[version_1.1]

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

Familiarize yourself with **Amazon EC2 pricing** and the **AWS Free Tier**.

Exercise 3: Launching an EC2 Instance

For this scenario, you create the employee directory application by using user data that you configure when you set up the EC2 instance.

In this exercise, you log in to the console as the IAM *Admin* user. You then launch an EC2 instance by using the IAM role that you created previously. Finally, after you create the employee directory application, you stop and then terminate the instance. Because this instance launch is a dry run, you terminate the instance to prevent additional costs from incurring.

Task 1: Launching an EC2 instance that uses a role

In this task, you will launch an EC2 instance that hosts the employee directory application.

1. If needed, log in to the AWS Management Console as your *Admin* user.
2. In the **Services** search bar, search for **EC2**, and open the service by choosing **EC2**.
3. In the navigation pane, under **Instances** choose **Instances**.
4. Choose **Launch instances**.
5. For **Name** use `employee-directory-app`.
6. Under **Application and OS Images (Amazon Machine Image)**, choose **Amazon Linux**.
7. Under **Instance type**, select **t2.micro**.
8. Under **Key pair (login)**, choose **Create a new key pair**.
9. For **Key pair name**, paste `app-key-pair`. Choose **Create key pair**. The required **.pem** file should automatically download for you.
10. Under **Network settings** and **Edit**: Keep the default VPC selection, which should

10. Under **Network settings** and **Edit**. Keep the default VPC selection, which should have *(default)* after the network name
    - **Subnet**: Choose the first subnet in the dropdown list
    - **Auto-assign Public IP**: *Enable*

11. Under **Firewall (security groups)** choose **Create security group** use `app-sg` for the **Security group name** and **Description**.

12. Under **Inbound security groups rules** choose **Remove** above the **ssh** rule.

13. Choose **Add security group rule**. For **Type** choose **HTTP**. Under **Source type** choose **Anywhere**.

14. Expand **Advanced details** and under **IAM instance profile** choose **S3DynamoDBFullAccessRole**.

15. In the **User data** box, paste the following code:

```
#!/bin/bash -ex
wget https://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/DEV-AWS-MO-GCNv2/Flask
unzip FlaskApp.zip
cd FlaskApp/
yum -y install python3 mysql
pip3 install -r requirements.txt
amazon-linux-extras install epel
yum -y install stress
export PHOTOS_BUCKET=${SUB_PHOTOS_BUCKET}
export AWS_DEFAULT_REGION=<INSERT REGION HERE>
export DYNAMO_MODE=on
FLASK_APP=application.py /usr/local/bin/flask run --host=0.0.0.0 --port=80
```

16. In the pasted code, change the following line to match your Region (your Region is listed at the top right, next to your user name):

```
export AWS_DEFAULT_REGION=<INSERT REGION HERE>
```

    Example:

    The following example uses the US West (Oregon) Region, or *us-west-2*.

```
export AWS_DEFAULT_REGION=us-west-2
```

    Note: In a later lab, you will modify this user data script again to use your Amazon Simple Storage Service (Amazon S3) bucket. For now, keep `${SUB_PHOTOS_BUCKET}` in the script.

17. Choose **Launch instance**.

18. Choose **View all instances**.

    The instance should now be listed under **Instances**.

19. Wait for the **Instance state** to change to *Running* and the **Status check** to change to *2/2 checks passed*.

Note: Often, the status checks update, but the console user interface (UI) might not update to reflect the most recent information. You can minimize waiting by refreshing the page after a few minutes.

Task 2. Viewing the application

In this task, you will view the application that's running on the instance in a web browser window.

1. Select the instance by selecting its check box.

   Instance information should load on the tabs in the pane.

2. On the **Details** tab, copy the **Public IPv4 address**.

   Note: Make sure that you only copy the address instead of choosing the **open address** link.

3. 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. Right now, you won't be able to interact with it yet because the application isn't connected to a database.

   Congratulations! You have successfully created an EC2 instance, which hosts the employee directory application.

   After you finish exploring the instance, you will stop and terminate your instance so that you don't incur future costs.

Task 3. Cleaning up

In this task, you will terminate the EC2 instance that you launched.

1. Go back to the AWS Management Console.

   The *employee-directory-app* instance should still be selected.

2. At the top of the console pane, choose **Instance state**, choose **Stop instance**, and choose **Stop**.

   The status in the **Instance state** column will eventually go into the *Stopped* state.

   Next, you will terminate the instance.

3. Make sure that check box next to the instance **Name** is selected.

4. Choose **Instance state**, choose **Terminate instance**, and choose **Terminate**.

feedback, or other questions? Contact us at https://support.aws.amazon.com/#/contacts/aws-training. All trademarks are the property of their owners.