

## AI-102T00 A: Designing and Implementing a Microsoft Azure AI Solution

### Course Summary

#### Description

AI-102 Designing and Implementing an Azure AI Solution is intended for software developers wanting to build AI infused applications that leverage Azure AI Services, Azure AI Search, and Microsoft Bot Framework. The course will use C# or Python as the programming language.

#### Topics

- Prepare to develop AI solutions on Azure
- Create and consume Azure AI Services
- Secure Azure AI Services
- Monitor Azure AI Services
- Deploy Azure AI services in containers
- Analyze Images
- Classify Images
- Detect, analyze, and recognize faces
- Read Text in images and documents with the Azure AI Vision Service
- Analyze video
- Analyze text with Azure AI Language
- Build a question answering solution
- Build a conversational language understanding model
- Create a custom text classification solution
- Create a custom named entity extraction solution
- Translate text with Azure AI Translator service
- Create speech-enabled apps with Azure AI services
- Translate speech with the Azure AI Speech service
- Create an Azure AI Search solution
- Create a custom skill for Azure AI Search
- Create a knowledge store with Azure AI Search
- Plan an Azure AI Document Intelligence solution
- Use prebuilt Azure AI Document Intelligence models
- Extract data from forms with Azure Document Intelligence
- Get started with Azure OpenAI Service
- Build natural language solutions with Azure OpenAI Service
- Apply prompt engineering with Azure OpenAI Service
- Generate code with Azure OpenAI Service
- Generate images with Azure OpenAI Service
- Use your own data with Azure OpenAI Service
- Fundamentals of Responsible Generative AI

#### Audience

Software engineers concerned with building, managing and deploying AI solutions that leverage Azure AI Services, Azure AI Search, and Azure OpenAI. They are familiar with C# or Python and have knowledge on using REST-based APIs to build computer vision, language analysis, knowledge mining, intelligent search, and generative AI solutions on Azure.

#### Prerequisites

Before attending this course, students must have:

- Knowledge of Microsoft Azure and ability to navigate the Azure portal
- Knowledge of either C# or Python
- Familiarity with JSON and REST programming semantics

#### Duration

Four days

## AI-102T00 A: Designing and Implementing a Microsoft Azure AI Solution

### Course Outline

- I. *Prepare to develop AI solutions on Azure*
  - A. Define artificial intelligence
  - B. Understand AI-related terms
  - C. Understand considerations for AI Engineers
  - D. Understand considerations for responsible AI
  - E. Understand capabilities of Azure Machine Learning
  - F. Understand capabilities of Azure AI Services
  - G. Understand capabilities of Azure OpenAI Service
  - H. Understand capabilities of Azure AI Search
- II. *Create and consume Azure AI Services*
  - A. Create Azure AI services resources in an Azure subscription.
  - B. Identify endpoints, keys, and locations required to consume an Azure AI services resource.
  - C. Use a REST API and an SDK to consume Azure AI services.
- III. *Secure Azure AI Services*
  - A. Consider authentication for Azure AI services
  - B. Manage network security for Azure AI services
- IV. *Monitor Azure AI Services*
  - A. Monitor Azure AI services costs.
  - B. Create alerts and view metrics for Azure AI services.
  - C. Manage Azure AI services diagnostic logging.
- V. *Deploy Azure AI services in containers*
  - A. Create containers for reuse
  - B. Deploy to a container and secure a container
  - C. Consume Azure AI services from a container
- VI. *Analyze Images*
  - A. Provision an Azure AI Vision resource
  - B. Analyze an image
  - C. Generate a smart-cropped thumbnail
- VII. *Classify Images*
  - A. Provision Azure resources for Azure AI Custom Vision
  - B. Understand image classification
  - C. Train an image classifier
- VIII. *Detect, analyze, and recognize faces*
  - A. Identify options for face detection, analysis, and identification
  - B. Understand considerations for face analysis
  - C. Detect faces with the Azure AI Vision service
  - D. Understand capabilities of the Face service
  - E. Compare and match detected faces
  - F. Implement facial recognition
- IX. *Read Text in images and documents with the Azure AI Vision Service*
  - A. Read text from images using OCR
  - B. Use the Azure AI Vision service Image Analysis with SDKs and the REST API
  - C. Develop an application that can read printed and handwritten text
- X. *Analyze video*
  - A. Describe Azure Video Indexer capabilities
  - B. Extract custom insights
  - C. Use Azure Video Indexer widgets and APIs
- XI. *Analyze text with Azure AI Language*
  - A. Detect language from text
  - B. Analyze text sentiment
  - C. Extract key phrases, entities, and linked entities
- XII. *Build a question answering solution*
  - A. Understand question answering and how it compares to language understanding
  - B. Create, test, publish and consume a knowledge base
  - C. Implement multi-turn conversation and active learning
  - D. Create a question answering bot to interact with using natural language

## AI-102T00 A: Designing and Implementing a Microsoft Azure AI Solution

### Course Summary (cont'd)

#### *XIII. Build a conversational language understanding model*

- A. Provision Azure resources for Azure AI Language resource
- B. Define intents, utterances, and entities
- C. Use patterns to differentiate similar utterances
- D. Use pre-built entity components
- E. Train, test, publish, and review an Azure AI Language model

#### *XIV. Create a custom text classification solution*

- A. Understand types of classification projects
- B. Build a custom text classification project
- C. Tag data, train, and deploy a model
- D. Submit classification tasks from your own app

#### *XV. Create a custom named entity extraction solution*

- A. Understand custom named entities and how they're labeled.
- B. Build a Language service project.
- C. Label data, train, and deploy an entity extraction model.
- D. Submit extraction tasks from your own app.

#### *XVI. Translate text with Azure AI Translator service*

- A. Provision a Translator resource
- B. Understand language detection, translation, and transliteration
- C. Specify translation options
- D. Define custom translations

#### *XVII. Create speech-enabled apps with Azure AI services*

- A. Provision an Azure resource for the Azure AI Speech service
- B. Use the Azure AI Speech to text API to implement speech recognition
- C. Use the Text to speech API to implement speech synthesis
- D. Configure audio format and voices
- E. Use Speech Synthesis Markup Language (SSML)

#### *XVIII. Translate speech with the Azure AI Speech service*

- A. Provision Azure resources for speech translation.
- B. Generate text translation from speech.
- C. Synthesize spoken translations.

#### *XIX. Create an Azure AI Search solution*

- A. Create an Azure AI Search solution
- B. Develop a search application

#### *XX. Create a custom skill for Azure AI Search*

- A. Implement a custom skill for Azure AI Search
- B. Integrate a custom skill into an Azure AI Search skillset

#### *XXI. Create a knowledge store with Azure AI Search*

- A. Create a knowledge store from an Azure AI Search pipeline
- B. View data in projections in a knowledge store

#### *XXII. Plan an Azure AI Document Intelligence solution*

- A. Describe the components of an Azure AI Document Intelligence solution.
- B. Create and connect to Azure AI Document Intelligence resources in Azure.
- C. Choose whether to use a prebuilt, custom, or composed model.

#### *XXIII. Use prebuilt Azure AI Document Intelligence models*

- A. Identify business problems that you can solve by using prebuilt models in Azure AI Document Intelligence.
- B. Analyze forms by using the General Document, Read, and Layout models.
- C. Analyze forms by using financial, ID, and tax prebuilt models

#### *XXIV. Extract data from forms with Azure Document Intelligence*

- A. Identify how Azure Document Intelligence's layout service, prebuilt models, and custom service can automate processes
- B. Use Azure Document Intelligence's Optical Character Recognition (OCR) capabilities with SDKs, REST API, and Azure Document Intelligence Studio
- C. Develop and test custom models

## AI-102T00 A: Designing and Implementing a Microsoft Azure AI Solution

---

### Course Summary (cont'd)

#### XXV. *Get started with Azure OpenAI Service*

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

#### XXVI. *Build natural language solutions with Azure OpenAI Service*

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

#### XXVII. *Apply prompt engineering with Azure OpenAI Service*

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

#### XXVIII. *Generate code with Azure OpenAI Service*

- A. Use natural language prompts to write code
- B. Build unit tests and understand complex code with AI models
- C. Generate comments and documentation for existing code

#### XXIX. *Generate images with Azure OpenAI Service*

- A. Describe the capabilities of DALL-E in the Azure openAI service
- B. Use the DALL-E playground in Azure OpenAI Studio

#### XXX. *Use your own data with Azure OpenAI Service*

- A. Describe the capabilities of Azure OpenAI on your data
- B. Configure Azure OpenAI to use your own data
- C. Use Azure OpenAI API to generate responses based on your own data

#### XXXI. *Fundamentals of Responsible Generative AI*

- A. Describe an overall process for responsible generative AI solution development
- B. Identify and prioritize potential harms relevant to a generative AI solution
- C. Measure the presence of harms in a generative AI solution
- D. Mitigate harms in a generative AI solution
- E. Prepare to deploy and operate a generative AI solution responsibly