*[version_1.0]*

**Note**

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

- AWS Identity and Access Management (IAM) roles
- Amazon Elastic Compute Cloud (Amazon EC2) instance

**The final exercise includes instructions to delete all the resources that you create in the exercises.**

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

# Exercise: Setting Up AWS CodeDeploy

In this exercise, you will create the IAM roles that are needed to complete the exercise. You will launch an EC2 instance and install the CodeDeploy agent onto that instance.

## Task 1: Setting up IAM

In this task, you create the required IAM roles and assign the necessary permissions to them.

1. In the AWS Management Console, choose **Services**, and search for and open **IAM**.

2. In the navigation pane, choose **Roles** and then choose **Create role**.

3. In the **Create role** page, configure the following settings.
   - **Select type of trusted entity**: Keep *AWS service* selected
   - **Choose a use case**: *CodeDeploy*
   - **Select your use case**: *CodeDeploy*

4. Choose **Next: Permissions**.

5. Skip the **Attached permissions policies** and **Add tags (optional)** pages by choosing **Next: Tags** and then choosing **Next: Review**.

6. For **Role name**, enter `CodeDeployServiceRole` and choose **Create role**.

7. Create another role by choosing **Create role** and configuring these role settings.
   - **Select type of trusted entity**: Again, keep *AWS service* selected
   - **Choose a use case**: *EC2*

8. Choose **Next: Permissions**.

9. In the **Filter policies** search box, enter `s3full` and select the **AmazonS3FullAccess** policy.
10. Search the policies again for `ssmmanaged` and select the **AmazonSSMManagedInstanceCore** policy.
11. Skip the next two pages by choosing **Next: Tags** and then choosing **Next: Review**.
12. For **Role name**, enter `EC2S3FullAccess` and choose **Create role**.

# Task 2: Setting up the EC2 instance

In this task, you configure and launch an EC2 instance to work with AWS Systems Manager Session Manager.

1. In the console, choose **Services**, and search for and open **EC2**. *Make sure you are in the Oregon (us-west-2) Region*.
2. In the navigation pane, choose **Instances** and then choose **Launch instances**.
3. From the list of Amazon Machine Images (AMIs), select the **Amazon Linux 2 AMI (HVM), SSD Volume Type**, and keep the *t2.micro* instance type selected.
4. Choose **Next: Configure Instance Details**.
5. In the **Configure Instance Details** page, configure the following settings, but keep the remaining default values.
   - **Auto-assign Public IP**: *Enable*
   - **IAM role**: *EC2S3FullAccess*
6. Choose **Next: Add Storage** and skip the **Add Storage** page by choosing **Next: Add Tags**.
7. Choose **Add Tag** and enter the following key-value pair.
   - **Key**: `Name`
   - **Value**: `TEST-environment`
8. Choose **Next: Configure Security Group**.
9. Configure the security group by selecting the following settings.
   - **Type**: *HTTP*
   - **Source**: *Anywhere*
10. Choose **Review and Launch**, and then choose **Launch**.
11. In the dialog box, select **Proceed without a key pair**, select the acknowledgement box, and choose **Launch Instances**.
12. In the **Launch Status** screen, open the list of EC2 instances by choosing the instance ID.
13. Wait for the **Instance state** column to change to *Running* and the **Status check** column to change to *passed*.

# Task 3: Installing the CodeDeploy agent

In this task you, install the CodeDeploy agent onto the EC2 instance.

1. From the list of instances, open the **TEST-environment** instance details by choosing the **Instance ID** link.

2. In the instance summary, choose **Connect**.

3. In **Connect to instance**, choose the **Session Manager** tab and then choose **Connect**.

4. In the terminal window, install the CodeDeploy agent by running the following commands. **Note**: *wget might already be installed.*

```
sudo yum update -y
sudo yum install ruby -y
sudo yum install wget -y
```

5. Install the latest version of the agent.

```
cd ~
wget https://aws-codedeploy-us-west-2.s3.us-west-2.amazonaws.com/latest/install
chmod +x ./install
sudo ./install auto
```

6. Verify that the agent is running.

```
sudo service codedeploy-agent status
```

**Note**: *If the agent does not show as running, try the following commands.*

```
``` bash
sudo service codedeploy-agent start
sudo service codedeploy-agent status
```
```

1. If you will work on the next exercise at a later time, you can choose to stop the instance so you don't incur any additional cost. To stop the instance:
   - In the **Instances** page, select the **TEST-environment** check box.
   - From the **Instance state** menu, select **Stop instance**.
   - You can restart the instance by choosing **Instance state** and selecting **Start instance**.

You will use this instance in the next exercise, where you will set up the CodeDeploy group and make some revisions to the code.