**Reinicia la terminal:** reset

**Eliminar Stack en Pulumi**

pulumi stack ls

pulumi stack select <nombre-del-stack>

pulumi destroy

pulumi stack rm <nombre-del-stack>

**Estructura:**

1. Crear una carpeta oculta:

mkdir .aws

1. Crear un archivo llamado credentials y pegar las credenciales de aws.

pico credentials

Guardar con ctrl + o. Luego, enter

Salir ctrl + x para salir

1. Crear una carpeta llamado pulumi-mv

mkdir pulumi-mv

1. Ejecutar el siguiente comando

sudo apt update

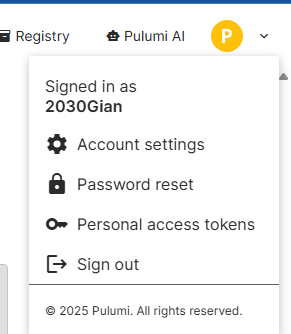
sudo apt install python3-venv

1. En caso de que no quiera inicializar el nuevo proyecto de Pulumi en Python ejecutar:

pulumi export PULUMI\_ACCESS\_TOKEN="<tu-token>"

aws-python --name crear-mv --force

El token se obtiene creando un **personal access token**

****

Para realizar esto primero tienes que crear una cuenta en Pulumi.

1. Instalar pip

sudo apt install python3-pip

1. Instalar aws CLI

sudo apt-get update

sudo snap install aws-cli –classic

**Paso 1: Instalación de Pulumi en ubuntu**

curl -fsSL https://get.pulumi.com | sh

Una vez realizado el paso 1, tienes dos opciones:

1. Cierra tu terminal de la máquina virtual
2. Añadir pulumi al PATH para que puedas usarlo directamente.

**Añadir Pulumi al PATH**

export PATH=$PATH:$HOME/.pulumi/bin

**Paso 2: Inicializa un nuevo proyecto Pulumi en Python**

pulumi new aws-python --name crear-mv --force

Esto crea Pulumi.yaml, requirements.txt, y el esqueleto en \_\_main\_\_.py.

Luego, activar el entorno virtual

source venv/bin/activate

Luego:

pip install -r requirements.txt

1. **Paso 3: Ejecutar el despliegue**

Vista previa de cambios

pulumi preview

Esto te mostrará qué recursos se crearán sin aplicarlos aún

pulumi up --yes

Pulumi creará tu Security Group y la MV en AWS

1. **Paso 4: Eliminar la MV**
   1. Obtén el ID de la instancia

pulumi stack output instance\_id

* 1. Teminar la instancia

aws ec2 terminate-instances --instance-ids <“Colocar el ID de la instacia ”>