

Note

The exercises in this course will have an associated charge in your AWS account. In this exercise, you create or use the following resources:

- AWS Identity and Access Management (IAM) policy and user (policies and users are AWS account features, offered at no additional charge)
- Amazon Elastic Compute Cloud (Amazon EC2) instance
- Amazon Simple Storage Service (Amazon S3) bucket

Familiarize yourself with [Amazon EC2 pricing](#), [Amazon S3 pricing](#), and the [AWS Free Tier](#).

Exercise 5: Creating an S3 Bucket and Modifying the EC2 Instance

For this scenario, you create the S3 bucket where the employee photos will be stored.

In this exercise, you create the S3 bucket, upload an object to it, and modify the bucket policy. You also launch an EC2 instance with updated user data so that the application uses the S3 bucket. Finally, you stop the EC2 instance to prevent future costs.

Task 1: Creating an S3 bucket

In this task, you will create an S3 bucket.

1. If needed, log in to the AWS Management Console with your *Admin* user.
2. In the search box, enter `s3` and open the Amazon S3 console by choosing **S3**.
3. Choose **Create bucket**.
4. For **Bucket name**, enter `employee-photo-bucket-<your initials>-<unique number>`.

Example:

```
employee-photo-bucket-al-907
```

5. Choose **Create bucket**.

Task 2: Uploading a photo

In this task, you will upload an object (a photo) to the S3 bucket.

1. Open the details of your newly created bucket by choosing the bucket name.
2. Choose **Upload**.
3. Choose **Add files**.
4. Choose a photo of your choice from your computer and choose **Open**.
5. Choose **Upload**.
At the top, you should see *Upload succeeded* in green.
6. Choose **Close**.

Task 3: Modifying the S3 bucket policy

In this task, you will update the bucket policy. The updated configuration allows the IAM role that you created previously to access the bucket.

1. Choose the **Permissions** tab.
2. Scroll down to **Bucket policy** and choose **Edit**.
3. In the box, paste the following policy:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowS3ReadAccess",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::<INSERT-ACCOUNT-NUMBER>:role/S3DynamoDBFullAccessRole"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::<INSERT-BUCKET-NAME>",
        "arn:aws:s3:::<INSERT-BUCKET-NAME>/*"
      ]
    }
  ]
}
```

4. Replace the `<INSERT-ACCOUNT-NUMBER>` placeholder with your account number.
5. Replace the `<INSERT-BUCKET-NAME>` placeholder with your bucket name.

Example:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowS3ReadAccess",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:role/S3DynamoDBFullAccessRole"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::employee-photo-bucket-al-907",
        "arn:aws:s3:::employee-photo-bucket-al-907/*"
      ]
    }
  ]
}
```

6. Choose **Save changes**.

Task 4: Modifying the application to use the S3 bucket

In this task, you will launch another EC2 instance. This time, you will modify the user data script so that the application uses the S3 bucket.

1. In the **Services** search box, enter `ec2` and open the service by choosing **EC2**.
2. In the navigation pane, under **Instances**, choose **Instances**.
3. Select the `employee-directory-app` instance, which should be in the *Stopped* state.
4. Choose **Actions** and then choose **Image and templates**, **Launch more like this**.
5. For **Name** and at the end of the **Value**, append `-s3`.

Example:

```
employee-directory-app-s3
```

6. For **Key pair name**, select **app-key-pair**.

- Under **Network settings** and **Auto-assign Public IP**, choose **Enable**.
- Scroll down and expand **Advanced Details**.
- In the **User data** box, update the values for the `PHOTOS_BUCKET` variable and (if needed) the `AWS_DEFAULT_REGION` variable.

```
#!/bin/bash --ex
wget https://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/DEV-AKS-PO-GOIN2/FlaskApp.zip
unzip FlaskApp.zip
cd FlaskApp/
yum -y install python3-pip
pip install -r requirements.txt
yum -y install stress
export PHOTOS_BUCKET=${SUB_PHOTOS_BUCKET}
export AWS_DEFAULT_REGION=<INSERT_REGION_HERE>
export DYNAMO_NODE=on
FLASK_APP=application.py /usr/local/bin/flask run --host=0.0.0.0 --port=80
```

Example:

This example uses a sample bucket name.

```
export PHOTOS_BUCKET=employee-photo-bucket-a1-907
```

- Choose **Launch instance**.
- Choose **View all instances**.
The new instance should now be in the **Instances** list.
- Wait for the **Instance state** to change to *Running* and the *Status check* to change to *2/2 checks passed*.
Note: You can refresh the page to update the instance status.
- If needed, clear the check box for the stopped instance that you created previously.
- Select the check box for the **employee-directory-app-s3** instance.
- Copy the **Public IPv4 address**.
Note: Make sure that you only copy the address instead of choosing the **open address** link.
- In a new browser window, paste the IP address that you copied. *Make sure to remove the 'S' after HTTP so you are using only HTTP instead.*

You should see an **Employee Directory** placeholder. You won't be able to interact with the application yet because it's not connected to a database.
Congratulations! You launched an EC2 instance that uses the S3 bucket you created.

Task 5: Deleting the object from the S3 bucket

In this task, you will delete the object that you uploaded to the S3 bucket.

- Open the Amazon S3 console by searching for and choosing **S3**.
- Open the bucket details by choosing the `employee-photo-bucket-` link.
- Select the check box for your object.
- Choose **Delete** and confirm the deletion by entering `permanently delete`.
- Choose **Delete objects** and then choose **Close**.

Task 6: Stopping your EC2 instance

In this task, you will now stop the instance to prevent future costs.

Note: Don't terminate the instance because you will use this instance in a later exercise.

- Return to the Amazon EC2 console by searching for and choosing **EC2**.
- If needed, in the navigation pane, choose **Instances**.
- Select the check box for the **employee-directory-app-s3** instance.
- Choose **Instance state** and then choose **Stop instance**.
- In the dialog box, choose **Stop**.

The **Instance state** will eventually go into the *Stopped* state.