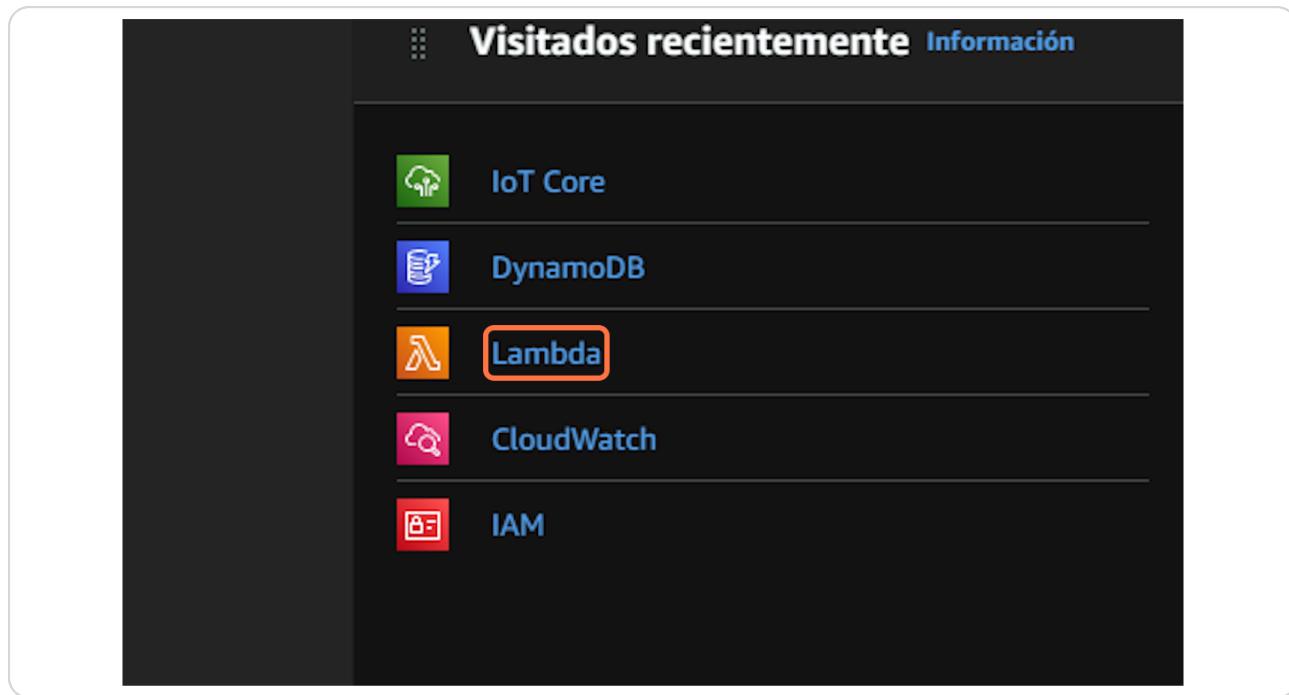


Go to <https://us-east-2.console.aws.amazon.com/console/home?region=us-east-2>

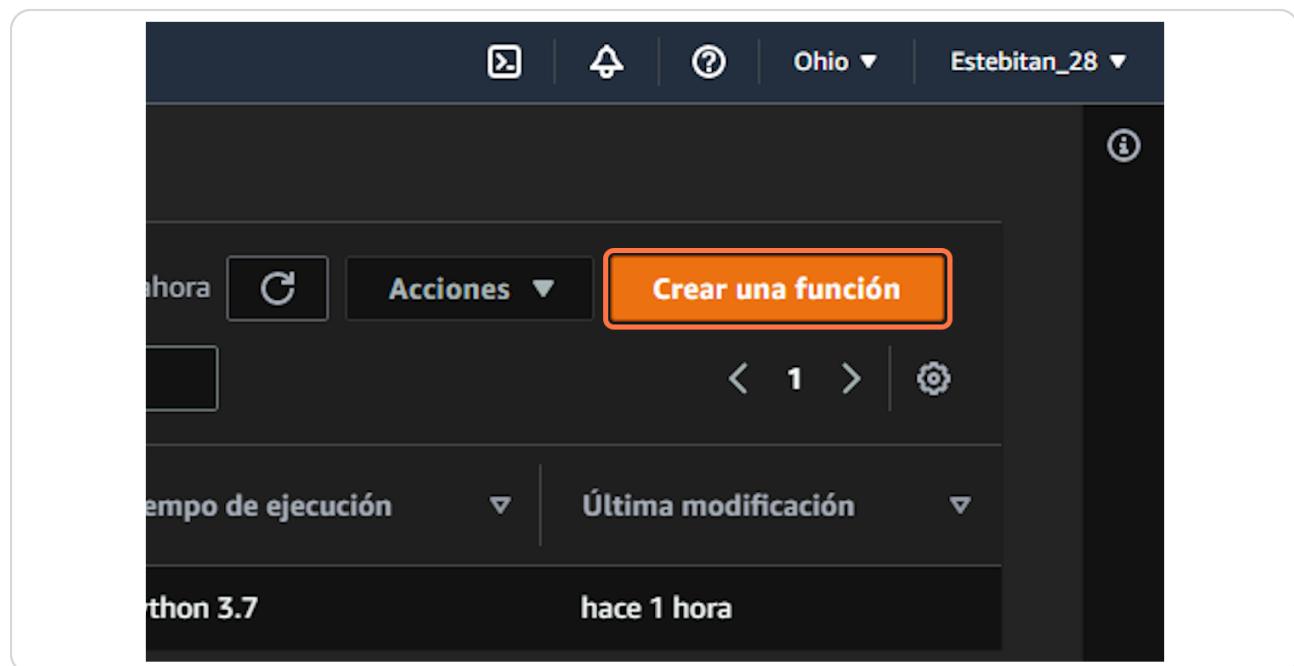
STEP 1

Haga clic en "Lambda"



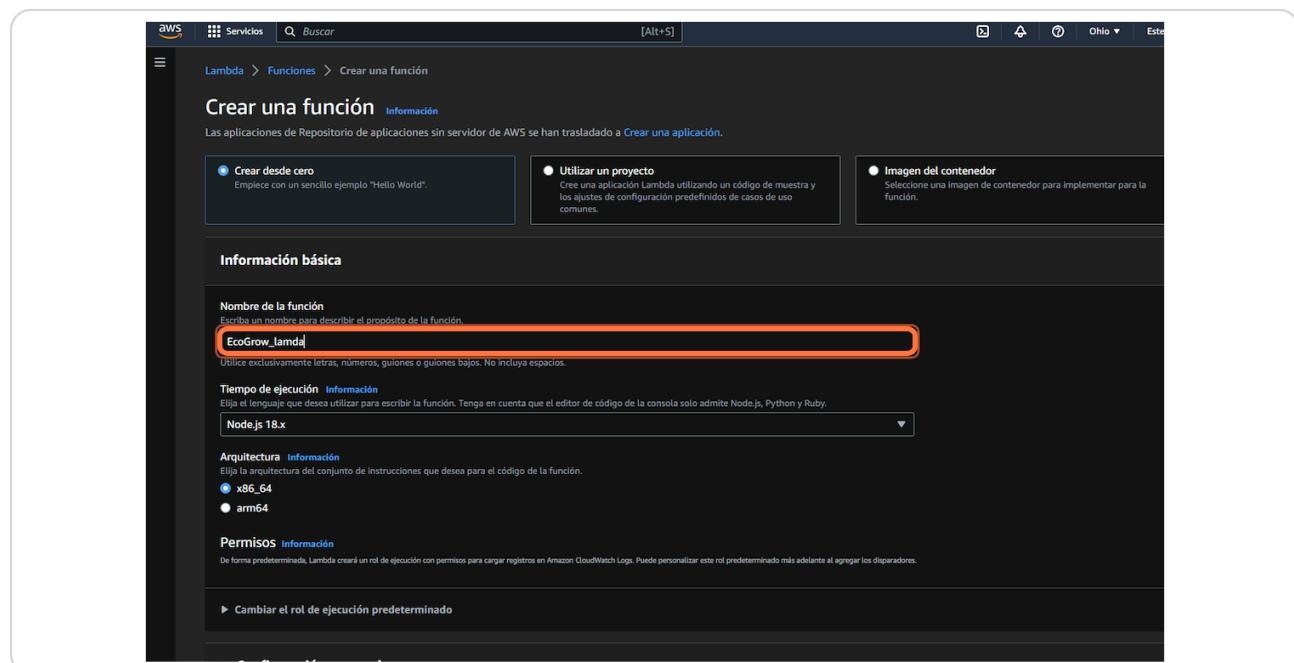
STEP 2

Haga clic en "Crear una función"



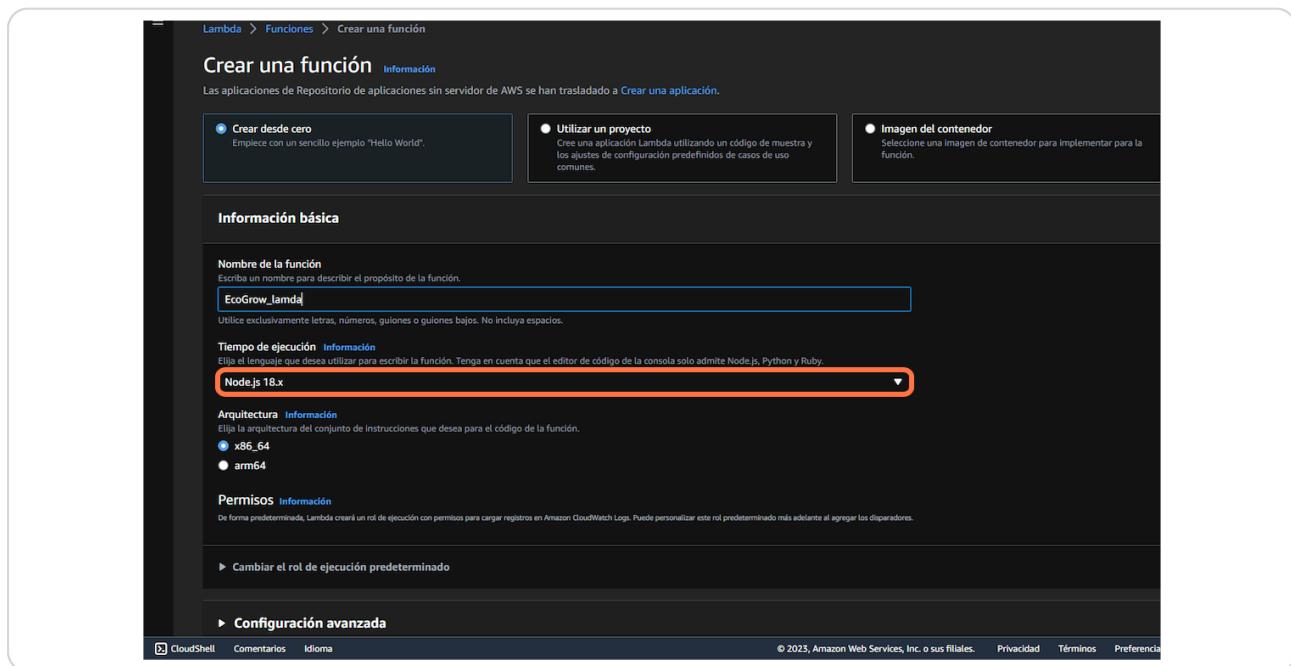
STEP 3

Ingrese como tipo "EcoGrow_lambda"



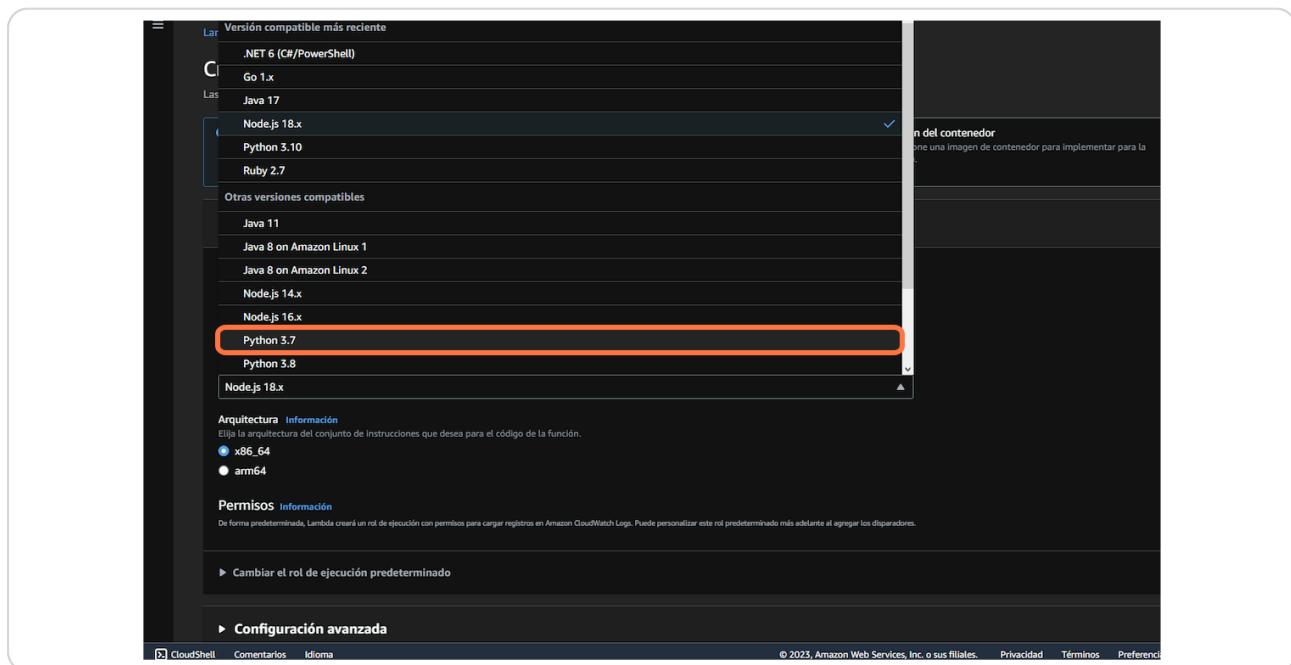
STEP 4

Haga clic en "Node.js 18.x"



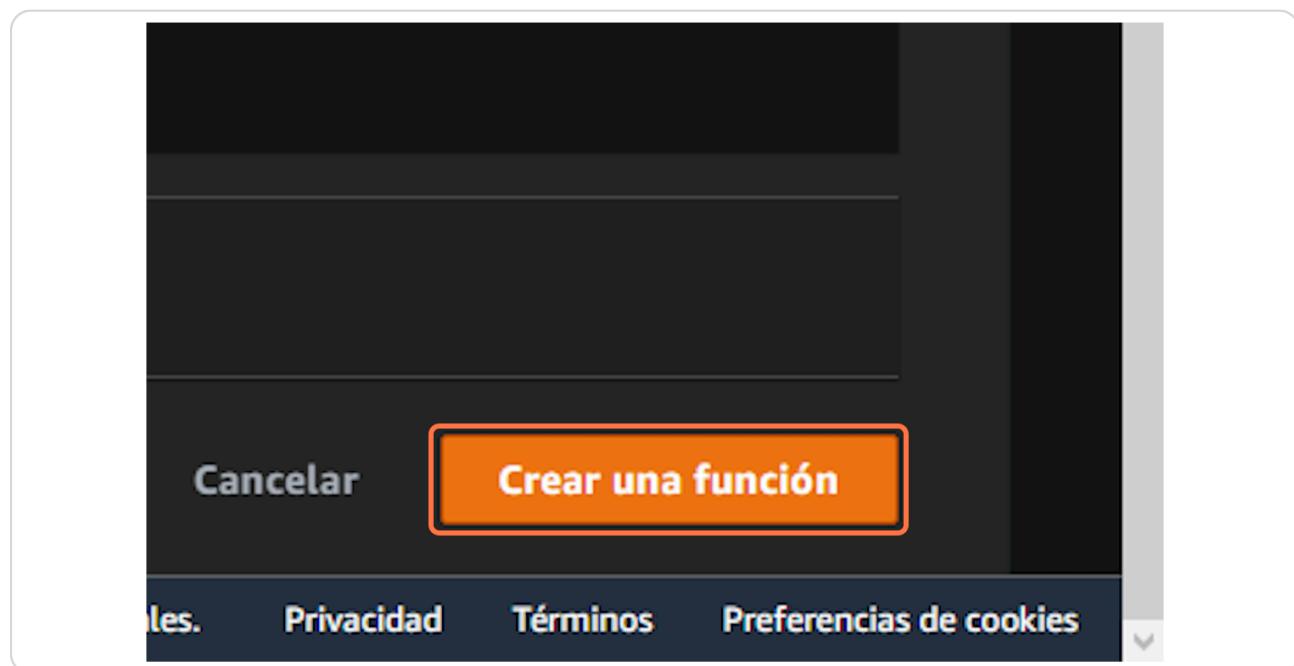
STEP 5

Haga clic en "Python 3.7"



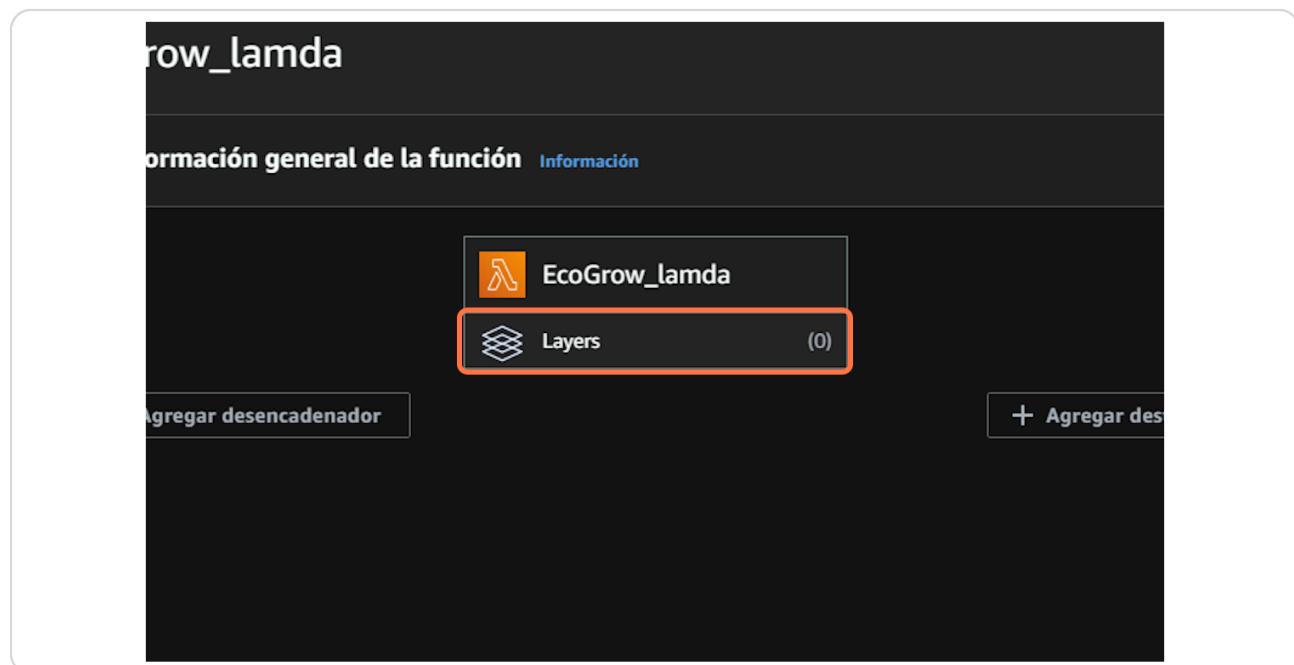
STEP 6

Haga clic en "Create a function"



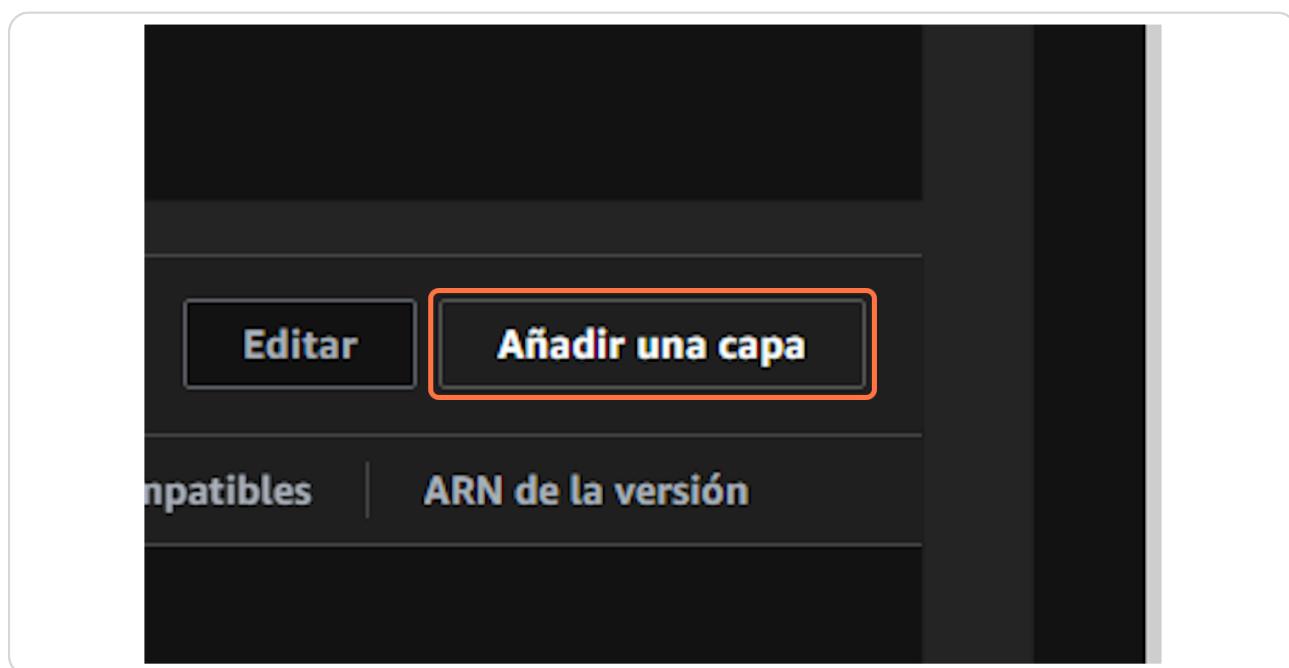
STEP 7

Haga clic en "Layers"



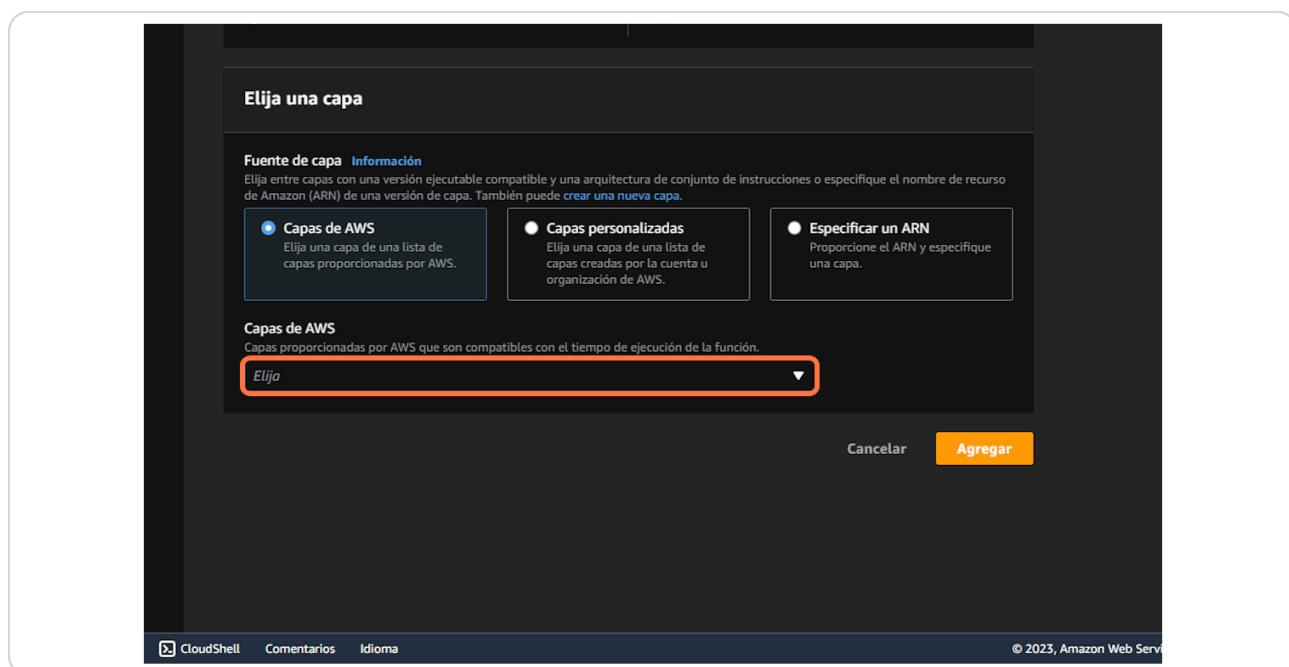
STEP 8

Haga clic en "Añadir una capa"



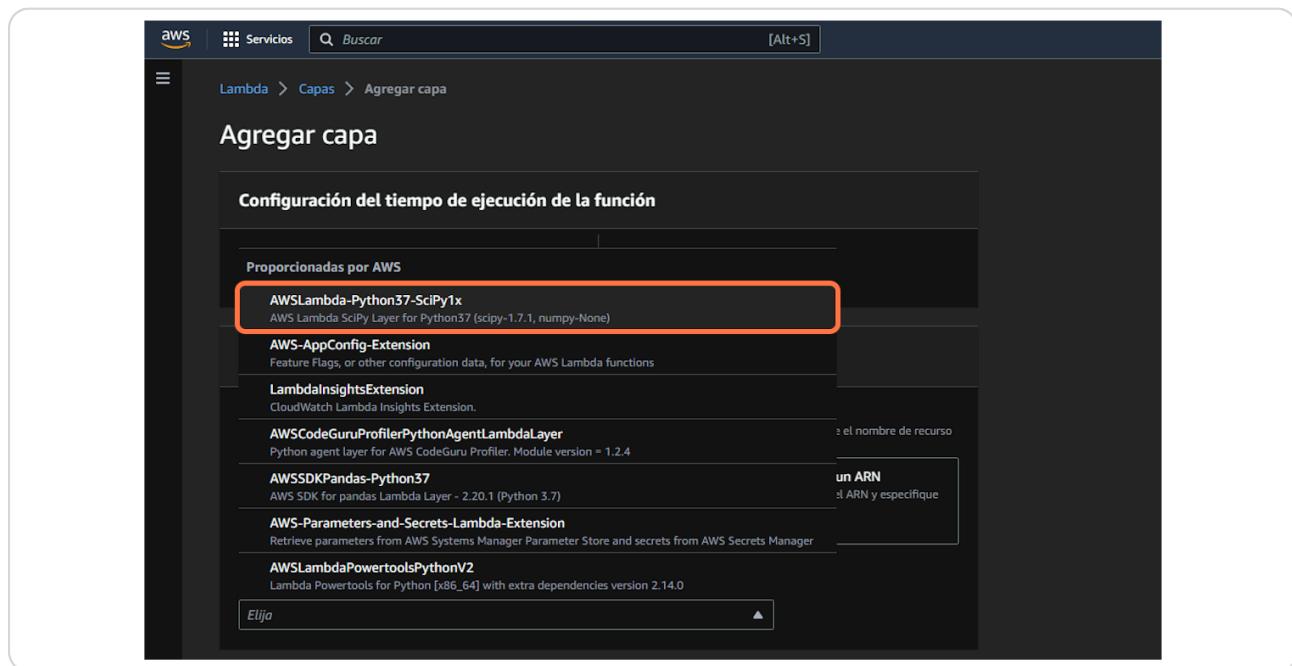
STEP 9

Haga clic en "Elija"



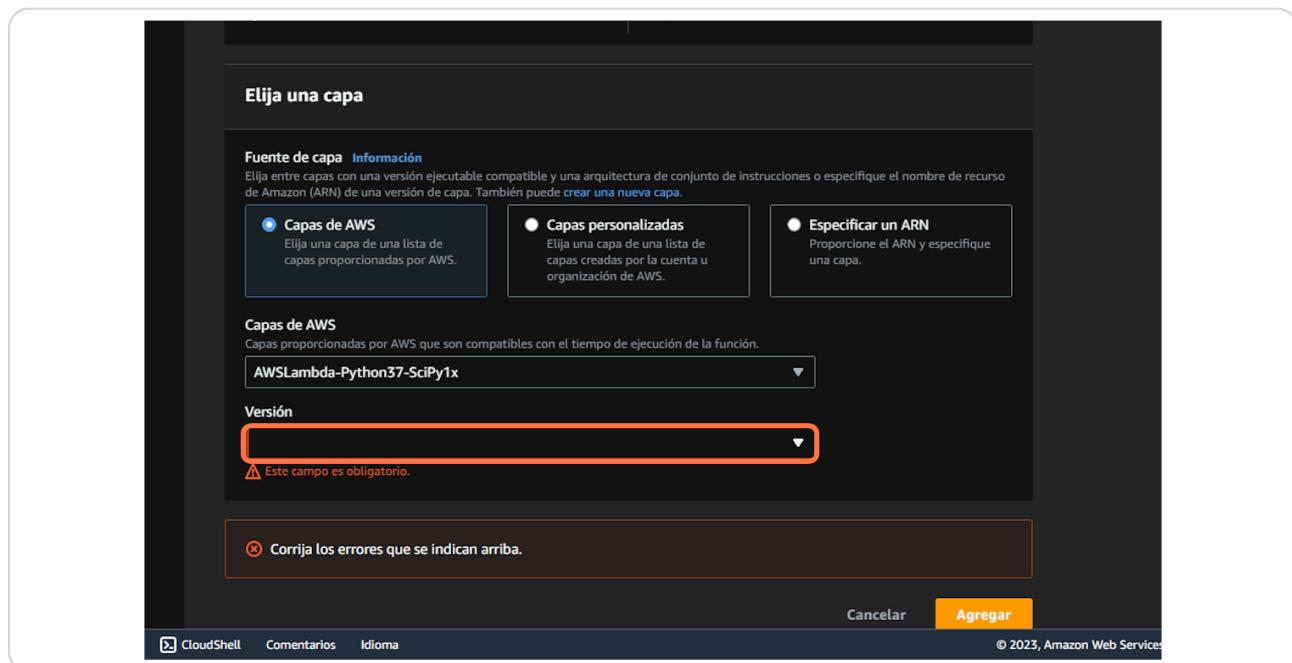
STEP 10

Haga clic en "AWSLambda-Python37-SciPy1x..."



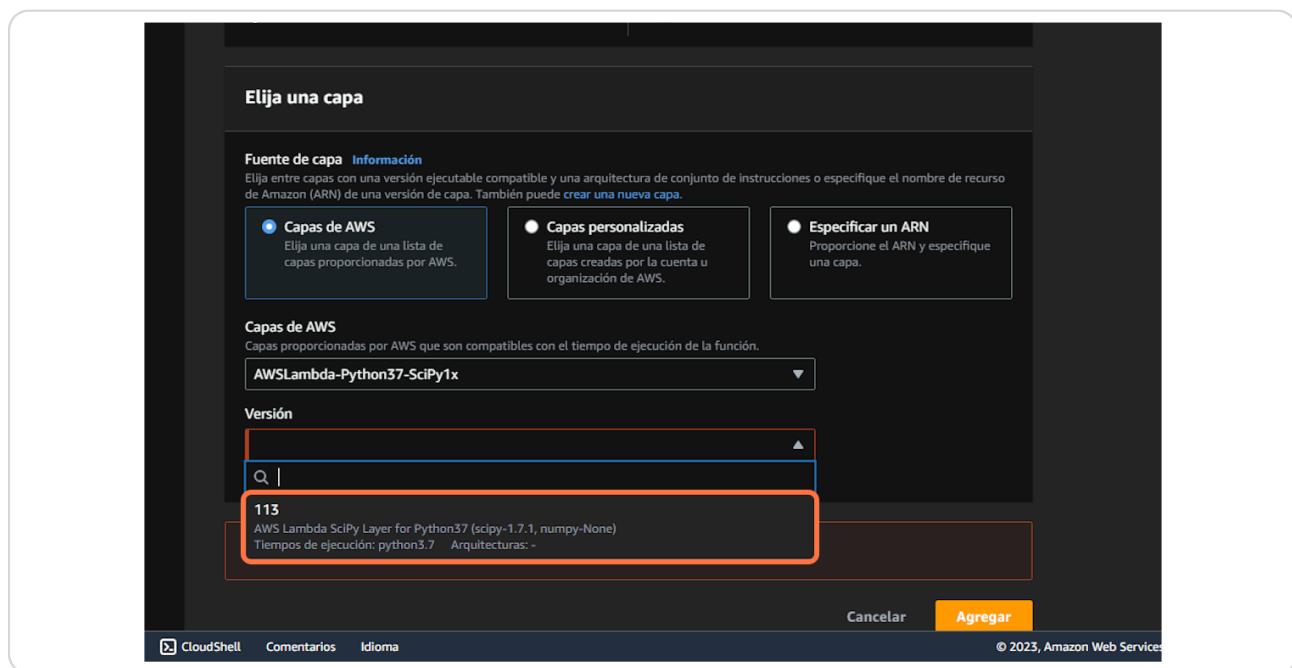
STEP 11

Haga clic en "Versión"



STEP 12

Haga clic en "113..."



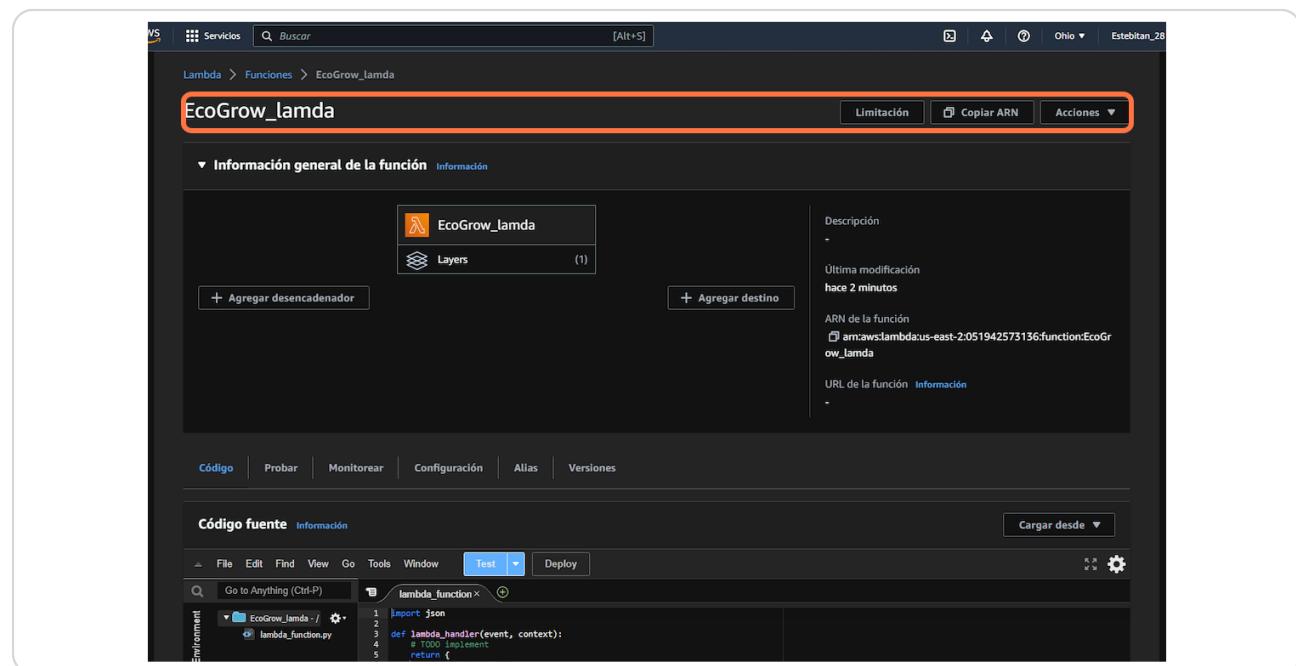
STEP 13

Haga clic en "Agregar"



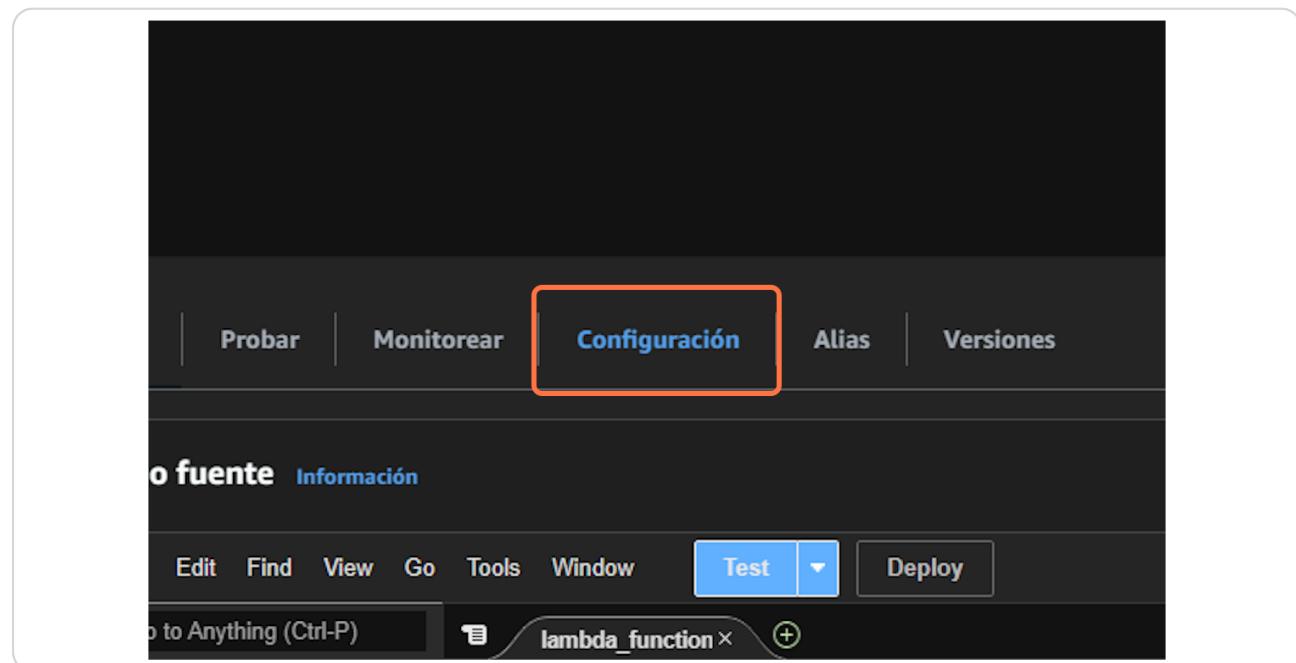
STEP 14

Haga clic en "EcoGrow_lamda..."



STEP 15

Haga clic en "Configuración"



STEP 16

Haga clic en "Permisos"

The screenshot shows the AWS Lambda function configuration interface. The top navigation bar includes tabs for Código, Probar, Monitorear, Configuración, Alias, and Versiones. The 'Configuración' tab is active. On the left, a sidebar lists several sections: Configuración general, Desencadenadores, Permisos (which is highlighted with a red box), Destinos, URL de la función, Variables de entorno, and Etiquetas. The main content area is titled 'Configuración general' and contains fields for Descripción (empty), Tiempo de espera (set to 0 min 3 s), and a large text input field below it.

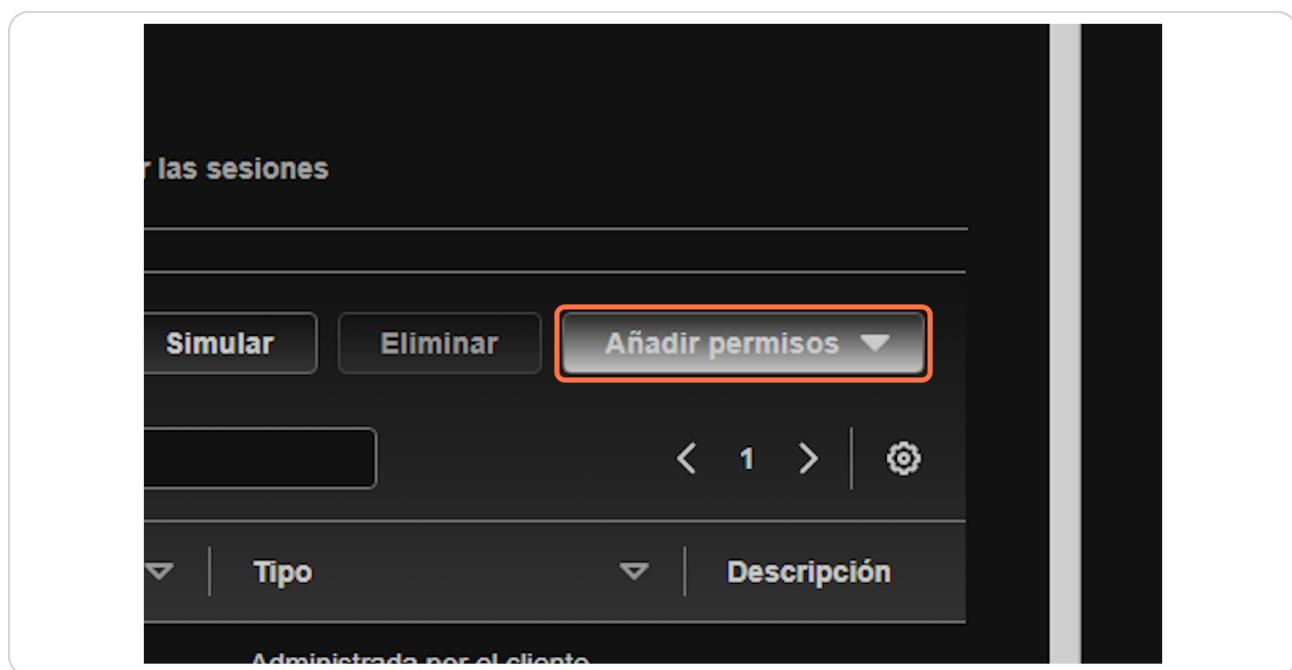
STEP 17

Haga clic en "EcoGrow_lambda-role-tkkrt9ho"

The screenshot shows the AWS Lambda function configuration interface. The top navigation bar includes tabs for Código, Probar, Monitorear, Configuración, Alias, and Versiones. The 'Configuración' tab is active. On the left, a sidebar lists several sections: Configuración general, Desencadenadores, Permisos (which is highlighted with a red box), Destinos, URL de la función, Variables de entorno, Etiquetas, and VPC. The main content area is titled 'Rol de ejecución' and displays the 'Nombre del rol' field, which contains the value 'EcoGrow_lambda-role-tkkrt9ho' with a blue selection highlight around it. Below this, there is a 'Resumen de recursos' section showing an icon for Amazon CloudWatch Logs and some associated text.

STEP 18

Haga clic en "Añadir permisos"



STEP 19

Haga clic en "Asociar políticas"



STEP 20

Ingrese el tipo como "dynamoDb"

The screenshot shows the AWS IAM Roles page. In the top navigation bar, 'Servicios' is selected. Below it, the path 'IAM > Roles > EcoGrow_lamba-role-tkkrt9ho > Añadir permisos' is shown. A search bar at the top right contains the text 'Buscar' and 'dynamoDb'. The main content area displays a table titled 'Otras políticas de permisos (849)'. The table has columns for 'Nombre de la política' (Policy Name), 'Tipo' (Type), and a small checkbox icon. One row is highlighted with a blue border: 'AmazonDynamoDBFullAccess' under 'Nombre de la política' and 'Administrada por AWS: función de trabajo' under 'Tipo'. The table includes a header row with filters for 'Nombre de la política' and 'Tipo'.

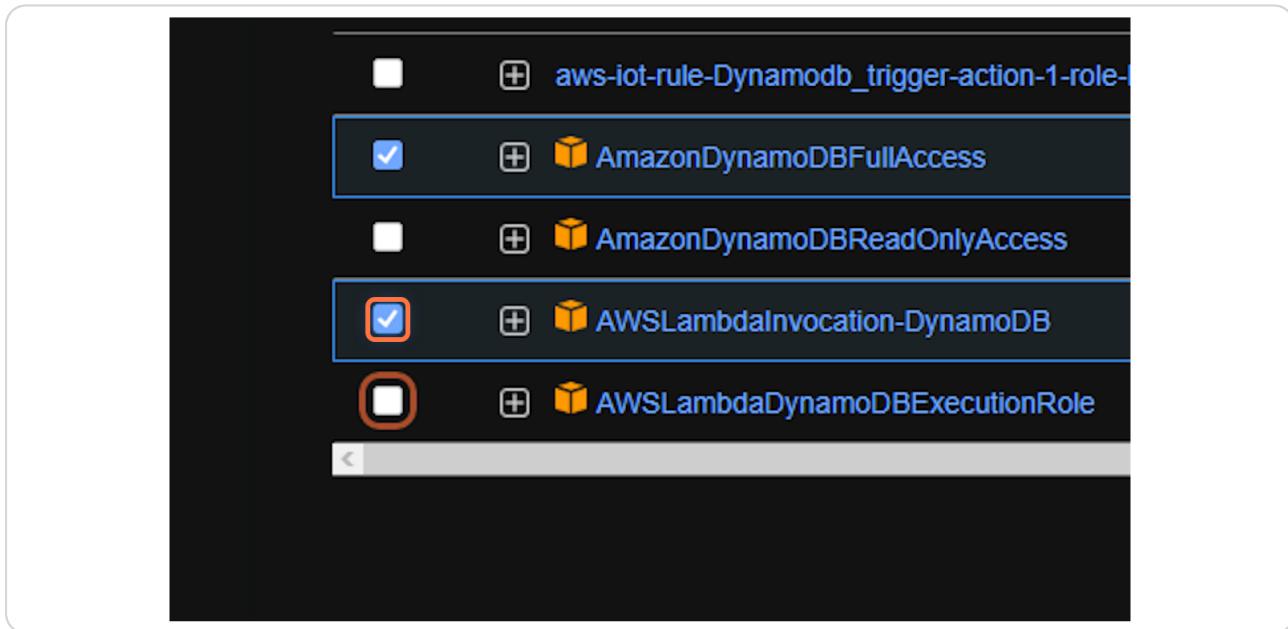
STEP 21

Seleccione la tabla "AmazonDynamoDBFullAccess", dadando clic sobre el cuadro blanco

The screenshot shows the 'Add permissions' dialog. At the top, there is a field labeled 'Nombre de la política' (Policy name) with a dropdown arrow. Below it is a list of policies. The policy 'AmazonDynamoDBFullAccess' is selected, indicated by a blue checkmark icon next to its name. Other policies listed include 'aws-iot-rule-DynamoDB_triger-action-1-role-Master_esp', 'aws-iot-rule-Dynamodb_trigger-action-1-role-IoT_dynamo', 'AmazonDynamoDBReadOnlyAccess' (which is also highlighted with a blue border around its name), 'AWSLambdaInvocation-DynamoDB', and 'AWSLambdaDynamoDBExecutionRole'. The background of the dialog is dark gray.

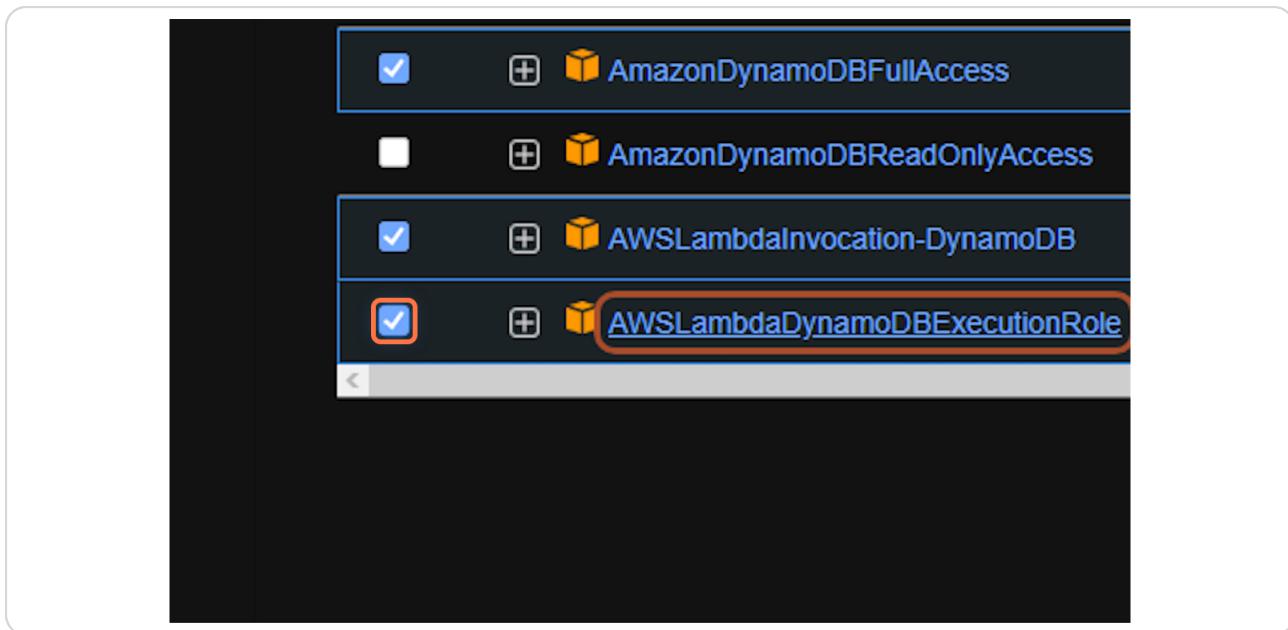
STEP 22

Seleccione la tabla "AWSLambdaInvocation-DynamoDB", dando clic sobre el cuadro blanco



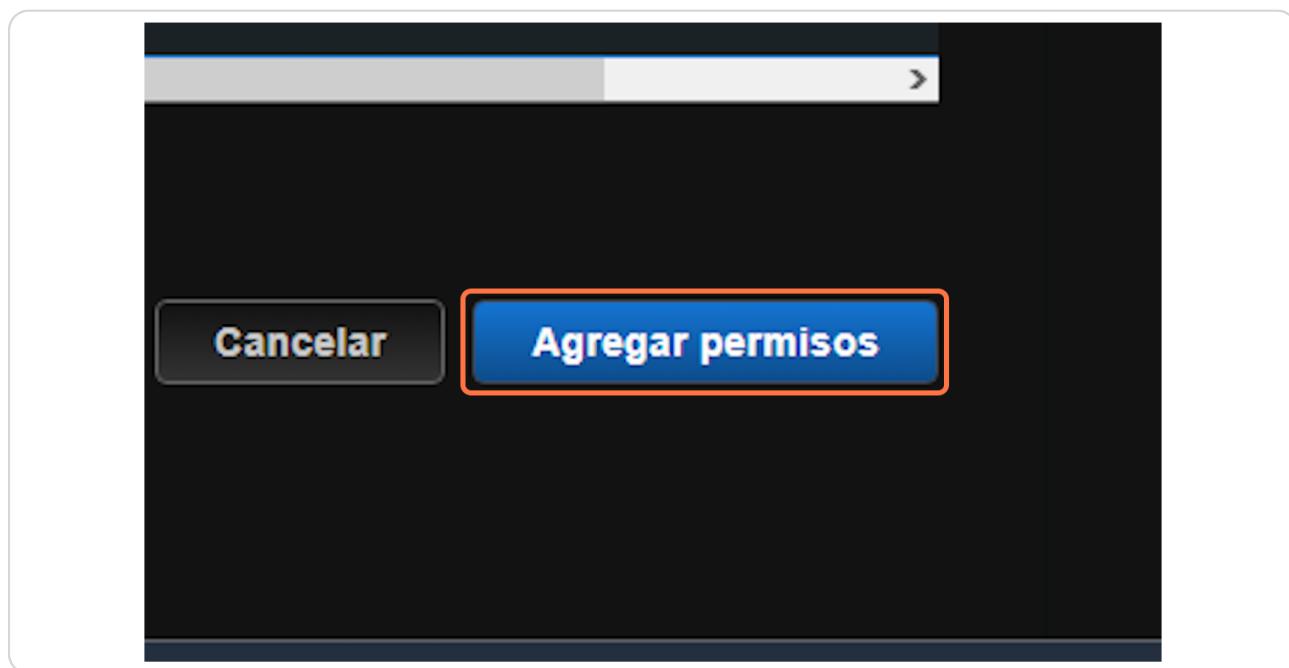
STEP 23

Seleccione la tabla "AWSLambdaDynamoDBExecutionRole", dando clic sobre el cuadro blanco



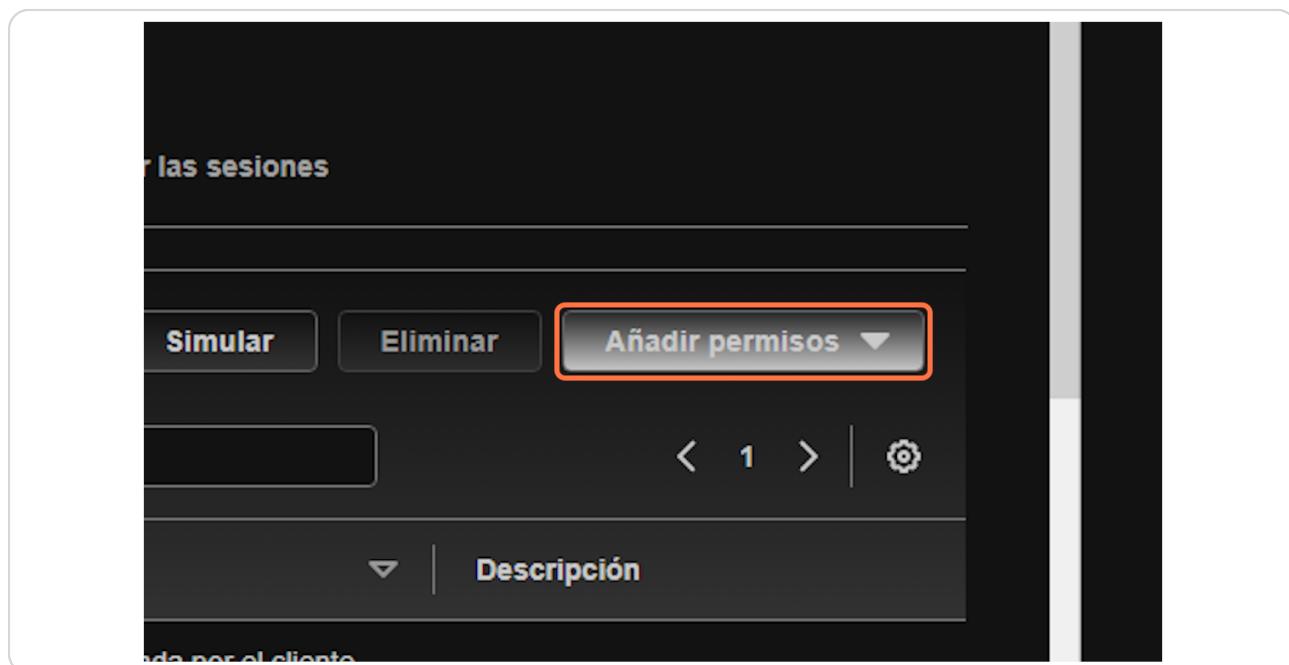
STEP 24

Haga click en "Agregar permisos"



STEP 25

Haga click en "Añadir permisos"



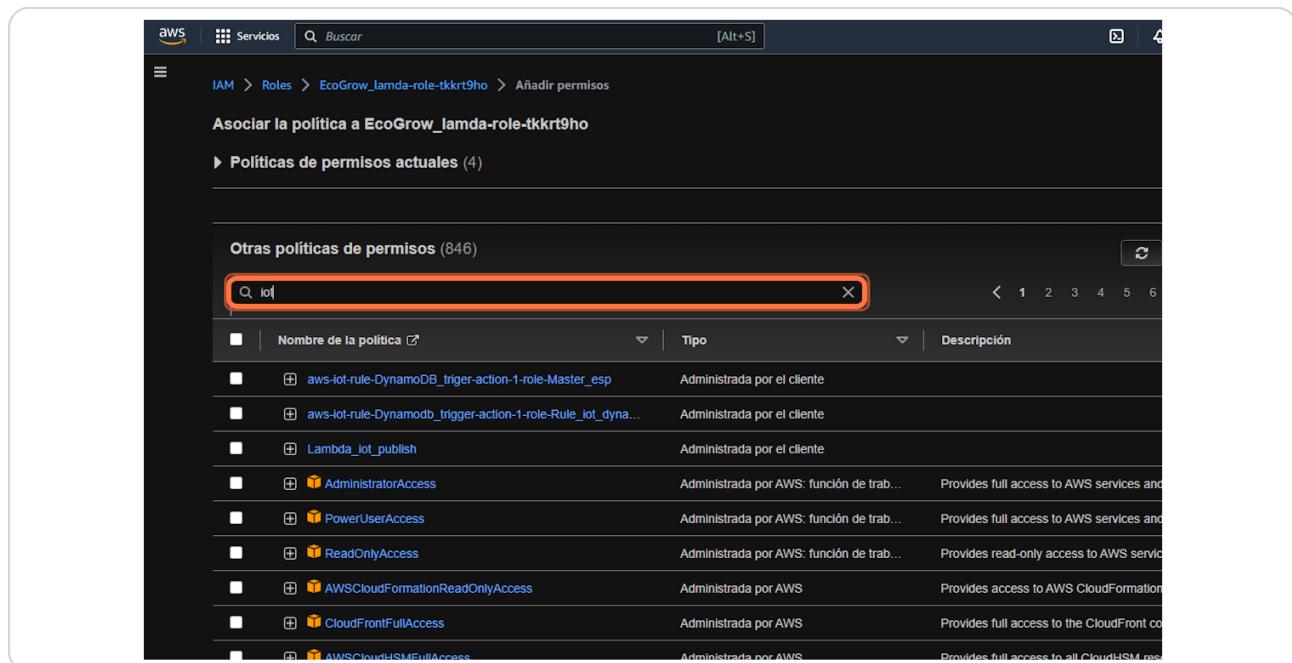
STEP 26

Haga click en "Asociar políticas"



STEP 27

Ingrese como tipo "iot"



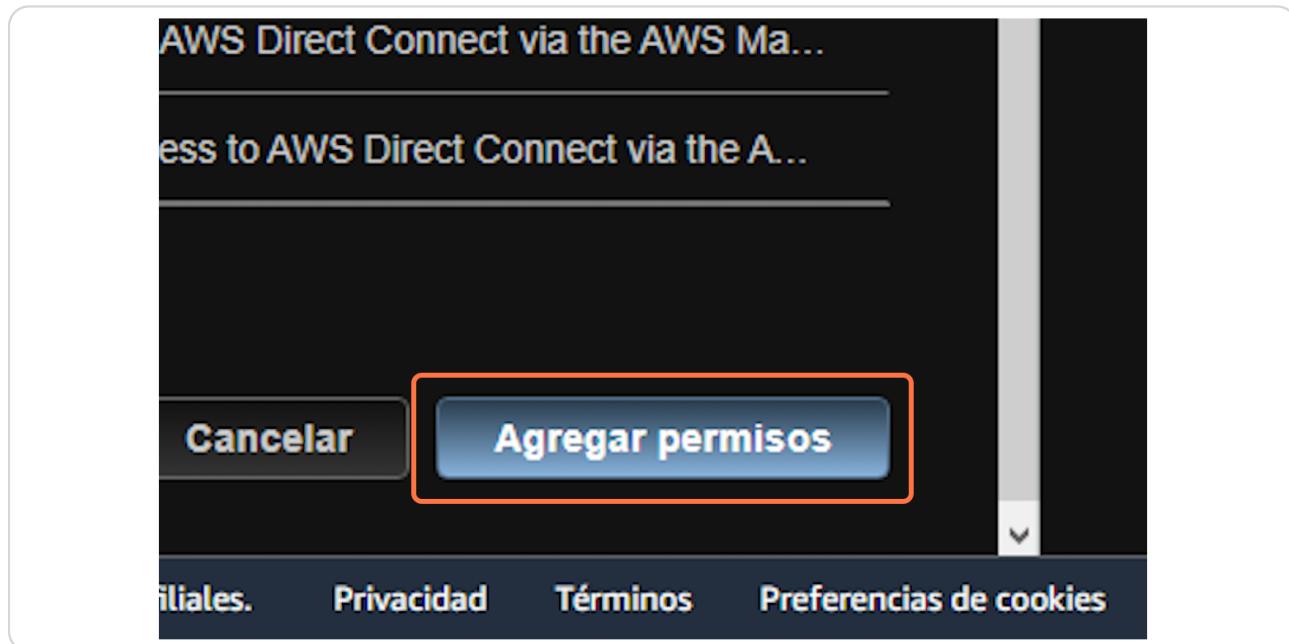
STEP 28

Seleccione la tabla "Lambda_iot_publish", "AWSIoTLogging" y "AWSIoTFullAccess", dando clic sobre el cuadro blanco



STEP 29

Haga click en "Agregar permisos"



STEP 30

Haga clic en la pestaña

The screenshot shows the AWS Lambda Role configuration page. At the top, there are tabs: Monitorear, **Configuración**, Alias, and Versiones. The **Configuración** tab is selected. Below the tabs, the title is "Rol de ejecución". A sub-section titled "Nombre del rol" contains the text "EcoGrow_lamda-role-tkkrt9ho" with a copy icon. A red box highlights the "Resumen de recursos" section. This section includes a "View role document (Ver el documento de roles)" button and a dropdown menu set to "Amazon CloudWatch Logs". Below the dropdown, it says "Para ver los recursos y las acciones a los que la función tiene permiso de acceso, elija un servicio." Under "Por acción" and "Por recurso", there is a table with one row: "Recurso" "am:awslogs:us-east-2:051942573136*" and "Acciones" "Allow: logs>CreateLogGroup". The bottom of the page includes standard AWS footer links: © 2023, Amazon Web Services, Inc. o sus filiales., Privacidad, Términos, Preferencias, and de cook.

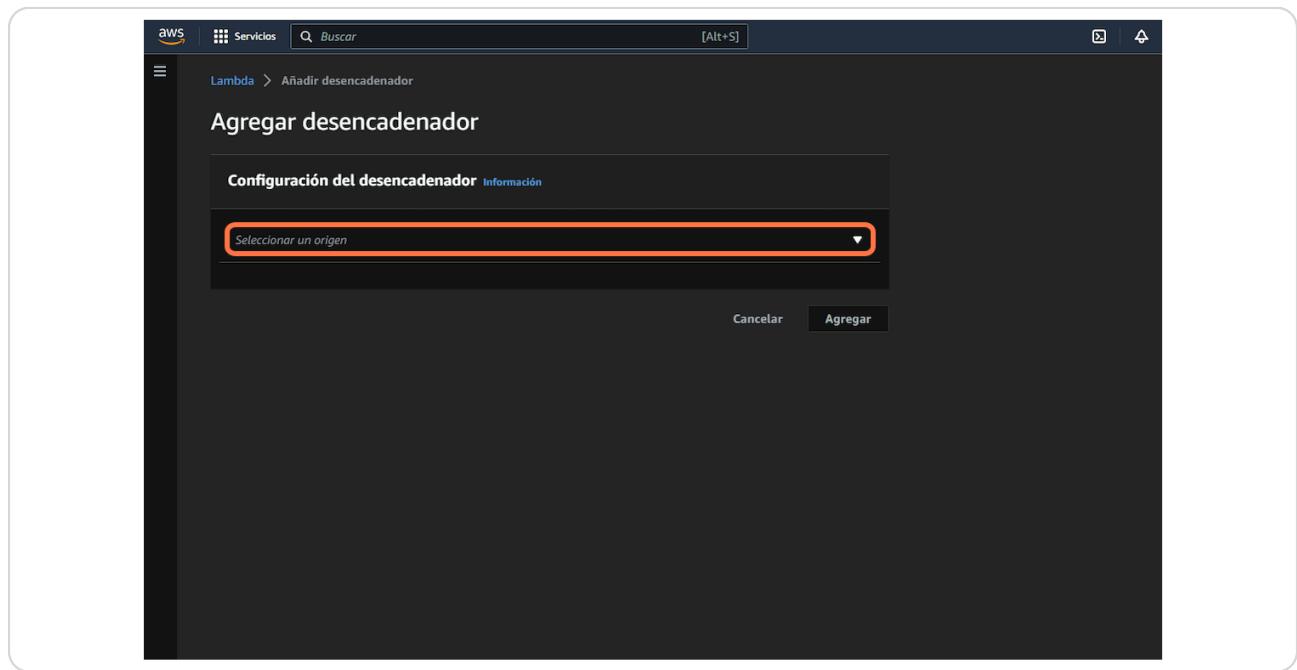
STEP 31

Haga clic en "Agregar desencadenador"

The screenshot shows the "Información general de la función" (General Function Information) page. It displays the function name "EcoGrow_lamda" and its associated "Layers" "(1)". Below this, there is a button labeled "+ Agregar desencadenador" (Add Trigger), which is highlighted with a red box. The background is dark, and the overall interface is consistent with the previous screenshot.

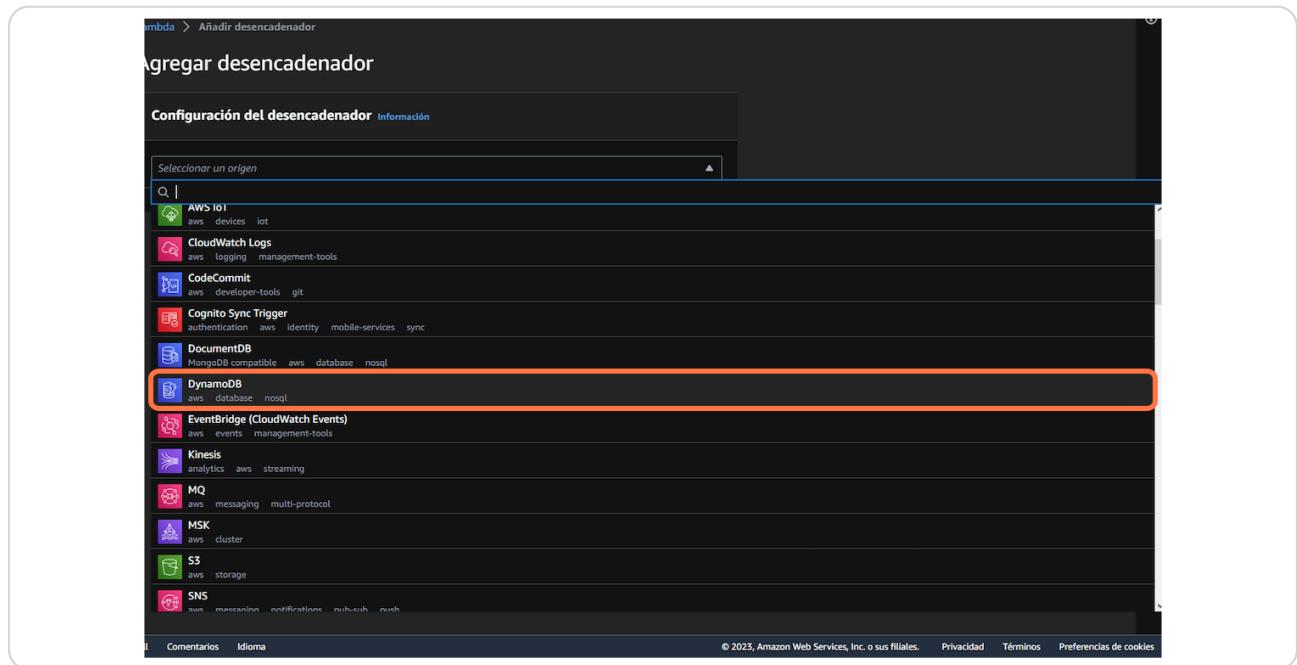
STEP 32

Haga clic en "Seleccionar un origen"



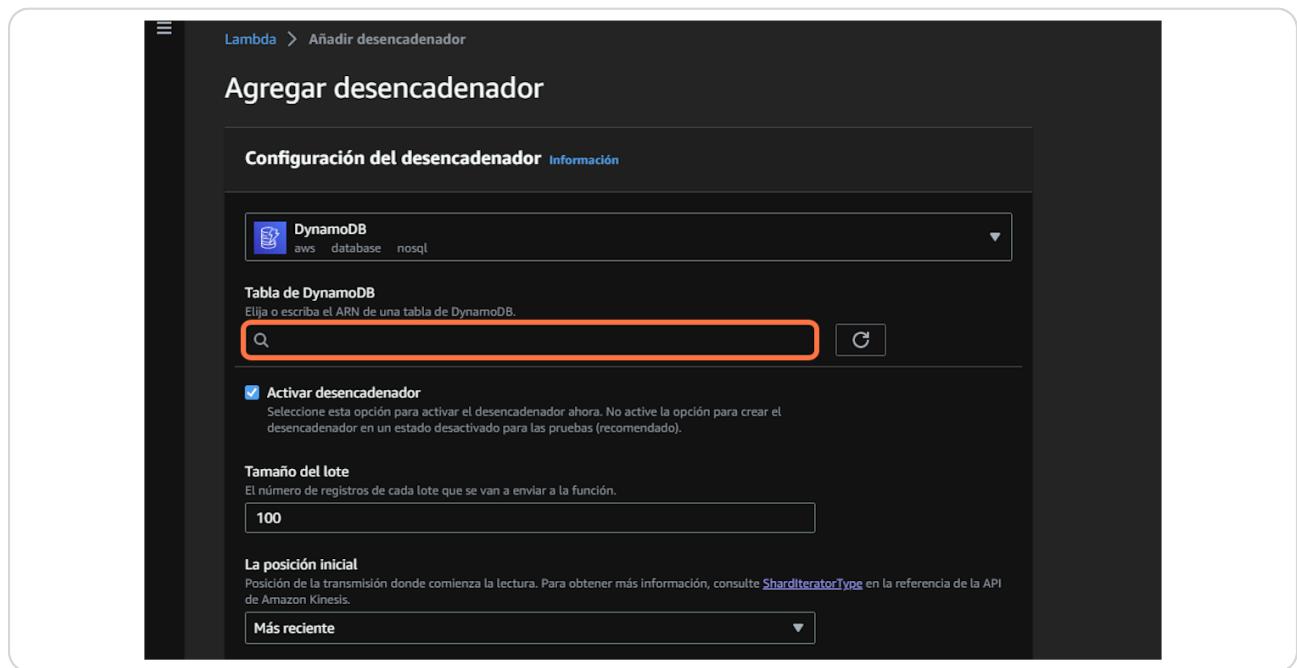
STEP 33

Haga clic en "DynamoDB..."



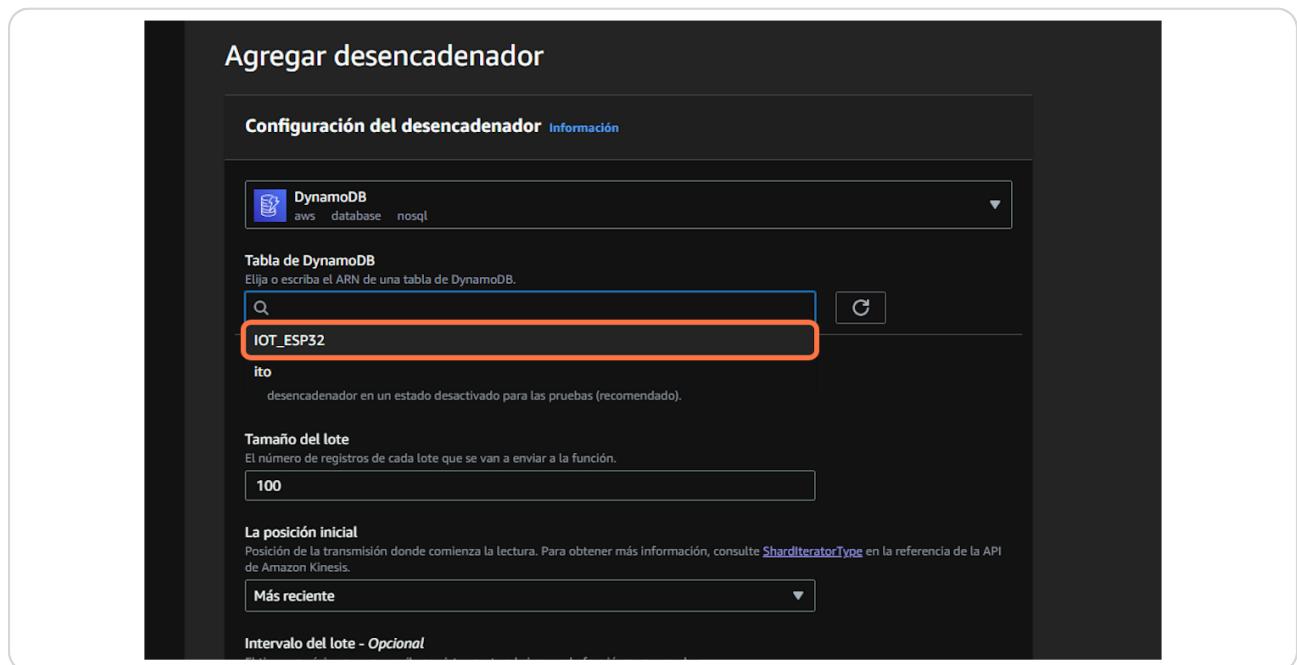
STEP 34

Haga clic en "Tabla de DynamoDB"



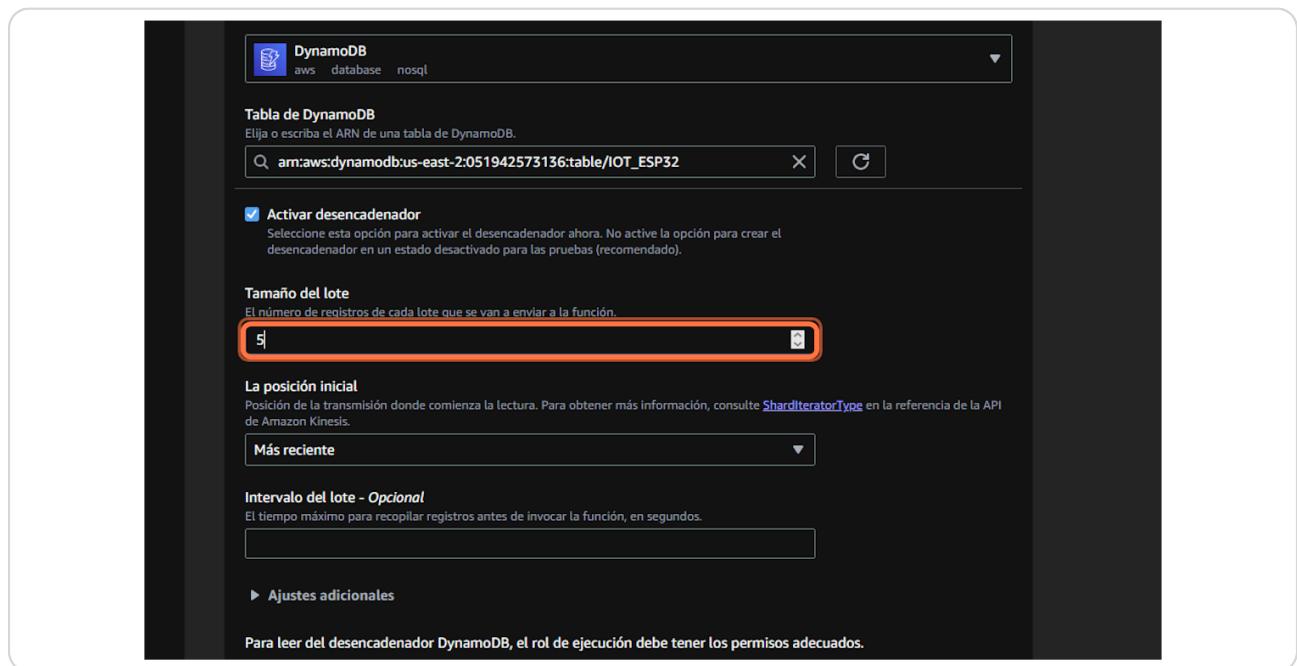
STEP 35

Haga clic en "IOT_ESP32..."



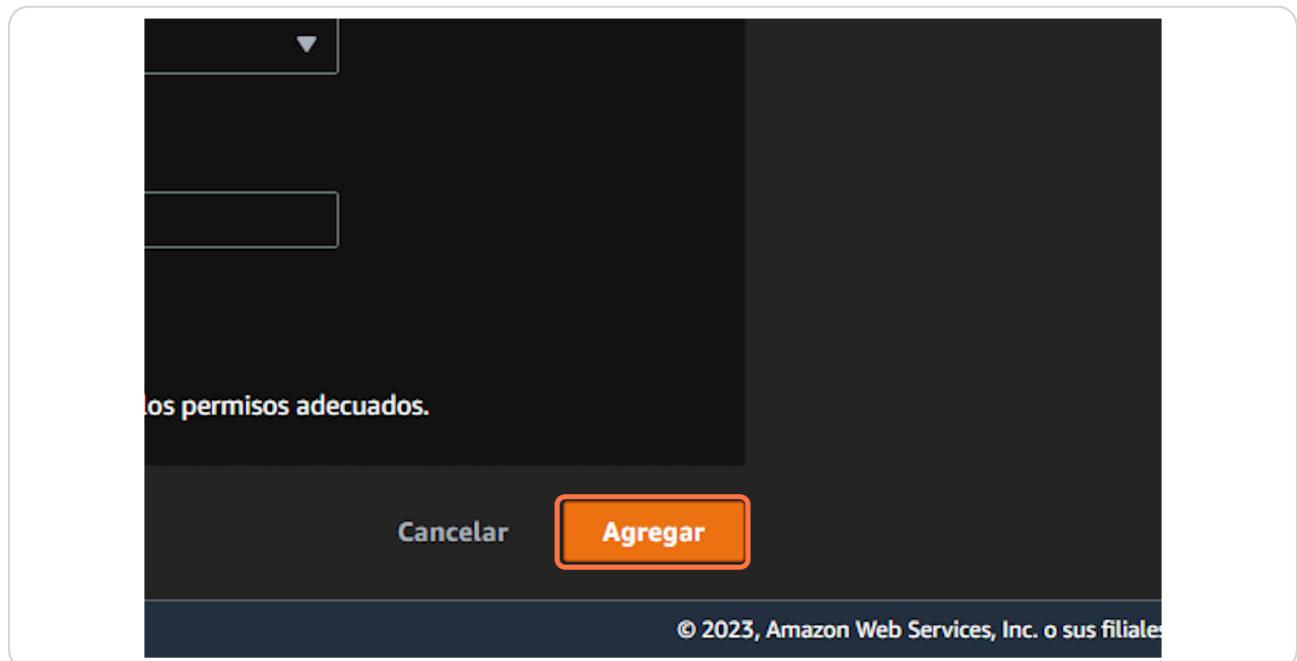
STEP 36

Ingrese como tipo "5"



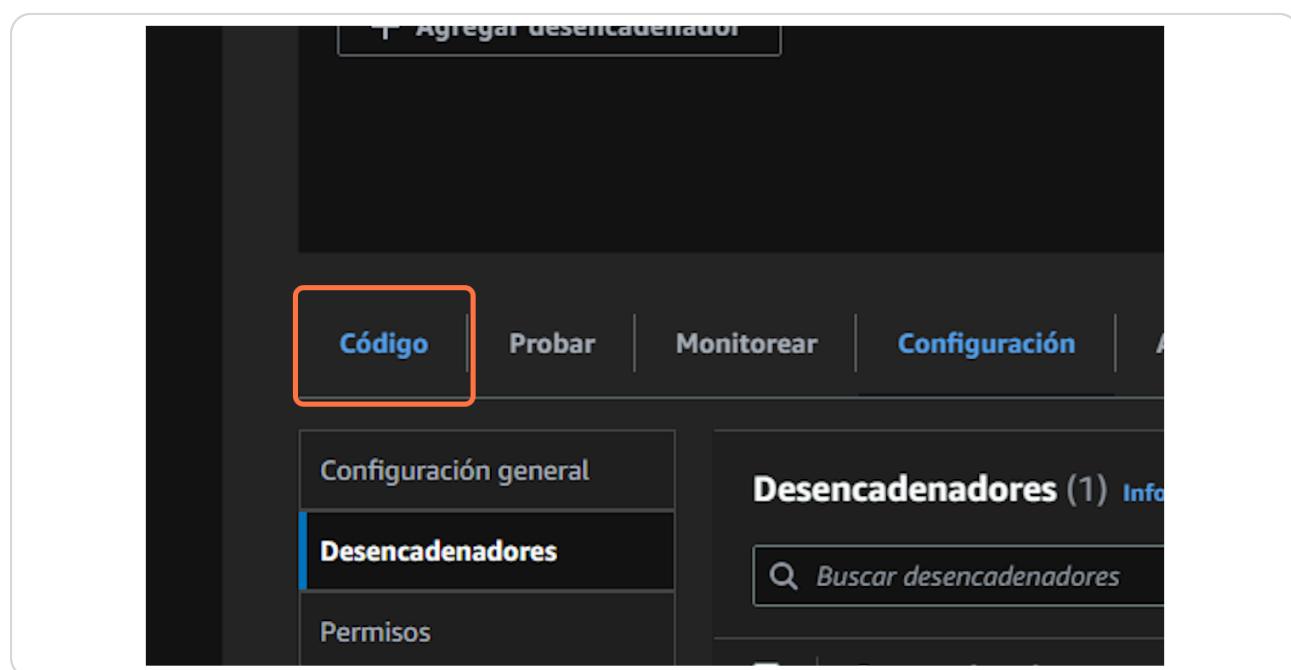
STEP 37

Haga clic en "Agregar"



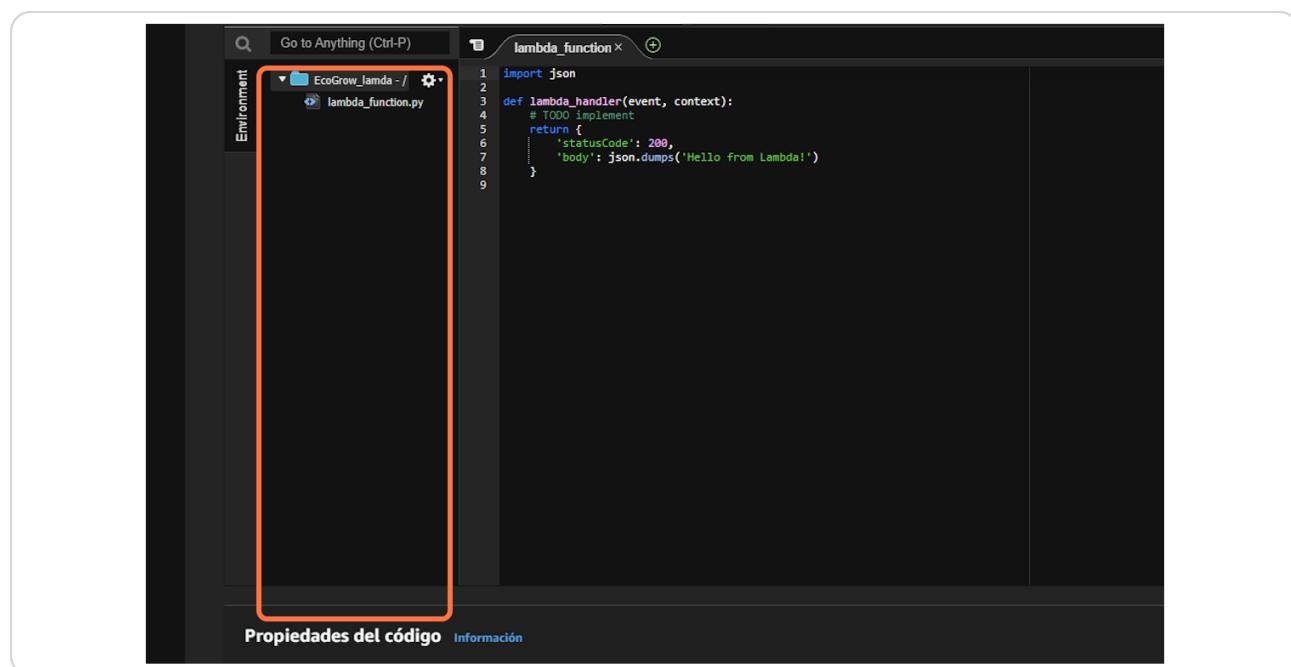
STEP 38

Haga clic en "Código"



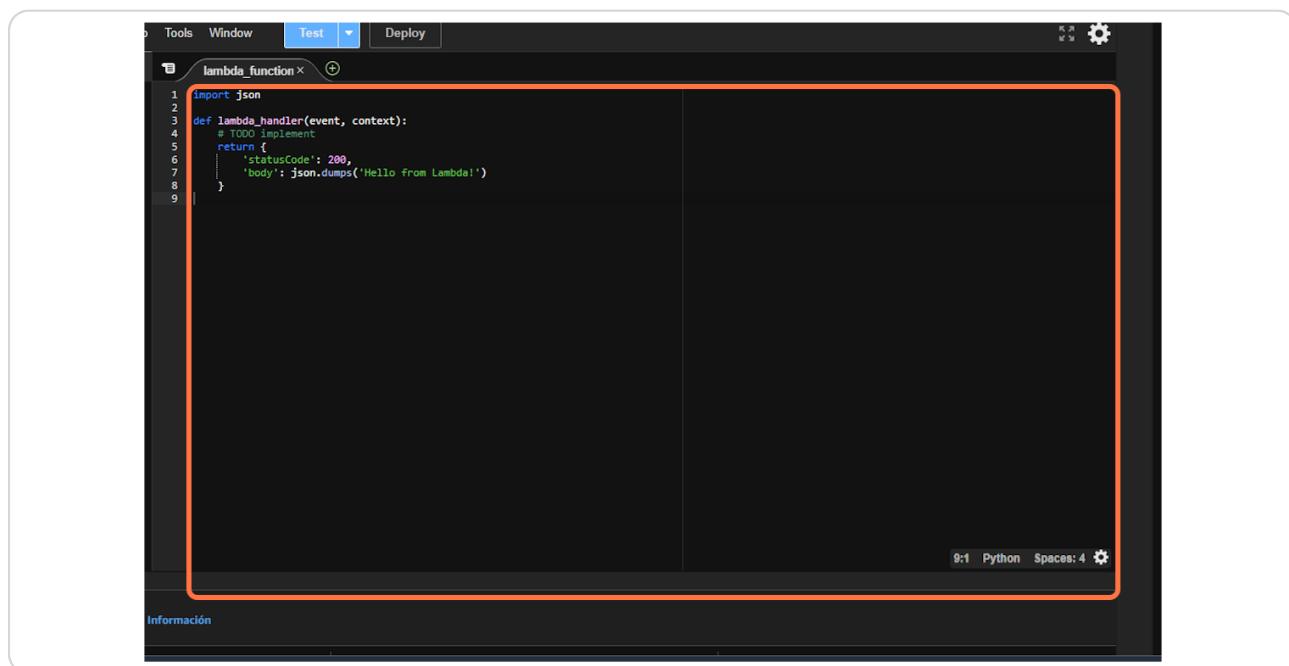
STEP 39

Haga clic en "EcoGrow_lambda - /..."



STEP 40

Encontra el siguiente código



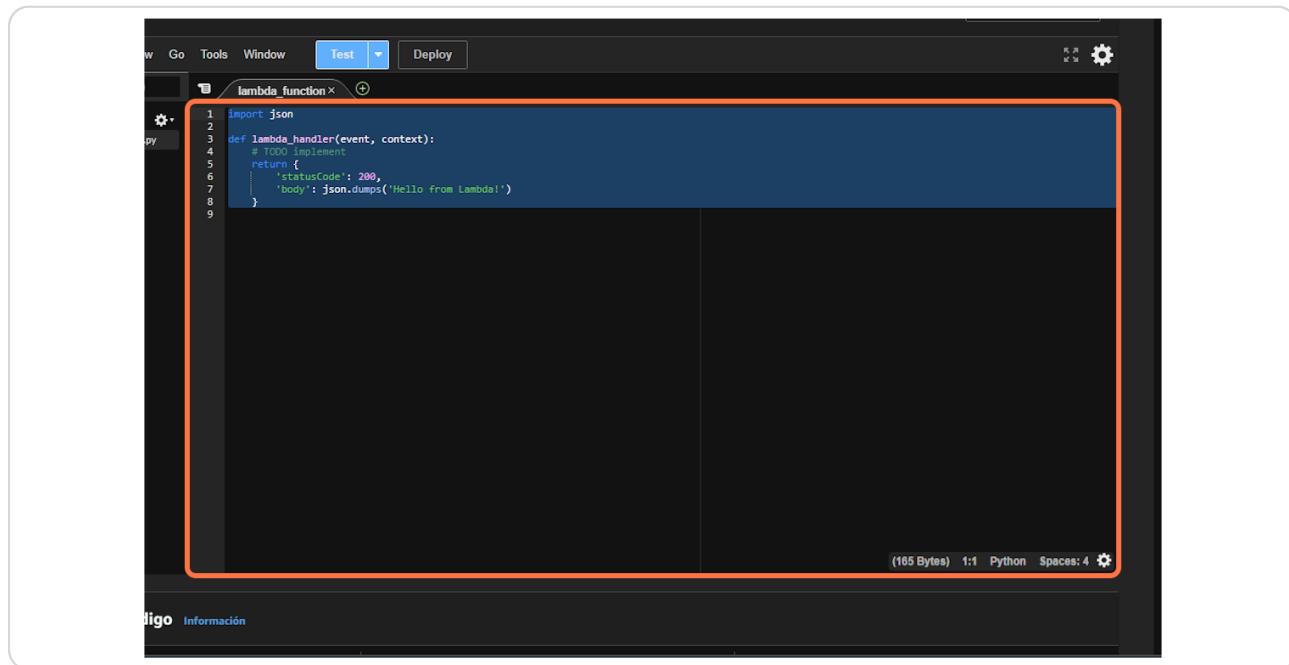
A screenshot of a code editor window titled "lambda_function.py". The window has tabs for "Tools", "Window", "Test", and "Deploy". The code editor shows the following Python code:

```
1 import json
2
3 def lambda_handler(event, context):
4     # TODO implement
5     return {
6         'statusCode': 200,
7         'body': json.dumps('Hello from Lambda!')
8     }
9
```

The code is highlighted with a thick orange border. The status bar at the bottom right shows "9:1 Python Spaces: 4".

STEP 41

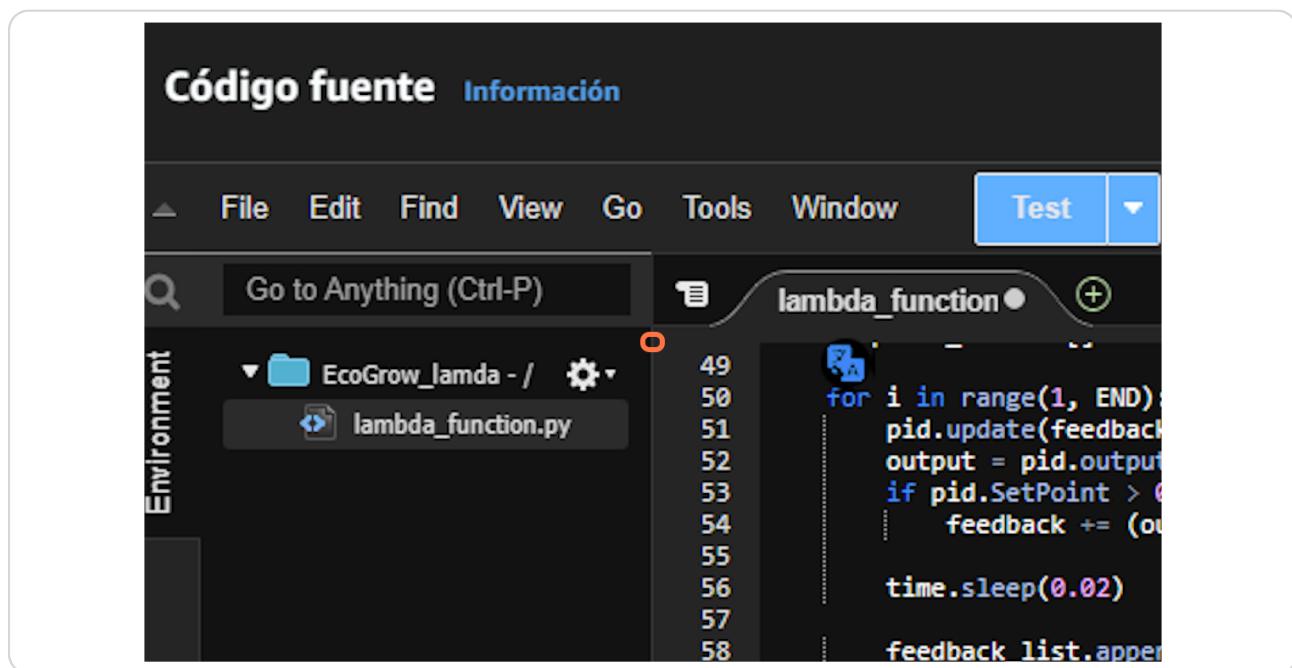
Seleccione el código



A screenshot of a code editor window titled "lambda_function.py". The window has tabs for "Tools", "Go", "Window", "Test", and "Deploy". The code editor shows the same Python code as in Step 40, with the entire code block highlighted by a thick orange border. The status bar at the bottom right shows "(165 Bytes) 1:1 Python Spaces: 4".

STEP 42

pegar el texto



```
for i in range(1, END):
    pid.update(feedback)
    output = pid.output
    if pid.SetPoint > 0:
        feedback -= (output - (1/i))
    time.sleep(0.02)

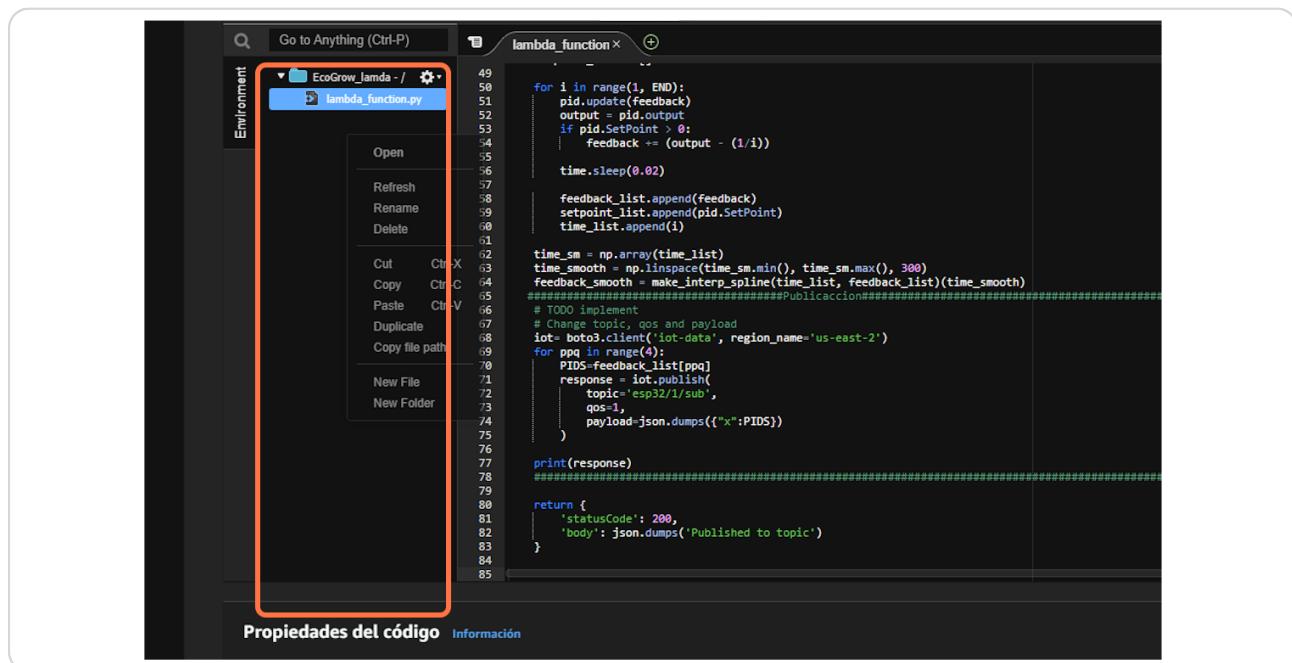
feedback_list.append(feedback)
setpoint_list.append(pid.SetPoint)
time_list.append(i)

time_sm = np.array(time_list)
time_smooth = np.linspace(time_sm.min(), time_sm.max(), 300)
feedback_smooth = make_interp_spline(time_list, feedback_list)(time_smooth)
#####
# TODO implement
# Change topic, qos and payload
iot= boto3.client('iot-data', region_name='us-east-2')
for ppq in range(4):
    PIDS=feedback_list[ppq]
    response = iot.publish(
        topic="esp32/1/sub",
        qos=1,
        payload=json.dumps({"x":PIDS})
    )
    print(response)
#####

return {
    'statusCode': 200,
    'body': json.dumps('Published to topic')
}
```

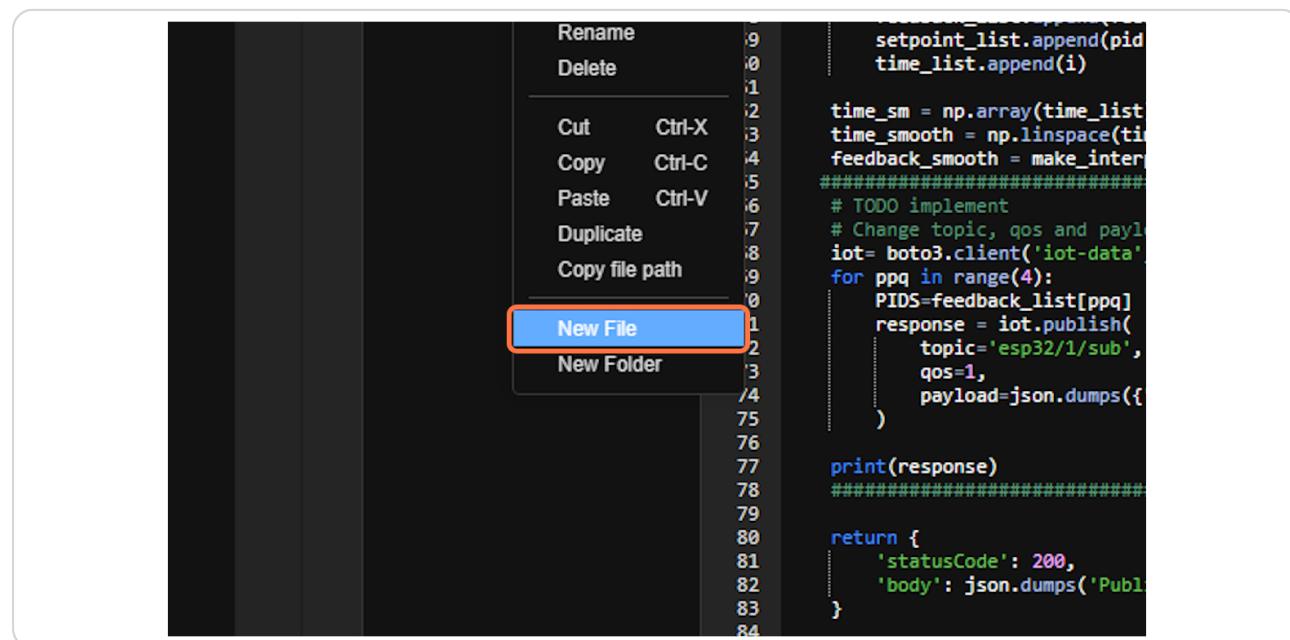
STEP 43

Haga clic derecho en "EcoGrow_lamda - /..."



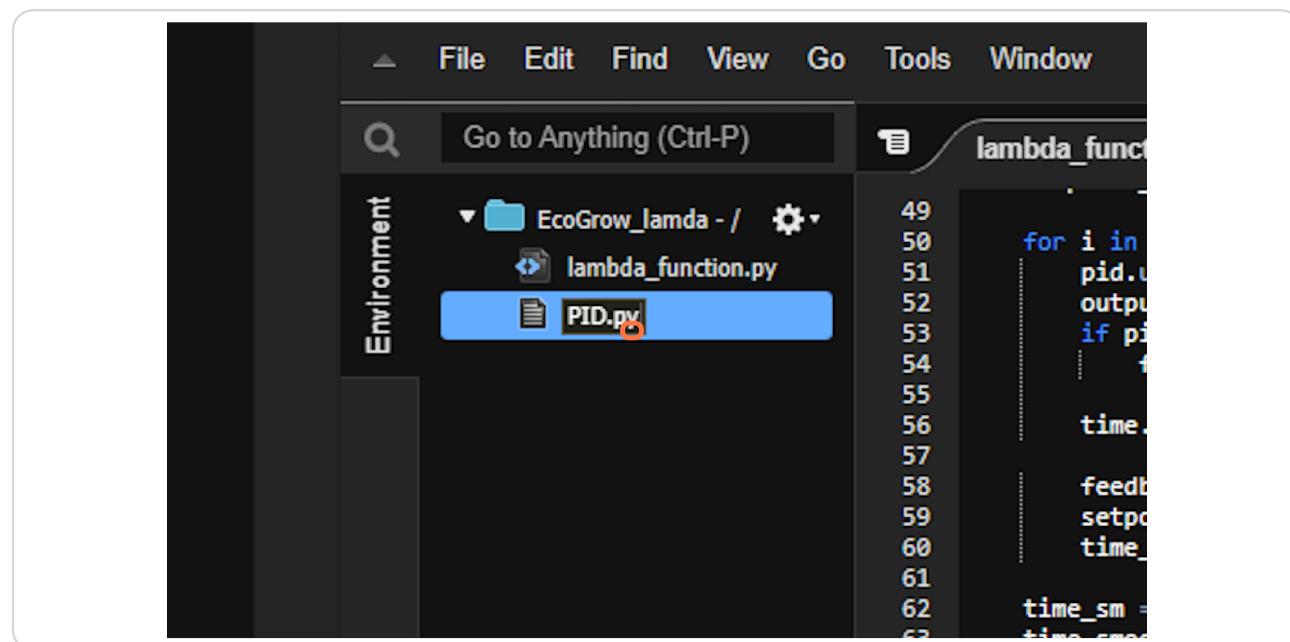
STEP 44

Haga clic en "New File"



STEP 45

Ingresar el nombre del nuevo documento "PID.py"



STEP 46

Haga clic en "EcoGrow_lambda - /..."

The screenshot shows the AWS Lambda function editor interface. On the left, there's a sidebar with a search bar and a tree view showing the project structure: 'EcoGrow_lambda -' > 'lambda_function' > 'PID.py'. The main area is titled 'lambda_function' and contains the Python code for the function. A red box highlights the 'lambda_function' tab. At the bottom, there are tabs for 'Propiedades del código' and 'Información'.

```
49
50     for i in range(1, END):
51         pid.update(feedback)
52         output = pid.output
53         if pid.SetPoint > 0:
54             feedback += (output - (1/i))
55
56         time.sleep(0.02)
57
58         feedback_list.append(feedback)
59         setpoint_list.append(pid.SetPoint)
60         time_list.append(i)
61
62         time_sm = np.array(time_list)
63         time_smooth = np.linspace(time_sm.min(), time_sm.max(), 300)
64         feedback_smooth = make_interp_spline(time_list, feedback_list)(time_smooth)
65 ######Publicacion#####
66 # TODO implement
67 # Change topic, qos and payload
68 iot = boto3.client('iot-data', region_name='us-east-2')
69 for ppq in range(4):
70     PIDS=feedback_list[ppq]
71     response = iot.publish(
72         topic='esp32/1/sub',
73         qos=1,
74         payload=json.dumps({"x":PIDS})
75     )
76
77     print(response)
78 #####
79
80     return {
81         'statusCode': 200,
82         'body': json.dumps('Published to topic')
83     }
84
85
```

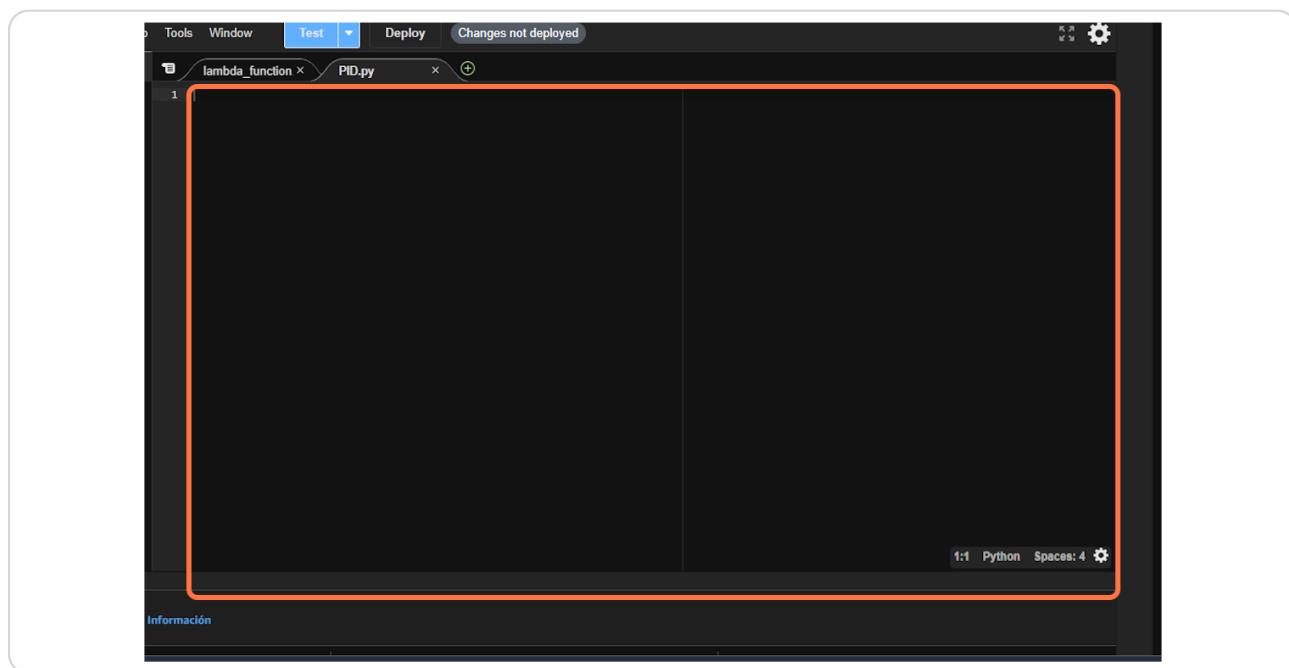
STEP 47

Haga clic en el cuadro de texto

The screenshot shows the AWS Lambda function editor interface. The main area is a code editor titled 'lambda_function' with the file 'PID.py' open. A red box highlights the large text input field where code is being typed. At the bottom right of the editor, there are status indicators: '1:1 Python Spaces: 4' and a gear icon. Below the editor, there are tabs for 'Propiedades del código' and 'Información'.

STEP 48

Haga clic en el cuadro de texto



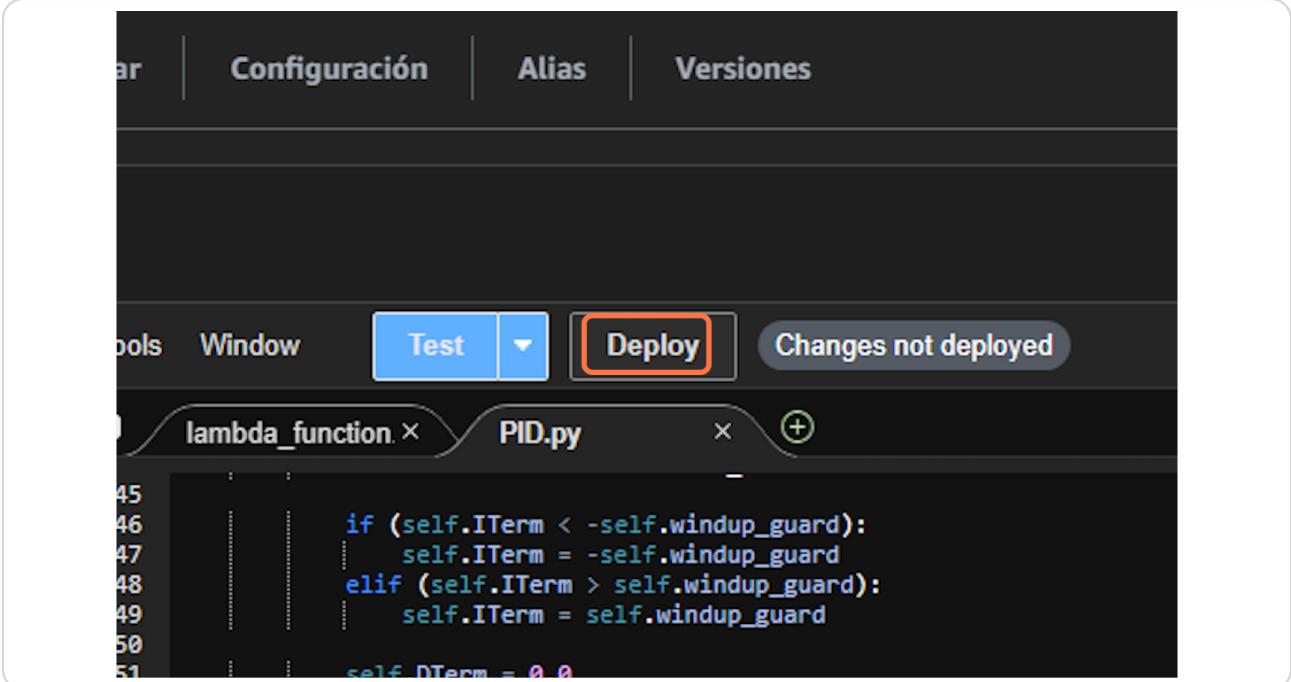
STEP 49

Pegar área de texto

A screenshot of a code editor window showing a block of Python code. The code includes lines 71 through 81, which define methods for setting gain, windup, and sample time. A red circle highlights the cursor at the end of line 81. Below the code editor, there is a status bar with the text "Medidas del código" and "Información". At the bottom of the screen, there are navigation links for "Contenidos" and "Idioma".

STEP 50

Haga clic en "Deploy"



The screenshot shows a software interface with a dark theme. At the top, there are tabs labeled "Configuración", "Alias", and "Versiones". Below these, a toolbar has buttons for "Tools", "Window", "Test" (which is highlighted in blue), "Deploy" (which is highlighted with a red border), and "Changes not deployed". The main area is a code editor with two tabs: "lambda_function" and "PID.py". The "PID.py" tab is active, showing the following code:

```
45
46     if (self.ITerm < -self.windup_guard):
47         self.ITerm = -self.windup_guard
48     elif (self.ITerm > self.windup_guard):
49         self.ITerm = self.windup_guard
50
51     self.DTerm = 0.0
```