

Advanced Data Engineering in Cloud

ASSIGNMENT-1

Hem Prakash Dev

Roll-G23AI1054

1. Design the high-level architecture of the data engineering platform, including the following components:

- Data Ingestion
- Data Storage
- Data Processing
- Data Aggregation
- Data Visualization

High-Level Architecture Design

1. Data Ingestion:

- **Tools:** AWS Kinesis, AWS S3, AWS Glue, Kafka
- **Functionality:** Captures data from various sources like databases, APIs, and streaming data sources.
For small to medium-sized data, S3 and Glue would suffice. For larger data, Kinesis or Kafka for real-time ingestion would be necessary.

2. Data Storage

- **Tools:** AWS S3, Amazon Redshift, Amazon RDS, Amazon DynamoDB
- **Functionality:** Stores raw and processed data. S3 can be used for raw data storage due to its scalability and cost-effectiveness. Redshift for structured, query-able data, RDS for transactional data, and DynamoDB for NoSQL storage needs.

3. Data Processing

- **Tools:** AWS Glue, AWS Lambda, Apache Spark (on EMR), AWS Fargate, AWS Batch
- **Functionality:** Processes data to transform, clean, and aggregate it. For small data sizes, Lambda and Glue are efficient. For larger data, Spark on EMR or AWS Batch for batch processing can be used.

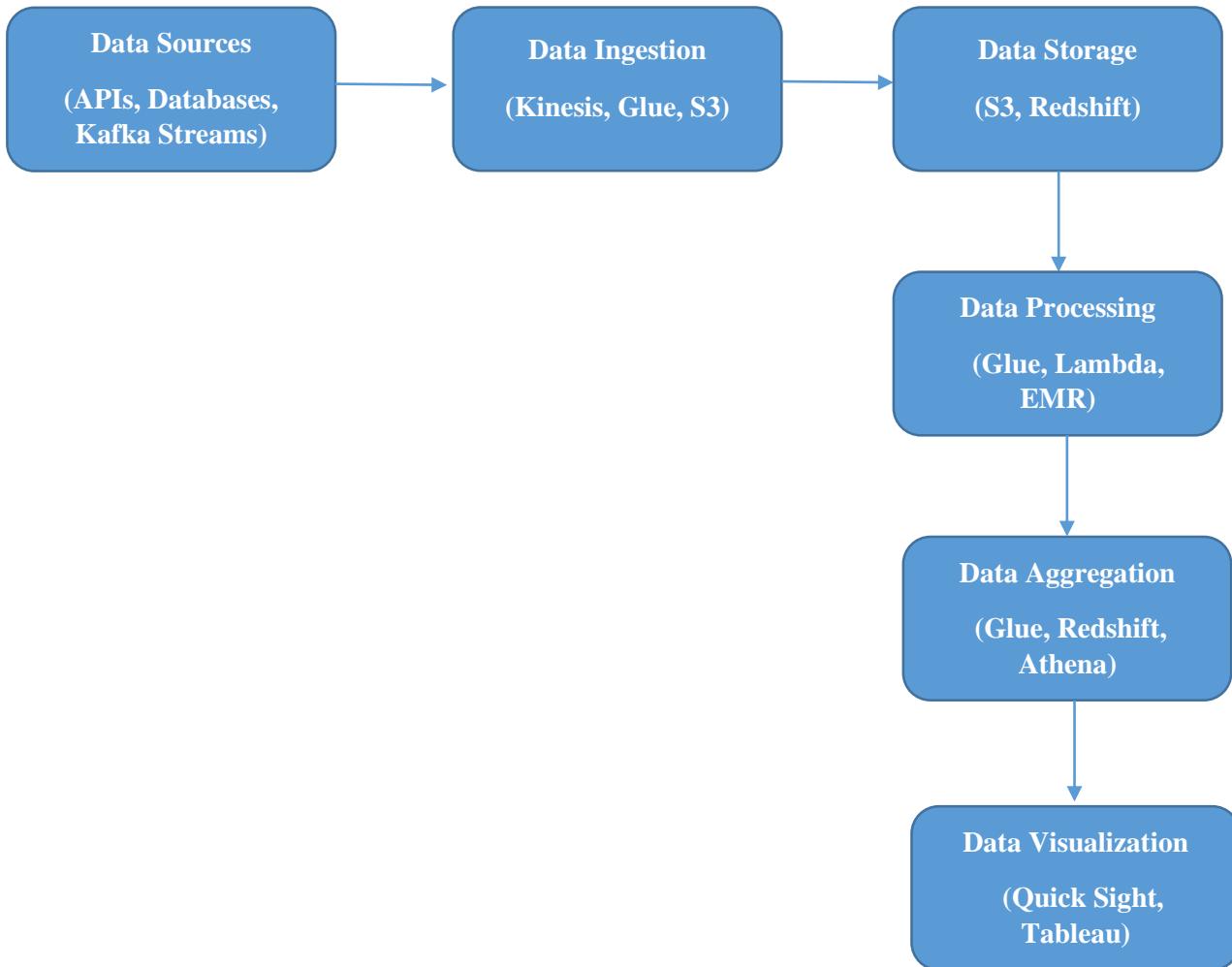
4. Data Aggregation

- **Tools:** AWS Glue, AWS Redshift, AWS Athena
- **Functionality:** Aggregates processed data for analysis. Glue and Redshift can be used for ETL and aggregation tasks. Athena provides a server less option to query data directly from S3.

5. Data Visualization

- **Tools:** Amazon QuickSight, Tableau, Power BI
- **Functionality:** Visualizes data for business intelligence. QuickSight is a native AWS tool that integrates well with other AWS services. Tableau and Power BI can be used for more advanced visualization needs.

Architecture Diagram



Q2. Create a GitHub repository to store and version control your code and configuration files.

The screenshot shows the GitHub 'New' repository creation interface. The top bar includes the GitHub logo, user icon, and navigation links. The main area is titled 'Home' and features a 'Updates to your homepage feed' section. Below it, there's a 'Start writing code' section with a 'Create new repository' button. To the right, there's an 'Explore repositories' sidebar listing several popular projects like 'webmin / webmin', 'getsentry / sentry-wizard', and 'electron / electron'. The central form for creating a new repository is filled with the following details:

- Repository name ***: Hem-Prakash-Dev-Bharadwaj
- Visibility**: Private (selected)
- Description**: A repository contains all of your project's files, revision history, and collaborator discussion.
- README**: Introduce yourself with a profile README
- Code Snippet**: A sample README.md file is shown with the following content:

```
1 - 🌎 Hi, I'm @Hem-Prakash-Dev-Bha
2 - 💬 I'm interested in ...
3 - 📚 I'm currently learning ...
4 - 🤝 I'm looking to collaborate o
5 - 📧 How to reach me ...
```

The screenshot shows a GitHub repository page for 'data-engineering' owned by 'Hem-Prakash-Dev-Bharadwaj'. The repository is public and has 13 commits. The README file has been updated. There are no releases or packages published.

Code | **Issues** | **Pull requests** | **Actions** | **Projects** | **Wiki** | **Security** | **Insights** | **Settings**

data-engineering (Public)

main | 1 Branch | 0 Tags

Go to file | Add file | Code

Hem-Prakash-Dev-Bharadwaj Update README.md | 142d0f9 · 1 minute ago | 13 Commits

Architecture Design and Setup | Rename Assignment 1: Architecture Design and Setup to A... | 5 days ago

Create GitHub Repository | Update Create GitHub Repository | 5 days ago

README.md | Update README.md | 1 minute ago

README

Data-Engineering

About

No description, website, or topics provided.

Readme | Activity | 0 stars | 1 watching | 0 forks

Releases

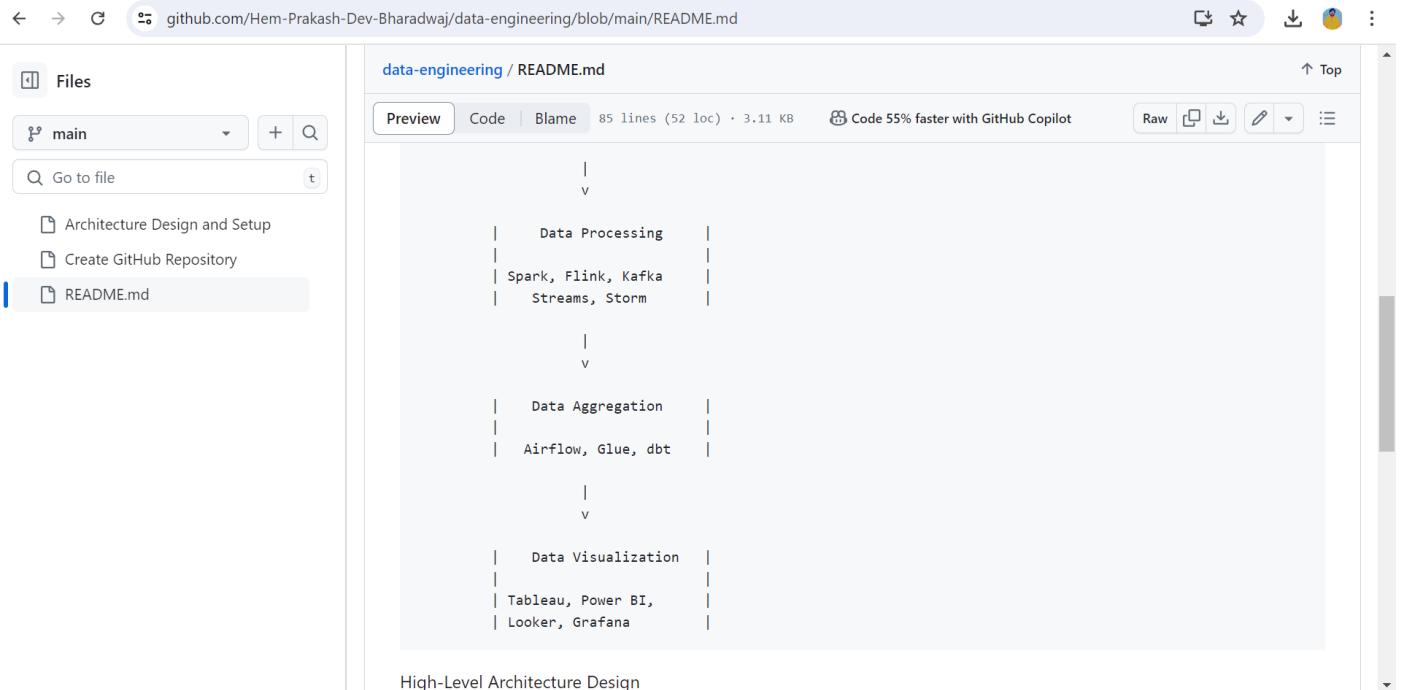
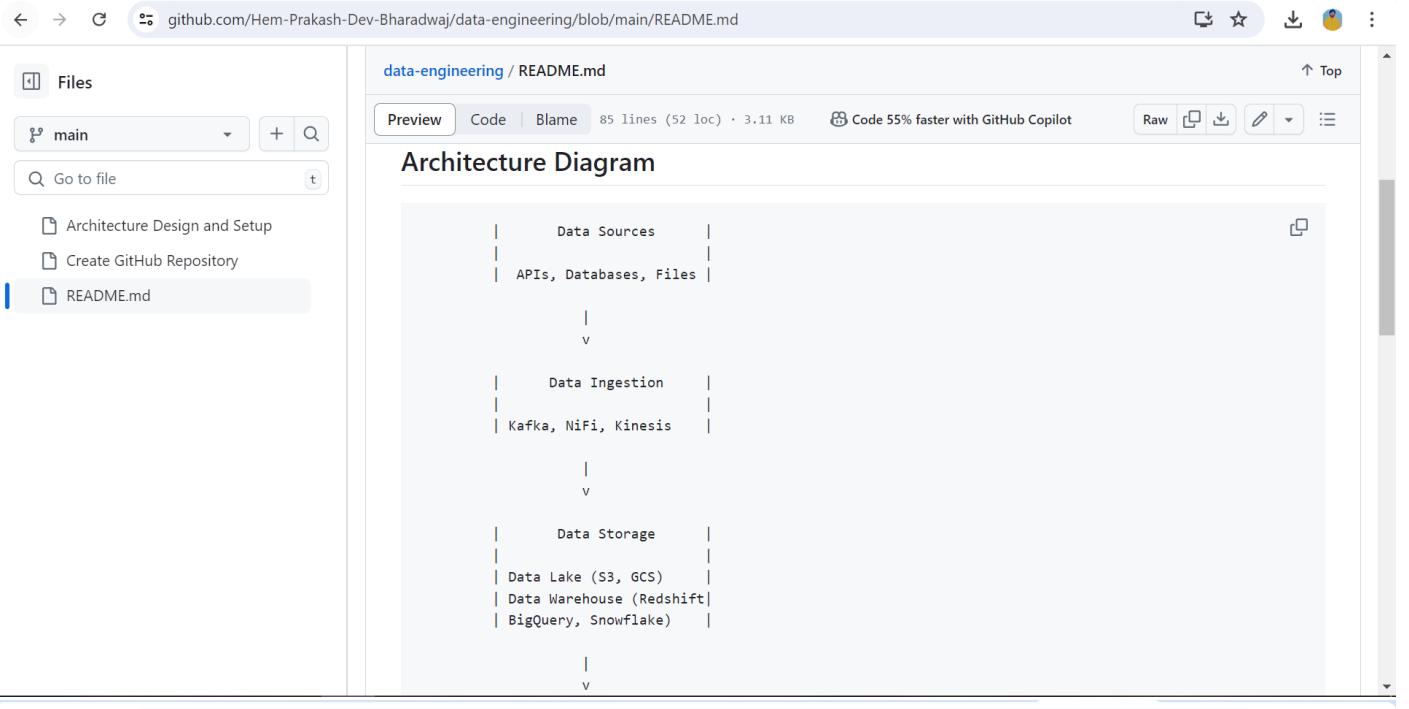
No releases published | Create a new release

Packages

3. Provide a brief README.md file in the repository with an overview of the project and the architecture diagram.

<https://github.com/Hem-Prakash-Dev-Bharadwai/data-engineering/blob/main/README.md>

The screenshot shows a GitHub repository page for 'data-engineering'. The URL is github.com/Hem-Prakash-Dev-Bharadwaj/data-engineering/blob/main/README.md. The repository owner is Hem-Prakash-Dev-Bharadwaj. The main navigation bar includes Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The 'Code' tab is selected. On the left sidebar, there's a 'Files' section with a dropdown menu showing 'main' (selected), 'Architecture Design and Setup', 'Create GitHub Repository', and 'README.md'. The main content area displays the README.md file, which has been updated by Hem-Prakash-Dev-Bharadwaj. The file contains sections for 'Data-Engineering' and 'Overview', and describes a platform for handling various data sizes using AWS services. A note at the bottom indicates that the code is 55% faster with GitHub Copilot.



High-Level Architecture Design

github.com/Hem-Prakash-Dev-Bharadwaj/data-engineering/blob/main/README.md

data-engineering / README.md

Preview Code Blame 85 lines (52 loc) · 3.11 KB Code 55% faster with GitHub Copilot Raw

High-Level Architecture Design

1. Data Ingestion:

- Tools: AWS Kinesis, AWS S3, AWS Glue, Kafka

Functionality: Captures data from various sources like databases, APIs, and streaming data sources. For small to medium-sized data, S3 and Glue would suffice. For larger data, Kinesis or Kafka for real-time ingestion would be necessary.

2. Data Storage

*Tools: AWS S3, Amazon Redshift, Amazon RDS, Amazon DynamoDB

- Functionality: Stores raw and processed data. S3 can be used for raw data storage due to its scalability and cost-effectiveness. Redshift for structured, query-able data, RDS for transactional data, and DynamoDB for NoSQL storage needs.

3. Data Processing

- Tools: AWS Glue, AWS Lambda, Apache Spark (on EMR), AWS Fargate, AWS Batch

Functionality: Processes data to transform, clean, and aggregate it. For small data sizes, Lambda and Glue are efficient. For larger data, Spark on EMR or AWS Batch for batch processing can be used.

github.com/Hem-Prakash-Dev-Bharadwaj/data-engineering/blob/main/README.md

data-engineering / README.md

Preview Code Blame 85 lines (52 loc) · 3.11 KB Code 55% faster with GitHub Copilot Raw

3. Data Processing

- Tools: AWS Glue, AWS Lambda, Apache Spark (on EMR), AWS Fargate, AWS Batch

Functionality: Processes data to transform, clean, and aggregate it. For small data sizes, Lambda and Glue are efficient. For larger data, Spark on EMR or AWS Batch for batch processing can be used.

4. Data Aggregation

- Tools: AWS Glue, AWS Redshift, AWS Athena

Functionality: Aggregates processed data for analysis. Glue and Redshift can be used for ETL and aggregation tasks. Athena provides a server less option to query data directly from S3.

5. Data Visualization

- Tools: Amazon QuickSight, Tableau, Power BI

Functionality: Visualizes data for business intelligence. QuickSight is a native AWS tool that integrates well with other AWS services. Tableau and Power BI can be used for more advanced visualization needs.

4. Set up the necessary AWS services and configure the basic infrastructure for the platform.

Amazon S3

1. Create an S3 Bucket for Data Storage

The screenshot shows the AWS S3 console interface. On the left, a sidebar titled 'Amazon S3' has a 'Buckets' section with various options like Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, and Block Public Access settings. Below this is a 'Storage Lens' section with options for Dashboards, Storage Lens groups, and AWS Organizations settings. The main content area is titled 'Amazon S3 > Buckets'. It features an 'Account snapshot - updated every 24 hours' card with a link to 'All AWS Regions'. Below it is a 'General purpose buckets' tab selected, showing a table with one row for 'hem-dev-bucket'. The table columns are Name, AWS Region, IAM Access Analyzer, and Creation date. The bucket details show it was created in 'Asia Pacific (Sydney) ap-southeast-2' on 'July 15, 2024, 17:32:51 (UTC+05:30)'. At the bottom, there are buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'.

Created 1st bucket

This screenshot shows the same AWS S3 console after a bucket has been created. A green notification bar at the top says 'Successfully created bucket "hem-dev-bucket"'. Below it, the 'General purpose buckets' table now shows two rows: 'hem-dev-bucket' and a new row for 'View analyzer for ap-southeast-2'. The 'View analyzer for ap-southeast-2' row includes a link to 'View analyzer for ap-southeast-2'. The rest of the interface is identical to the previous screenshot, including the sidebar and the 'Create bucket' button.

Once we created bucket, we can upload file(S3-Object) inside the bucket

The screenshot shows the AWS S3 console interface. At the top, the URL is ap-southeast-2.console.aws.amazon.com/s3/buckets/hem-dev-bucket?region=ap-southeast-2&bucketType=general&tab=objects. The top navigation bar includes the AWS logo, Services, a search bar, and account information for 'Hem_Prakash_Dev'. Below the navigation, the breadcrumb path shows 'Amazon S3 > Buckets > hem-dev-bucket'. The main title is 'hem-dev-bucket Info'. A horizontal menu bar below the title includes 'Objects' (which is selected), 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access Points'. The main content area is titled 'Objects (0) Info'. It features a toolbar with buttons for 'C' (Create), 'Copy S3 URI', 'Copy URL', 'Download', 'Open', 'Delete', 'Actions', 'Create folder', and 'Upload'. A message states: 'Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)'. Below this is a search bar labeled 'Find objects by prefix'. A table header row includes columns for Name, Type, Last modified, Size, and Storage class. A message 'No objects' indicates 'You don't have any objects in this bucket.' A large orange 'Upload' button is centered at the bottom of the object list. The footer contains links for CloudShell, Feedback, and copyright information: '© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences'.

Uploaded sample object in S3 Bucket

The screenshot shows the AWS S3 upload confirmation page. The URL is ap-southeast-2.console.aws.amazon.com/s3/upload/hem-dev-bucket?region=ap-southeast-2&bucketType=general. The top navigation bar and account information are identical to the previous screenshot. The main message is 'Upload succeeded' with a green checkmark icon and the text 'View details below.'. Below this, a summary table shows the destination 's3://hem-dev-bucket' with 'Succeeded' status (1 file, 113.5 KB (100.00%)) and 'Failed' status (0 files, 0 B (0%)). A horizontal menu bar below the summary includes 'Files and folders' (selected) and 'Configuration'. The 'Files and folders' section is titled 'Files and folders (1 Total, 113.5 KB)'. It features a search bar 'Find by name' and a table with columns: Name, Folder, Type, Size, Status, and Error. The single item listed is 'TRUST PAN...' with a download icon, '-' under Folder, 'image/jpeg' under Type, '113.5 KB' under Size, 'Succeeded' under Status, and '-' under Error. The footer contains links for CloudShell, Feedback, and copyright information: '© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences'.

2. Set Up Amazon RDS for Structured Data Storage

The screenshot shows the AWS RDS Databases page. On the left, there's a sidebar with options like Dashboard, Databases (which is selected), Query Editor, etc. The main area shows a success message: "Consider creating a Blue/Green Deployment to minimize downtime during upgrades". It says you may want to consider using Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides a staging environment for changes to production databases. There are links to the RDS User Guide and Aurora User Guide. Below this, there's a table titled "Databases (0)" with a "Create database" button.

Database Creation

The screenshot shows the AWS RDS Databases page after a database has been created. A green banner at the top says "Successfully created database hem-database-1". It says you can use settings from hem-database-1 to simplify configuration of suggested database add-ons while we finish creating your DB for you. Below this, there's a notifications bar with 0 notifications. The main area shows a success message: "Consider creating a Blue/Green Deployment to minimize downtime during upgrades". It says you may want to consider using Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides a staging environment for changes to production databases. There are links to the RDS User Guide and Aurora User Guide. Below this, there's a table titled "Databases (3)" showing one database entry: "hem-database-1" (Available, Regional cluster, Aurora MySQL, ap-southeast-2, 2 instances).

3. Set Up Amazon Redshift for Data Warehousing

Successfully created cluster in amazon redshift

ap-southeast-2.console.aws.amazon.com/redshiftv2/home?region=ap-southeast-2#/clusters

Services Search [Alt+S] Sydney Hem_Prakash_Dev

Query data using Redshift query editor

Work with your client tools

Choose your JDBC or ODBC driver

Cluster redshift-cluster-1

Driver JDBC 4.2 without AWS SDK (.jar)

Copy JDBC URL Copy ODBC URL Download driver

Clusters (1) Info

Filter clusters by property or value

Cluster	Status	Cluster namespace	Availability Zone	Multi-AZ	Storage capacity us...	CPU utilize...
redshift-cluster-1 ra3.4xlarge 2 nodes 256 ...	Available	2214cd...-4a48-4e2b-...	ap-southeast-2c	No		

Query editor v2 provides new features such as multistatement query execution, query parameterization, query versioning, visualizations, and query sharing. [Learn more about query editor v2](#)

Amazon Redshift > Clusters > redshift-cluster-1

redshift-cluster-1

Actions Edit Add partner integration Query data

General information Info

Cluster identifier redshift-cluster-1	Status Available	Node type ra3.4xlarge	Endpoint redshift-cluster-1.ckcd532mevyt.ap-southeast-2.redshift.amazonaws.com:5439/dev
Custom domain name -	Date created July 15, 2024, 21:09 (UTC+05:30)	Number of nodes 2	JDBC URL jdbc:redshift://redshift-cluster-1.ckcd532mevyt.ap-southeast-2.redshift.amazonaws.com:5439/dev
Cluster namespace ARN arn:aws:redshift:ap-southeast-2:905418436402:namespace:2214cd...-4a48-4e2b-b809-3055b5fa53ce	Storage used -	Patch version Patch 181	ODBC URL Driver={Amazon Redshift (x64)};Server=redshift-cluster-1.ckcd532mevyt.ap-southeast-2.redshift.amazonaws.com;
Cluster configuration Production	Multi-AZ No		

ap-southeast-2.console.aws.amazon.com/redshiftv2/home?region=ap-southeast-2#/query-editor:

Amazon Redshift > Query editor

Editor Query history Saved queries Scheduled queries

Connect to a database to run queries and view results.

Resources Info Connect to database

Select database Info To view schemas, select a database.

Select schema Info To view tables, select a schema.

Filter tables

No resources No resources to display

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ap-southeast-2.console.aws.amazon.com/redshiftv2/home?region=ap-southeast-2#/query-editor:

Amazon Redshift > Query editor

Editor | Query history | Saved queries | Scheduled queries

Status: Connected | database: dev | user: awsuser | Change connection

Resources Info | Select database Info | Select schema Info | Filter tables | category | date | event

Query 1

```
1 create table shoes( shoetype varchar (10), color varchar(10));  
2 insert into shoes values  
3 ('loafers', 'brown'),  
4 ('sandals', 'black');
```

CloudShell Feedback

ap-southeast-2.console.aws.amazon.com/redshiftv2/home?region=ap-southeast-2#/query-editor:

Amazon Redshift > Query editor

Editor | Query history | Saved queries | Scheduled queries

Status: Connected | database: dev | user: awsuser | Change connection

Resources Info | Select database Info | Select schema Info | Filter tables | category | date | event | listing | sales | shoes

Query 1

```
1 create table shoes( shoetype varchar (10), color varchar(10));  
2 insert into shoes values  
3 ('loafers', 'brown'),  
4 ('sandals', 'black');
```

Run | Save | Schedule | Clear | Send feedback

Query results | Table details

CloudShell Feedback

Configure AWS Glue

ap-southeast-2.console.aws.amazon.com/glue/home?region=ap-southeast-2#/v2/getting-started

AWS Glue > Welcome to AWS Glue

Welcome to AWS Glue

Get started by setting up your account and users, cataloging your data, and building ETL jobs to prepare data for analytics.

Prepare your account for AWS Glue

Admins: Grant access to AWS Glue and set a default IAM role.

Set up roles and users

Catalog and search for datasets

View your databases & tables and catalog data using Crawlers.

Go to the Data Catalog

Move and transform data

Transform data using a visual, notebook, or code interface.

Author and edit ETL jobs

Resources and tutorials

Getting started with AWS Glue: Documentation | AWS Training

Video on working with AWS Glue Studio: Part 1 | Part 2 | Part 3

Using connectors and connections

AWS Glue Documentation home

Examples: AWS Glue blog posts | AWS Glue on GitHub

Data integration and management

Monitor & debug ETL jobs and track usage

Go to job run monitoring

Connect to your data stores

Go to connections

CloudShell Feedback

ap-southeast-2.console.aws.amazon.com/glue/home?region=ap-southeast-2#/v2/data-catalog/databases

AWS Services Search [Alt+S] Sydney Hem_Prakash_Dev

AWS Glue

- Getting started
- ETL jobs
- Visual ETL
- Notebooks
- Job run monitoring
- Data Catalog tables
- Data connections
- Workflows (orchestration)

▶ Data Catalog

▶ Data Integration and ETL

▶ Legacy pages

What's New Documentation AWS Marketplace

Enable compact mode Enable new navigation

Databases (1)

Last updated (UTC) July 15, 2024 at 16:02:50

Name	Description	Location URI	Created on (UTC)
dev	-	-	July 15, 2024 at 16:02:49

ap-southeast-2.console.aws.amazon.com/glue/home?region=ap-southeast-2#/v2/data-catalog/tables/view/dev?database=dev&catalogId=9054184...

AWS Services Search [Alt+S] Sydney Hem_Prakash_Dev

AWS Glue

- Getting started
- ETL jobs
- Visual ETL
- Notebooks
- Job run monitoring
- Data Catalog tables
- Data connections
- Workflows (orchestration)

▶ Data Catalog

▶ Data Integration and ETL

▶ Legacy pages

What's New Documentation AWS Marketplace

Enable compact mode Enable new navigation

⌚ One table successfully created
The following table is now created: "dev (db:dev)"

AWS Glue > Tables > dev

dev Last updated (UTC)
July 15, 2024 at 16:04:39 Version 0 (Current version) Actions

Table overview Data quality New

Name	Description	Database	Classification
dev	-	dev	JSON

Table details Advanced properties

Name dev	Description -	Database dev	Classification JSON
Location -	Connection -	Deprecated -	Last updated July 15, 2024 at 16:04:39
Input format org.apache.hadoop.mapred.TextInputFormat	Output format org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat	Serde serialization lib org.openx.data.jsonserde.JsonSerDe	

Schema Partitions Indexes

Schema (0) Edit schema as JSON Edit schema

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Set Up Amazon EMR for Big Data Processing

ap-southeast-2.console.aws.amazon.com/emr/home?region=ap-southeast-2#/clusters

aws Services Search [Alt+S]

Amazon EMR X

Amazon EMR > EMR on EC2: Clusters

Clusters (0) Info

Filter clusters by status Find clusters Filter clusters by creation date-time

No Clusters

No Clusters to display.

Clusters Cluster ID Cluster name Status Creation time (UTC+05:30)

EMR Serverless

EMR on EC2 Clusters Notebooks and Git repos Events Block public access Security configurations

EMR on EKS Virtual clusters

EMR Studio Getting Started Studios Workspaces (Notebooks)

What's New Video tour

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ap-southeast-2.console.aws.amazon.com/emr/home?region=ap-southeast-2#/createCluster

aws Services Search [Alt+S]

Amazon EMR > EMR on EC2: Clusters > Create cluster

Create cluster Info

▼ Name and applications - required Info

Name your cluster and choose the applications that you want to install to your cluster.

Name My cluster

Amazon EMR release Info

A release contains a set of applications which can be installed on your cluster.

emr-7.1.0

Application bundle

Spark Interactive Core Hadoop Flink HBase Presto Trino Custom

AmazonCloudWatchAgent 1.300032.2 Flink 1.18.1 HBase 2.4.17

HCatalog 3.1.3 Hadoop 3.3.6 Hive 3.1.3

Hue 4.11.0 JupyterEnterpriseGateway 2.6.0 JupyterHub 1.5.0

Summary Info

Provisioning configuration

Core size: 1 instance Task size: 1 instance

Networking - required

VPC vpc-000d75683... Subnet subnet-01f62d...

Cluster termination

Cluster termination Terminate cluster after idle time Idle time: 1 hour

Cluster logs

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Identity and Access Management (IAM)

Role AWSServiceRoleForAmazonEMRServerless created.

Search

Role name Trusted entities

Roles Anywhere Info

Authenticate your non AWS workloads and securely provide access to AWS services.

Access AWS from your non AWS workloads

Operate your non AWS workloads using the same authentication and authorization strategy that you use within AWS.

X.509 Standard

Use your own existing PKI infrastructure or use [AWS Certificate Manager Private Certificate Authority](#) to authenticate identities.

Temporary credentials

Use temporary credentials with ease and benefit from the enhanced security they provide.

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/roles/details/AWSServiceRoleForAmazonEMRServerless?section=permissions

Identity and Access Management (IAM)

IAM > Roles > AWSServiceRoleForAmazonEMRServerless

AWSServiceRoleForAmazonEMRServerless Info

Allows access to other AWS service resources that are required to run Amazon EMR Serverless

Summary

Creation date: July 15, 2024, 21:59 (UTC+05:30) ARN: arn:aws:iam::905418436402:role/aws-service-role/ops.emr-serverless.amazonaws.com/AWSServiceRoleForAmazonEMRServerless

Last activity: - Maximum session duration: 1 hour

Permissions Trust relationships Tags Access Advisor

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/users

Identity and Access Management (IAM)

User created successfully

You can view and download the user's password and email instructions for signing in to the AWS Management Console.

IAM > Users

Users (1) Info

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

Search User name Path Group Last activity MFA Pass

User name	Path	Group	Last activity	MFA	Pass
hem-dev	/	0	-	-	-

View user Delete Create user

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/users/details/hem-dev?section=permissions

aws Services Search [Alt+S] Global ▾ Hem_Prakash_Dev ▾

Identity and Access Management (IAM)

IAM > Users > hem-dev

hem-dev Info Delete

Summary

ARN arn:aws:iam::905418436402:user/he m-dev	Console access Disabled	Access key 1 Create access key
Created July 15, 2024, 22:01 (UTC+05:30)	Last console sign-in -	

Permissions Groups Tags Security credentials Access Advisor

Permissions policies (0) Add permissions ▾

Permissions are defined by policies attached to the user directly or through groups.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Create AWS Lambda Functions

us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/begin

aws Services Search [Alt+S] N. Virginia ▾ Hem_Prakash_Dev ▾

Compute

AWS Lambda

lets you run code without thinking about servers.

You pay only for the compute time that you consume — there is no charge when your code is not running. With Lambda, you can run code for virtually any type of application or backend service, all with zero administration.

Get started

Author a Lambda function from scratch, or choose from one of many preconfigured examples.

Create a function

How it works

Run Next: Lambda responds to events

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/functions/lemdaconfigure?newFunction=true&tab=code

Services Search [Alt+S] N. Virginia Hem_Prakash_Dev

Successfully created the function **lemdaconfigure**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

Code Test Monitor Configuration Aliases Versions

Code source Info Upload from ▾

File Edit Find View Go Tools Window Test Deploy Environment Vari

Go to Anything (Ctrl-P)

lambda_function Environment Vari λ

```
1 import json
2
3 print('Loading function')
4
5
6 def lambda_handler(event, context):
7     #print("Received event: " + json.dumps(event, indent=2))
8     print("value1 = " + event['key1'])
9     print("value2 = " + event['key2'])
10    print("value3 = " + event['key3'])
11    return event['key1'] # Echo back the first key value
12    #raise Exception('Something went wrong')
```

1:1 Python Spaces: 4

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Successfully updated the function **test_event**.

Code source Info Upload from ▾

File Edit Find View Go Tools Window Test Deploy Changes not deployed Environment Vari

Go to Anything (Ctrl-P)

lambda_function Environment Vari adder.py λ

```
1 import json
2 from adder import add
3
4 print('Loading function')
5
6
7 def lambda_handler(event, context):
8     print("Received event: " + json.dumps(event, indent=2))
9     print(f"the sum of 5 and 4 is {add(5,4)}")
10    |
```