de desenvolvimento](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/dev-setup/dev-mac/step/1)

### 1.4 [Configurando sua estação de trabalho Linux (CentOS) como um ambiente de desenvolvimento](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/dev-setup/dev-centos/step/1)

### 1.5 [Configurando sua estação de trabalho Linux (Ubuntu) como um ambiente de desenvolvimento](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/dev-setup/dev-ubuntu/step/1)

Seguindo a instalação das ferramentas :



Pede para fazer o clone do github 🡪 Aplicar no git-bash

git **clone** <https://github.com/CiscoDevNet/hello_network>

## 2 - Verifique o seu ambiente de laboratório

[2.1 - Verifique e prepare seu DevNet Express Lab Pod](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/meraki-verify/meraki-verify/step/1)

Falha no lab Verify Your Lab Environment

#### Step 3: Setup Python Virtual Environment

Install the lab requirements and run the Meraki Cloud Simulator

pip install -r requirements.txt 🡪 Apresenta erro no wheel e o netifaces

**cd** ~/code/meraki-code/meraki\_cloud\_simulator

python meraki\_cloud\_simulator.py 🡪 Apresenta erro ao carregar o módulo request

* Tive que estalar os pacotes todos no Python do clone no diretório /
* C:\Users\maxwe\code\meraki-code\venv\Scripts
* Também instalei o requeriments.txt manualmente. As falhas ficaram somente com o netifaces, tive sucesso com o request 🡪 o problema era a versão que não podia ficar vazio
* Teste com o simulator ok
* Naveguei no  <http://localhost:5001/go>. E vi o reporte pelo git-bash
* pip install -r requirements.txt cd ~/code/meraki-code/meraki\_cloud\_simulator python meraki\_cloud\_simulator.py

#### Step 5: Prepare Lab Environment Files

Dentro do arquivo : C:\Users\maxwe\code\meraki-code\ env\_user.py.

Token dura 12 horas. Este aí é o de 20/10/20

# User Input  
  
WT\_ACCESS\_TOKEN = "MTAzOWE1NGQtY2QxNi00M2I3LWI4NDktNGI5MDg2MWFiOGNjOTEzYzVkYjYtNDYx\_PF84\_consumer"  
WT\_ROOM\_ID = "Y2lzY29zcGFyazovL3VzL1JPT00vNTM1Mjc4NjAtMTI0MS0xMWViLWE2NjMtMGJhYjc3OGQxZmFi"  
MERAKI\_API\_KEY = "6bec40cf957de430a6f1f2baa056b99a4fac9ea0"

Token dura 12 horas. Este aí é o de 21/10/20

WT\_ACCESS\_TOKEN =

YmNjYzZlZWItMjUzMy00OTM1LTkwZWUtZjk1YThjMGRlZjFiYTJkNjAxZGEtNDk0\_PF84\_consumer

#### Etapa 6: execute o script de verificação

Ativando ambiente virtual pelo git bash

cd ~/code/meraki-code $ source venv/Scripts/activate

## 3 - Introdução aos fundamentos de codificação

### [3.1 - Introdução à codificação e APIs](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/fundamentals/intro-coding-and-apis/step/1)

### 3.2 - [Uma breve introdução ao Git](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/fundamentals/git-basic-workflows/step/1)

#### Step 2: Basic Git workflows

Sobre o Git Passos e comandos.

Git diff 🡪 ver as diferenças

Gig status 🡪 status

Git fetch 🡪 fazer as atalizações

Git checkout (nome de brach) fazer alteração de branch

Git branch –(d)elete –force <nome do brach> -> deleta o branch

Git checkout (nome do arquivo) 🡪 Recupera a última versão do arquivo

git switch <master> -- > alterna de branch

Git log 🡪 Mostra as alterações

gitk 🡪 Abre versão gráfica do Git

git commit -a 🡪 Confirma as alterações Abrirá um texto para preecher.

git commit -m “texto de alteração” 🡪 confirma alterações Evitará abrir o editor de txt

alice $ git pull </home/bob/ myrepo> master 🡪 Busca e mescla as alterações de um branch

<https://git-scm.com/docs/gittutorial> --> OUTROS COMANDOS e explicações sobre o git

#### Step 3: The DevNet sample-code workflow

| **Degrau** | **Açao** | **Comandos Git** |
| --- | --- | --- |
| 1 | Clone o Repositório Remoto | git clone <url> |
| 2 | Criar e verificar uma filial local | git checkout -b <new branch name> |
| 3 | Confirmar alterações incrementalmente | git add <new or modified file> git commit -m "Commit Message" |

### [3.3 Introdução ao Python - Parte 1](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/fundamentals/intro-python-part1/step/1)

### [3.4 Introdução ao Python - Parte 2](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/fundamentals/intro-python-part2/step/1)

#### Etapa 3: Loops Python

#### Iterando por meio de um dicionário

fruit\_inventory = {"apples": 5, "pears": 2, "oranges": 9}

>>> **for** fruit **in** fruit\_inventory:

... print(fruit)

... oranges apples pears

>>> list(fruit\_inventory.items())

[('oranges', 9), ('apples', 5), ('pears', 2)]

>>> **for** fruit **in** fruit\_inventory.items():

... print(fruit) ...

('oranges', 9) ('apples', 5) ('pears', 2)

>>> **for** fruit, quantity **in** fruit\_inventory.items():

... print("You have {} {}.".format(quantity, fruit))

... You have 5 apples. You have 2 pears. You have 9 oranges.

#### Etapa 6: exercício prático

##### Seu código funcionou?

Impressionante. Não se esqueça de usar

git add <filename>e git commit -m "Your commit message"salvar seu sucesso!

#### Onde ir para aprender mais

##### Para iniciantes:

* [Cursos introdutórios de Python edx.org](https://www.edx.org/course/?search_query=Python&level=Introductory)

O autor deste laboratório recomenda enfaticamente o seguinte curso, onde ele aprendeu Python:

[MITx 6.00.1x: introdução à ciência da computação e programação usando Python](https://www.edx.org/course/introduction-computer-science-mitx-6-00-1x-11)

* [Coursera.org Cursos de Python](https://www.coursera.org/courses?_facet_changed_=true&domains=computer-science&languages=en&query=Python&userQuery=Python)
* [codecademy.com Aprenda Python](https://www.codecademy.com/learn/learn-python)
* [Aprenda Python da maneira mais difícil](https://learnpythonthehardway.org/)

##### Para codificadores intermediários:

* [Cursos intermediários de Python edx.org](https://www.edx.org/course/?search_query=Python&level=Intermediate)
* [O Guia do Mochileiro para Python!](http://docs.python-guide.org/en/latest/)
* [Python eficaz](https://effectivepython.com/)
* [Full Stack Python](https://www.fullstackpython.com/)

### [3.5 Analisando JSON com Python](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/fundamentals/parsing-json-python/step/1)

#### Introdução

##### Benefícios dos formatos de intercâmbio de dados padronizados

Existem vários formatos de intercâmbio de dados padronizados proeminentes em uso hoje, sendo os seguintes os mais difundidos:

1. [JSON (JavaScript Object Notation)](https://www.json.org/)
2. [XML (eXtensible Markup Language)](https://www.w3.org/XML/)
3. [YAML (YAML Ain't Markup Language)](http://yaml.org/)

#### Onde ir para aprender mais

Quer saber mais sobre JSON e o jsonmódulo Python ? Confira os seguintes recursos:

* [json.org](https://www.json.org/)
* [Codificador e decodificador JSON - nos documentos Python](https://docs.python.org/3/library/json.html)

## 4 Fundamentos da API REST

### [4.1 - O que é REST? O que são APIs?](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/rest-api-fundamentals/what-are-rest-apis/step/1)

#### Etapa 1: O que é REST? O que é uma API?

#### Etapa 2: uma olhada sob o capô em REST

#### Etapa 3: payloads REST

#### Etapa 4: alguns exemplos de API REST

##### API Cisco Webex Teams

Acesse <https://developer.webex.com/> e faça login para ver um token de desenvolved

1. Depois de se inscrever e fazer login no [site Cisco Webex for Developers,](https://developer.webex.com/) você pode obter seu token de API na [seção Autenticação da página de Introdução](https://developer.webex.com/docs/api/getting-started#accounts-and-authentication) .
2. Em seguida, vá para Referência da API> Pessoas> Obter meus próprios detalhes para encontrar o seu personIdnessa página da web:

Meu Personal ID em 02/11/20 :

02/11/20 🡪

Zjg4NTIyNTktY2E1My00OTNjLTg4ODgtNjFhYWQ0OTkwNThjMDgxNDQyZmItYmNh\_PF84\_consumer

Exemplo 🡪

Y2lzY29zcGFyazovL3VzL1BFT1BMRS83MDY4MmE4My04MzdlLTQzZDgtYTBhYS0xNDVlYjdkZmY5ODc

Agora, abra uma janela do Terminal e execute este exemplo para enviar uma mensagem "Hi from DevNet". Substitua o toPersonEmail=someone@cisco.comwith the email associated with your Webex Teams account, and put in your token value instead ofMDg ... zRh4:

curl https://webexapis.com/v1/messages -X POST -H "Authorization:Bearer Zjg4NTIyNTktY2E1My00OTNjLTg4ODgtNjFhYWQ0OTkwNThjMDgxNDQyZmItYmNh\_PF84\_consumer" --data "maxwell.gomes@telefonica.com" --data "text=Ola%20dem%20DevNet"

curl https://webexapis.com/v1/messages -X POST -H "Authorization:Bearer **Zjg4NTIyNTktY2E1My00OTNjLTg4ODgtNjFhYWQ0OTkwNThjMDgxNDQyZmItYmNh\_PF84\_consumer**" --data "toPersonEmail=maxwell.gomes@telefonica.com" --data "text=Hi%20from%20DevNet"

Não tive sucesso. Apresentou a seguinte falha:

maxwe@DESKTOP-R4GM4RK MINGW64 ~/code/meraki-code (meucodigo)

$ curl https://webexapis.com/v1/messages -X POST -H "Authorization:Bearer Zjg4NTIyNTktY2E1My00OTNjLTg4ODgtNjFhYWQ0OTkwNThjMDgxNDQyZmItYmNh\_PF84\_consumer" --data "toPersonEmail=maxwell.gomes@telefonica.com" --data "text=Hi%20from%20DevNet"

% Total % Received % Xferd Average Speed Time Time Time Current

Dload Upload Total Spent Left Speed

100 213 100 147 100 66 105 47 0:00:01 0:00:01 --:--:-- 153{"message":"Failed to create room.","errors":[{"description":"Failed to create room."}],"trackingId":"ROUTER\_5FA03EA2-EC24-01BB-1FD8-AC12FAA61FD8"}

### [4.2 – Introdução às APIs REST?](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/rest-api-fundamentals/what-are-rest-apis/step/1)

#### Etapa 7: ferramenta REST API com exemplos

##### Prática com cURL: cliente para URLs

Juntando tudo, você pode executar este curlcomando a partir de uma janela de Terminal no Mac ou em uma janela do Git Bash no Windows 10:

$ curl -X GET https://webexapis.com/v1/teams -H "Authorization:Bearer Zjg4NTIyNTktY2E1My00OTNjLTg4ODgtNjFhYWQ0OTkwNThjMDgxNDQyZmItYmNh\_PF84\_consumer" | python -m json.tool

% Total % Received % Xferd Average Speed Time Time Time Current

Dload Upload Total Spent Left Speed

100 305 0 305 0 0 219 0 --:--:-- 0:00:01 --:--:-- 219

{

"items": [

{

"id": "Y2lzY29zcGFyazovL3VzL1RFQU0vNjAwZDU5MTAtZjM2Ny0xMWVhLWE5ZTMtOTFkYTQxNDNiYzlk",

"name": "DevNet Express - S\u00c4\udc81o Paulo - VIRTUAL - MERAKI - October 20-21 2020",

"creatorId": "Y2lzY29zcGFyazovL3VzL1BFT1BMRS8zMWJlZTdjZi1mNTlmLTRlYjgtYmY5Ny1jZjkxOWYxMjRhZDY",

"created": "2020-09-10T13:13:10.689Z"

}

]

}

##### Solicitações Python - uma biblioteca Python

Python Requests é uma biblioteca Python simples, com documentação e exemplos em <http://docs.python-requests.org/> .

##### OpenAPI/Swagger: dynamic API documentation

<https://editor.swagger.io/> 🡪 Auxilia na geração de APIs

### 4.3 - [Exercício prático: usando o Postman para interagir com APIs REST](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/rest-api-fundamentals/hands-on-postman/step/1)

## 5 - Introdução às Integrações Meraki

### [5.1 - Introdução às integrações Meraki](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/meraki-integrations/meraki-01-integrations/step/1)

## 6 - API Meraki Dashboard

### [6.1 Documentos da API Meraki REST](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/meraki-dashboard-api/meraki-09-meraki-docs/step/1)

### 6.2 [Meraki Dashboard API usando Postman](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/meraki-dashboard-api/meraki-02-dashboard-api/step/1) [6.3 Meraki Wireless Health](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/meraki-dashboard-api/meraki-06-wireless-health/step/1) [6.4 Missão Meraki](https://developer.cisco.com/learning/devnet-express/devnet-express-meraki/meraki-dashboard-api/meraki-05-mission/step/1)

## 7 - Monitoramento Meraki e Integrações Analíticas

## 8 - Integrações Meraki Captive Portal