Here's a detailed step-by-step lab guide based on **"Automating EC2 with AWS Lambda Using Boto3 – Lab 2"** : **stops running EC2 instances across all AWS regions**:

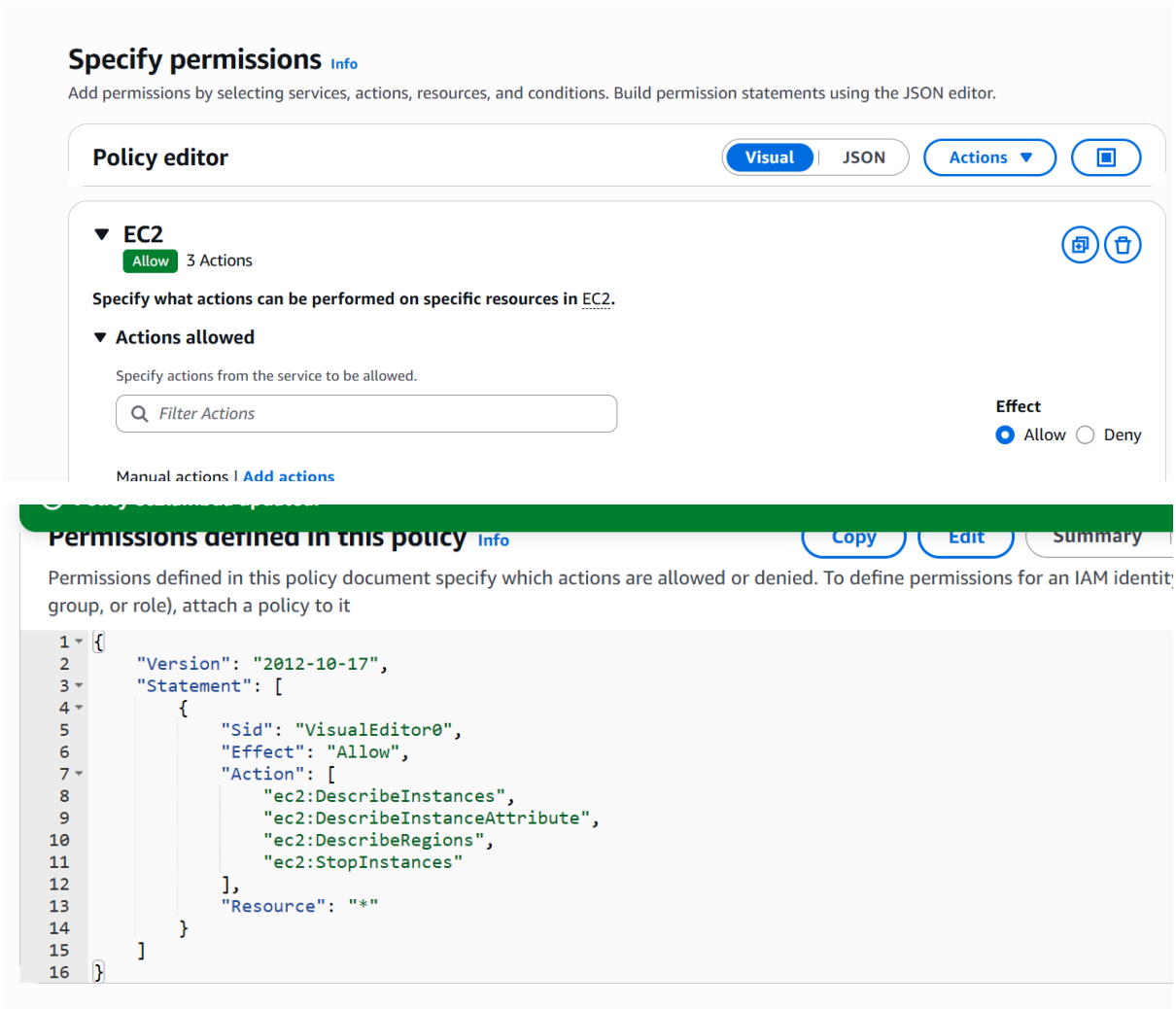
Objectives

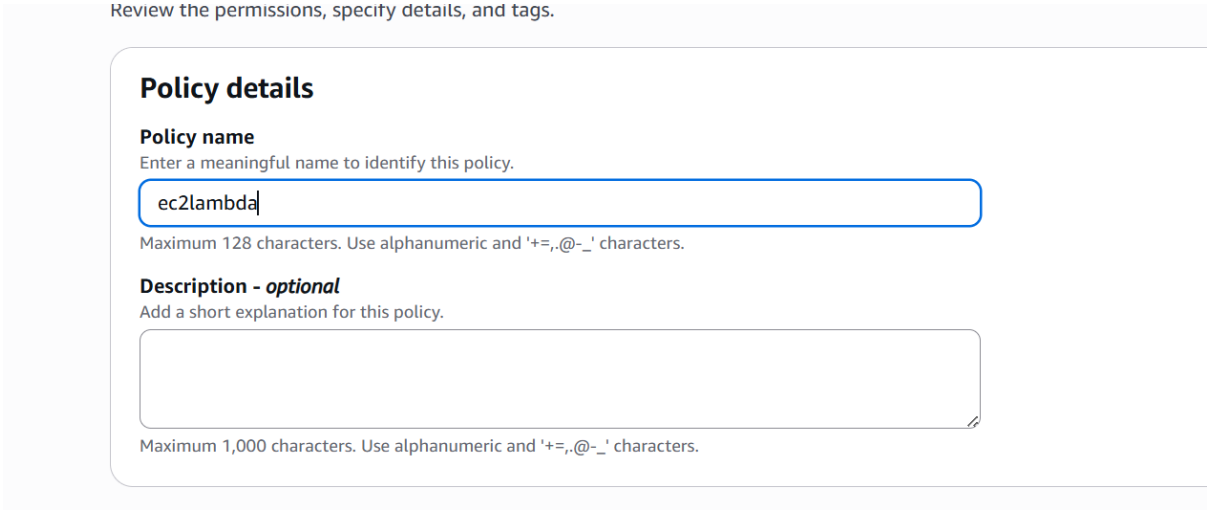
- Build a Lambda function that lists **all AWS regions**, identifies **running EC2 instances**, and stops them.

Step-by-Step Lab

1. Prerequisites

- AWS account with permissions to **describe and stop EC2 instances** (ec2:DescribeRegions, ec2:DescribeInstances, ec2:StopInstances).

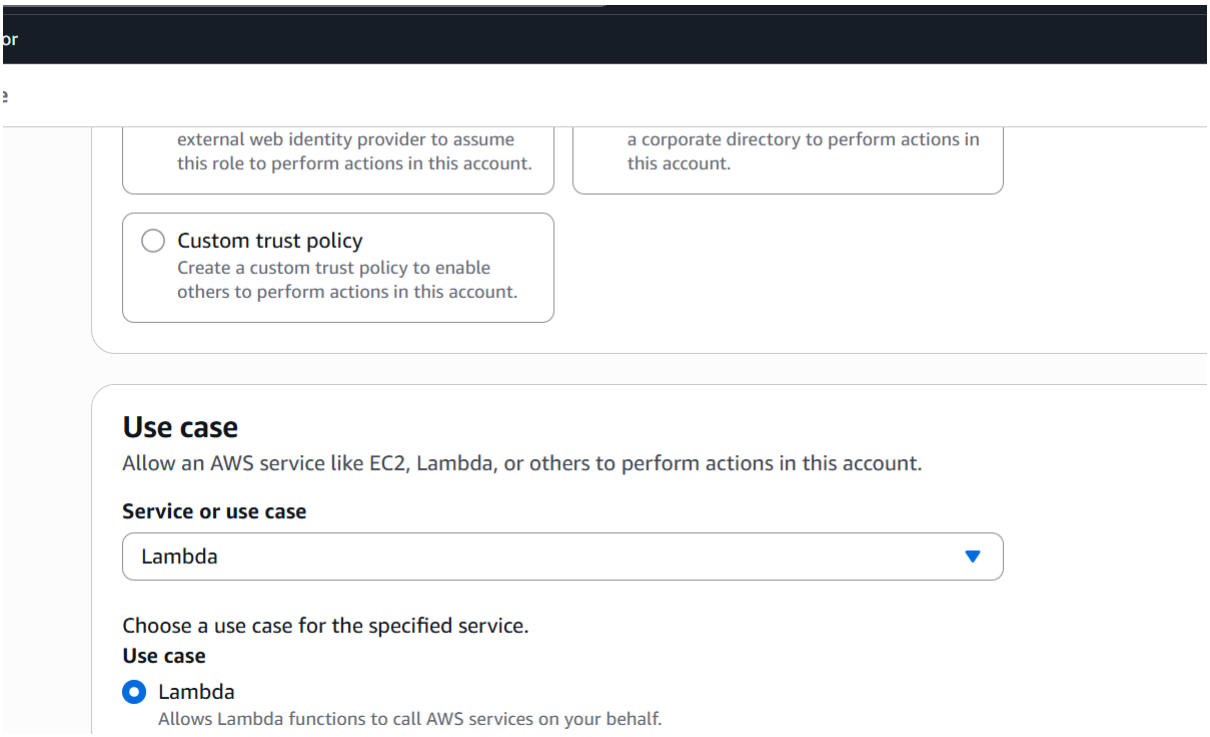




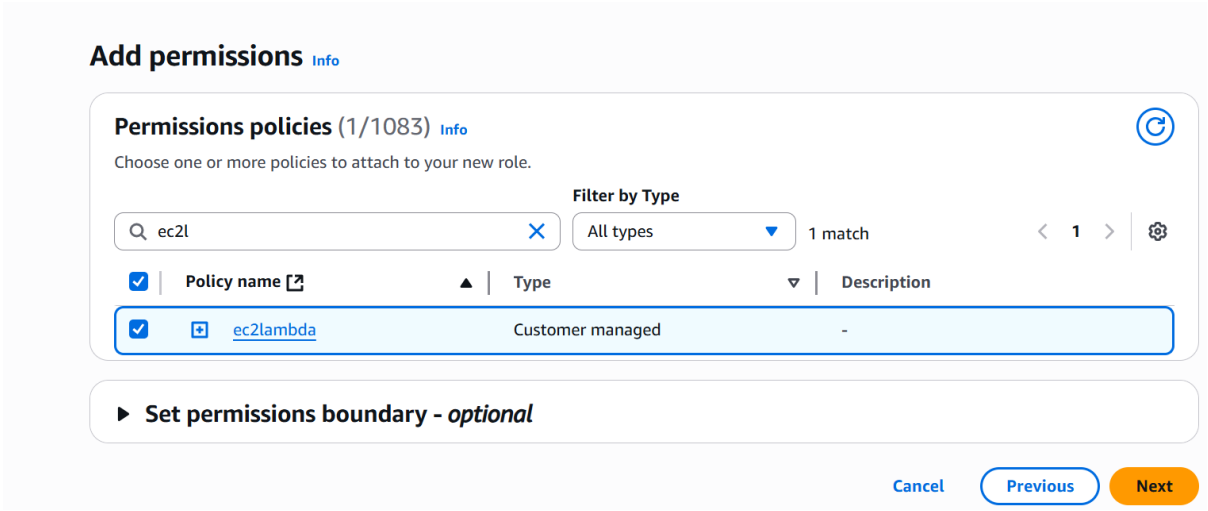
---

2. Create IAM Role for Lambda

1. Go to **IAM → Roles → Create role**.
2. Choose **Lambda** as trusted entity.



3. Attach policy with:
    - ec2:DescribeRegions
    - ec2:DescribeInstances
    - ec2:StopInstances

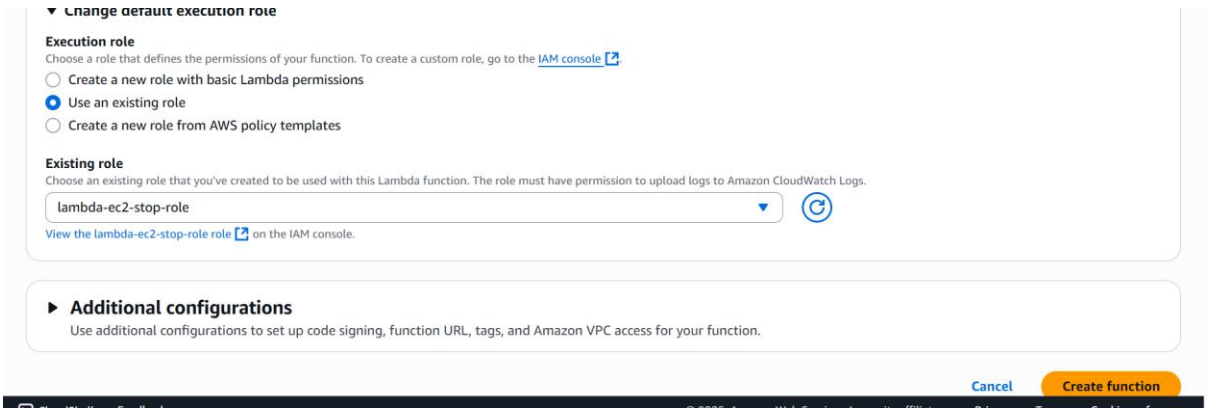


4. Name role: e.g. lambda-ec2-stop-role.

---

3. Create the Lambda Function

1. Open **AWS Lambda Console → Create Function**.
2. Choose:
    - Author from scratch
    - Runtime: **Python 3.9+**
    - Role: **Use existing role** → select lambda-ec2-stop-role



3.

4. Name function: e.g., StopAllRunningEC2s.

---

4. Add the Python Code

Paste this into the Lambda inline editor:

```
import boto3

def lambda_handler(event, context):
```

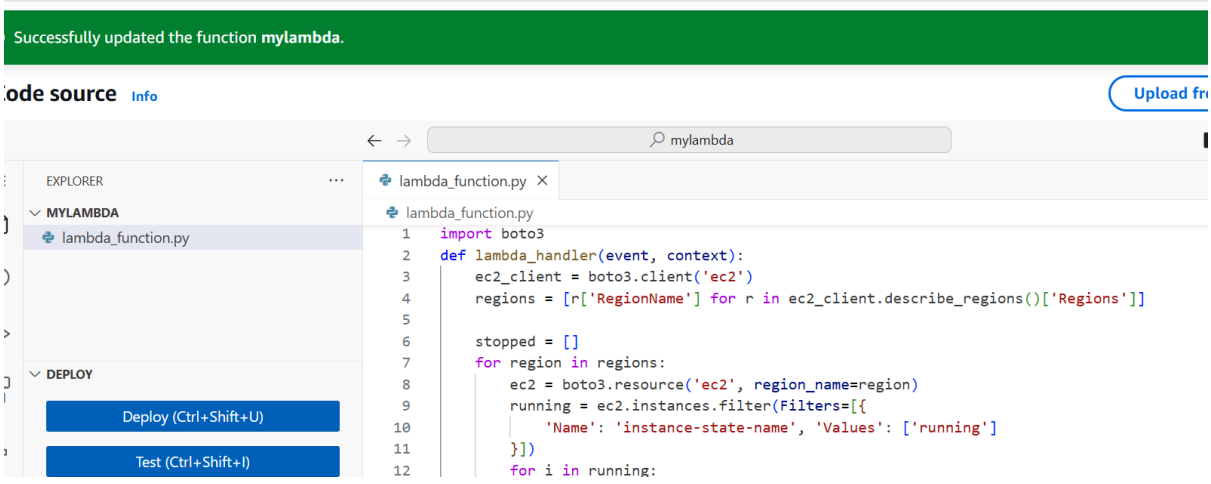
```
ec2_client = boto3.client('ec2')
regions = [r['RegionName'] for r in ec2_client.describe_regions()['Regions']]


stopped = []
for region in regions:
    ec2 = boto3.resource('ec2', region_name=region)
    running = ec2.instances.filter(Filters=[{
        'Name': 'instance-state-name', 'Values': ['running']
    }])
    for i in running:
        i.stop()
        stopped.append({'Region': region, 'InstanceId': i.id})
        print(f"Stopped {i.id} in {region}")


return {'StoppedInstances': stopped}
```

5. Deploy the Code

- Click **Deploy** in Lambda console to save the function.



---

6. Test the Function

1. Click **Test → Create Event**.

2. Use a simple template like:

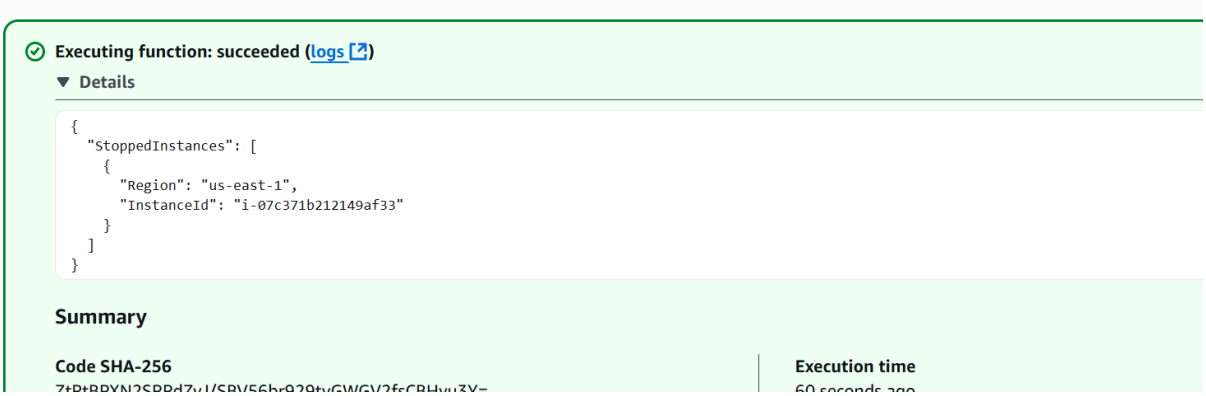
{}

3. Run the test.
4. Check logs in **CloudWatch** for output like "Stopped i-0123456789abcdef0 in us-east-1".

---

7. Inspect Results

- Navigate to the EC2 console.
- Verify previously running instances are now in **“stopping” or “stopped”** states across regions.



---

8. Schedule Lambda Execution

(Optional: automate periodic shutdowns)

1. Go to **Lambda → Configuration → Triggers → Add trigger**.
2. Choose **EventBridge (CloudWatch Events)**.
3. Create rule:
    - Type: **Schedule expression**
    - Expression: cron(0 22 * * ? *) → e.g., 10 PM UTC daily.
4. Click **Add**.

.