Evolution of a website: Going from single server to serverless

**SPL-TF-100-NWSSTS-1-EN - Version 1.0.8**

© 2024 Amazon Web Services, Inc. or its affiliates. All rights reserved. This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited. All trademarks are the property of their owners.

Note: Do not include any personal, identifying, or confidential information into the lab environment. Information entered may be visible to others.

Corrections, feedback, or other questions? Contact us at [*AWS Training and Certification*](https://support.aws.amazon.com/#/contacts/aws-training).

**Lab Overview**

AnyCompany Ice Cream hires you to create and maintain their website. In this lab, you follow the growth of the business and how it evolves their website’s infrastructure.

You start by creating the company’s static website hosted on Amazon Simple Storage Service (S3) that displays the ice cream flavors. The business starts to gain popularity among the locals and flavors start to sell out before the day is up. The business owners ask you to migrate to a server where they can have you push updates. They decide to migrate their static site to Amazon Elastic Compute Cloud (EC2) with a Linux, Apache, MySQL, PHP (LAMP) stack built on it.

This will allow the website to update and react dynamically. Demand for AnyCompany’s ice cream surges after a viral video and the infrastructure and maintenance becomes difficult to manage. To scale with demand, you suggest migrating sections of their LAMP stack to serverless services, starting with the MySQL database. The MySQL database will be replaced with Amazon DynamoDB serverless database.

LAB OBJECTIVES

In this lab, you:

* Upload files to Amazon S3 to host a static website.
* Migrate the dynamic website hosted on EC2 from using a MySQL database to using Amazon DynamoDB.
* Update the DynamoDB table with a new ice cream flavor.
* Refresh the website to reveal your new ice cream flavor.

DURATION

This lab requires approximately *45* minutes to complete.

ICON KEY

Various icons are used throughout this lab to call attention to different types of instructions and notes. The following list explains the purpose for each icon:

* **Command:** A command that you must run.
* **Expected output:** A sample output that you can use to verify the output of a command or edited file.
* **Note:** A hint, tip, or important guidance.
* **CAUTION:** Information of special interest or importance (not so important to cause problems with the equipment or data if you miss it, but it could result in the need to repeat certain steps).

**Start lab**

1. To launch the lab, at the top of the page, choose **Start lab**.

**Caution:** You must wait for the provisioned AWS services to be ready before you can continue.

1. To open the lab, choose **Open Console**.

You are automatically signed in to the AWS Management Console in a new web browser tab.

**WARNING:** **Do not change the Region unless instructed.**

COMMON SIGN-IN ERRORS

**Error: You must first sign out**



If you see the message, **You must first log out before logging into a different AWS account:**

* Choose the **click here** link.
* Close your **Amazon Web Services Sign In** web browser tab and return to your initial lab page.
* Choose **Open Console** again.

**Error: Choosing Start Lab has no effect**

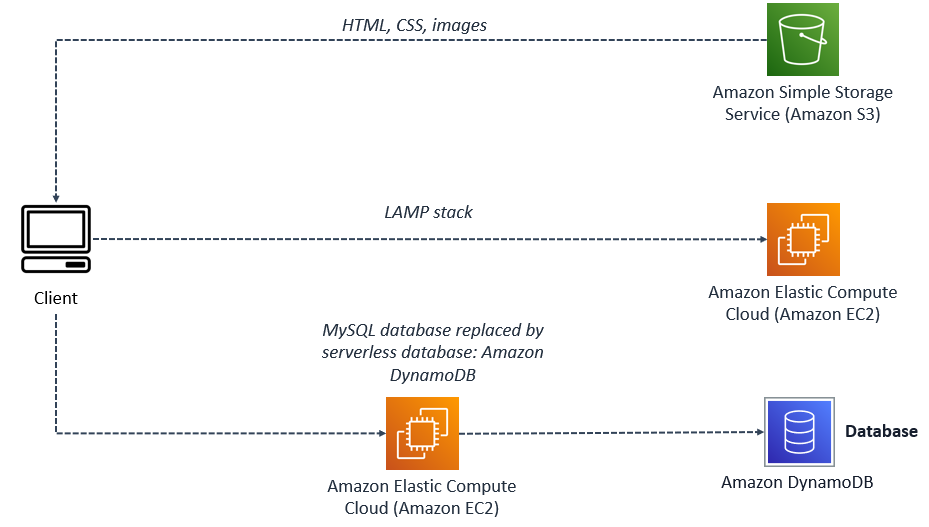
In some cases, certain pop-up or script blocker web browser extensions might prevent the **Start Lab** button from working as intended. If you experience an issue starting the lab:

* Add the lab domain name to your pop-up or script blocker’s allow list or turn it off.
* Refresh the page and try again.

APPLICATION ARCHITECTURE

The application architecture uses the following services:

* Static site: Amazon S3 which contains HTML, CSS, and images.
* Dynamic site: Amazon EC2 Linux, Apache web server, MySQL database, and PHP (LAMP) stack will come pre installed. This will allow the owner or developer to update the flavors available for the day.
* Serverless: Amazon DynamoDB. This will replace the MySQL database and make the database portion serverless.



*Figure: The preceding architecture shows three different ways the client is accessing information. The client can access static content through Amazon S3 which holds the HTML, CSS, and image files. Dynamic content can be accessed through the dynamic site hosted on Amazon EC2. The site has a database, and is able to update content through its server. The last way to access this site is keeping the Amazon EC2 instance, and migrating the database to a serverless database, which is seen in this diagram.*

**Task 1: Upload files to Amazon S3**

In this task, AnyCompany Ice Cream would like you to create a static website which displays their ice cream flavors. To help speed up this step, this task is already configured with an S3 bucket and the following actions have already been completed.

* The [website hosting](https://docs.aws.amazon.com/AmazonS3/latest/userguide/EnableWebsiteHosting.html) feature is enabled to allow this bucket to host a website.
* Additionally, a **bucket policy** was added to allow **s3:GetObject** permissions to this bucket and the files within the bucket.
* The **index document** property has been set to **index.html** in the root of the bucket. This is a file you just copied to the bucket.

The static website files have already been created including the **index.html** file, **css** folder, and an **images** folder. In this task, you download the zip file containing these necessary files. Then you will upload them to the **s3htmlbucket** to finish configuring the website hosted using an Amazon S3 bucket.

TASK 1.1: DOWNLOAD THE STATIC-SITE.ZIP FILE

In this task you download the static website files to your local machine.

1. Download the **static-site.zip** file containing the static website files to your local machine *(Save link as…)*.

* [Static-Site.zip](https://us-west-2-aws-training.s3.amazonaws.com/courses/SPL-TF-100-NWSSTS/v1.0.8.prod-ff989be4/scripts/static-site.zip)

1. **Unzip** the files to a temporary directory or the desktop.

TASK 1.2: UPLOAD THE FILES TO AMAZON S3

In this task, you use the S3 management console to upload the static site files to the **s3htmlbucket** that will host your website.

1. At the top of the AWS Management Console, in the search bar, search for and choose

S3

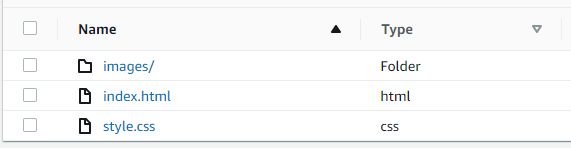
.

1. Choose the text link for the bucket with **s3htmlbucket** in the name.
2. Open the location where you unzipped the static website files to on your machine. Select all of the files and folders and drag them to the **Files and folders** area of the **Object** page.
3. Choose **Upload** at the bottom of the page.

**Expected service output:**

**Upload succeeded**

1. Choose the **Close** button.
2. Your bucket listing should look similar to the example below:



*Figure: The index.html document is in the root of the bucket. The****images****folder has the ice cream image inside of it and the****css****style sheet gives the****index.html****document colors, fonts, and size.*

TASK 1.3: TEST THE STATIC S3 WEBSITE

In this task, you test the static S3 website to ensure it works as expected.

1. Choose the **Objects** tab located at the top of the page.
2. Navigate to the **static-site/index.html** section, located at the end of the page.
3. Copy and paste the object URL in a new browser tab to view the website.

**Congratulations!** You have successfully crated a static website for AnyCompany Ice Cream.

**Task 2: Verify the dynamic website on Amazon EC2 is running**

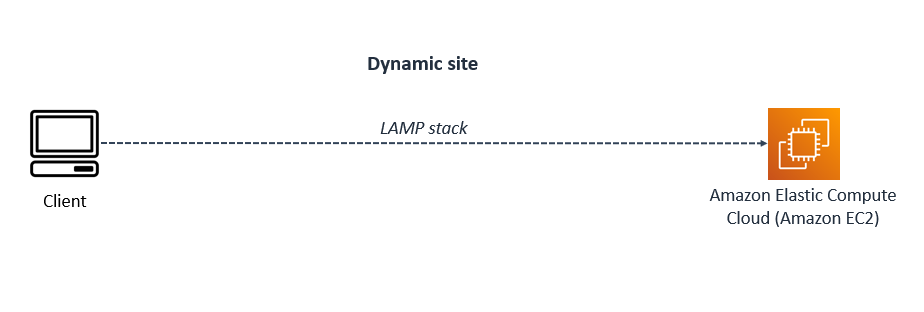
The business starts to grow, and customers want a feature on the website where they can see an updated flavor of the day, or when a flavor sells out. You suggest to migrate the server to an Amazon EC2 instance with LAMP installed on it. This allows the site to generate dynamic content using a database and allows you to update flavors as inventory is added or runs out.

In this task, you verify the new dynamic website is working and that data about the inventory for available ice cream flavors is returned from the database. You copy the IP address for the EC2 instance and paste it into a new browser tab to see if the website pulls up.

To help speed up this step, this task is already configured with a Amazon VPC and its networking resources, and an Amazon EC2 with a LAMP stack configured. The following actions have already been completed.

* The [LAMP stack](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-lamp-amazon-linux-2.html) is installed as a web server to host AnyCompany’s dynamic website.
* Additionally, an **IAM Policy** is used to define the permissions for each service used.
* Flavors and prices are now updated through a MySQL database that dynamically updates the website.

**Architecture Overview**



*Figure: In this architecture, the Operating System (OS) for this Amazon EC2 instance is Linux, the HTTP server is Apache, the MySQL database is MariaDB, and the scripting language used is PHP. All of these configured together, run the dynamic website you open shortly. A dynamic site is information or web pages that are generated in real time. With a LAMP stack, customers will be able to input orders, and submit them.*

1. At the top of the AWS Management Console, in the search bar, search for and choose

EC2

.

1. Choose **Instances** from the left navigation pane.
2. Check the box next to the instance with the name  **Ice Cream**.

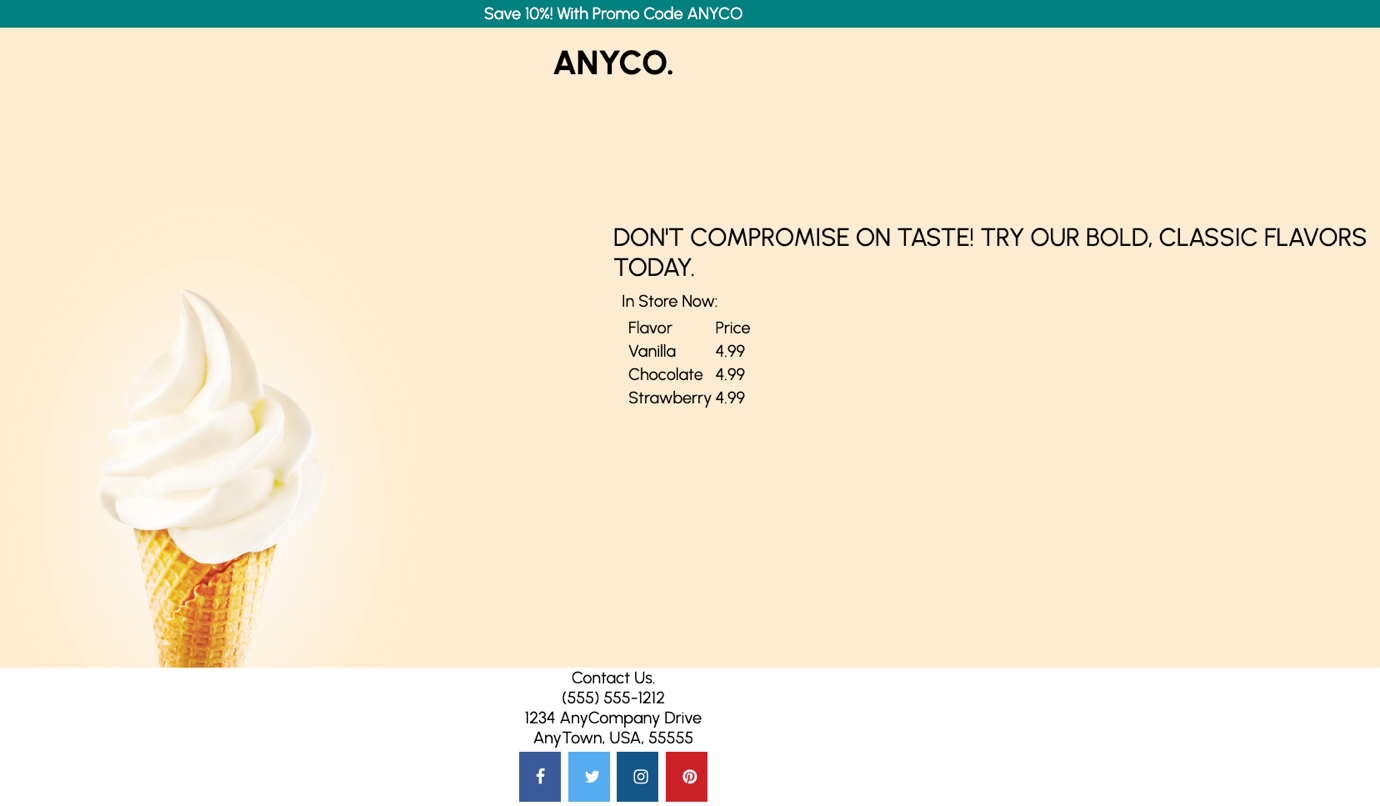
**Note:** A detailed notification box appears following the selected instance. Each tab gives you an overview of the instance such as its *public or private IPv4 address*, *Public IPv4 DNS name*, the *VPC* it is assigned to, *security associations*, *status* of the instance, and *networking* associations.

1. In the **Details** tab, copy the **Public IPv4 IP address** value.
2. Paste the **Public IPv4 IP address** into a new browser tab to see the dynamic website running on EC2. Leave this tab **open** for **task 3**.

 Use HTTP and not HTTPS in the browser.

http://<PUBLIC IPv4 ADDRESS>

**Expected website:**



*Figure: Depicted is the home page for AnyCompany Ice Cream with the photo of a waffle cone filled with vanilla ice cream and a listing of available ice cream flavors and prices.*

**Congratulations!** You have successfully verified the dynamic website using a MySQL database to show inventory of available ice cream flavors is working as expected.

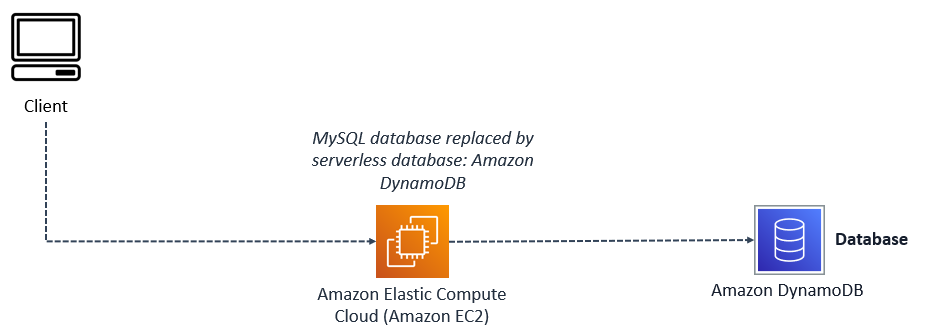
**Task 3: Update dynamic website to a serverless solution using DynamoDB for the MySQL database**

Business has been steadily growing but with a recent video of a customer’s experience that has since gone viral, the AnyCompany Ice Cream website has experienced more web traffic than the current MySQL database server can keep up with. Unfortunately, customers are getting errors accessing the website. To remedy this, you are tasked with migrating the local MySQL database to Amazon DynamoDB since it can scale with the demands of customer web traffic to ensure customers continue to reach the AnyCompany Ice Cream website without interruptions.

**Note:** The Amazon DynamoDB database has already been created and the existing MySQL table has already been imported.

* The [Amazon DynamoDB](https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html) table is installed, however, the items in the table need to be created.
* A script is created to replace the old index.php with a new index.php.
* IAM user, and roles are adjusted to allow permissions to access Amazon DynamoDB.

**Architecture Overview**



*Figure: In this architecture, the Operating System (OS) for this Amazon EC2 instance is Linux, the HTTP server is Apache, the MySQL database has been replaced with an Amazon DynamoDB serverless database, and the scripting language used is PHP. All of these configured together run the dynamic website you open shortly. A dynamic site is information or web pages that are generated in real time.*

In this task, you complete the following steps:

* Update the database with a new flavor using the Amazon DynamoDB console.
* Update the **index.php** file to point to the Amazon DynamoDB database using a bash script.
* Refresh the website to see that it is now pulling in content from the Amazon DynamoDB database revealing the the new flavor you just added.

TASK 3.1: UPDATE THE AMAZON DYNAMODB PRODUCTS TABLE WITH A NEW ICE CREAM FLAVOR

In this task, you create a new ice cream flavor in the Products table using the Amazon DynamoDB console.

1. At the top of the AWS Management Console, in the search bar, search for and choose

DynamoDB

.

1. In the left navigation pane, choose **Tables**.
2. Choose the text link for the table named Products.
3. Choose **Explore table items** .
4. In the **Items returned** section, choose **Create item** and configure the following options:

* **Attribute name:** id - Partition key
* **Value:**

1

1. Choose **Add new attribute** **>** **String** and then configure the following options:

* **Attribute name:**

ProductFlavor

* **Value:**

Cinnamon-spiced Chocolate

**Note:** You can put in any flavor name you want for the **Value** field.

1. Choose **Add new attribute** **>** **String** and then configure the following options:

* **Attribute name:**

ProductPrice

* **Value:**

7.99

1. Choose **Create Item** .

TASK 3.2: UPDATE THE WEBSITE TO USE AMAZON DYNAMODB

In this task, you connect to your EC2 instance using Session Manager and run a script to migrate the website from using the local MySQL database to using the serverless database Amazon DynamoDB.

1. At the top of the AWS Management Console, in the search bar, search for and choose

EC2

.

1. Choose **Instances** from the left navigation pane.
2. Check the box next to the instance with the name  **Ice Cream**.
3. In the top section of the console, choose **Connect** .
4. From the **Connect to instance** page, choose the **Session Manager** tab, then choose **Connect** .
5. **Command:** From the terminal session copy and paste the following command to migrate the website from using the MySQL database to Amazon DynamoDB.

sudo /var/www/html/migrate-database.sh

**Expected output:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\* This is OUTPUT ONLY. \*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Renaming index.php to compute-index.php

Renaming serverless-index.php to index.php

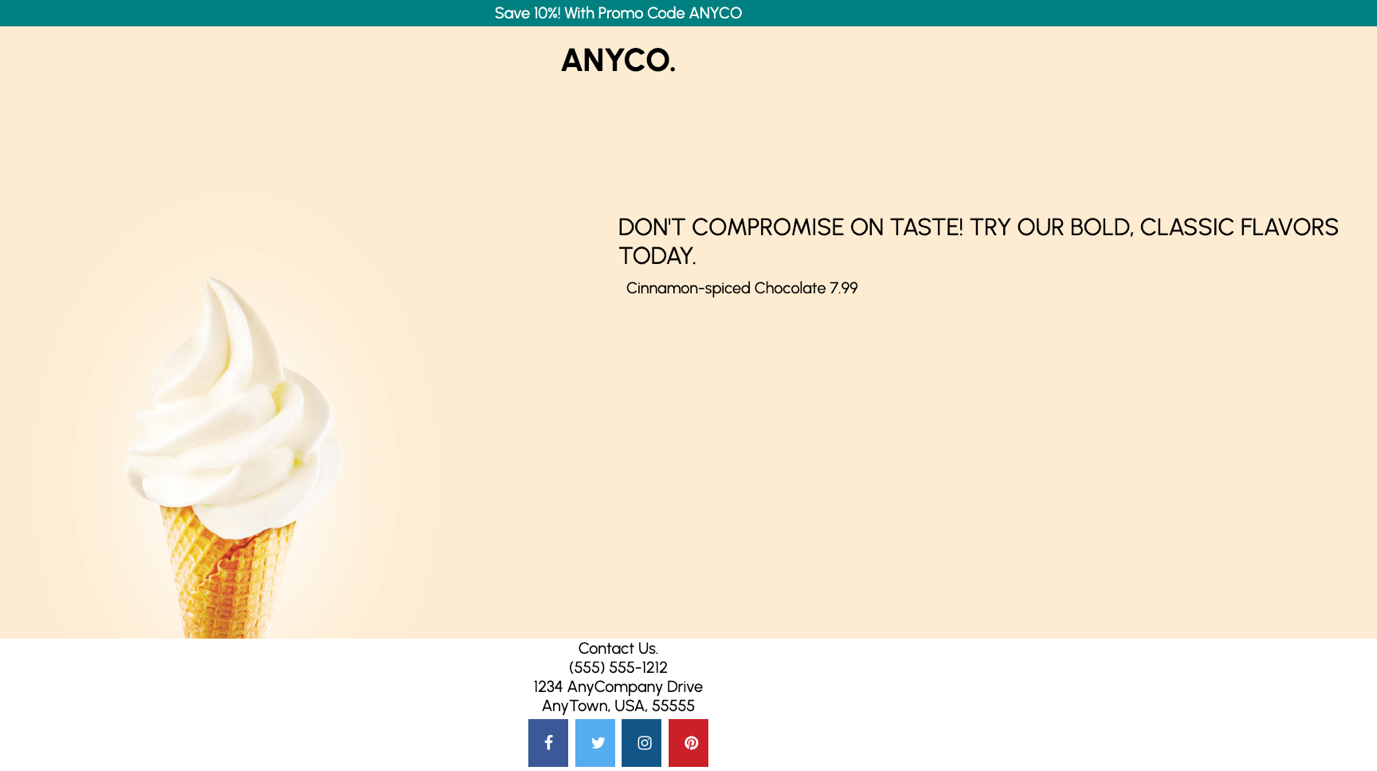
Restarting Apache

TASK 3.3 REFRESH WEBSITE TO SEE YOUR NEW ICE CREAM FLAVOR

In this task, you refresh the browser to show the new ice cream flavor you added to the Amazon DynamoDB Products table.

1. Switch back to the browser tab opened to the AnyCompany Ice Cream website and choose the  refresh icon to update the website and see your new ice cream flavor.

**Expected website:**



*Figure: The website is updated with the new flavor and price listed from the Amazon DynamoDB Products table that was just created.*

**Congratulations!** You have successfully added a new ice cream flavor and price to the Amazon DynamoDB Products table, migrated the website to use Amazon DynamoDB instead of MySQL for the database, and refreshed the website to reveal your new ice cream flavor.

**Conclusion**

**Congratulations!** You now have successfully:

* Uploaded files to Amazon S3 to host a static website.
* Migrated the dynamic website hosted on EC2 from using a MySQL database to using Amazon DynamoDB.
* Updated the DynamoDB table with a new ice cream flavor.
* Refreshed the website to reveal your new ice cream flavor.

**End lab**

Follow these steps to close the console and end your lab.

1. Return to the **AWS Management Console**.
2. At the upper-right corner of the page, choose **AWSLabsUser**, and then choose **Sign out**.
3. Choose **End lab** and then confirm that you want to end your lab.

**Additional Resources**

* For more information about Amazon DynamoDB, see [Amazon DynamoDB](https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html).
* For more information about how to configure and enable static website hosting on Amazon S3, see [Amazon S3](https://docs.aws.amazon.com/AmazonS3/latest/userguide/EnableWebsiteHosting.html).
* For more information about how to install and configure a LAMP stack on Amazon EC2, see [Amazon EC2](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/install-LAMP.html).

For more information about AWS Training and Certification, see [*https://aws.amazon.com/training/*](https://aws.amazon.com/training/).

*Your feedback is welcome and appreciated.*  
If you would like to share any feedback, suggestions, or corrections, please provide the details in our [*AWS Training and Certification Contact Form*](https://support.aws.amazon.com/#/contacts/aws-training).