



Serverless E-Learning Application

1. In this lab we are going to make a Serverless E-learning application using AWS Lambda, API gateway, and Bedrock foundational models. We will also S3 bucket where we will upload all the necessary files.
2. Below is some important information.
 - **Amazon Bedrock Knowledge Base** supports building **generative AI applications** using **retrieval-augmented generation (RAG)** by automating data ingestion workflows.
 - In an e-learning solution, content like **PDFs** stored in an **S3 bucket** can be broken down into smaller **chunks** (e.g., 300 characters) for vector embeddings.
 - **Embedding models** such as **Cohere** or **Titan** can create embeddings from the PDF chunks, which are stored in a **vector database** like **Amazon OpenSearch** or **Pinecone**.
 - Amazon Bedrock Knowledge Base simplifies the process with **just a few clicks**, automating steps like data loading, chunking, embedding creation, and vector storage.
 - Currently, it supports **S3** as the data source, chunking strategies, Cohere and Titan embedding models, and vector databases like **Amazon OpenSearch**, **Aurora**, **Pinecone**, and **Redis**.
3. So, there is a prerequisite for this lab and that is you need to create an IAM user with administrator privileges and then provide this user the Console access and then you need to log in with this user to work with this lab.
4. Now the first thing we need to do is create an S3 bucket and upload all the relevant files to it.

Name	Type	Last modified	Size	Storage class
Amazon EBS FAQs Amazon Web Services.pdf	pdf	October 26, 2024, 15:37:07 (UTC+05:30)	273.7 KB	Standard
Amazon EC2 FAQs.pdf	pdf	October 26, 2024, 15:37:09 (UTC+05:30)	2.2 MB	Standard
Amazon Elastic Container Service FAQs.pdf	pdf	October 26, 2024, 15:37:10 (UTC+05:30)	275.7 KB	Standard
Amazon Simple Storage Service.pdf	pdf	October 26, 2024, 15:37:11 (UTC+05:30)	1.5 MB	Standard
AWS Certified Solutions Architect Associate Exam Guide.pdf	pdf	October 26, 2024, 15:37:12 (UTC+05:30)	172.3 KB	Standard

5. Then search for Amazon Bedrock and navigate to it, from the left pane expand builder tools choose knowledge bases, and click on Create knowledge base. **Also keep in mind that using a knowledge base will cost you \$0.50 cents per hour.**

▼ Builder tools

Prompt management [Preview](#)

Knowledge bases

Agents

Prompt flows [Preview](#)

6. In step 1, you need to give it a name then scroll down, then for IAM choose Create New Role and for the data source choose S3. Click on next.

[Amazon Bedrock](#) > [Knowledge bases](#) > Create knowledge base

Step 1 **Provide knowledge base details**

Step 2 Configure data source

Step 3 Select embeddings model and configure vector store

Step 4 Review and create

Provide knowledge base details

Knowledge base details

Knowledge base name
e-learning-application

Valid characters are a-z, A-Z, 0-9, _ (underscore) and - (hyphen). The name can have up to 50 characters.

Knowledge base description - optional
demo for creating an e-learning application

Valid characters are a-z, A-Z, 0-9, _ (underscore) and - (hyphen). The name can have up to 200 characters.

 

IAM permissions

Certain permissions are necessary to access other services or perform actions in order to create this resource. For more information, see [service role](#) for Amazon Bedrock

Runtime role

- Create and use a new service role
 Use an existing service role

Service role name

AmazonBedrockExecutionRoleForKnowledgeBase_t4kp

Choose data source

Select the data source that you want to configure in the next step. You can add up to 5 data sources in a knowledge base.

 Amazon S3 	 Web Crawler - Preview <input type="radio"/>
Object storage service that stores data as objects within buckets.	Web page crawler that extracts content from public web pages you are authorized to crawl.

7. After that you need to choose your bucket and click on next.



Amazon S3 Info

Provide details to connect Amazon Bedrock to your S3 data source.

▼ Data source: knowledge-base-quick-start-b0iyq-data-source

[Delete](#)

Data source name

Valid characters are a-z, A-Z, 0-9, _ (underscore) and - (hyphen). The name can have up to 100 characters.

Data source location

- This AWS account
- Other AWS account

S3 URI

To increase the accuracy and relevance of your responses, add a .metadata.json file containing metadata for your data source to your S3 bucket. [Info](#)

S3 URI

[View](#) [Browse S3](#)

- Add customer-managed KMS key for S3 data - *optional*

If you encrypted your S3 data, provide the KMS key here so that Bedrock can decrypt it.

Chunking and parsing configurations [Info](#)

Choose between default or advanced customization.

- Default

Uses default parsing and chunking strategy.

- Custom

Customize the parsing and chunking strategy, including using advanced parsing.

► Advanced settings - *optional*

8. In step 3, choose Titan embeddings G1 as your embedding model, and for vector base choose Quick create and click on next.
9. Then from the review page you need to click on create knowledge base.

[Amazon Bedrock](#) > [Knowledge bases](#) > Create knowledge base

- Step 1
 Provide knowledge base details
- Step 2
 Configure data source
- Step 3
 Select embeddings model and configure vector store
- Step 4
 Review and create

Select embeddings model and configure vector store

Choose an embeddings model to convert the data that you will provide in the next step, and provide details for a vector data store in which Bedrock can store, manage, and update your embeddings. The embeddings model and vector store cannot be changed after creation of knowledge base.

Embeddings model

Select an embeddings model to convert your data into an embedding. Pricing depends on the model. [Learn more](#)



Titan Text Embeddings v2

By Amazon



Titan Embeddings G1 - Text v1.2



By Amazon



Embed English v3

By Cohere



Embed Multilingual v3



By Cohere

Vector dimensions

Select the vector dimension size for your embeddings model to balance accuracy, cost, and latency. Higher dimensions improves overall accuracy and requires more vector storage. [Learn more](#)

1536



Vector database

Let Amazon create a vector store on your behalf or select a previously created store to allow Bedrock to store, update and manage embeddings. You will be billed directly from the vector store provider. [Learn more](#)

Select how you want to create your vector store.

Quick create a new vector store - *Recommended*

We will create an Amazon OpenSearch Serverless vector store on your behalf. This cost-efficient option is intended only for development and can't be migrated to production workload later. [Learn more](#)

Choose a vector store you have created

Select Amazon OpenSearch Serverless, Amazon Aurora, MongoDB Atlas, Pinecone or Redis Enterprise Cloud and provide field mappings.

Enable redundancy (active replicas) - *optional*

The default configuration has active replicas disabled, which is optimal for development workloads. Enable this option if you want to enable redundant active replicas, which may increase storage costs.

Add customer-managed KMS key for Amazon OpenSearch Serverless vector - optional

If you encrypted your OpenSearch data, provide the KMS key here so that Bedrock can decrypt it.

[Cancel](#)

[Previous](#)

[Next](#)

10. The creation of a knowledge base will take 4-5 minutes or more. Below you can see that our knowledge base has been created.

Knowledge base 'e-learning-application' is created successfully. Sync one or more data sources to index your content for searching. Syncing can take from a few minutes to a few hours.

[Go to data sources](#)

X

[Amazon Bedrock](#) > [Knowledge bases](#) > e-learning-application

e-learning-application

[Test](#)

[Delete](#)

Knowledge base overview

[Edit](#)

Knowledge base name
e-learning-application

Knowledge base ID
 WYXGKANEJC

Knowledge base description
demo for creating an e-learning application

Status
 Available

Service Role
[AmazonBedrockExecutionRoleForKnowledgeBase_ti4kp](#)

Created date
October 26, 2024, 15:53 (UTC+05:30)

Log Deliveries
Configure log deliveries and event logs in the [Edit](#) page.

11. Now scroll down to the data source section and select your data source then click on sync.

Data source (1)

Data sources contain information returned when querying a Knowledge base.

Sync Stop sync Add

Find data source

< 1 >

Data so...	Status	Data sour...	Account ID	Source Link
knowledge...	Available	S3	46364677...	s3://e-lea...

12. Once your data source has been synced then click on select model and choose a model of your choice.

Test knowledge base



Generate responses



Test your knowledge base by running a query to generate responses. To disable response generation and only see retrieved information stored from your vector store, turn off Generate

Select model

1. Category

Model providers

-  Amazon
-  Anthropic
-  Cohere
-  Mistral AI

2. Model

Models with access (5)

- Claude Instant v1.2**
Text model | Context size = up to 100k
- Claude v2.1**
Text model | Context size = up to 200k
- Claude v2**
Text model | Context size = up to 100k
- Claude 3 Sonnet v1**
Text & vision model | Context size = up to 200k
- Claude 3 Haiku v1**
Text & vision model | Context size = up to 200k

Models without access (1) [Request access](#)

3. Throughput

On-demand

No provisioned throughput options available. [Purchase Provisioned Throughput](#) to continue.

[Cancel](#)

[Apply](#)

13. Here you can see that we get the answer to our question and we can also see the source of information.

Generate responses  

Claude Instant v1.2 | On-demand

[Change](#)



which ebs volume in aws I can use
for high throughput



For high throughput workloads, you can use Throughput Optimized HDD (st1) or Cold HDD (sc1) volume types. ST1 volumes are ideal for frequently accessed, throughput intensive workloads with large datasets and large I/O sizes. SC1 volumes provide lower cost storage and are ideal for less frequently accessed workloads with large, cold datasets.^[1]

[Show source details >](#)

Enter your message here

 Run

14. If you click on show source details then you can see source chunks too.

Test Knowledge base

The screenshot shows the Amazon Bedrock interface for testing a knowledge base. On the left, there's a sidebar with a toggle for "Generate responses" and a section for "Claude Instant v1.2 | On-demand". Below that is a "Configure your retrieval and responses" panel with a note to customize search strategy. The main area contains a query card: "which ebs volume in aws I can use for high throughput". To the right, under "Source details (1)", it shows a "Query configurations (1)" section and a "Source chunk 1" section. The "Source chunk 1" section provides detailed information about EBS volumes, mentioning ST1 and SC1 types, their performance characteristics, and cost considerations.

15. Now we need to create an AWS Lambda function to move forward with our lab. Here give your function a name choose runtime environment as Python and click on create.

The screenshot shows the "Create function" wizard in the AWS Lambda console. It starts with a choice between "Author from scratch", "Use a blueprint", and "Container image". The "Author from scratch" option is selected. The "Basic information" step is shown, where the function name is set to "e-learning-app". The "Runtime" is chosen as "Python 3.11", and the "Architecture" is "x86_64". The "Runtime" dropdown has a note about supported languages: Node.js, Python, and Ruby.

16. Once your function is created you need to increase the time out time from the general configuration.

The screenshot shows the AWS Lambda Configuration page. The 'General configuration' tab is selected. On the left, there's a sidebar with 'General configuration', 'Triggers', 'Permissions', 'Destinations', and 'Environment variables'. In the main area, under 'General configuration', there are fields for 'Description' (empty), 'Memory' (128 MB), 'Ephemeral storage' (512 MB), 'SnapStart' (None), and 'Timeout' (1 min 3 sec). A red box highlights the 'Edit' button at the top right and the 'Timeout' field.

17. Then we need to provide the required permission to our lambda function. So, to your role attach the bedrock full access permission.

The screenshot shows the AWS Lambda Configuration page with the 'Configuration' tab selected. On the left, there's a sidebar with 'General configuration', 'Triggers', 'Permissions' (which is highlighted with a red box), and 'Destinations'. In the main area, under 'Permissions', there's a 'Role name' field containing 'e-learning-app-role-bdmsieuq' (highlighted with a red box). Below it, there's a table showing 'Permissions policies' with two entries: 'AmazonBedrockFullAccess' (AWS managed) and 'AWSLambdaBasicExecutionRole-7366f68...' (Customer managed).

Policy name	Type	Attached entities
AmazonBedrockFullAccess	AWS managed	2
AWSLambdaBasicExecutionRole-7366f68...	Customer managed	1

18. Then in your lambda function come to the code section use the code given below and deploy it.

```

import json
#1. import boto3
import boto3
#2 create client connection with bedrock
client_bedrock_knowledgebase = boto3.client('bedrock-agent-runtime')
def lambda_handler(event, context):
    #3 Store the user prompt
    print(event['prompt'])
    user_prompt=event['prompt']
    # 4. Use retrieve and generate API

```

```

client_knowledgebase
client_bedrock_knowledgebase.retrieve_and_generate(
    input={
        'text': user_prompt
    },
    retrieveAndGenerateConfiguration={
        'type': 'KNOWLEDGE_BASE',
        'knowledgeBaseConfiguration': {
            'knowledgeBaseId': 'WYXGKANEJC',
            'modelArn': 'arn:aws:bedrock:us-east-1::foundation-
model/anthropic.claude-instant-v1'
        }
    }
)

# print(client_knowledgebase)
#print(client_knowledgebase['output']['text'])

#print(client_knowledgebase['citations'][0]['generatedResponsePart']['textRespo
nsePart'])
    response_kbase_final=client_knowledgebase['output']['text']
    return {
        'statusCode': 200,
        'body': response_kbase_final
    }
}

```

The screenshot shows the AWS Lambda function editor interface. The top navigation bar includes File, Edit, Find, View, Go, Tools, Window, Test (which is selected), and Deploy. Below the navigation is a search bar labeled 'Go to Anything (Ctrl-P)'. The main workspace is titled 'lambda_function' and contains environment variables. On the left, there's a sidebar for 'Environment' with a tree view showing 'e-learning-app /' and 'lambda_function.py'. The code editor displays the Python script 'lambda_function.py' with line numbers from 1 to 31. The code itself is identical to the one provided above.

```

1 import json
2 #1. import boto3
3 import boto3
4 #2 create client connection with bedrock
5 client_bedrock_knowledgebase = boto3.client('bedrock-agent-runtime')
6 def lambda_handler(event, context):
7     #3 Store the user prompt
8     print(event['prompt'])
9     user_prompt=event['prompt']
10    # 4. Use retrieve and generate API
11    client_knowledgebase = client_bedrock_knowledgebase.retrieve_and_generate(
12        input={
13            'text': user_prompt
14        },
15        retrieveAndGenerateConfiguration={
16            'type': 'KNOWLEDGE_BASE',
17            'knowledgeBaseConfiguration': {
18                'knowledgeBaseId': '041RCI046A',
19                'modelArn': 'arn:aws:bedrock:us-west-2::foundation-model/anthropic.claude-instant-v1'
20            }
21        }
22    )
23    # print(client_knowledgebase)
24    #print(client_knowledgebase['output']['text'])
25    #print(client_knowledgebase['citations'][0]['generatedResponsePart']['textResponsePart'])
26    response_kbase_final=client_knowledgebase['output']['text']
27    return {
28        'statusCode': 200,
29        'body': response_kbase_final
30    }
31

```

19. In the above code in line number 18 you need to provide the knowledge base ID and you can get this ID from your knowledge base, so go to your knowledge base, and there in the overview you can get this ID.

e-learning-application

Test

Delete

Knowledge base overview

Edit

Knowledge base name

e-learning-application

Knowledge base ID

WYXGKANEJC

Knowledge base description

demo for creating an e-learning application

Status

Available

Service Role

AmazonBedrockExecutionRoleForKnowledgeBase_t4kp

Created date

October 26, 2024, 15:53 (UTC+05:30)

Log Deliveries

Configure log deliveries and event logs in the

Edit page.

20. Now after deploying your code, you need to create a test event. In the event JSON area, you need to write your prompt like this to get the proper output. Click on save and then on test.

```
{  
  "prompt": "Which EBS volume for high IOPS"  
}
```

Event JSON

```
1 {  
2   "prompt": "Which EBS volume for high IOPS"  
3 }
```

21. Below you can see that we got the response as expected.

Code | **Test** | Monitor | Configuration | Aliases | Versions

Executing function: succeeded ([logs](#))

▼ Details

The area below shows the last 4 KB of the execution log.

```
{ "statusCode": 200, "body": "Provisioned IOPS SSD (io1 and io2) volumes are designed for transactional, IOPS-intensive database workloads, boot volumes, and workloads that require high IOPS." }
```

Summary

Code SHA-256 A+nbaS2fP335Kcx3400Zv7dJGmLXWoG1rL7juzQ8iPM=	Execution time 26 seconds ago
Request ID 393c3927-9111-49f2-b32d-0c634e05e9a7	Function version \$LATEST
Init duration 407.55 ms	Duration 2357.73 ms

22. So, we have completed our work on the lambda function now we are going to integrate our lambda function with API Gateway.
23. Search and navigate to the API gateway and click on build REST API. Then give it a name and click on build.

REST API

Develop a REST API where you gain complete control over the request and response along with API management capabilities.

Works with the following:
Lambda, HTTP, AWS Services

[Import](#) [Build](#)

Create REST API

API details

New API
Create a new REST API.

Clone existing API
Create a copy of an API in this AWS account.

Import API
Import an API from an OpenAPI definition.

Example API
Learn about API Gateway with an example API.

API name

Description - *optional*

24. So, on our default resource we are going to create a GET method. Click on create method.

The screenshot shows the AWS API Gateway 'Resources' page. At the top, there's a breadcrumb navigation: API Gateway > APIs > Resources - e-LearningApp (zea7tos4oa). Below the breadcrumb is a 'Resources' section with a 'Create resource' button. To the right is a 'Resource details' panel showing the path '/' and Resource ID 'ydx5pi8y4m'. At the top right of the details panel are 'API actions' and 'Deploy API' buttons. Below the details panel is a 'Methods (0)' section with a 'Create method' button highlighted by a red box. The 'Methods' table has columns for Method type, Integration type, Authorization, and API key. A message below the table says 'No methods defined.'

25. First choose method type as GET and then choose your lambda function.

The screenshot shows the 'Method type' configuration screen. It starts with a 'Method type' dropdown set to 'GET'. Below it is an 'Integration type' section with five options: 'Lambda function' (selected), 'HTTP', 'Mock', 'AWS service', and 'VPC link'. Each option has a description and a corresponding icon. Under 'Lambda function', there's also a 'Lambda proxy integration' section with a note about sending requests to a Lambda function as a structured event. At the bottom, there's a 'Lambda function' input field where 'us-east-1' is selected and a search bar containing 'arn:aws:lambda:us-east-1:463646775279:function:e-le'.

26. In your GET method choose method request and click on edit.

27. Here you need to choose the **same validator in the request validator** expand **URL query string parameters** and click on **add query string** then in the name **write prompt** and click on save.

Name	Required	Caching	
prompt	<input type="checkbox"/>	<input type="checkbox"/>	<button>Remove</button>

28. Go to integration request in GET method and click on edit.

29. Scroll down to the bottom expand mapping templates and click on add mapping templates then in the content type and template body write the same thing as you see below.

```
{  
  "prompt": "$input.params('prompt')"  
}
```

The screenshot shows the 'Mapping templates' section of the AWS API Gateway configuration. It includes fields for Content type (application/json), Generate template (a dropdown menu), and Template body (a code editor containing the JSON template provided above). The code editor has line numbers 1, 2, and 3, with line 3 being the selected line.

30. After that choose your resource and click on deploy API.

The screenshot shows the 'Resources' page in the AWS API Gateway. It displays a single resource path '/'. The 'Methods' section shows one GET method configured with Lambda integration and no authorization. The 'Deploy API' button in the top right corner is highlighted with a red box.

31. Choose new stage and give it a name then click on deploy.

Deploy API



Create or select a stage where your API will be deployed. You can use the deployment history to revert or change the active deployment for a stage. [Learn more](#)

Stage

New stage



Stage name

dev

i A new stage will be created with the default settings. Edit your stage settings on the [Stage page](#).

Deployment description

Cancel

Deploy

32. Now come to GET method and choose Test then in the query strings write your prompt in the same way and click on test.

Create resource

Method request Integration request Integration response Method response **Test**

Test method
Make a test call to your method. When you make a test call, API Gateway skips authorization and directly invokes your method.

Query strings
prompt='Which is the best Compute Resource'

Headers
Enter a header name and value separated by a colon (:). Use a new line for each header.
header1:value1
header2:value2

33. Below you can see that you get the response as expected.

/ - GET method test results		
Request	Latency ms	Status
/?prompt='Which is the best Compute Resource'	5115	200
Response body		
<pre>{"statusCode": 200, "body": "There are different types of compute optimized EC2 instances that are best suited for different types of compute workloads. Compute Optimized instances have proportionally more CPU resources than memory and are well suited for compute-intensive and high performance computing applications. Within the Compute Optimized instance category, the choice of which instance type is best depends on the specific application's resource utilization characteristics."}</pre>		

34. Then in your API gateway choose stages and then come to the GET stage and from here copy the invoke URL.

The screenshot shows the AWS API Gateway interface. In the top navigation bar, the path is: API Gateway > APIs > e-LearningApp (zea7tos4oa) > Stages. On the right side of the screen, there is a button labeled "Create stage". Below the navigation, the word "Stages" is displayed. On the left, a tree view shows a "dev" stage expanded, with a single child node labeled "/". Underneath this node, the "GET" method is selected, indicated by a blue underline. To the right of the tree view, there is a section titled "Method overrides". It contains a note: "By default, methods inherit stage-level settings. To customize settings for a method, configure method overrides." Below this note, a message states: "This method inherits its settings from the 'dev' stage." At the bottom of this section, the "Invoke URL" is listed as "https://zea7tos4oa.execute-api.us-east-1.amazonaws.com/dev/".

35. After that you need to open Post Man and create a new workspace here then choose the GET method and paste the invoke URL here.
 36. In the key you need to write the prompt and, in the value, you need to ask it a question then you will see that it was able to give the response.

The screenshot shows a Postman request to `https://zea7tos4oa.execute-api.us-east-1.amazonaws.com/dev/?prompt=Which is the best AWS Compute Service`. The response status is `200 OK` with a response time of `2.77 s` and a size of `600 B`. The response body is:

```
1 {
2     "statusCode": 200,
3     "body": "AWS Lambda is the best AWS compute service for event driven workloads. Lambda allows you to run code without provisioning or managing servers. It is serverless, so it is a good fit for workloads that are unpredictable and irregular."
4 }
```

37. Once you are done with the lab you need to delete your knowledge base from AWS Bedrock. So, go to bedrock then to knowledge choose it and click on delete after that delete all the other resources.

The screenshot shows the AWS Bedrock Knowledge Bases page with one item listed:

Name	Status	Description	Source	Created	Last sync
e-learning...	Available	demo for ...	7	October 2...	-