

Document Generated: 07/20/2024

Learning Style: Virtual Classroom

Provider:

Difficulty: Intermediate

Course Duration: 1 Day

## Develop Generative AI Solutions with Azure OpenAI Service (AI-050)



If you enroll in this course at the listed price, you receive a Free Official Exam Voucher for the **AI-050T00** Exam. This course does not include Exam Voucher if enrolled within the Master Subscription, however, you can request to purchase the Official Exam Voucher separately.

## About this Course:

This course provides a comprehensive exploration of the fascinating realm of prompt engineering using the Azure OpenAI Service. It immerses participants in a learning experience that combines practical exercises and illustrative examples, enabling them to grasp the art of tailoring AI responses to closely match real-world scenarios. Upon completing this course, participants will have acquired the expertise and proficiency required to craft potent Generative AI Solutions on the Azure platform, making them well-equipped to harness the capabilities of AI for practical applications

## Course Objectives

- Describe Azure OpenAI workloads and access the Azure OpenAI Service
- Understand generative AI models.
- Understand Azure OpenAI's language, code, and image capabilities.
- Understand Azure OpenAI's responsible AI practices and limited access policies.
- Create an Azure OpenAI Service resource and understand types of Azure OpenAI base models.
- Use the Azure OpenAI Studio, console, or REST API to deploy a base model and test it in the Studio's playgrounds.
- Generate completions to prompts and begin to manage model parameters.
- Integrate Azure OpenAI into your application
- Differentiate between different endpoints available to your application
- Generate completions to prompts using the REST API and language specific SDKs
- Understand the concept of prompt engineering and its role in optimizing Azure OpenAI models' performance.
- Know how to design and optimize prompts to better utilize AI models.
- Include clear instructions, request output composition, and use contextual content to improve the quality of the model's responses.

## Audience:

The audience for this course includes software developers and data scientists who need to use large language models for generative AI. Some programming experience is recommended, but the course will be valuable to anyone seeking to understand how the Azure OpenAI service can be used to implement generative AI solutions.

## Prerequisites:

**Before starting this learning path, you should already have:**

- Familiarity with Azure and the Azure portal.
- Experience programming with C# or Python

## Course Outline:

### Module 1: Introduction to Azure OpenAI Service

Get to know the connection between artificial intelligence (AI), Responsible AI, and text, code, and image generation. Understand how you can use Azure OpenAI to build solutions against AI models within Azure.

#### Learning Objectives

- Describe Azure OpenAI workloads and access the Azure OpenAI Service
- Understand generative AI models.
- Understand Azure OpenAI's language, code, and image capabilities.
- Understand Azure OpenAI's responsible AI practices and limited access policies.

#### Lessons

- What is generative AI
- Describe Azure OpenAI
- How to use Azure OpenAI
- Understand OpenAI's natural language capabilities.
- Understand OpenAI code generation capabilities.
- Understand OpenAI's image generation capabilities.
- Describe Azure OpenAI's access and responsible AI policies.

### Module 2: Get started with Azure OpenAI Service

This module provides engineers with the skills to begin building an Azure OpenAI Service solution.

#### Learning Objectives

- Create an Azure OpenAI Service resource and understand types of Azure OpenAI base models.
- Use the Azure OpenAI Studio, console, or REST API to deploy a base model and test it in the Studio's playgrounds.
- Generate completions to prompts and begin to manage model parameters.

#### Lessons

- Access Azure OpenAI Service
- Use Azure OpenAI Studio
- Explore types of generative AI models
- Deploy generative AI models
- Use prompts to get completions from models
- Test models in Azure OpenAI Studios playgrounds

## **Module 3: Build natural language solutions with Azure OpenAI Service**

This module provides engineers with the skills to begin building apps that integrate with the Azure OpenAI Service.

### **Learning Objectives:**

- Integrate Azure OpenAI into your application
- Differentiate between different endpoints available to your application
- Generate completions to prompts using the REST API and language specific SDKs

### **Lessons**

- Integrate Azure OpenAI into your app
- Use Azure OpenAI REST API
- Use Azure OpenAI SDK

## **Module 4: Apply prompt engineering with Azure OpenAI Service**

Prompt engineering in Azure OpenAI is a technique that involves designing prompts for natural language processing models. This process improves accuracy and relevancy in responses, optimizing the performance of the model.

### **Learning Objectives**

- Understand the concept of prompt engineering and its role in optimizing Azure OpenAI models' performance.
- Know how to design and optimize prompts to better utilize AI models.
- Include clear instructions, request output composition, and use contextual content to improve the quality of the model's responses.

### **Lessons**

- Understand prompt engineering
- Write more effective prompts
- Provide context to Improve accuracy

## **Credly Badge:**

### **Display your Completion Badge And Get The Recognition You Deserve.**

Add a completion and readiness badge to your LinkedIn profile, Facebook page, or Twitter account to validate your professional and technical expertise. With badges issued and validated by Credly, you can:



- Let anyone verify your completion and achievement by clicking on the badge
- Display your hard work and validate your expertise
- Display each badge's details about specific skills you developed.

Badges are issued by QuickStart and verified through Credly.

[Find Out More](#) or [See List Of Badges](#)