**Tutorial para fazer o nosso hands on na AWS.  
(Criação de conta até o deploy do site manualmente)**

1. **Vamos começar o nosso tutorial, o primeiro passo vai ser criar a conta na AWS.**

Clicando no link abaixo:

[https://aws.amazon.com/pt/free/?all-free-tier.sort-by=item.additionalFields.SortRank&all-free-tier.sort-order=asc&awsf.Free%20Tier%20Types=\*all&awsf.Free%20Tier%20Categories=\*all](https://aws.amazon.com/pt/free/?all-free-tier.sort-by=item.additionalFields.SortRank&all-free-tier.sort-order=asc&awsf.Free%20Tier%20Types=*all&awsf.Free%20Tier%20Categories=*all)

Abaixo vamos em **“Crie uma conta da AWS”**

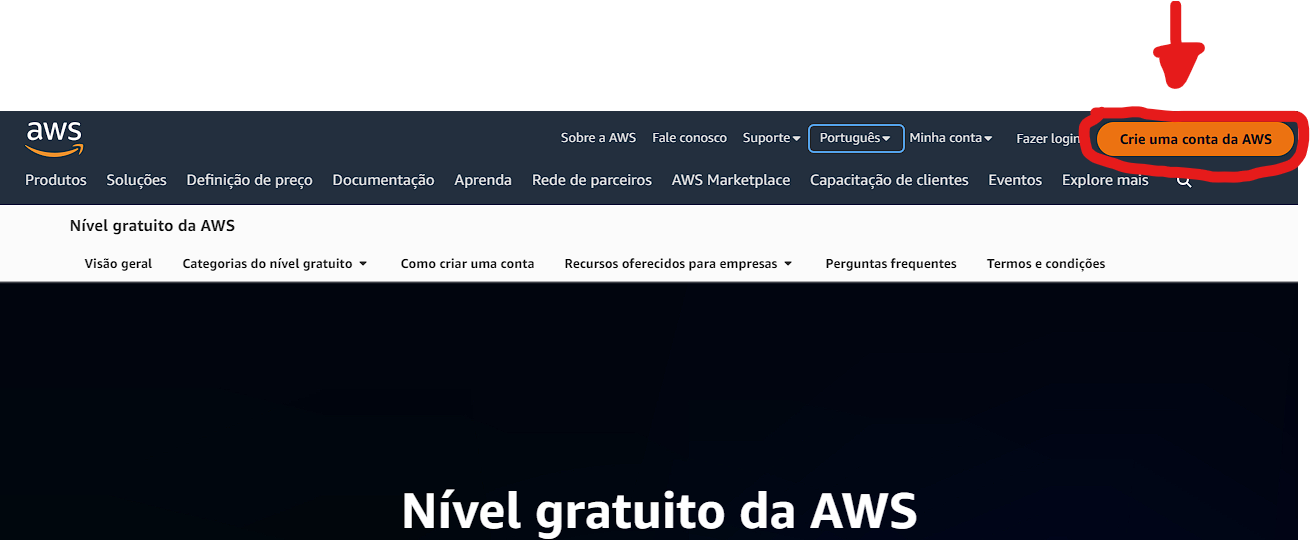
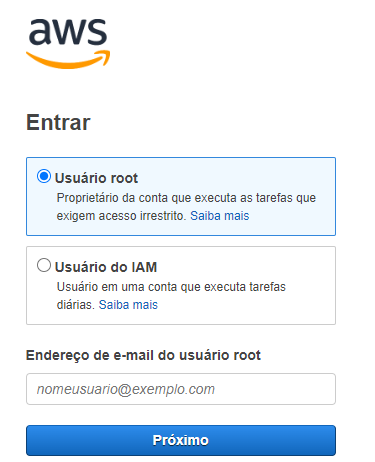


Figura 1 Criando uma conta na AWS

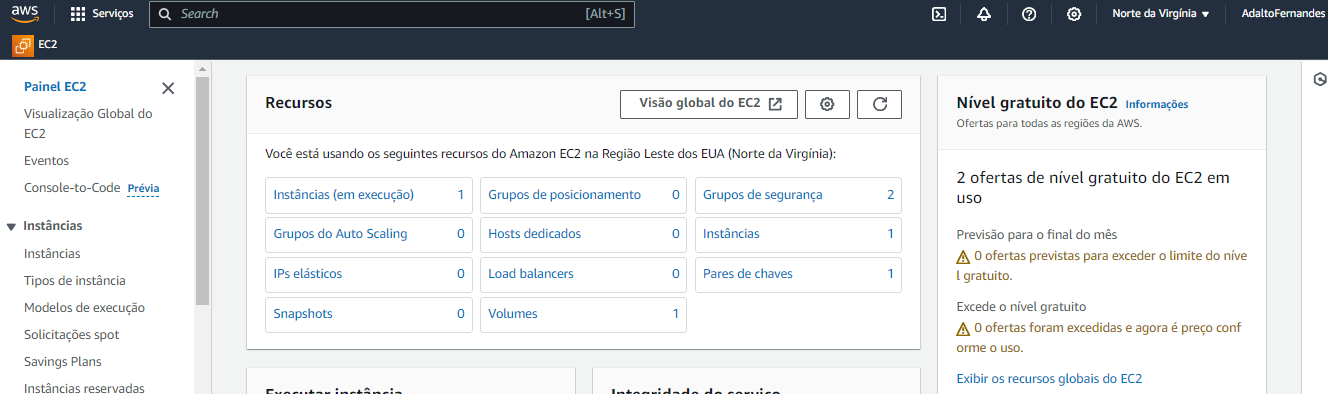
**Seguir o passo a passo para a criação da conta de acordo com o site da aws.**

1. **Segundo passo:** **Fazer Login no console da AWS.**

Vamos realizar o login como “usuário root” colocando e-mail e senha.

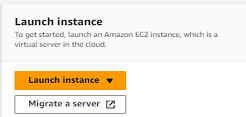


Vamos agora até o console da AWS, onde temos o controle dos serviços:

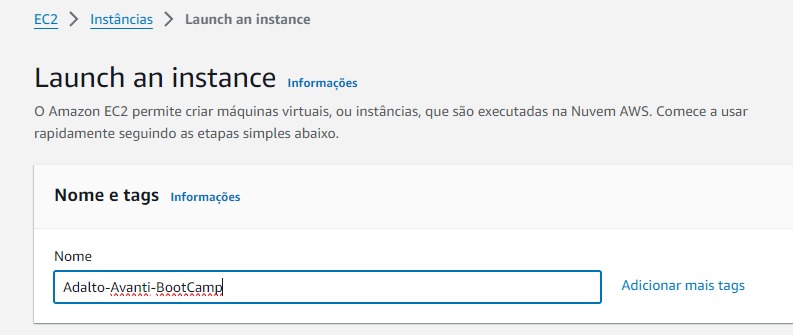


1. Terceiro Passo: Criação que é o launch da nossa instancia.

Vamos clicar ali em **Launch instance**, ou **Executar instância** em português.

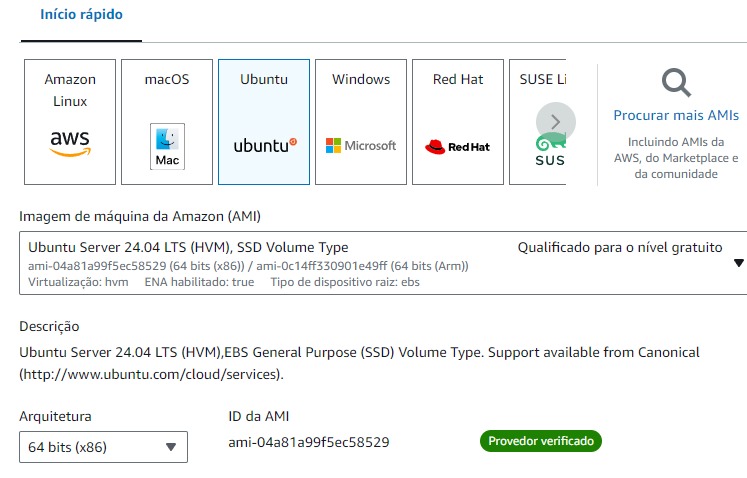


Agora vamos dar nome a nossa instancia. Só preencher o campo abaixo.

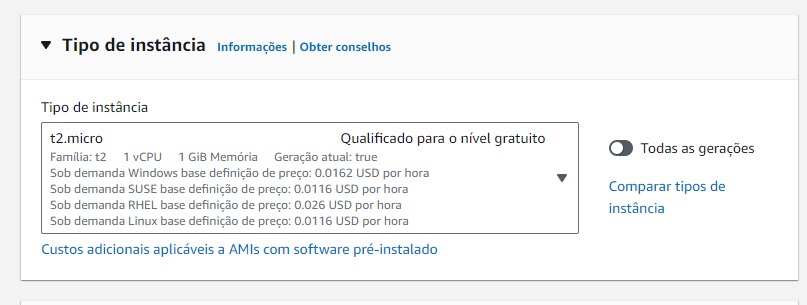


1. O quarto passo**:** **Escolher nosso sistema operacional para nossa EC2** da AWS. (Vamos usar apenas as free tier para não ter gastos)

Abaixo vamos escolher o sistema: **Ubuntu Server 24.04 LTS.**



1. Quinto passo: Usar a instância que é de graça, a **t2.micro.**



1. Sexto passo: **Criar o par de chaves**  
   Agora não tem muito segredo é escolher o nome que deseja e guardar o formato **.pem** numa pasta onde deseja para acessar posteriormente se desejar.



1. Setimo passo: **Configurar o grupo de segurança.  
   (Passo muito importante)**

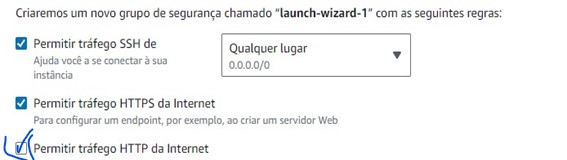
Estes grupos de segurança desempenham o papel de firewalls virtuais,

Conseguimos personalizar esses grupos para autorizar ou

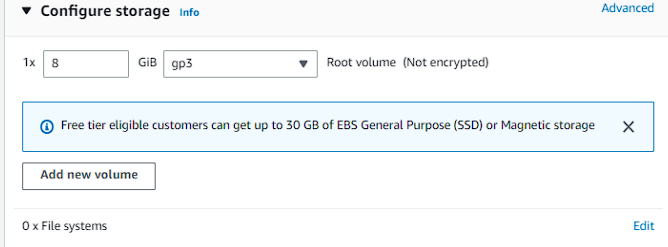
Bloquear o tráfego de endereços IP específicos, protocolos e portas.

**Agora devemos criar um novo grupo de segurança ou escolher um já existente.**

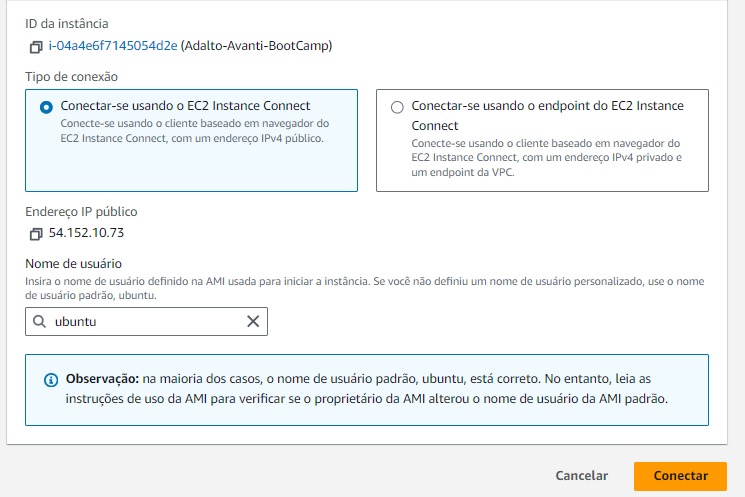
Abaixo vamos deixar as configurações nesse padrão. (**Eu esqueci de marcar o Permitir Tráfego HTTP, acabei enfrentando um problema que mostro como resolver no final)**   
Fazendo dessa forma marcando todos abaixo, não vamos passar pelo o que eu passei.

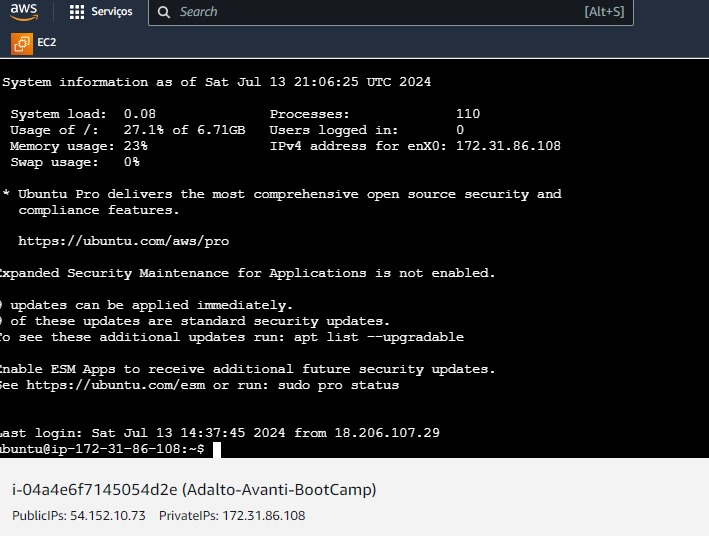
****

1. Oitavo passo: Escolha do disco de armazenamento.



1. Nono passo: Conectar a instância.   
   Vamos apertar em **“Conectar”**

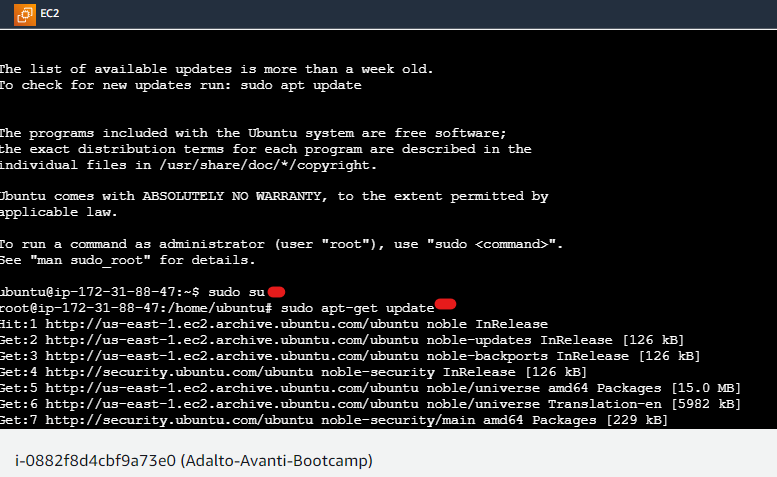


**Após conectar, está é a tela que temos que ver:**  


* **Agora vamos configurar e instalar o Apache**

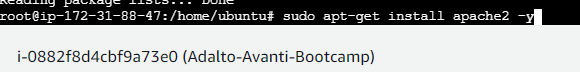
Vamos digitar no console: **sudo su**

E depois vamos digitar: **sudo apt-get update**



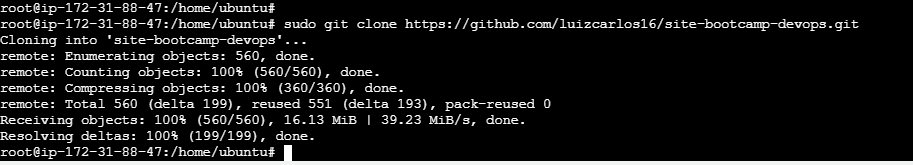
* Instalando o **Apache**

Após o update, digitar no console: **sudo apt-get install apache2 -y**



* Agora vamos baixar o projeto do Github

No console digitar: **sudo git clone https://github.com/luizcarlos16/site-bootcamp-devops.git**

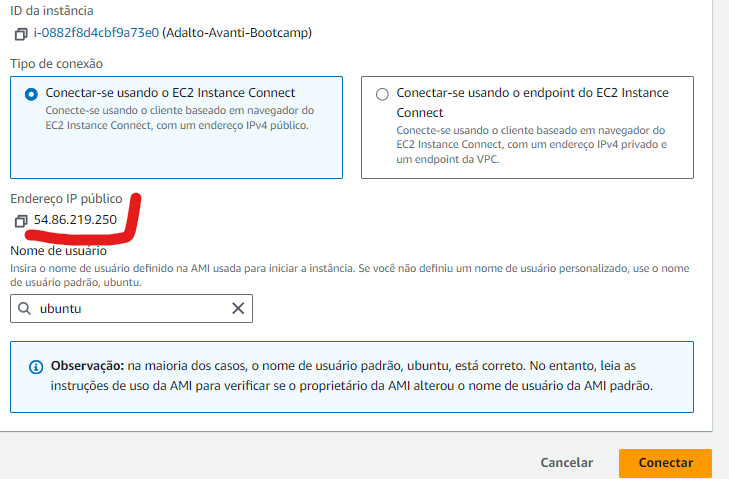


* Verificando se o clone foi feito, no console digitar: **ll**  
  Depois vamos copiar os dados do clone e jogar no html, digitando :

**sudo cp site-bootcamp-devops/\* -R /var/www/html/**

* **Agora vamos conectar no site que copiamos.**

Para pegar o nosso endereço de **IP Público** é só ir onde conectamos.



* **Vamos abrir o navegador e digitar o nosso endereço de IP Público.**



**Pronto conseguimos fazer o nosso deploy manual.**

* Neste processo eu esqueci de habilitar o HTTP na configuração da máquina na AWS. **(Resolvendo a porta 80)**

**Vou ensinar como eu resolvi o problema, minha porta 80 não estava habilitada.**

* No painel de instancias, vou até o nome do grupo de segurança, para visualizar qual estava sendo usado. Vi que estou usando o launch-wizard-2 e default.
* Agora vamos ali na esquerda em “**Rede e Segurança**” clicar em “**Security Groups**”

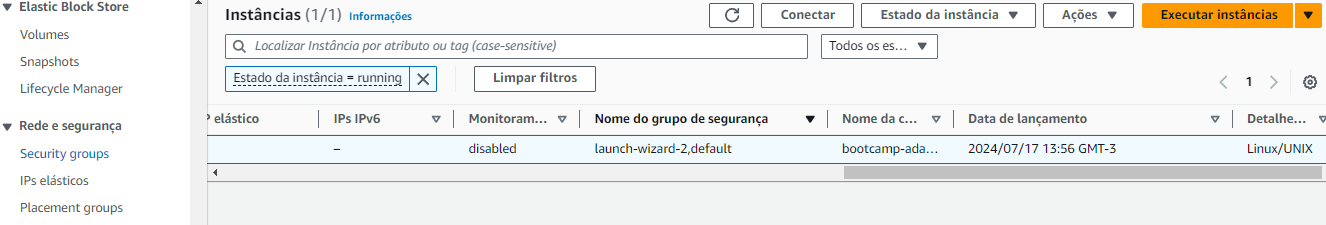
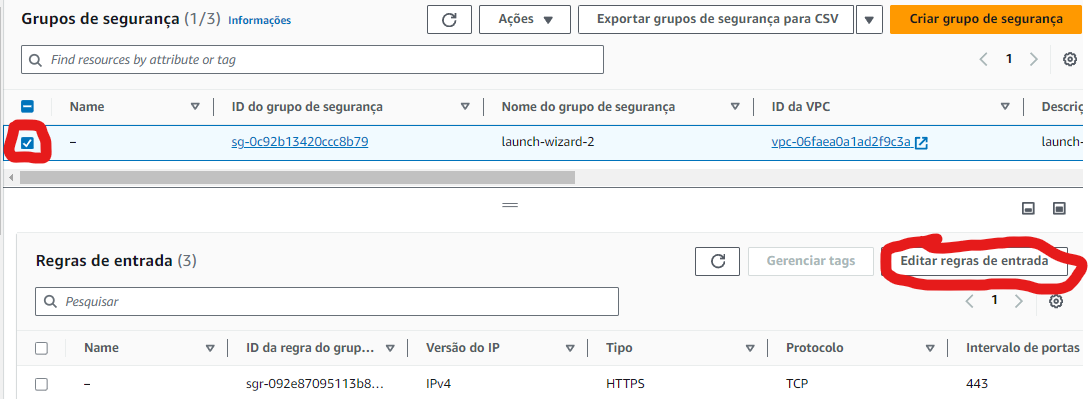
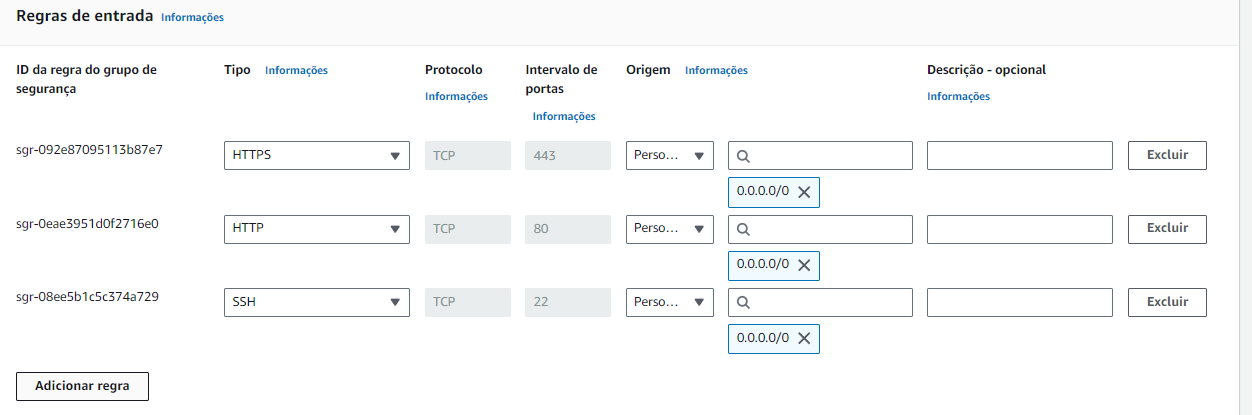


Figura 2 - Nome do Grupo de Segurança e Security Groups

* Agora vamos selecionar o grupo de segurança e editar a **“Regras de entrada”,** dessa forma vamos liberar a porta 80.

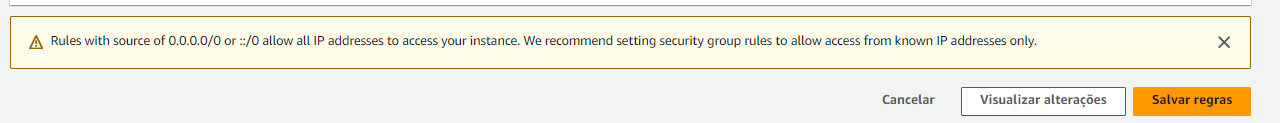


* Na tela abaixo, vamos escolher **HTTP** no Intervalor de portas vamos colocar a porta **80** e **Habilitar o 0.0.0.0/0.**



Deve ficar assim, com a porta 80. Pronto agora o site vai funcionar e vamos conseguir nos conectar ao nosso IP Público corretamente.

**Pronto só vamos salvar as regras.**

****

**Tutorial Hands on Terraform e AWS**

**(Da instalação ao Deploy)**

* **Agora vamos instalar o Terraform**

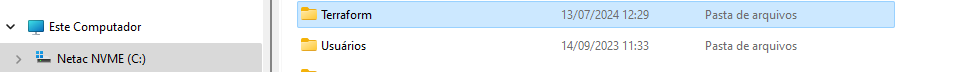
Vamos acessar o site para download do Terraform (Só escolher seu sistema operacional).

[https://developer.hashicorp.com/terraform/install?product\_intent=terraform#windows](https://developer.hashicorp.com/terraform/install?product_intent=terraform%23windows)

****

Figura 3 - Baixei a versão AMD64

* Agora vamos criar uma pasta com nome **Terraform** no disco c.



Agora ficou fácil vamos extrair o que a gente baixou, dentro dessa pasta.

Ou seja, vamos extrair os arquivos na pasta **C:\Terraform.**

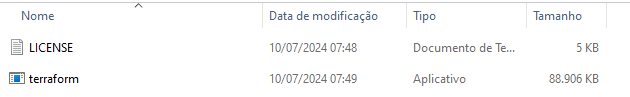
****

Figura 4 - A pasta vai ficar assim.

* Agora vamos configurar a “Instalação” do Terraform no Windows.

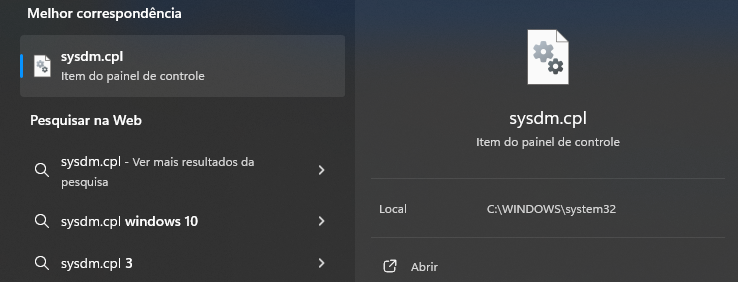
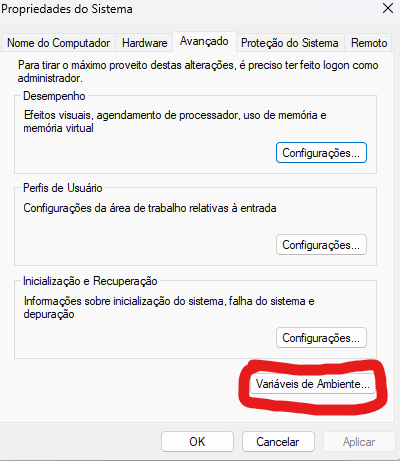
**No pesquisar vamos digitar: sysdm.cpl**  


Figura 5 - Agora é abrir

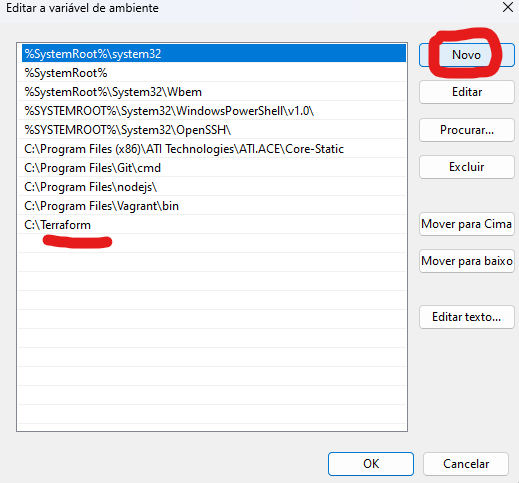
* No ambiente de propriedades, vamos clicar em avançado e nessa aba vamos em **Variáveis do Ambiente**.



* Na segunda janela na Variáveis do sistema, vamos **selecionar o Path** e ir em **Editar**.

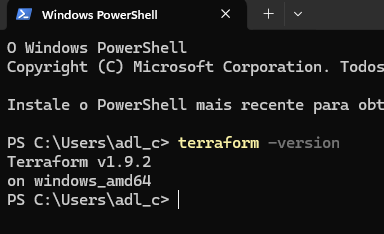


* Clicar em Novo e adicionar: **C:\Terraform, agora é ir no Ok.**



* Vamos abrir o Powershell e ver se a instalação está correta.

Vamos digitar no console: **terraform -version**



**Pronto nosso Terraform está instalado corretamente.**

* **Instalação da AWS CLI no Windows.**

No site: <https://docs.aws.amazon.com/pt_br/cli/latest/userguide/getting-started-install.html>  
vamos seguir o passo a passo via CMD, também pode baixar o **instalador AWS CLI MSI para Windows 64bits.**

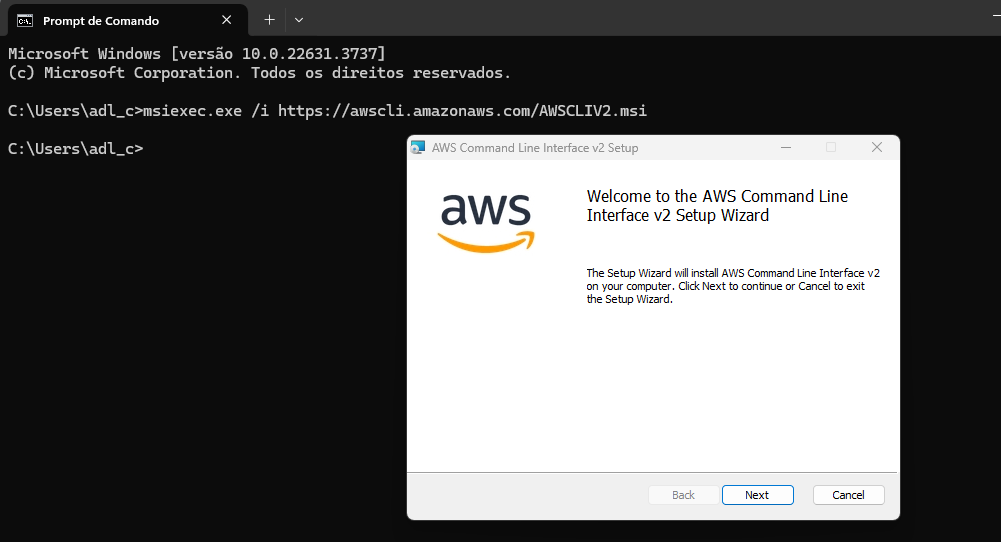
O link para download**:** [https://awscli.amazonaws.com/ AWSCLIV 2.msi](https://awscli.amazonaws.com/AWSCLIV2.msi)

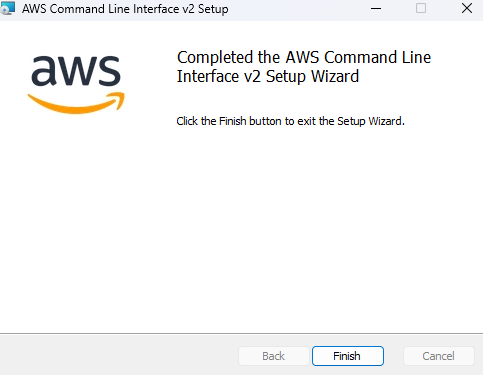
**(Se quiser instalar visualmente)**

**Vamos trabalhar via console (CMD).**

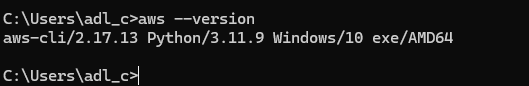
* **Primeiro passo:** abrir o **CMD**

Lá no cmd vamos digitar:   
**msiexec.exe /i** [**https://awscli.amazonaws.com/AWSCLIV2.msi**](https://awscli.amazonaws.com/AWSCLIV2.msi)- Clicar em **Next** e seguir a instalação.

****

****

* Com a instalação do CLI, vamos confirmar se está tudo ok no cmd.Digitando no cmd: **aws –version**

****

**Pronto, está tudo certo.**

**Terraform e AWS - Criação de Usuário na AWS**

1° - Acessar o site da AWS para criação da **“Identity and Acessar Management”**, ou seja, o IAM.

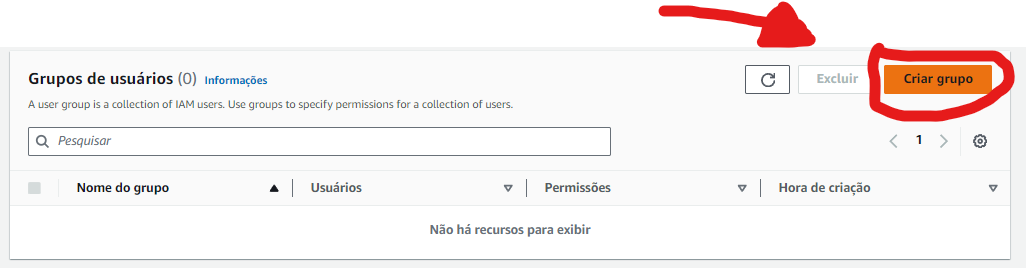
Link: [**https://us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/home**](https://us-east-1.console.aws.amazon.com/iam/home?region=us-east-1%23/home)

2° - Vamos criar um **“Grupo de Usuário”**

Como na imagem abaixo, vamos no Grupos de Usuários



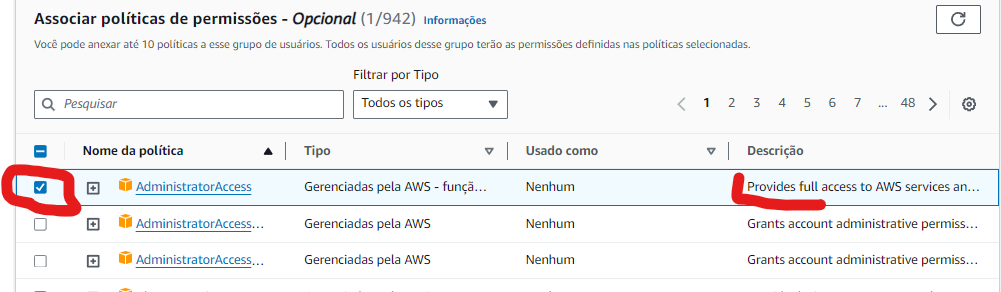
* Agora é ir em **“Criar Grupo”**

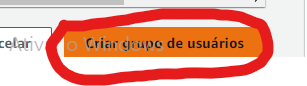


* Definir o nome do grupo, o meu coloquei adaltec, você coloca o que desejar.

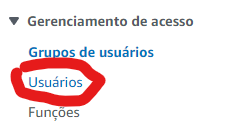


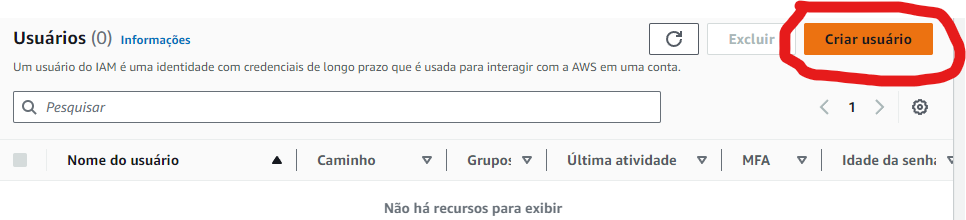
* Aqui é a parte sensível, porque vamos habilitar o acesso “Full” para nosso grupo. Ou seja, vamos liberar tudo para este grupo. Abaixo marcamos o **Administrator com Descrição Provides full** e no final da página apertar em **Criar grupo de usuários**.



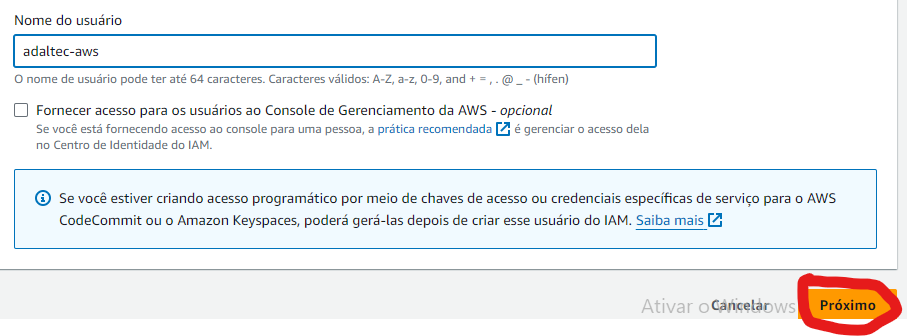


* Agora é ir na Aba **“Usuários”** e depois em **“Criar Usuário”**

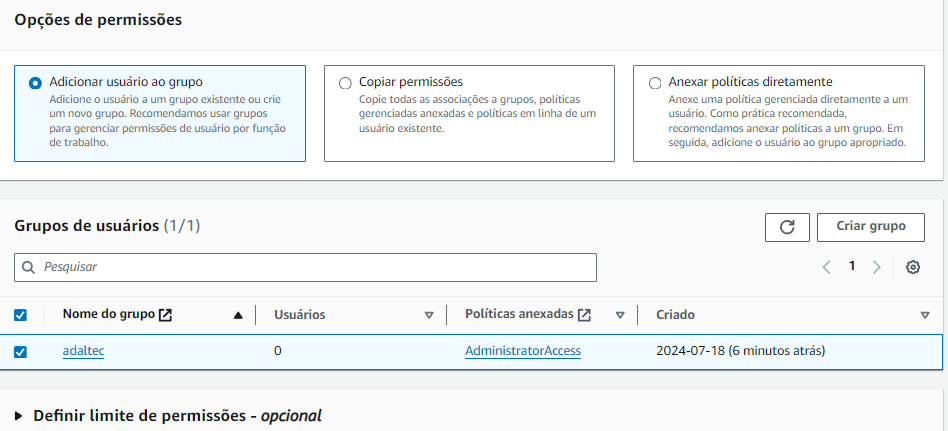




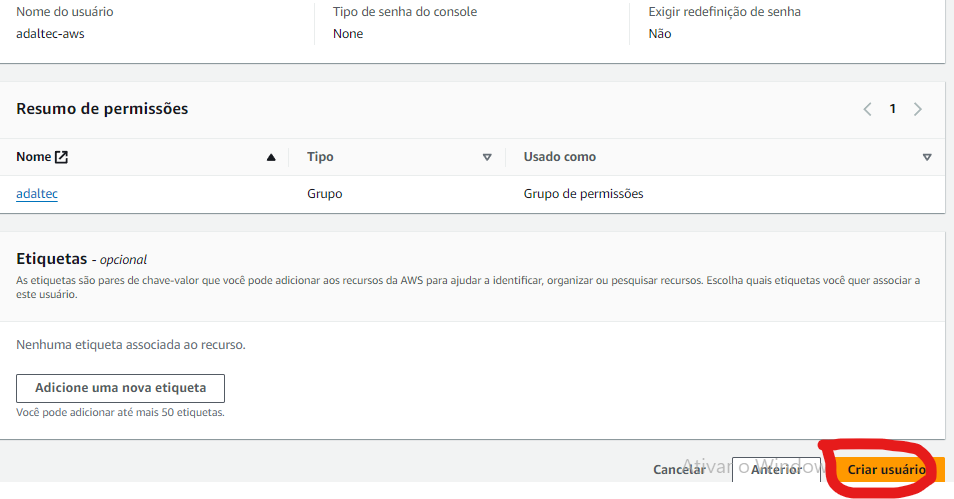
* Vamos adicionar o **nome do usuário** e aperta em **Próximo**



* Neste momento vamos adicionar o nosso **Usuário** ao **Grupo** que criamos.



* No final da página vamos apertar em **Próximo** e **Criar usuário**



* Agora com o usuário criado, **vamos selecionar o usuário (adaltec)** para criar a chave de acesso, para usarmos na configuração da nossa máquina via prompt com o AWS CLI. Ou seja, agora vamos conseguir criar a estrutura, ec2 pelo CMD via Terraform.  
  E **selecionar** também criar **chave de acesso**.

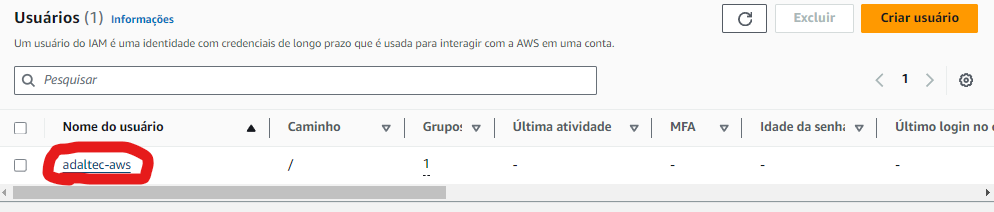
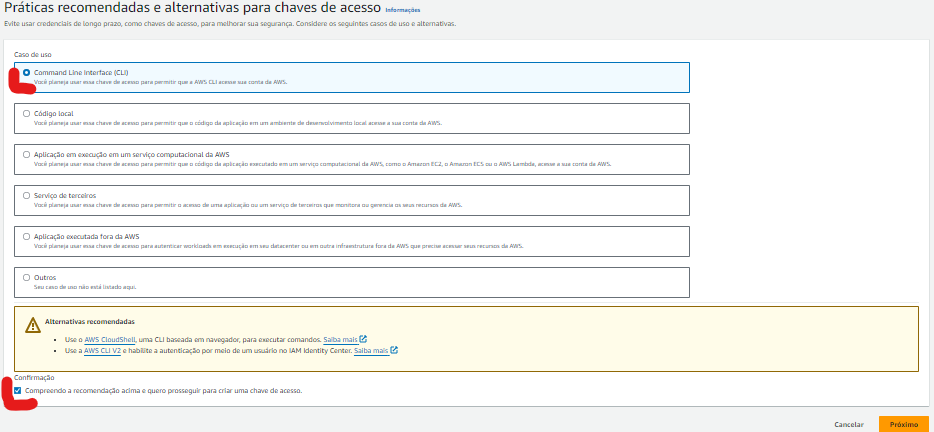


Figura 6 - Selecionando o nome do usuário



Figura 7 - Selecionando criar chave de acesso

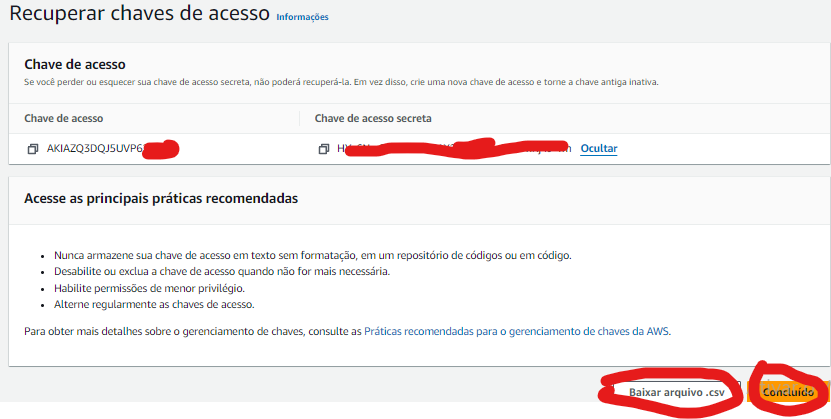
* Após clicar em **“Criar chave de acesso”,** vamos selecionar o **“Command Line Interface (CLI)”** e no final da página **selecionar** também **“Compreendo a recomendação acima e quero prosseguir para criar uma chave de acesso.”  
  E clicar em Próximo no final da página.**



* Agora é clicar em **“Criar chave de acesso”**



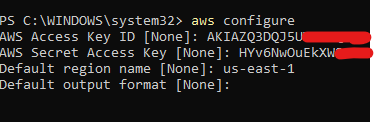
* Pronto agora com a chave criada, vamos guardar ela e “**Baixar** **arquivo .csv”** e “**Concluído”.**



* Agora vamos abrir o **CMD** para vincular nosso usuário ao **AWS CLI**

No console vamos digitar: **aws configure**

Depois vamos digitar nossa: **CHAVE DE ACESSO (AWS KEY ID),   
CHAVE DE ACESSO SECRETA (AWS SECRET ACCESS KEY)  
Região do servidor: us-east-1 (a minha região)  
Default output format (só apertar enter)**

****

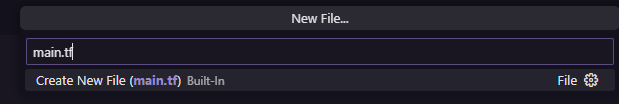
**Pronto, agora vamos executar no terraform é criar nossa ec2.**

* Fazendo a automação no **VISUAL STUDIO CODE** para rodar no Terraform.

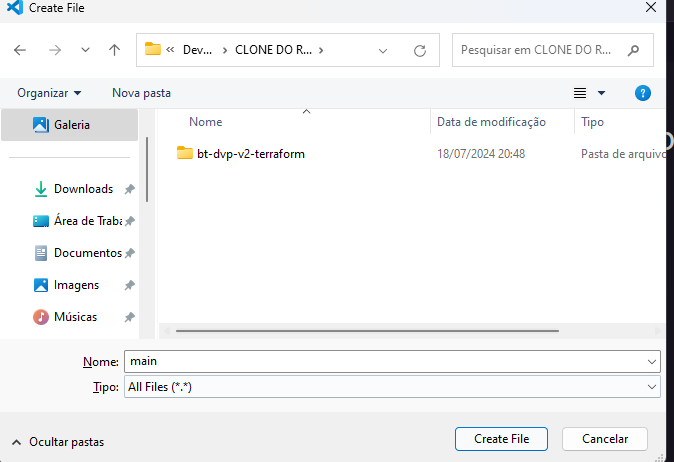
- Vamos abrir o **VS Code** e **criar 1** arquivo de nome **main.tf**

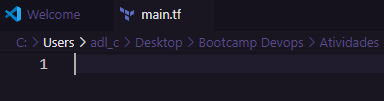
**-> Em File –** clicar em **New File** e colocar o nome **main.tf** e apertar enter.



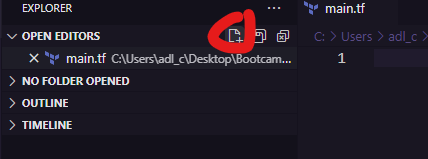


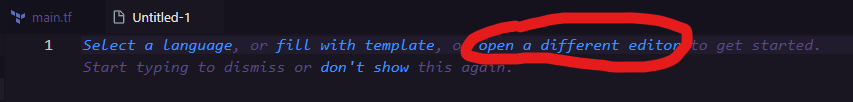
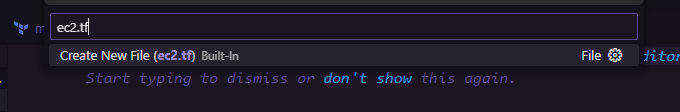
* Aqui vamos salvar o arquivo do terraform, num lugar onde desejamos.

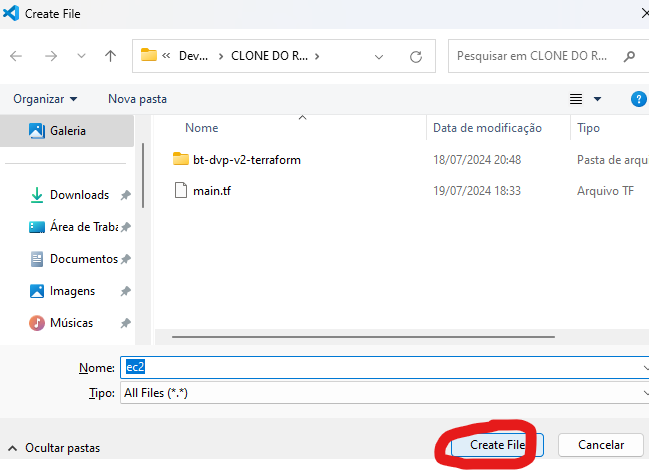




* Pronto, agora vamos fazer outra aba de nome ec2.tf vamos seguir o passo a passo abaixo, como foi feito no main.

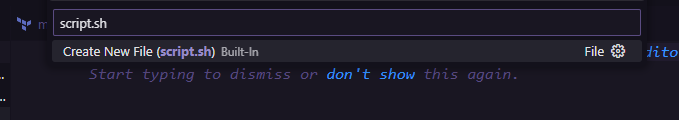
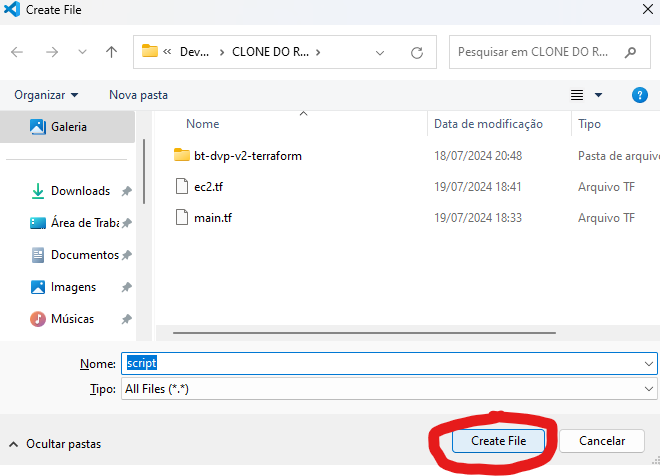


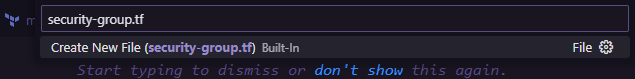


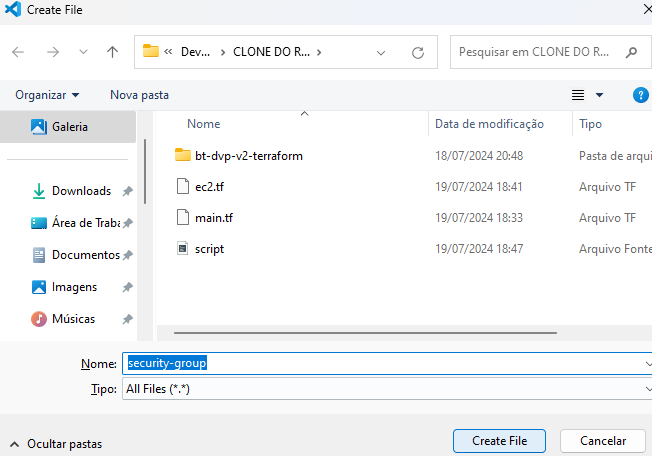
* Pronto agora falta o nosso **script.sh,**

Vamos criar da mesma forma, siga abaixo.

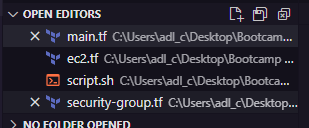
 

* Vamos **criar o ultimo arquivo** o **security-group.tf**, da mesma forma que os outros.





* Como deve ficar nosso VS Code para preenchermos.



* Vamos começar a editar o nosso **script.sh**

Abaixo o meu script.sh (Com explicações comentadas no código para ficar mais claro)

Ou seja, vamos digitar no Vscode:

**- Sudo apt-get uptade**

**- Sudo apt-get upgrade –y**

**- Sudo apt-get install apache2 –y**

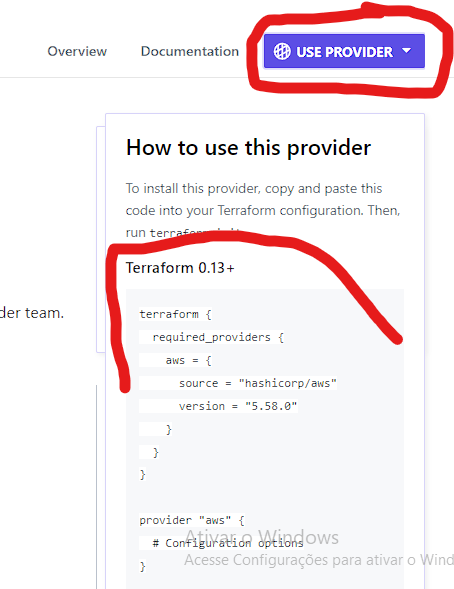
**- Sudo git clone** [**https://github.com/luizcarlos16/site-bootcamp-devops.git**](https://github.com/luizcarlos16/site-bootcamp-devops.git)

**- Sudo cp site-bootcamp-devops/\* -R /var/www/html**



**Pronto, agora vamos salvar com Ctrl S e ir para o próximo.**

* **Editando o** **main.tf**, vamos no site: [registry.terraform.io/providers/hashicorp/aws/latest](https://registry.terraform.io/providers/hashicorp/aws/latest)
* Clicar em **“USE PROVIDER”** e copiar o **Terraform 0.13+**

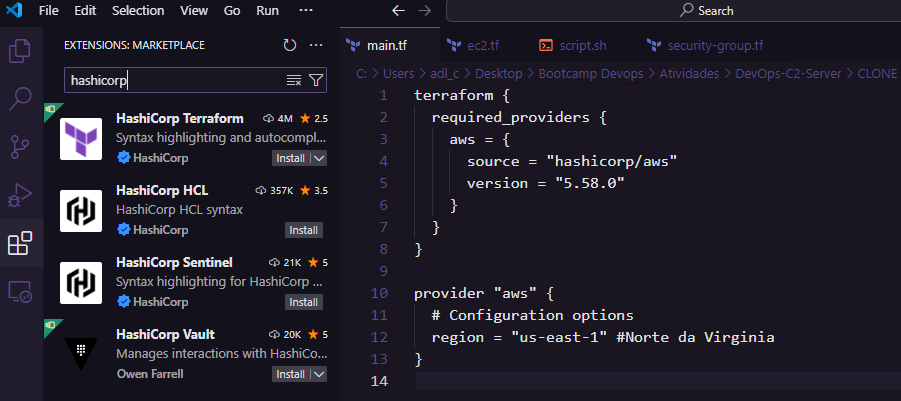


* Com o código copiado vamos colar no **main.tf**, como foi feito abaixo, **adicionei uma linha ao código.**

**A linha 12 não tinha**, acabei inserindo para ter a região que costumo criar e ter EC2. **O código é:**  
**region = “us-east-1” #Norte da Virginia**

****

**OBS: Quando colei o código vi que não tinha a extensão do Terraform no VSCODE. Só procurar e instalar o Hashicorp Terraform.**

****

**Pronto agora é continuar. (Esse é o meu código antes da extensão instalada)**

* Editando agora a **ec2.tf**, a documentação está no link:

[**registry.terraform.io/providers/hashicorp/aws/latest/docs/data-sources/ec2\_host**](https://registry.terraform.io/providers/hashicorp/aws/latest/docs/data-sources/ec2_host)

* **Só preencher como no meu print e abaixo explico o que cada código faz.**

**Explicação do código para ficar didático.**

**resource** “aws\_instance” “adaltec-maquina”

**(O segundo campo onde está “adaltec-maquina” é para dar nome a máquina que vamos criar)**

**Ami** = “ami -0a0e5d9c7acc336f1”

**(Ami é onde vamos mostrar qual sistema operacional queremos para a máquina virtual. Ou seja, a imagem que a gente queria do Ubuntu, a gente coloca o código aqui para ele instalar.)**

**Key\_name = “bootcamp-adalto”**

**(Local onde devemos colocar a Keypairs da aws)**

**Vpc\_security\_groups\_ids = [aws\_security\_group.adaltec-server.id]  
(devemos alterar aqui, de acordo com o nome que temos no security-group)**

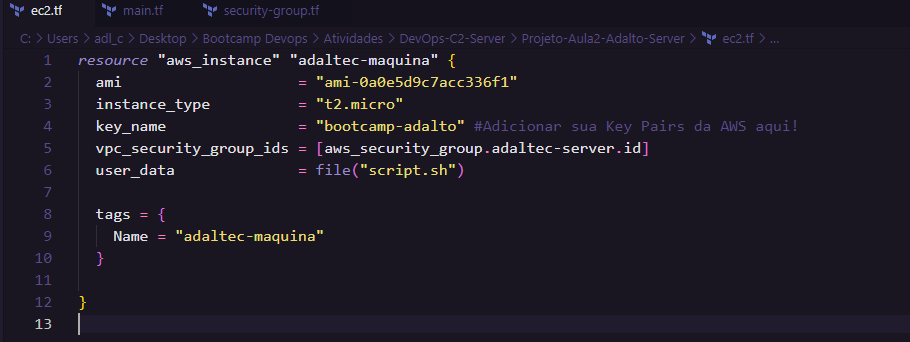
**user\_data = file(“script.sh”)**

**(aqui é onde usamos o sudo de forma automática, instalando o apache e etc.)**

**tags = `{**

**Name = “adaltec-maquina”  
}**

**(Como no primeiro campo aqui literalmente é onde colocamos o nome.)**

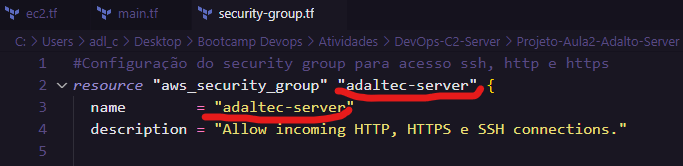
****

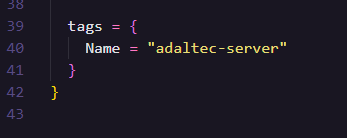
* Última parte de código no VSCODE, editar o **security-group.tf**

Vamos pegar do repositório do professor, e editar para usarmos.

<https://github.com/luizcarlos16/bt-dvp-v2-terraform/blob/main/security-group.tf>

- **Copiar todo código e editar** **as linhas** **1, 2 e 40.**   
(Vamos colocar nestas linhas o que está na linha 5 do ec2.tf, em o marcado em amarelo. [aws\_security\_group.**adaltec-server**.id])

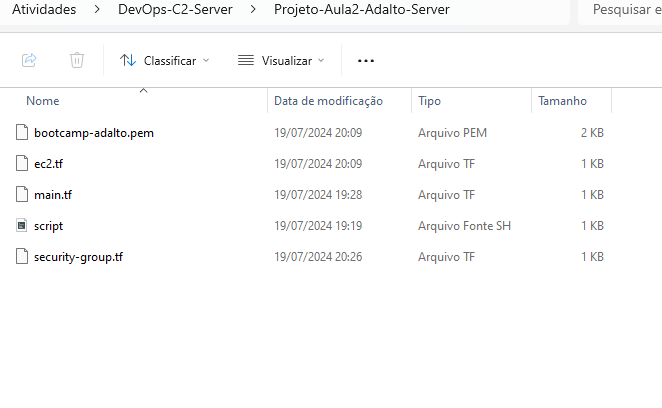




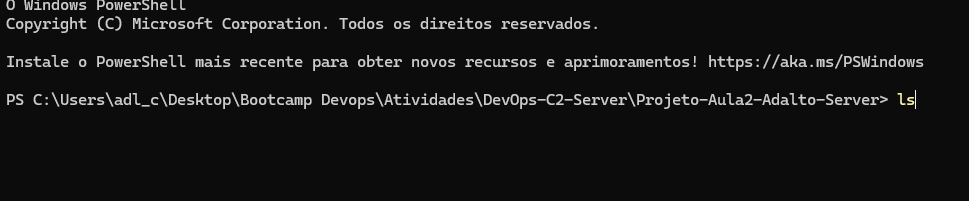
Eu coloquei de adaltec-server, você pode colocar o que bem gostar.

* **Agora vamos até a pasta onde tem os arquivos que criamos no vscode e abrir o cmd.**

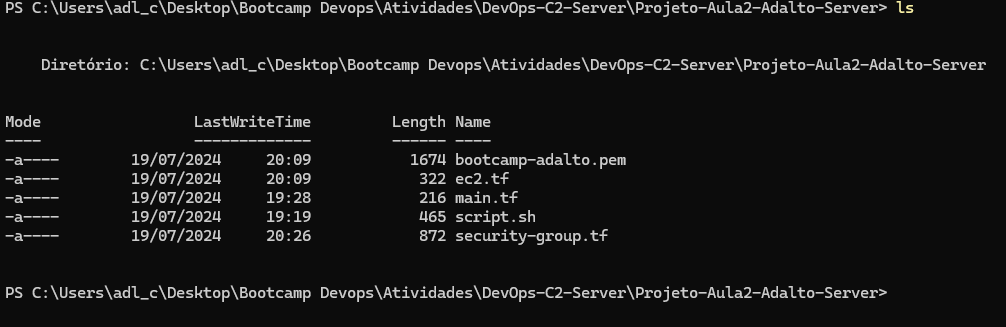
**-** Abaixo estou dentro da pasta, e com o botão direito do mouse vou em **“Abrir no Terminal”**



* No terminal vamos digitar: **ls**

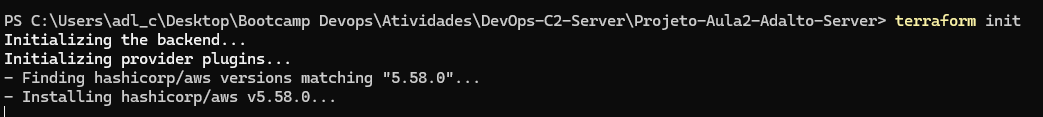


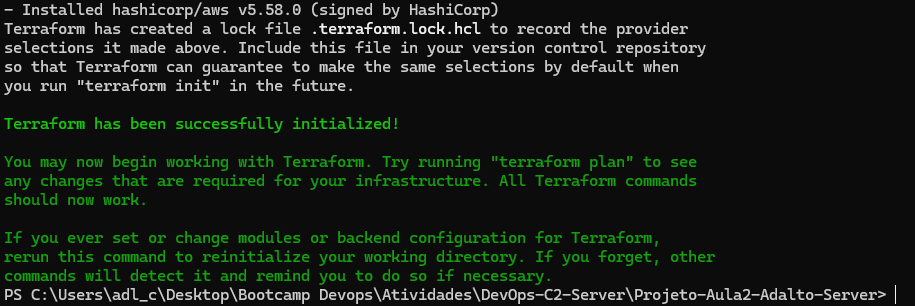
- O comando **ls** é usado para mostrar tudo que tem na pasta.



* Agora vamos iniciar o Terraform, digitando no console:

**terraform init**

****

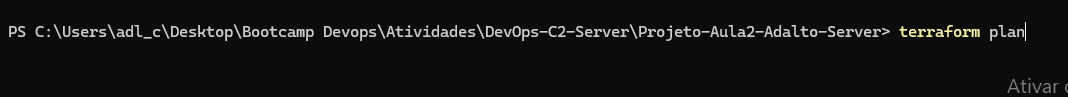
****

**- Digito: ls** no console para ver o que foi criado pelo terraform.

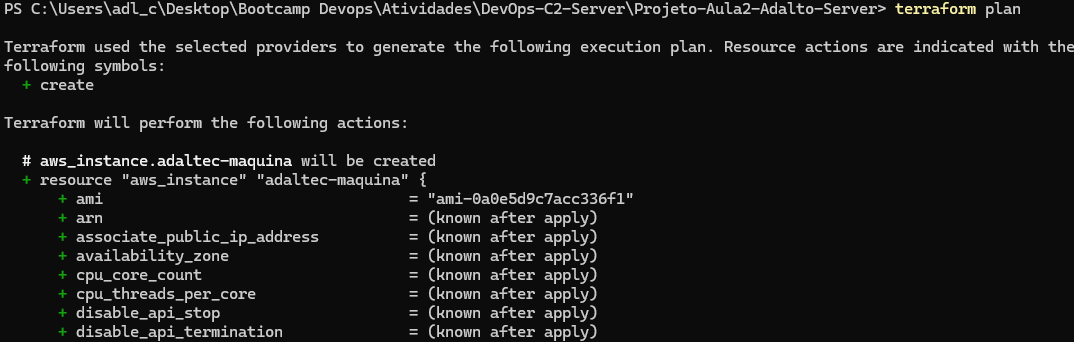
Conseguimos visualizar 2 arquivos.

****

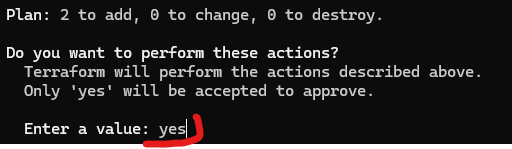
**-** Vamos digitar agora: **terraform plan**

****

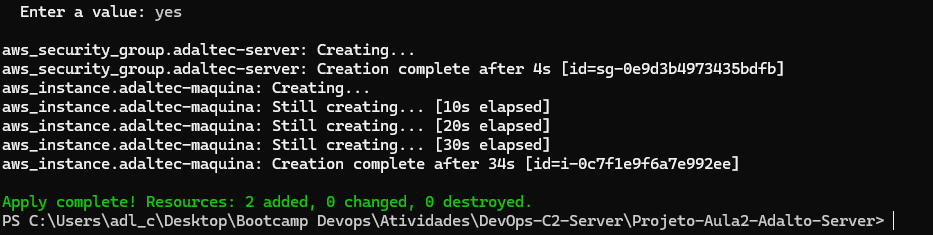
- O terraform plan é para ver o que vai ser feito.



* Vamos para criação, digitando no console: **terraform apply**

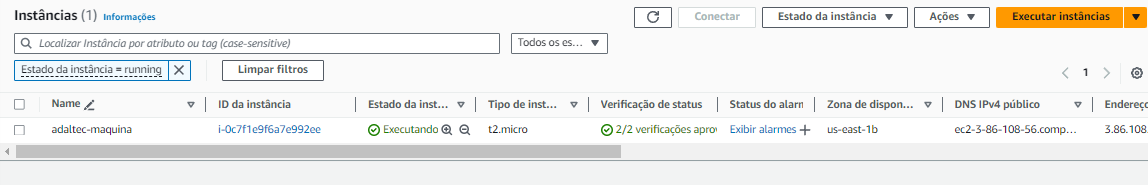


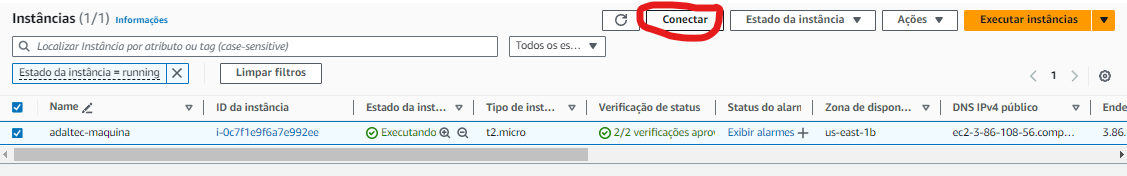
- Agora ele nos pergunta se queremos criar, vamos digitar no console: **yes**



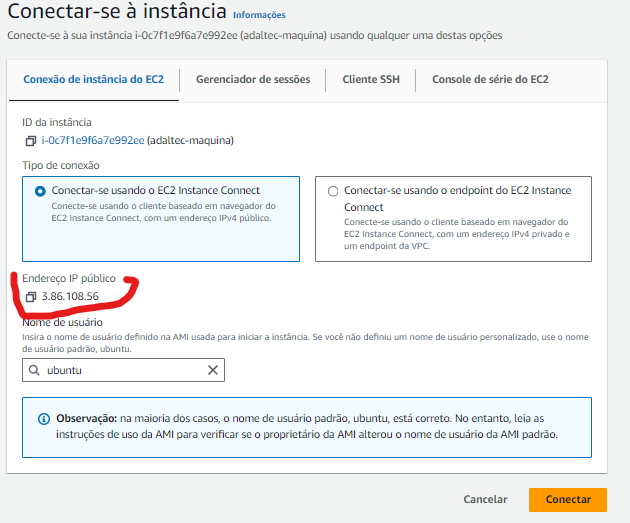
* **Pronto, o terraform fez todo aquele processo manual que faríamos parar criar nossa EC2.**
* **Vamos visualizar na AWS nossa nova instancia e acessar para ver o projeto rodando.**

**1° Passo - ir no site da AWS no console EC2 para visualização da nossa máquina virtual e selecionar a maquina.**

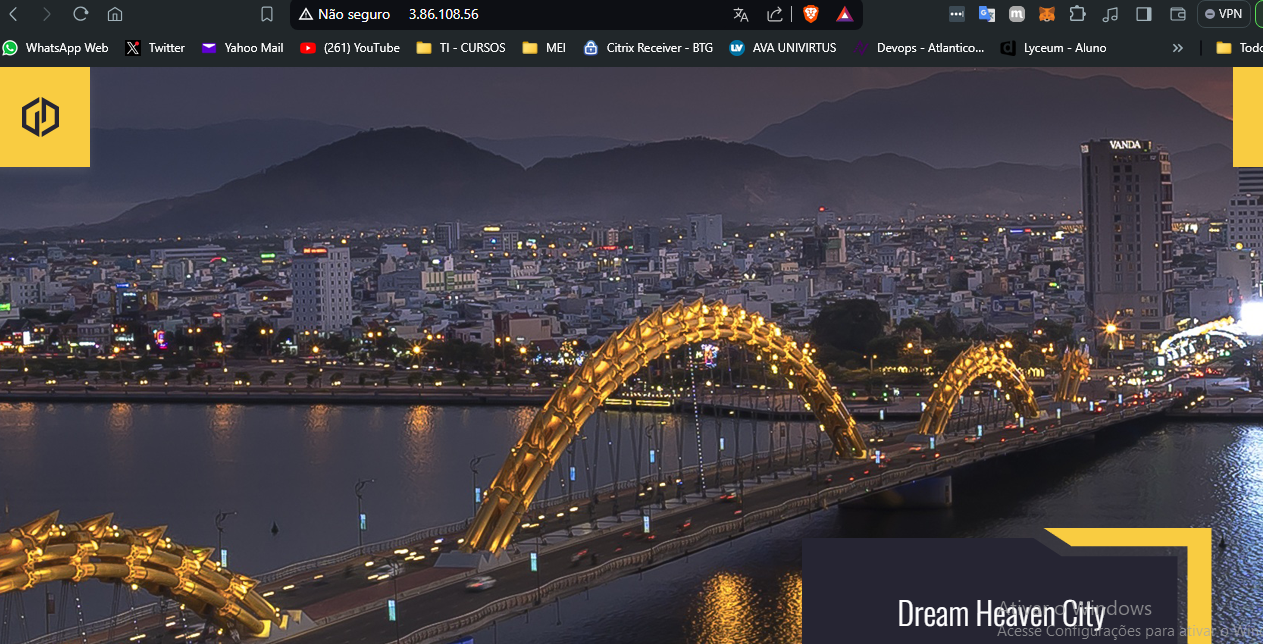
****

****

**2° Passo – pegar o ip público e abrir no navegador.**

****

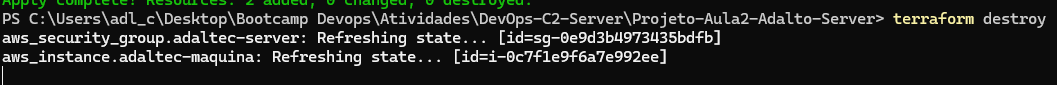
**- Site rodando no navegador com o IP Publico.**

****

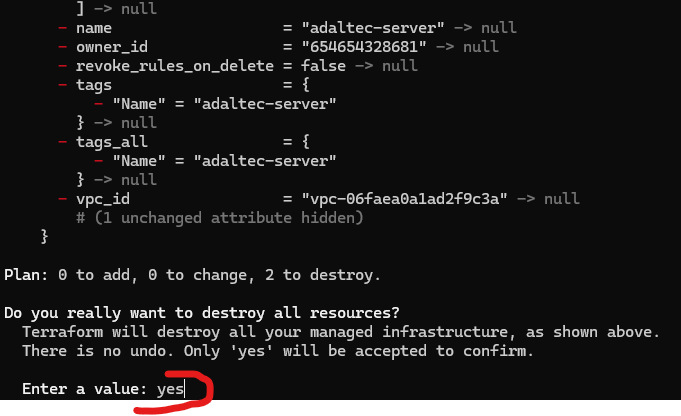
* **Ainda não acabou, vamos aprender a usar o Terraform Destroy.**

(Para a aws não cobrar a gente pelo servidor rodando !!)

- No console vamos digitar: **terraform destroy**

****

**- digitar: yes**

****

**Fim !!!**