

warning "Mise à jour du backend (octobre 2026)" L'inférence GitHub Models a été retirée le 30 juillet 2026. Les références ci-dessous sont conservées uniquement comme contexte historique pendant la migration de ce lab. Utilisez le Lab 013 ? Backends LLM gratuits pour configurer Foundry Local ou Ollama avant de continuer.

Lab 009 : Démarrage rapide avec Azure OpenAI Service

Niveau : L100 Parcours : ? Microsoft Foundry Durée : ~30 min ? Coût : Azure Free ? Compte Azure gratuit, utilisation minimale

Ce que vous apprendrez

Ce qu'est Azure OpenAI Service et en quoi il diffère d'OpenAI et de GitHub Models Comment créer une ressource Azure OpenAI et déployer un modèle Comment appeler Azure OpenAI via le SDK Python et l'API REST L'authentification par clé API vs. Microsoft Entra ID (identité managée) Quand utiliser Azure OpenAI vs. GitHub Models vs. Ollama

Introduction

Azure OpenAI Service fournit l'accès aux modèles d'OpenAI (GPT-4o, GPT-4o-mini, o1 et embeddings) via l'infrastructure Azure. Cela signifie :

Conformité entreprise : Les données restent dans votre tenant Azure, régies par vos

politiques Réseau privé : Déploiement derrière un VNet, sans exposition à Internet public

Intégrations Azure : Fonctionne nativement avec Azure AI Search, Azure AI Foundry, Key Vault, et plus encore SLA : Garanties de disponibilité de niveau entreprise

info "Azure OpenAI vs. GitHub Models vs. OpenAI" | | GitHub Models | OpenAI (direct) |

Azure OpenAI | |-----|-----|-----| | Coût | Niveau gratuit

| Paiement par token | Paiement par token | | Authentification | GitHub PAT | Clé API | Clé

API ou identité managée | | Résidence des données | GitHub/Azure | OpenAI US | Votre région

Azure | | Fonctionnalités entreprise | ? | Limitées | ? Complètes | | Limites de débit |

Faibles (dev uniquement) | Standard | Configurables | | Idéal pour | Apprentissage, dev |

Applications grand public | Production entreprise |

Pour tous les labs L200+ de ce hub, vous pouvez utiliser GitHub Models (gratuit). Azure OpenAI est le parcours recommandé pour passer en production.

Configuration des prérequis

Créer un compte Azure

? Compte Azure gratuit ? 200 \$ de crédit, 12 mois de services gratuits

warning "Limitations du niveau gratuit" Azure OpenAI dans le niveau gratuit dispose d'un quota très limité. Pour ces exercices, quelques appels API suffisent.

Demander l'accès à Azure OpenAI

Azure OpenAI nécessite une demande : 1. Rendez-vous sur aka.ms/aoai/access 2. Remplissez le formulaire avec votre cas d'utilisation 3. Attendez l'approbation (généralement quelques heures à 1 jour)

tip "En attendant l'approbation" Complétez les sections conceptuelles de ce lab et utilisez GitHub Models pour les exercices pratiques ? le code est identique.

Étape 1 : Créer une ressource Azure OpenAI

1. Dans le Portail Azure, recherchez Azure OpenAI 2. Cliquez sur + Créer 3. Remplissez : -

Abonnement : votre abonnement - Groupe de ressources : créer nouveau ? rg-learning-hub -

Région : East US 2 (meilleure disponibilité des modèles) - Nom :

aoai-learning-hub-[votrenom] (doit être unique à l'échelle mondiale) - Niveau tarifaire :

Standard S0 4. Cliquez sur Vérifier + Créer ? Créer

Étape 2 : Déployer un modèle

1. Ouvrez votre ressource Azure OpenAI 2. Cliquez sur "Accéder à Azure OpenAI Studio" ?

oai.azure.com 3. Cliquez sur Déploiements ? + Créer un nouveau déploiement 4. Configurez :

- Modèle : gpt-4o-mini - Nom du déploiement : gpt-4o-mini (même nom que le modèle, plus facile à retenir) - Tokens par minute : 10K (suffisant pour les labs) 5. Cliquez sur Déployer

Déployez également un modèle d'embedding : Modèle : text-embedding-3-small Nom du déploiement : text-embedding-3-small

Étape 3 : Obtenir votre point de terminaison et vos clés

Depuis votre ressource Azure OpenAI : 1. Cliquez sur Clés et point de terminaison dans le menu de gauche 2. Copiez la CLÉ 1 et l'URL du Point de terminaison

Stockez-les comme variables d'environnement :

```
=== "Windows (PowerShell)" [diagram or code sample]
```

```
=== "macOS / Linux" [diagram or code sample]
```

danger "Ne commitez jamais de clés API" Utilisez des variables d'environnement, Azure Key Vault ou des fichiers .env (ajoutez .env à .gitignore). Ne codez jamais de clés API en dur dans le code source.

Étape 4 : Votre première complétion de chat

[diagram or code sample]

[diagram or code sample]

tip "Même API, client différent" Le code est identique à OpenAI ? il suffit d'utiliser AzureOpenAI au lieu de OpenAI, d'ajouter azure_endpoint et api_version. C'est tout.

Étape 5 : Embeddings avec Azure OpenAI

[diagram or code sample]

Étape 6 : Basculer entre les backends avec une seule variable

Ce pattern vous permet d'écrire le code une seule fois et de l'exécuter avec GitHub Models (gratuit) en développement et Azure OpenAI en production :

[diagram or code sample]

Utilisation : [diagram or code sample]

Étape 7 : Identité managée (bonne pratique de production)

En production, n'utilisez jamais de clés API. Utilisez plutôt l'identité managée :

[diagram or code sample]

[diagram or code sample]

Attribuez à votre application le rôle Cognitive Services OpenAI User dans Azure RBAC.

Azure OpenAI dans Azure AI Foundry

Azure AI Foundry fournit un portail unifié pour gérer Azure OpenAI et d'autres services d'IA :

1. Rendez-vous sur ai.azure.com 2. Créez un Hub (une fois par organisation) et un Projet (une fois par application) 3. Déployez des modèles via le Catalogue de modèles 4. Utilisez le Playground pour tester de manière interactive 5. Accédez aux outils d'Évaluation et de Surveillance

? Consultez le Lab 030 ? Foundry Agent Service + MCP pour un guide complet de Foundry.

? Vérification des connaissances

question "1. Quelle est la différence de code principale entre l'utilisation directe

d'OpenAI et Azure OpenAI ?" Utilisez AzureOpenAI au lieu de OpenAI, et fournissez azure_endpoint et api_version. Le paramètre model prend le nom de votre déploiement, pas le nom du modèle. Tout le reste (messages, temperature, etc.) est identique.

question "2. Pourquoi devriez-vous utiliser l'identité managée plutôt que les clés API en production ?" Les clés API sont des secrets statiques qui peuvent être divulgués, volés ou accidentellement committés dans le contrôle de source. L'identité managée utilise le système d'identité intégré d'Azure ? aucun secret n'est jamais stocké dans le code ou les

fichiers de configuration. L'accès est contrôlé via Azure RBAC, est automatiquement renouvelé et laisse des traces d'audit.

question "3. Quand choisiriez-vous GitHub Models plutôt qu'Azure OpenAI ?" GitHub Models est idéal pour le développement, l'apprentissage et le prototypage ? c'est gratuit avec juste un compte GitHub, sans abonnement Azure nécessaire, sans déploiement à configurer. Choisissez Azure OpenAI quand vous avez besoin de : exigences de résidence des données, réseau privé, SLA entreprise, intégration avec Azure AI Search ou Foundry, ou fiabilité de niveau production.

Résumé

Concept | Point clé | Client AzureOpenAI | Même API qu'OpenAI, juste des paramètres de constructeur différents | Nom du déploiement | Vous déployez un modèle avec un nom ? utilisez ce nom, pas le nom du modèle | Version de l'API | Paramètre obligatoire ? vérifiez la dernière version sur aka.ms/oai/docs | Identité managée | À utiliser en production ? pas de secrets statiques | Foundry | Le portail qui englobe Azure OpenAI avec des outils d'évaluation, de surveillance et d'agents |

Prochaines étapes

Construire un agent complet avec Foundry : ? Lab 030 ? Foundry Agent Service + MCP Ajouter le RAG à votre application Azure OpenAI : ? Lab 031 ? Recherche sémantique pgvector sur Azure Alternative gratuite pour les labs : ? Lab 013 ? GitHub Models

Modernisation de Phase 3

warning "Verified 2026-10-27" Utilisez Microsoft Foundry et Agent Service GA fondé sur Responses API. Les évaluations sont GA ; les hosted agents restent en preview et doivent être signalés. Ne dépendez pas des Workflows du portail retirés le 01/12/2026.

[diagram or code sample]

Foundry Agent Service GA announcement Foundry Agent Service Foundry evaluation Foundry Workflows retirement

Kit d'accompagnement

Le build du site génère des téléchargements PDF et ZIP stables. Le ZIP contient le guide du lab et tous les fichiers d'accompagnement déclarés.

Télécharger le PDF généré Télécharger le ZIP généré

Supporting file: CHEATSHEET.md

Fichiers d'accompagnement: Lab 009 : Démarrage rapide avec Azure OpenAI Service Commencez par le parcours local ou gratuit documenté. Consignez les hypothèses et les preuves de validation. Gardez les services hébergés facultatifs sauf exigence explicite du lab.

Supporting file: README.md

Lab 009 : Démarrage rapide avec Azure OpenAI Service Utilisez ces fichiers avec le guide du lab et le kit généré.