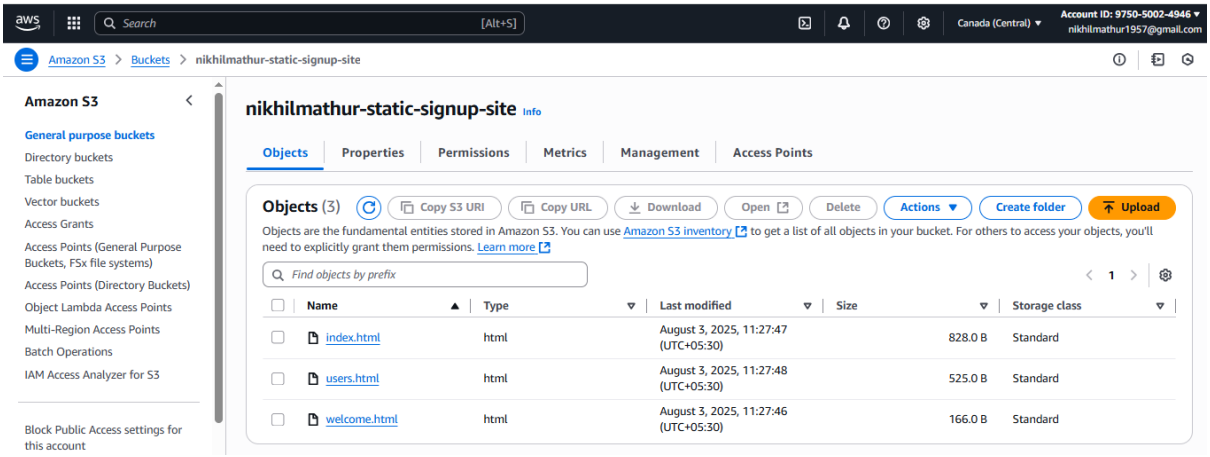
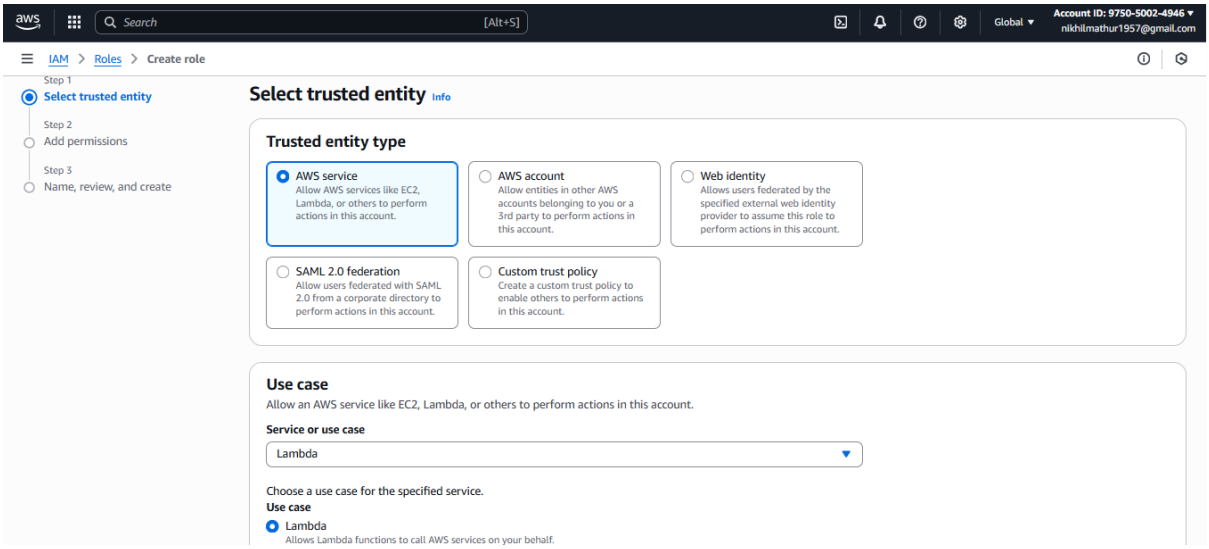
**Assignment 2: Automated S3 Bucket Cleanup Using AWS Lambda and Boto3**

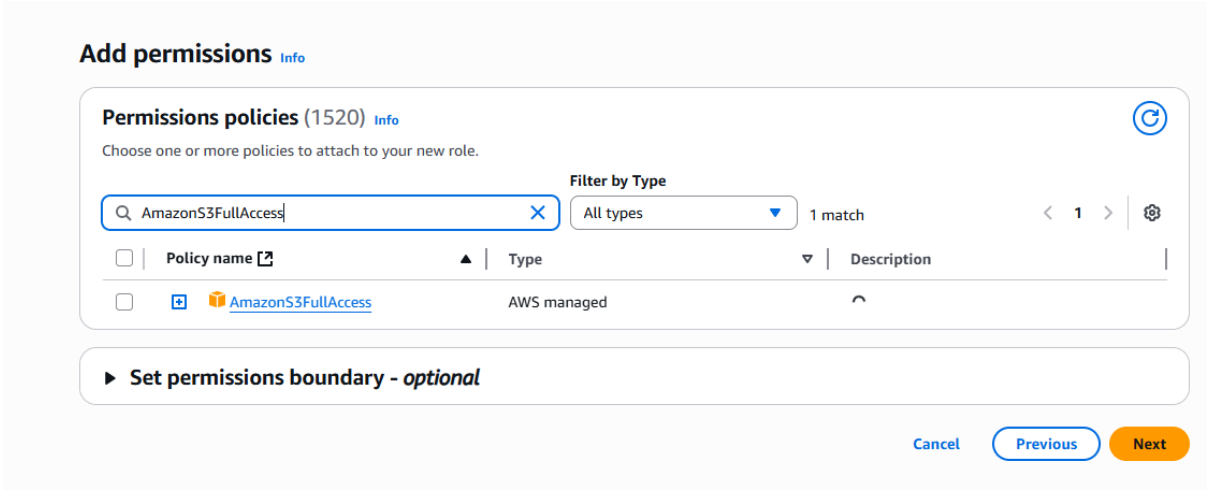
**Task: Automate the deletion of files older than 30 days in a specific S3 bucket.**

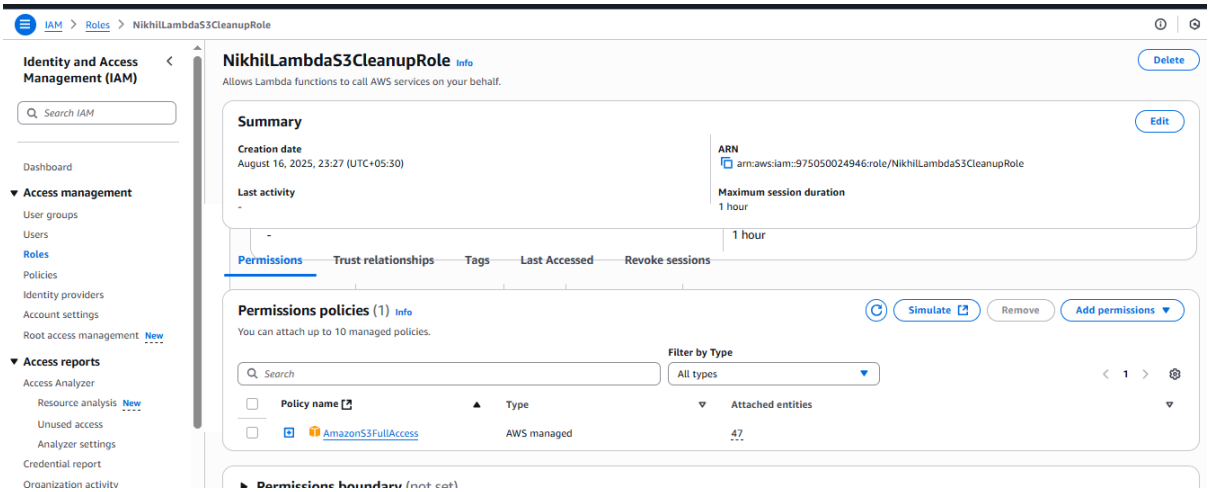
1. **S3 Setup: (I used old bucket for the task)**
    a. **Navigate to the S3 dashboard and create a new bucket.**
        i. Click the orange Create bucket button.
        ii. Bucket name: **nikhilmathur-static-signup-site** (Must be globally unique)
        iii. Choose your preferred region (Keep it consistent with other resources (like EC2, Lambda))
        iv. Open your bucket by clicking its name.
        v. Click Upload → Add files (choose any file from your computer).
        vi. Click Upload at the bottom.




2. **Lambda Function:**
    a. **Create an IAM Role for Lambda**
        i. **AWS Console => IAM =>Roles => Create role.**
        ii. **Trusted Entity type: AWS Services**
        iii. **Use Case: Lambda**
        iv. **Click Next**
        v. **Permissions policies: AmazonS3FullAccess**
        vi. **Role name: NikhilLambdaS3CleanupRole**



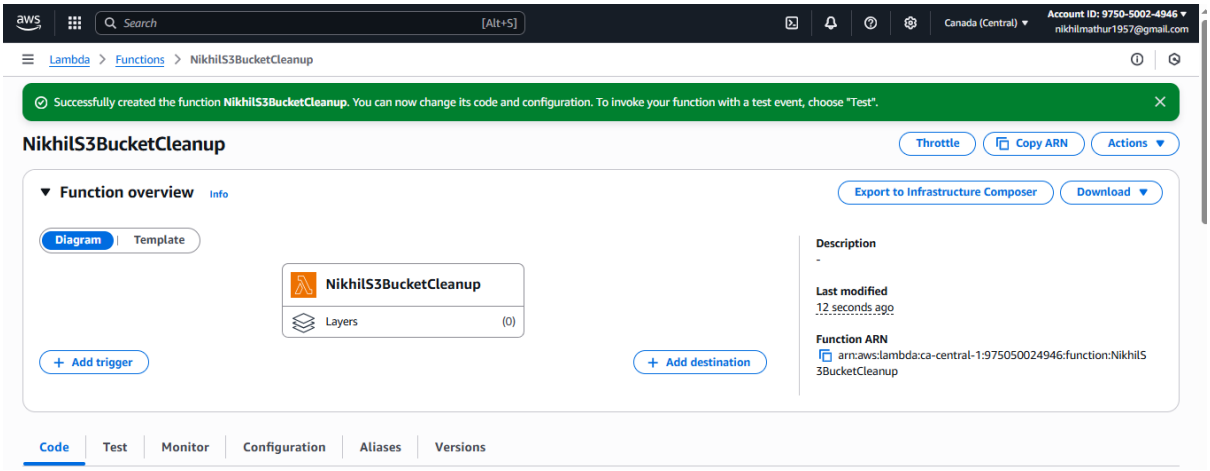




**b. Create the Lambda Function**

- **i. Go to AWS Console => Lambda.**
- **ii. Click Create function.**
- **iii. Select Author from scratch**

iv. **Function name: NikhilS3BucketCleanup**
v. **Runtime: Python 3.13**
vi. **Permissions:**
vii. **Expand Change default execution role.**
viii. **Select Use an existing role.**
ix. **Choose NikhilLambdaS3CleanupRole from the dropdown.**
x. **Note => I choose the role prashantb12-role-9p53470y for permission access to run the code.**
xi. **Click Create function.**




## Code

```python
import boto3
import datetime

def lambda_handler(event, context):
    s3 = boto3.client('s3')
    bucket_name = "nikhilmathur-static-signup-site"
    days_threshold = 10   #10 days old files

    # Calculate cutoff date
    cutoff_date = datetime.datetime.now(datetime.timezone.utc) - datetime.timedelta(days=days_threshold)

    # List all objects in the bucket
    response = s3.list_objects_v2(Bucket=bucket_name)

    if 'Contents' not in response:
        print("Bucket is empty.")
        return {"Deleted": [], "Message": "No files found."}

    deleted_files = []

    for obj in response['Contents']:
        if obj['LastModified'] < cutoff_date:
            s3.delete_object(Bucket=bucket_name, Key=obj['Key'])
            deleted_files.append(obj['Key'])
            print(f"Deleted: {obj['Key']}")

    if not deleted_files:
        print("No files older than threshold found.")
    else:
        print(f"Deleted {len(deleted_files)} files.")

    return {"Deleted": deleted_files}
```

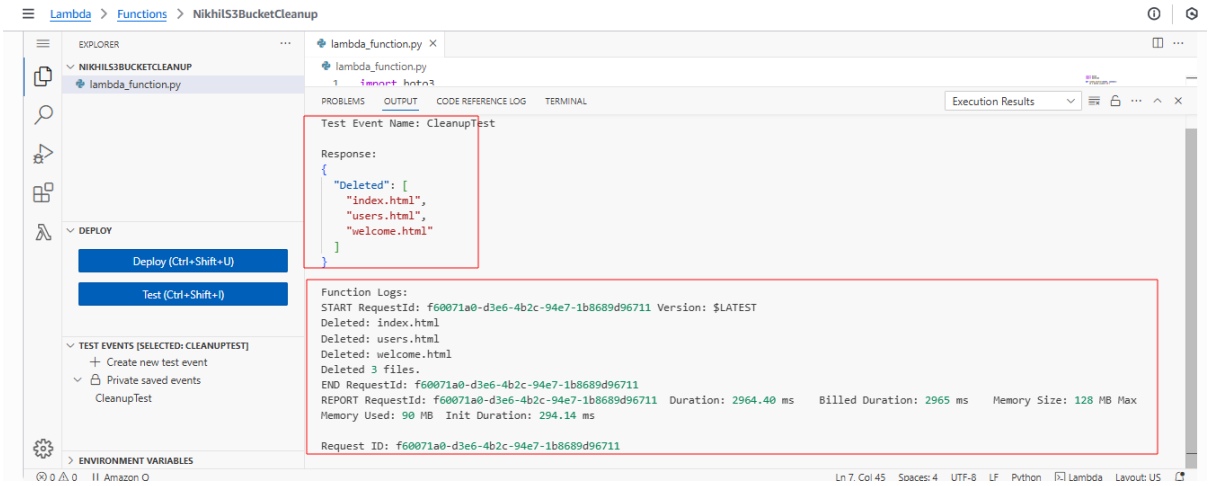
**Click for deploy the code**

**Click Test → Create test event → name it CleanupTest.**

**Keep event JSON as {}.**

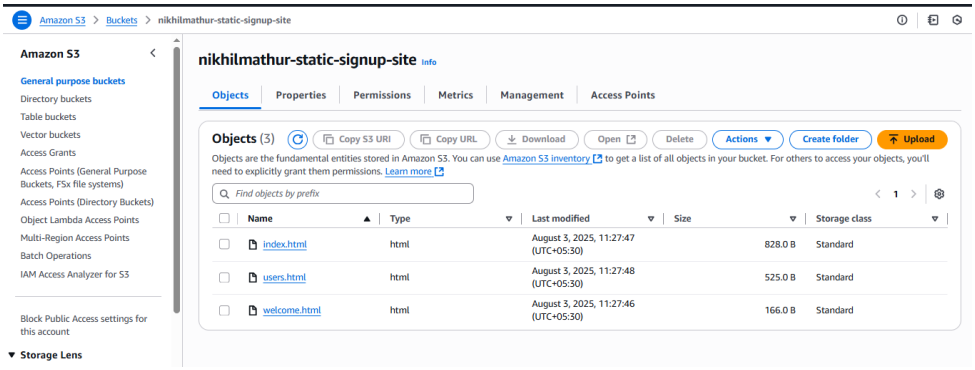
**Run the test.**

**Check the Logs for deleted files list.**




## 3. Manual Invocation:

### Before function run




### After run the function