Top 25 AWS Services

Friday, November 22, 2024

9:51 AM

import com.amazonaws.AmazonServiceException;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;

import com.amazonaws.regions.Regions;

// EC2

import com.amazonaws.services.ec2.AmazonEC2;

import com.amazonaws.services.ec2.AmazonEC2ClientBuilder;

import com.amazonaws.services.ec2.model.DescribeInstancesRequest;

import com.amazonaws.services.ec2.model.DescribeInstancesResult;

// S3

import com.amazonaws.services.s3.AmazonS3;

import com.amazonaws.services.s3.AmazonS3ClientBuilder;

import com.amazonaws.services.s3.model.Bucket;

// RDS

import com.amazonaws.services.rds.AmazonRDS;

import com.amazonaws.services.rds.AmazonRDSClientBuilder;

import com.amazonaws.services.rds.model.DescribeDBInstancesRequest;

import com.amazonaws.services.rds.model.DescribeDBInstancesResult;

// DynamoDB

import com.amazonaws.services.dynamodbv2.AmazonDynamoDB;

import com.amazonaws.services.dynamodbv2.AmazonDynamoDBClientBuilder;

import com.amazonaws.services.dynamodbv2.document.DynamoDB;

import com.amazonaws.services.dynamodbv2.document.Table;

// Lambda

import com.amazonaws.services.lambda.AWSLambda;

import com.amazonaws.services.lambda.AWSLambdaClientBuilder;

import com.amazonaws.services.lambda.model.InvokeRequest;

import com.amazonaws.services.lambda.model.InvokeResult;

// VPC

import com.amazonaws.services.ec2.model.DescribeVpcsRequest;

import com.amazonaws.services.ec2.model.DescribeVpcsResult;

// CloudFront

import com.amazonaws.services.cloudfront.AmazonCloudFront;

import com.amazonaws.services.cloudfront.AmazonCloudFrontClientBuilder;

import com.amazonaws.services.cloudfront.model.ListDistributionsRequest;

import com.amazonaws.services.cloudfront.model.ListDistributionsResult;

// SQS

import com.amazonaws.services.sqs.AmazonSQS;

import com.amazonaws.services.sqs.AmazonSQSClientBuilder;

import com.amazonaws.services.sqs.model.CreateQueueRequest;

import com.amazonaws.services.sqs.model.GetQueueUrlResult;

// SNS

import com.amazonaws.services.sns.AmazonSNS;

import com.amazonaws.services.sns.AmazonSNSClientBuilder;

import com.amazonaws.services.sns.model.CreateTopicRequest;

import com.amazonaws.services.sns.model.CreateTopicResult;

// Kinesis

import com.amazonaws.services.kinesis.AmazonKinesis;

import com.amazonaws.services.kinesis.AmazonKinesisClientBuilder;

import com.amazonaws.services.kinesis.model.PutRecordRequest;

import com.amazonaws.services.kinesis.model.PutRecordResult;

// Glue

import com.amazonaws.services.glue.AWSGlue;

import com.amazonaws.services.glue.AWSGlueClientBuilder;

import com.amazonaws.services.glue.model.StartJobRunRequest;

import com.amazonaws.services.glue.model.StartJobRunResult;

// Redshift

import com.amazonaws.services.redshift.AmazonRedshift;

import com.amazonaws.services.redshift.AmazonRedshiftClientBuilder;

import com.amazonaws.services.redshift.model.DescribeClustersRequest;

import com.amazonaws.services.redshift.model.DescribeClustersResult;

// Elasticsearch (OpenSearch)

import com.amazonaws.services.opensearch.AWSOpenSearch;

import com.amazonaws.services.opensearch.AWSOpenSearchClientBuilder;

import com.amazonaws.services.opensearch.model.DescribeDomainsRequest;

import com.amazonaws.services.opensearch.model.DescribeDomainsResult;

// Cognito

import com.amazonaws.services.cognitoidentity.AmazonCognitoIdentity;

import com.amazonaws.services.cognitoidentity.AmazonCognitoIdentityClientBuilder;

import com.amazonaws.services.cognitoidentity.model.ListIdentityPoolsRequest;

import com.amazonaws.services.cognitoidentity.model.ListIdentityPoolsResult;

// Route 53

import com.amazonaws.services.route53.AmazonRoute53;

import com.amazonaws.services.route53.AmazonRoute53ClientBuilder;

import com.amazonaws.services.route53.model.ListHostedZonesRequest;

import com.amazonaws.services.route53.model.ListHostedZonesResult;

// IAM

import com.amazonaws.services.identitymanagement.AmazonIdentityManagement;

import com.amazonaws.services.identitymanagement.AmazonIdentityManagementClientBuilder;

import com.amazonaws.services.identitymanagement.model.ListUsersRequest;

import com.amazonaws.services.identitymanagement.model.ListUsersResult;

// CloudWatch

import com.amazonaws.services.cloudwatch.AmazonCloudWatch;

import com.amazonaws.services.cloudwatch.AmazonCloudWatchClientBuilder;

import com.amazonaws.services.cloudwatch.model.PutMetricDataRequest;

import com.amazonaws.services.cloudwatch.model.PutMetricDataResult;

import com.amazonaws.services.cloudwatch.model.MetricDatum;

import com.amazonaws.services.cloudwatch.model.StandardUnit;

// EBS

import com.amazonaws.services.ec2.model.DescribeVolumesRequest;

import com.amazonaws.services.ec2.model.DescribeVolumesResult;

// Auto Scaling

import com.amazonaws.services.autoscaling.AmazonAutoScaling;

import com.amazonaws.services.autoscaling.AmazonAutoScalingClientBuilder;

import com.amazonaws.services.autoscaling.model.DescribeAutoScalingGroupsRequest;

import com.amazonaws.services.autoscaling.model.DescribeAutoScalingGroupsResult;

// EKS

import com.amazonaws.services.eks.AmazonEKS;

import com.amazonaws.services.eks.AmazonEKSClientBuilder;

import com.amazonaws.services.eks.model.DescribeClusterRequest;

import com.amazonaws.services.eks.model.DescribeClusterResult;

// Fargate (ECS)

import com.amazonaws.services.ecs.AmazonECS;

import com.amazonaws.services.ecs.AmazonECSClientBuilder;

import com.amazonaws.services.ecs.model.ListTasksRequest;

import com.amazonaws.services.ecs.model.ListTasksResult;

// Textract

import com.amazonaws.services.textract.AmazonTextract;

import com.amazonaws.services.textract.AmazonTextractClientBuilder;

import com.amazonaws.services.textract.model.Document;

import com.amazonaws.services.textract.model.S3Object;

import com.amazonaws.services.textract.model.DetectDocumentTextRequest;

import com.amazonaws.services.textract.model.DetectDocumentTextResult;

// Rekognition

import com.amazonaws.services.rekognition.AmazonRekognition;

import com.amazonaws.services.rekognition.AmazonRekognitionClientBuilder;

import com.amazonaws.services.rekognition.model.DetectLabelsRequest;

import com.amazonaws.services.rekognition.model.DetectLabelsResult;

import com.amazonaws.services.rekognition.model.Image;

import com.amazonaws.services.rekognition.model.S3Object;

// Kendra

import com.amazonaws.services.kendra.AmazonKendra;

import com.amazonaws.services.kendra.AmazonKendraClientBuilder;

import com.amazonaws.services.kendra.model.QueryRequest;

import com.amazonaws.services.kendra.model.QueryResult;

import java.nio.ByteBuffer;

import java.nio.charset.StandardCharsets;

import java.util.List;

public class AWSAllServicesDemo {

public static void main(String[] args) {

// Credentials and Regions

ProfileCredentialsProvider credentialsProvider = new ProfileCredentialsProvider();

Regions region = Regions.US\_EAST\_1;

// Example interactions with AWS services

// 1. Amazon EC2

AmazonEC2 ec2Client = AmazonEC2ClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

DescribeInstancesRequest describeInstancesRequest = new DescribeInstancesRequest();

DescribeInstancesResult describeInstancesResult = ec2Client.describeInstances(describeInstancesRequest);

System.out.println("EC2 Instances: " + describeInstancesResult.getReservations());

// 2. Amazon S3

AmazonS3 s3Client = AmazonS3ClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

List<Bucket> buckets = s3Client.listBuckets();

System.out.println("S3 Buckets: " + buckets);

// 3. Amazon RDS

AmazonRDS rdsClient = AmazonRDSClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

DescribeDBInstancesRequest describeDBInstancesRequest = new DescribeDBInstancesRequest();

DescribeDBInstancesResult describeDBInstancesResult = rdsClient.describeDBInstances(describeDBInstancesRequest);

System.out.println("RDS Instances: " + describeDBInstancesResult.getDBInstances());

// 4. Amazon DynamoDB

AmazonDynamoDB dynamoDBClient = AmazonDynamoDBClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

DynamoDB dynamoDB = new DynamoDB(dynamoDBClient);

Table table = dynamoDB.getTable("your-table-name");

System.out.println("DynamoDB Table: " + table.getTableName());

// 5. AWS Lambda

AWSLambda lambdaClient = AWSLambdaClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

InvokeRequest invokeRequest = new InvokeRequest()

.withFunctionName("your-lambda-function")

.withPayload("{ \"key\": \"value\" }");

InvokeResult invokeResult = lambdaClient.invoke(invokeRequest);

System.out.println("Lambda invoke result: " + new String(invokeResult.getPayload().array(), StandardCharsets.UTF\_8));

// 6. Amazon VPC

DescribeVpcsRequest describeVpcsRequest = new DescribeVpcsRequest();

DescribeVpcsResult describeVpcsResult = ec2Client.describeVpcs(describeVpcsRequest);

System.out.println("VPCs: " + describeVpcsResult.getVpcs());

// 7. Amazon CloudFront

AmazonCloudFront cloudFrontClient = AmazonCloudFrontClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

ListDistributionsRequest listDistributionsRequest = new ListDistributionsRequest();

ListDistributionsResult listDistributionsResult = cloudFrontClient.listDistributions(listDistributionsRequest);

System.out.println("CloudFront Distributions: " + listDistributionsResult.getDistributionList());

// 8. Amazon SQS

AmazonSQS sqsClient = AmazonSQSClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

CreateQueueRequest createQueueRequest = new CreateQueueRequest("MyQueue");

GetQueueUrlResult queueUrl = sqsClient.createQueue(createQueueRequest);

System.out.println("SQS Queue URL: " + queueUrl.getQueueUrl());

// 9. Amazon SNS

AmazonSNS snsClient = AmazonSNSClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

CreateTopicRequest createTopicRequest = new CreateTopicRequest("MyTopic");

CreateTopicResult createTopicResult = snsClient.createTopic(createTopicRequest);

System.out.println("SNS Topic ARN: " + createTopicResult.getTopicArn());

// 10. Amazon Kinesis

AmazonKinesis kinesisClient = AmazonKinesisClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

PutRecordRequest putRecordRequest = new PutRecordRequest()

.withStreamName("YourStream")

.withData(ByteBuffer.wrap("Data".getBytes(StandardCharsets.UTF\_8)))

.withPartitionKey("partitionKey");

PutRecordResult putRecordResult = kinesisClient.putRecord(putRecordRequest);

System.out.println("Kinesis record: " + putRecordResult.getSequenceNumber());

// 11. AWS Glue

AWSGlue glueClient = AWSGlueClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

StartJobRunRequest startJobRunRequest = new StartJobRunRequest().withJobName("MyGlueJob");

StartJobRunResult startJobRunResult = glueClient.startJobRun(startJobRunRequest);

System.out.println("Glue Job ID: " + startJobRunResult.getJobRunId());

// 12. Amazon Redshift

AmazonRedshift redshiftClient = AmazonRedshiftClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

DescribeClustersRequest describeClustersRequest = new DescribeClustersRequest();

DescribeClustersResult describeClustersResult = redshiftClient.describeClusters(describeClustersRequest);

System.out.println("Redshift Clusters: " + describeClustersResult.getClusters());

// 13. Amazon OpenSearch (formerly Elasticsearch)

AWSOpenSearch openSearchClient = AWSOpenSearchClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

DescribeDomainsRequest describeDomainsRequest = new DescribeDomainsRequest().withDomainNames("MyDomain");

DescribeDomainsResult describeDomainsResult = openSearchClient.describeDomains(describeDomainsRequest);

System.out.println("OpenSearch Domains: " + describeDomainsResult.getDomainStatusList());

// 14. Amazon Cognito

AmazonCognitoIdentity cognitoClient = AmazonCognitoIdentityClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

ListIdentityPoolsRequest listIdentityPoolsRequest = new ListIdentityPoolsRequest().withMaxResults(10);

ListIdentityPoolsResult listIdentityPoolsResult = cognitoClient.listIdentityPools(listIdentityPoolsRequest);

System.out.println("Cognito Identity Pools: " + listIdentityPoolsResult.getIdentityPools());

// 15. Amazon Route 53

AmazonRoute53 route53Client = AmazonRoute53ClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

ListHostedZonesRequest listHostedZonesRequest = new ListHostedZonesRequest();

ListHostedZonesResult listHostedZonesResult = route53Client.listHostedZones(listHostedZonesRequest);

System.out.println("Route 53 Hosted Zones: " + listHostedZonesResult.getHostedZones());

// 16. AWS IAM

AmazonIdentityManagement iamClient = AmazonIdentityManagementClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

ListUsersRequest listUsersRequest = new ListUsersRequest();

ListUsersResult listUsersResult = iamClient.listUsers(listUsersRequest);

System.out.println("IAM Users: " + listUsersResult.getUsers());

// 17. Amazon CloudWatch

AmazonCloudWatch cloudWatchClient = AmazonCloudWatchClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

MetricDatum datum = new MetricDatum()

.withMetricName("MyMetric")

.withUnit(StandardUnit.None)

.withValue(1.0);

PutMetricDataRequest request = new PutMetricDataRequest()

.withNamespace("MyNamespace")

.withMetricData(datum);

PutMetricDataResult response = cloudWatchClient.putMetricData(request);

System.out.println("CloudWatch Metric Data Result: " + response);

// 18. Amazon EBS

DescribeVolumesRequest describeVolumesRequest = new DescribeVolumesRequest();

DescribeVolumesResult describeVolumesResult = ec2Client.describeVolumes(describeVolumesRequest);

System.out.println("EBS Volumes: " + describeVolumesResult.getVolumes());

// 19. Auto Scaling

AmazonAutoScaling autoScalingClient = AmazonAutoScalingClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

DescribeAutoScalingGroupsRequest describeAutoScalingGroupsRequest = new DescribeAutoScalingGroupsRequest();

DescribeAutoScalingGroupsResult describeAutoScalingGroupsResult = autoScalingClient.describeAutoScalingGroups(describeAutoScalingGroupsRequest);

System.out.println("Auto Scaling Groups: " + describeAutoScalingGroupsResult.getAutoScalingGroups());

// 20. Amazon EKS

AmazonEKS eksClient = AmazonEKSClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

DescribeClusterRequest describeClusterRequest = new DescribeClusterRequest().withName("MyCluster");

DescribeClusterResult describeClusterResult = eksClient.describeCluster(describeClusterRequest);

System.out.println("EKS Cluster: " + describeClusterResult.getCluster());

// 21. AWS Fargate (ECS)

AmazonECS ecsClient = AmazonECSClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

ListTasksRequest listTasksRequest = new ListTasksRequest().withCluster("MyCluster");

ListTasksResult listTasksResult = ecsClient.listTasks(listTasksRequest);

System.out.println("ECS Tasks: " + listTasksResult.getTaskArns());

// 22. Amazon Textract

AmazonTextract textractClient = AmazonTextractClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

Document document = new Document().withS3Object(new S3Object().withBucket("your-s3-bucket").withName("document.png"));

DetectDocumentTextRequest detectDocumentTextRequest = new DetectDocumentTextRequest().withDocument(document);

DetectDocumentTextResult detectDocumentTextResult = textractClient.detectDocumentText(detectDocumentTextRequest);

System.out.println("Detected Document Text: " + detectDocumentTextResult.getBlocks());

// 23. Amazon Rekognition

AmazonRekognition rekognitionClient = AmazonRekognitionClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

Image image = new Image().withS3Object(new S3Object().withBucket("your-s3-bucket").withName("image.jpg"));

DetectLabelsRequest detectLabelsRequest = new DetectLabelsRequest().withImage(image);

DetectLabelsResult detectLabelsResult = rekognitionClient.detectLabels(detectLabelsRequest);

System.out.println("Rekognition Labels: " + detectLabelsResult.getLabels());

// 24. Amazon Kendra

AmazonKendra kendraClient = AmazonKendraClientBuilder.standard()

.withCredentials(credentialsProvider)

.withRegion(region)

.build();

QueryRequest queryRequest = new QueryRequest().withIndexId("your-kendra-index-id").withQueryText("What is AWS?");

QueryResult queryResult = kendraClient.query(queryRequest);

System.out.println("Kendra Query Results: " + queryResult.getResultItems());

}

}