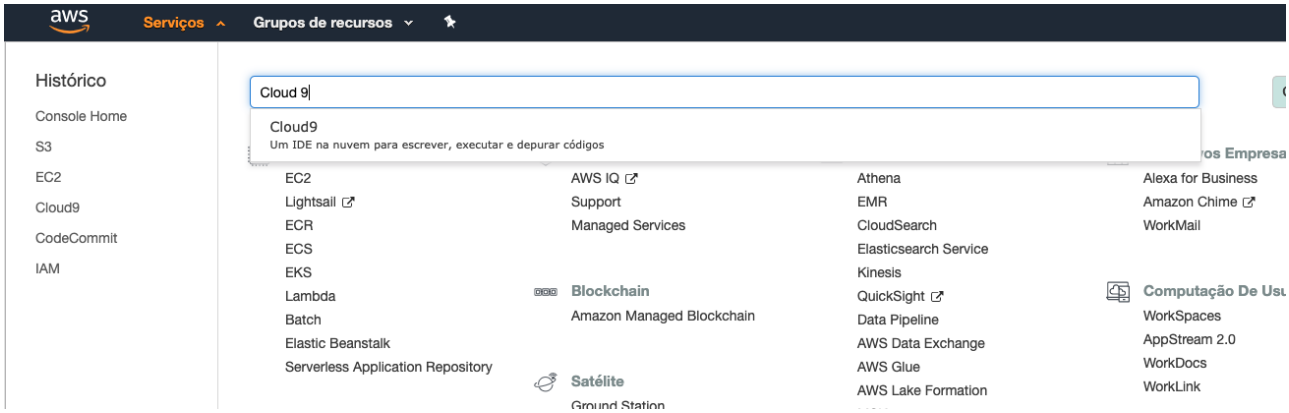
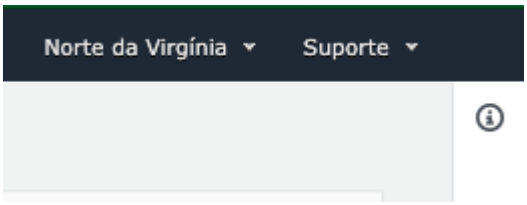
01 - Setup e Configuração de ambiente

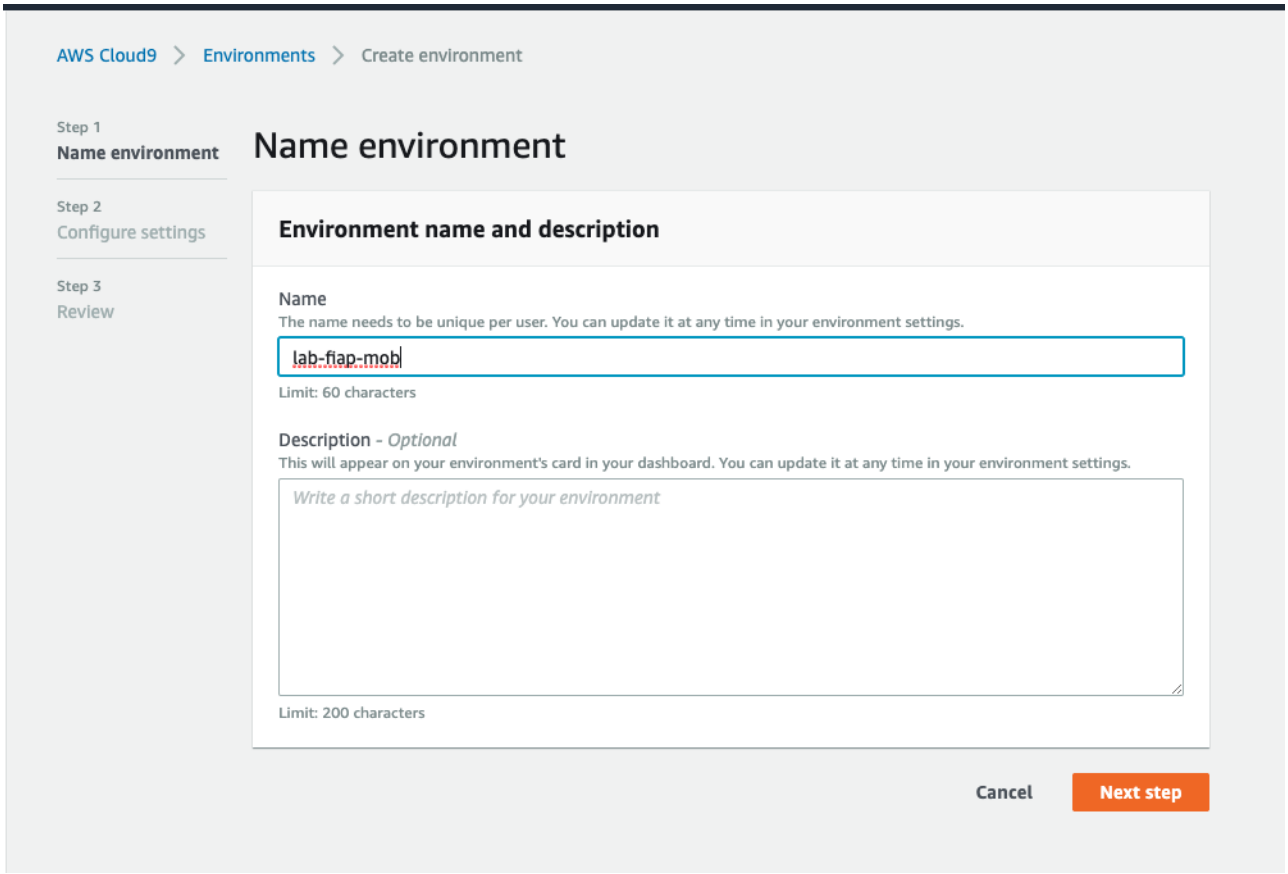
1. Abra o console da AWS e va para o serviço `Cloud 9`.



2. garanta que a região que esta utilizando é `us-east-1/ Norte da Virgínia`. Você consegue ver isso no canto superior direiro da tela. 

3. Clique em `create environment`.

4. Coloque o nome `lab-fiap` e avance.



5. Deixe as configurações como na imagem a seguir. Se atente ao tipo da maquina que deve ser t2.medium:

Configure settings

Environment settings

Environment type **Info**

Choose between creating a new EC2 instance for your new environment or connecting directly to your server over SSH.

- (•) **Create a new instance for environment (EC2)**
  Launch a new instance in this region to run your new environment.
- ( ) **Connect and run in remote server (SSH)**
  Display instructions to connect remotely over SSH and run your new environment.

Instance type

- ( ) **t2.micro (1 GiB RAM + 1 vCPU)**
  Free-tier eligible. Ideal for educational users and exploration.
- ( ) **t3.small (2 GiB RAM + 2 vCPU)**
  Recommended for small-sized web projects.
- ( ) **m5.large (8 GiB RAM + 2 vCPU)**
  Recommended for production and general-purpose development.
- (•) **Other instance type**
  Select an instance type.

t2.medium ▼

Platform

- ( ) **Amazon Linux**
- (•) **Ubuntu Server 18.04 LTS**

Cost-saving setting

Choose a predetermined amount of time to auto-hibernate your environment and prevent unnecessary charges. We recommend a hibernation settings of half an hour of no activity to maximize savings.

After 30 minutes (default) ▼

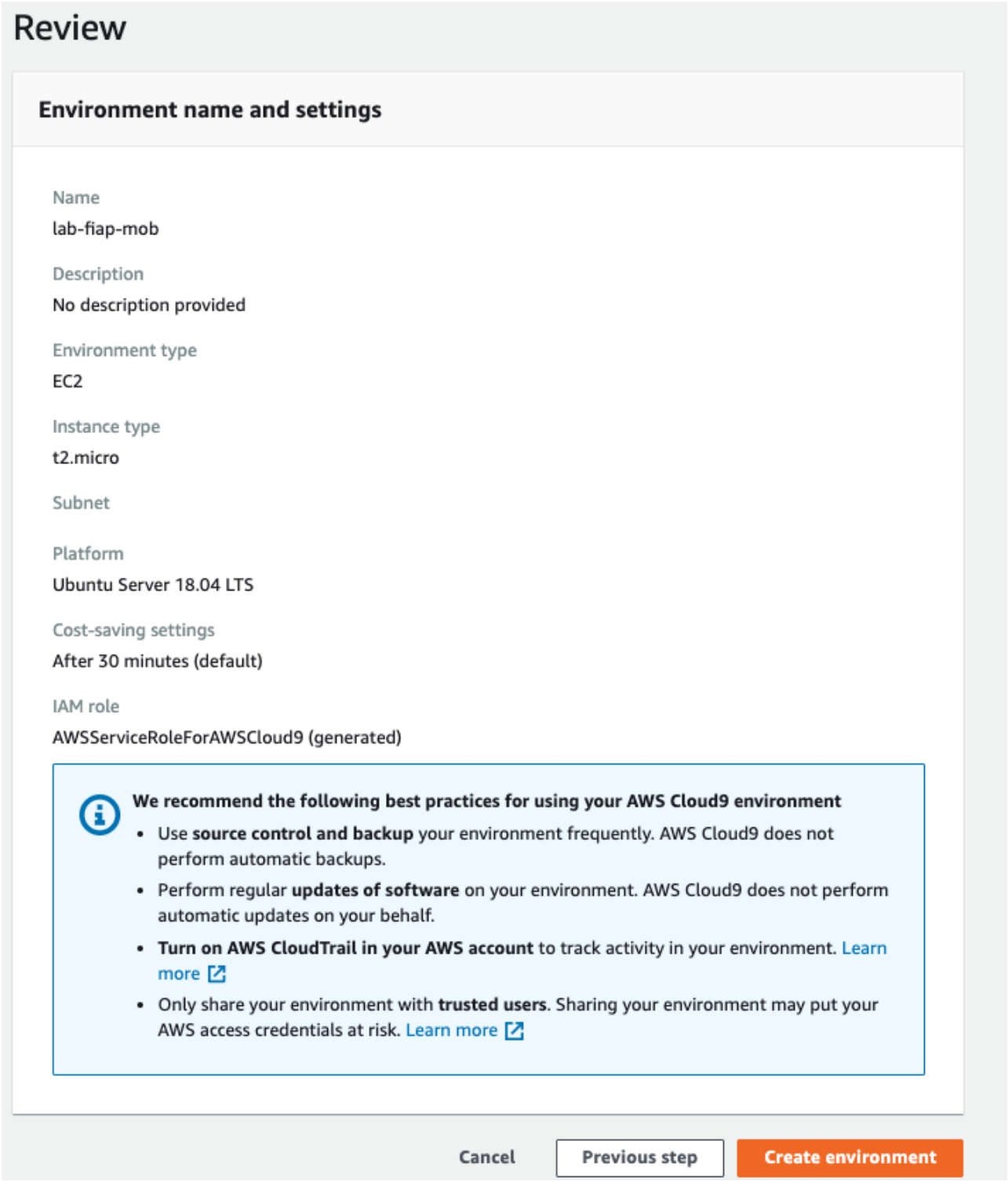
IAM role

AWS Cloud9 creates a service-linked role for you. This allows AWS Cloud9 to call other AWS services on your behalf. You can delete the role from the AWS IAM console once you no longer have any AWS Cloud9 environments. **Learn more**

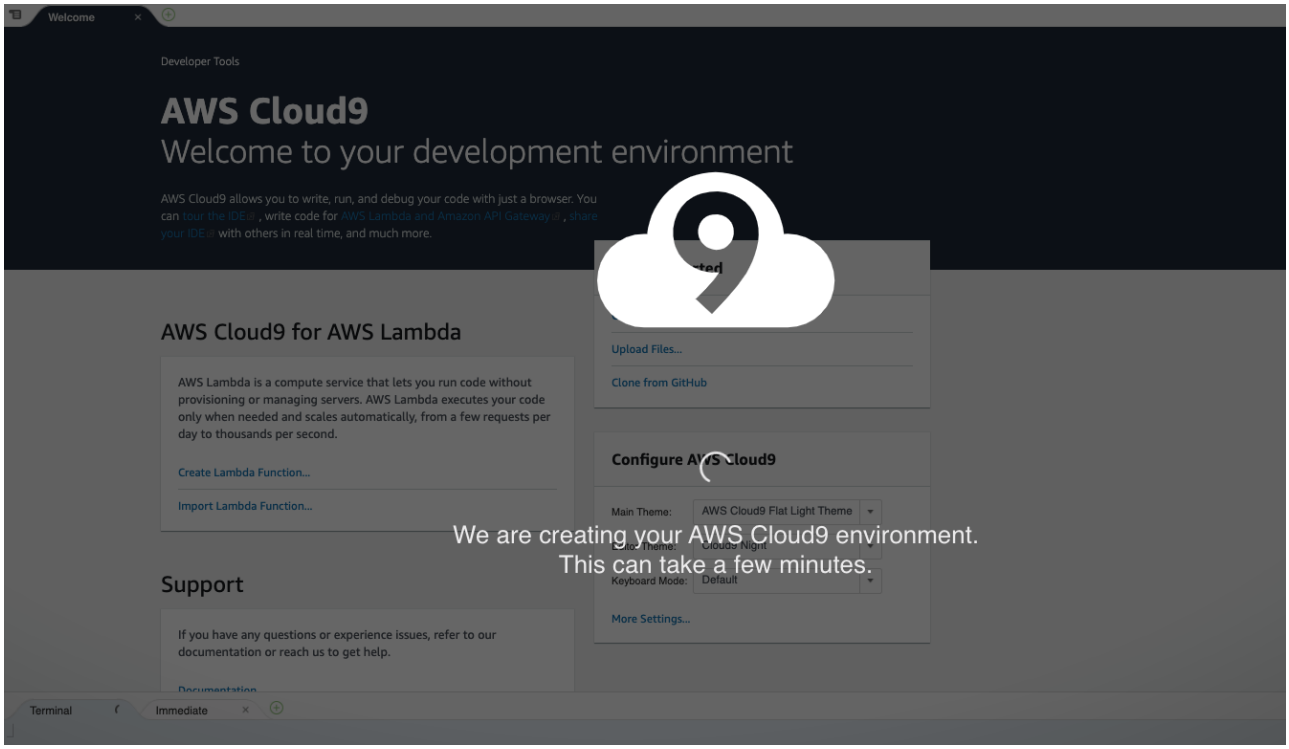
AWSServiceRoleForAWSCloud9

▶ **Network settings (advanced)**

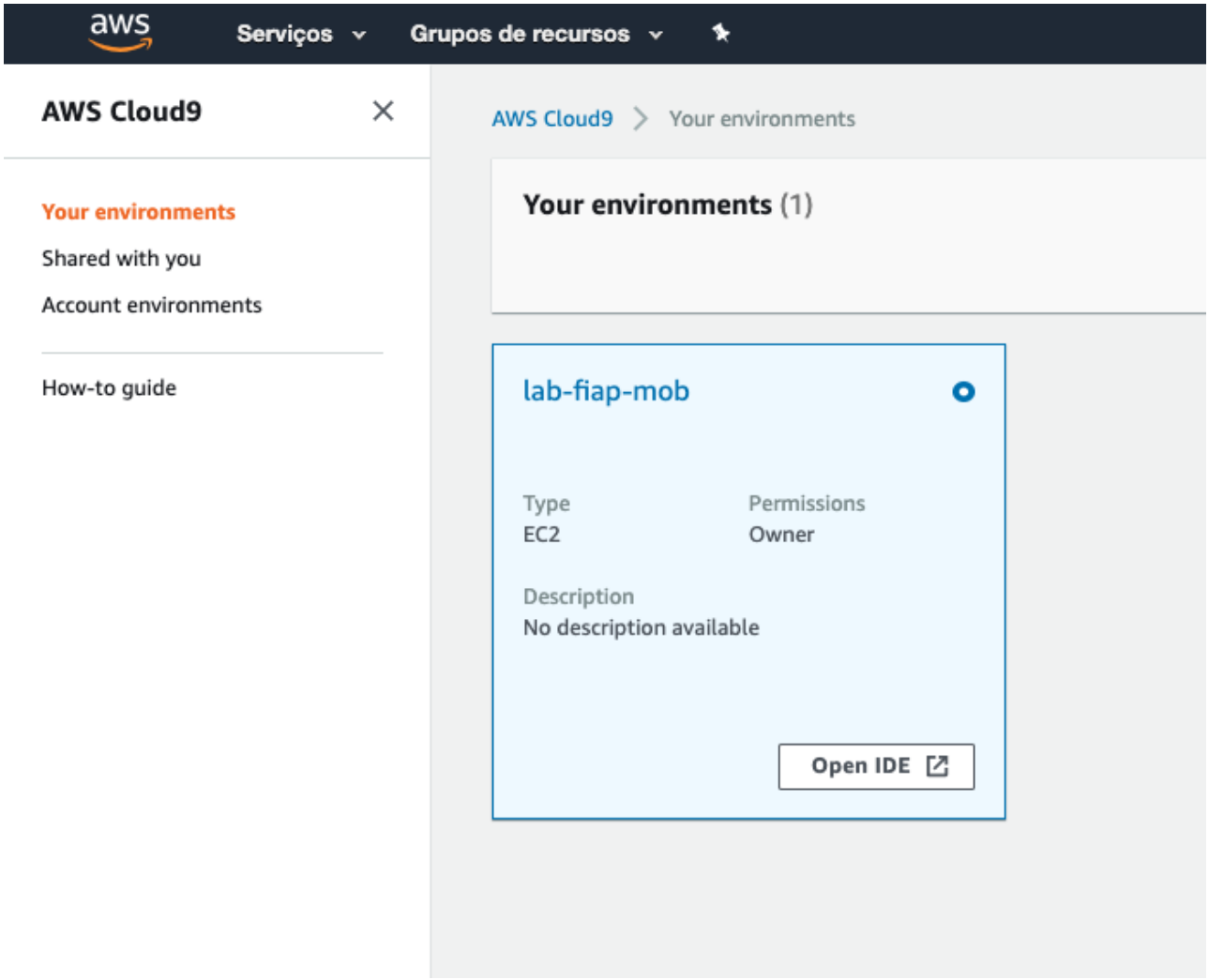
6. Caso os parametros estejam como na imagem a seguir clique em `Create Environment`



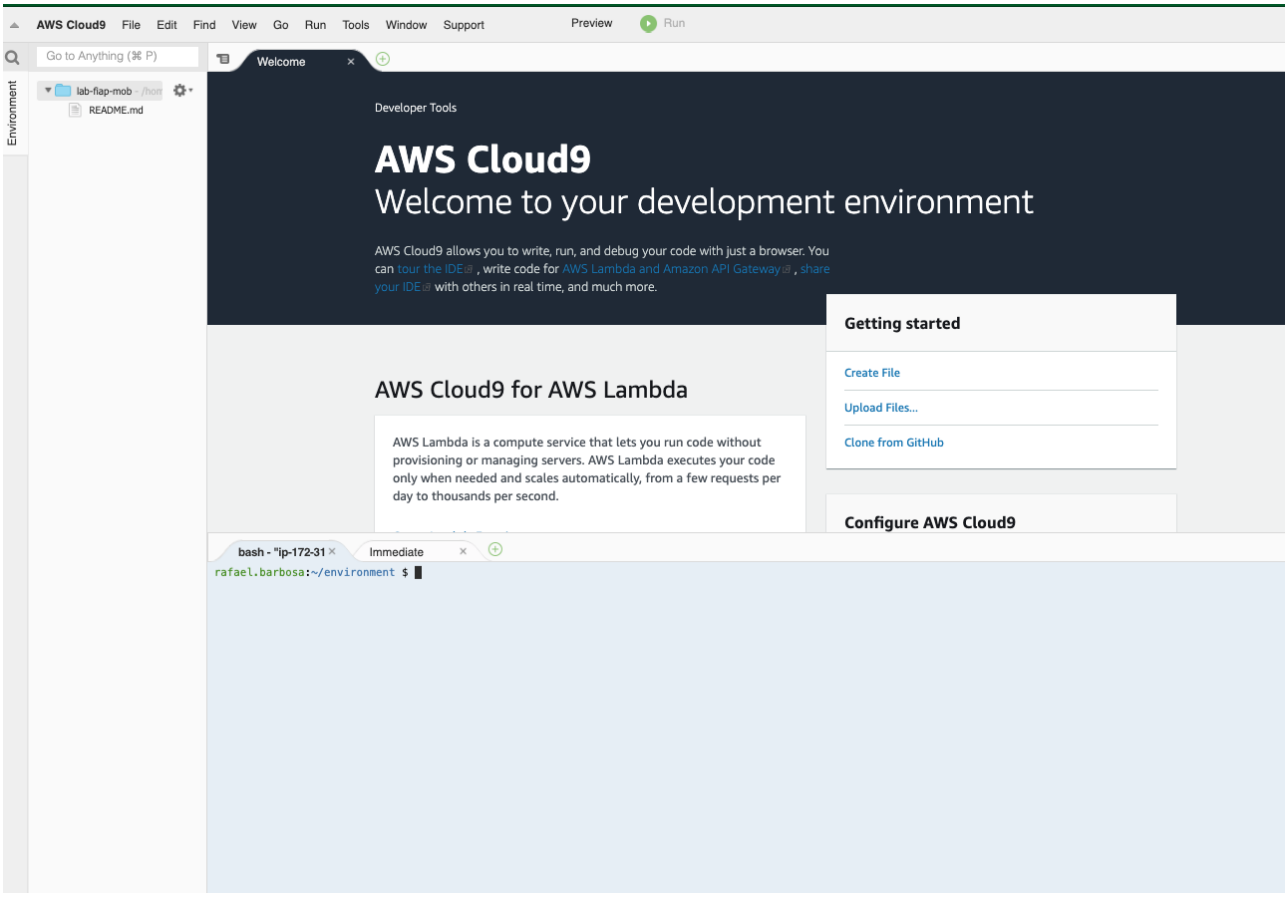
7. A criação do ambiente pode levar alguns minutos.



8. Após a criação clique em `abrir IDE`, caso o IDE não tenha aberto automaticamente.



9. Para os próximos comandos utilize o console bash que fica no canto inferior do seu IDE.



10. Execute o comando `npm install -g serverless` para instalar o serverless framework.

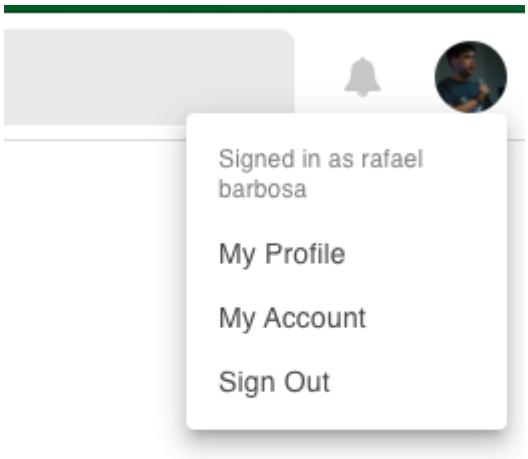
11. Execute o comando `sudo apt install jq -y` para instalar o software que irá nos ajudar a ler e manipular Jsons no terminal

12. Para utilizar o SDK em python da AWS instale com o comando `pip3 install boto3`

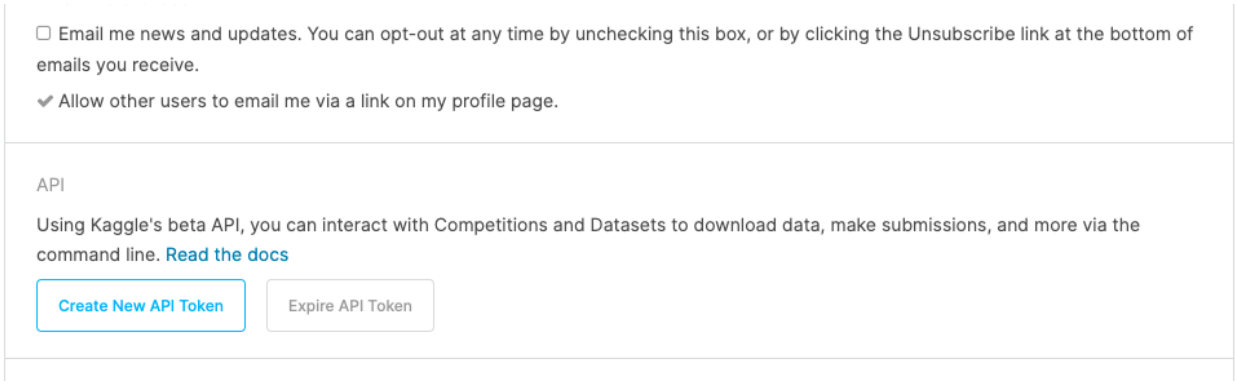
13. Iremos utilizar um dataset do kaggle para fazer o bootcamp, instale o sdk da api para conseguir fazer o download via terminal `pip install kaggle`

14. Em outra aba do navegador entre na sua conta do kaggle - login

15. No canto direito superior clique na sua foto e clique em 'My account':



16. Desça a tela até a opção API e clique em 'Create new API Token'



17. Um arquivo, kaggle.json ,será baixado para sua maquina, abra o arquivo e copie o conteúdo.

18. De volta ao terminal do Cloud9, execute o comando `mkdir -p ~/.kaggle`

19. Execute o comando abaixo copiando o conteudo do kaggle.json que baixou e colocando entre aspas simples como na imagem:

```
echo 'SEU JSON USER COM TOKEN' >> ~/.kaggle/kaggle.json
```

20. Execute o comando `chmod 600 ~/.kaggle/kaggle.json` para colcoar a permissão certa no token

21. Vamos criar o lugar onde ficarão os dados na maquina. Para isso crie a pasta com o comando `mkdir ~/environment/seattle-library-collection-inventory` e entre nela `cd ~/environment/seattle-library-collection-inventory`

22. Vamos baixar o dataset na maquina com o comando `kaggle datasets download -d city-of-seattle/seattle-library-collection-inventory`

23. AO termino do download execute o comando `unzip seattle-library-collection-inventory.zip` para descompactar o conteúdo.

24. Apague o zip que baixou e já não é mais necessário com comando `rm seattle-library-collection-inventory.zip`