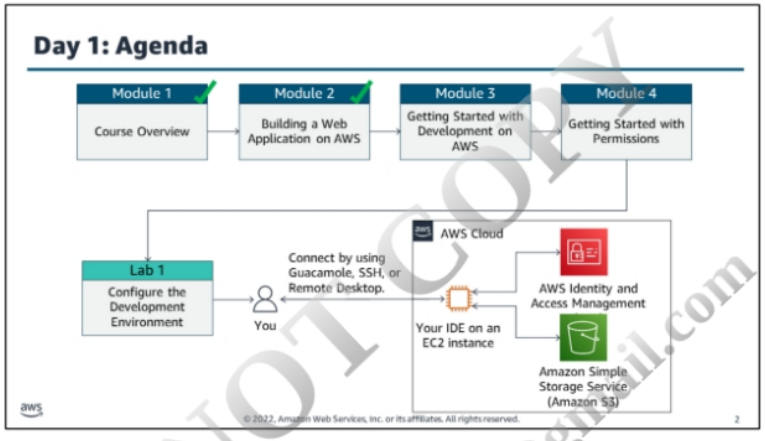
Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification

Module 3: Getting Started with Development on AWS

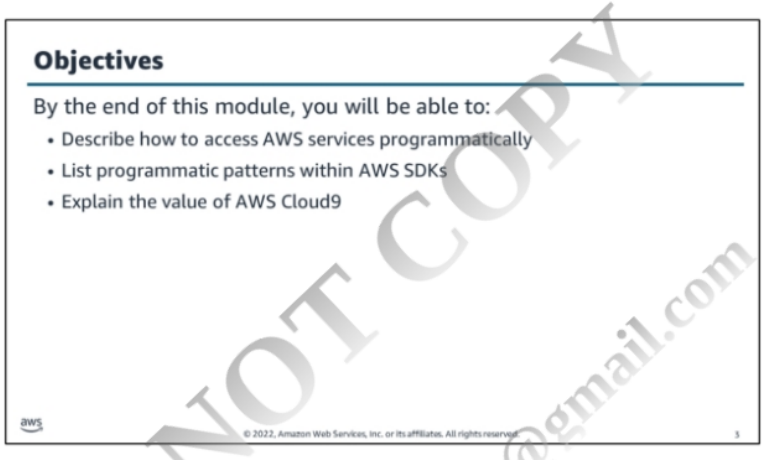


In this module, you will focus on foundational knowledge required to start your development with AWS resources.

The module concludes with a brief demonstration of AWS Cloud9. The demonstration explores the AWS Cloud9 UI and building with AWS services from the command line interface.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



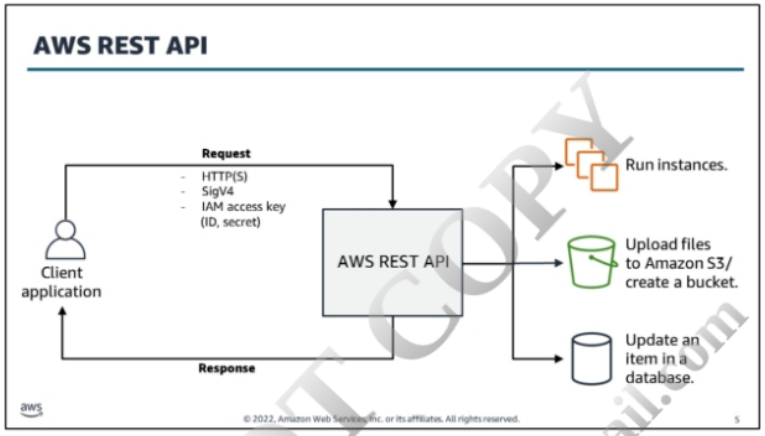
DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



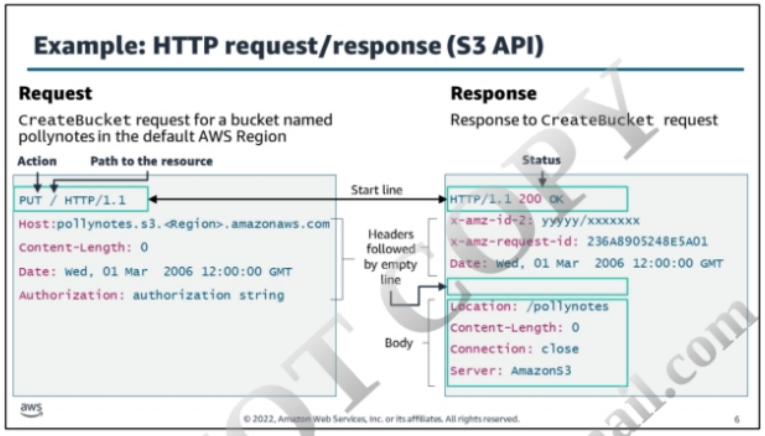
All AWS services support a dedicated application programming interface (API) to expose their features. To connect programmatically to an AWS service, you use an AWS service endpoint. An endpoint is the URL of the entry point for an AWS offering where the API is exposed.

For security, most requests to AWS must be signed with an access key, which consists of an access key ID and secret access key. AWS supports Signature Version 4 (SigV4) as the process to add authentication information to AWS requests sent by HTTP. Signing does more than allow AWS to identify who sent the request. It also prevents the following:
Tampering with the request while in transit (uses hash values as part of the request)
Potential replay attacks (uses timestamps to expire events that take too long to arrive)

DO NOT COPY sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS



APIs provide a standard way of communicating. The structure of the request and the response messages are similar. They each have a Start line, and optionally headers and body sections.

Request and response start lines
The start line of the request message indicates the action to be taken on the server. The start line includes the path to the resource on the web server. The resource path must have a value. If it is not specified, it is indicated by a forward slash (/), which represents the server root.

The start line of the response indicates the success or failure of the request.

Headers provide metadata associated with an API request and response. Some headers—such as Host, which indicates the domain of the server—are well defined. Headers can also be custom. Headers are followed by an empty line to indicate the end of the header fields.

The format of a header is as follows: <header field> ":" <field-value>

DO NOT COPY sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Types of headers:

General – Apply to both request and response
Example: Date: Wed, 01 Mar 2006 12:00:00 GMT
Client Request – Apply to the request message
Example: Host:pollynotes.s3.<Region>.amazonaws.com
Server Response – Apply to the respond message
Example: x-amz-request-id: 236A8905248E5A01
Resource – Contain metadata about the "body" or the resource identified in the request
Example: Content-Type: text/plain

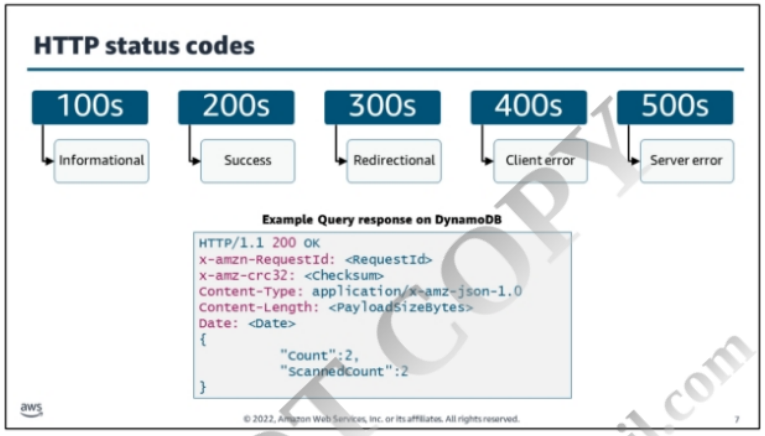
The body, sometimes called a payload, contains content that will be transmitted between the server and the client.

Reference
The examples are from "CreateBucket" in the Amazon Simple Storage Service (S3) API Reference (https://docs.aws.amazon.com/AmazonS3/latest/API/API_CreateBucket.html).

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS

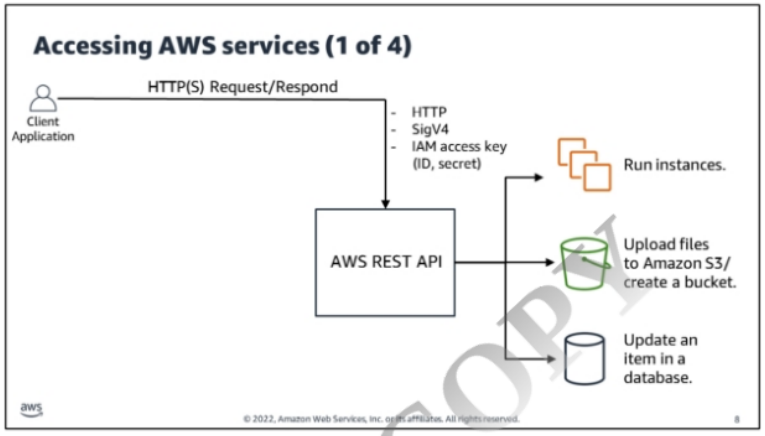


When a service responds to a request, the response header contains a status and a status message.

HTTP status code classes:
1xx – Informational response indicates that everything is OK so far.
2xx – Server successfully received the request and successfully processed the request.
3xx – Redirectional response status indicates that a further action is necessary to complete the request.
4xx – Client error, indicating possibly incorrect syntax, or a different user error that prevents fulfilling the request.
5xx – Server error that presented what seems to be a valid request from being fulfilled.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



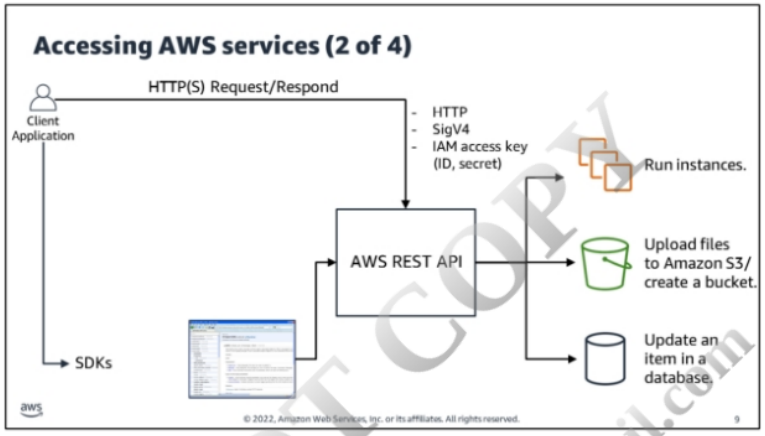
You can use several ways to access AWS services and features. All four methods are built on the REST API of the AWS services.

Application programming interface (API) – All AWS services support dedicated APIs to expose their features, and you can use these APIs directly. However, formatting requests, interpreting, and responding directly can be significant work, especially because each service has its own specific details.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

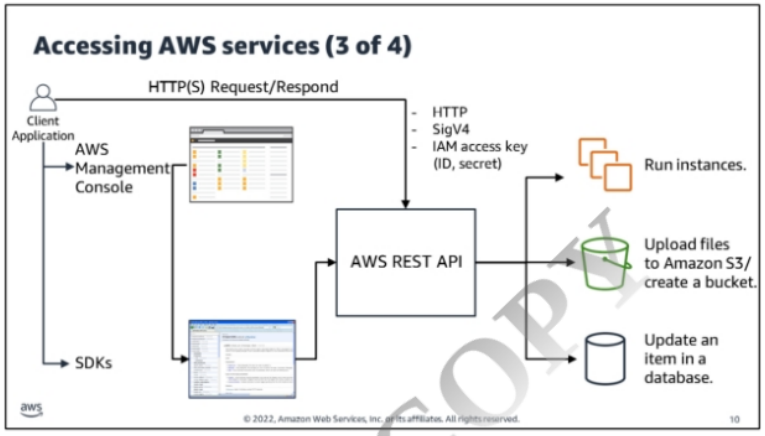
AWS Training and Certification Module 3: Getting Started with Development on AWS



Software Development Kits (SDKs) – AWS has created SDKs to simplify the programmatic use of the APIs. Consider the need for HTTP requests to be signed with your AWS access key. A developer can format the API request with this information. However, when requests are made using the AWS CLI or the AWS SDK, requests are automatically signed. AWS CLI and SDKs also handle retries and errors so that you don't have to manually program that logic. AWS provides SDK packages that provide functionality to access AWS in a variety of popular programming languages. This flexibility simplifies the use of AWS in your existing applications. It also enables creating applications to deploy and monitor complex systems entirely through code.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

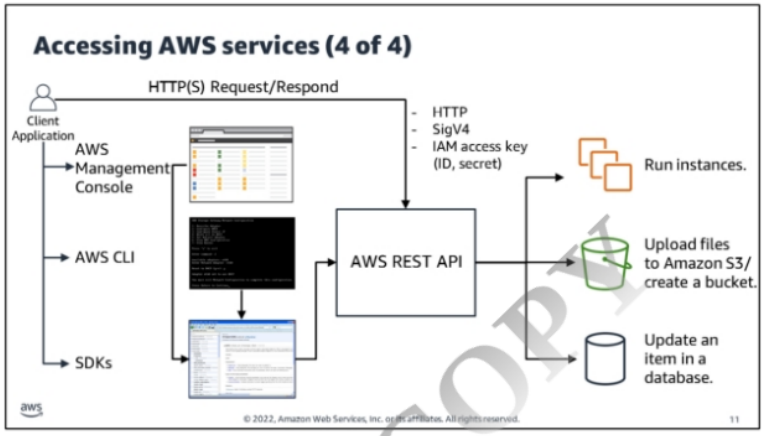


AWS Management Console – The console provides a rich graphical interface to a majority of the features AWS offers. However, it is often more efficient to work with AWS services programmatically instead of through the console.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS

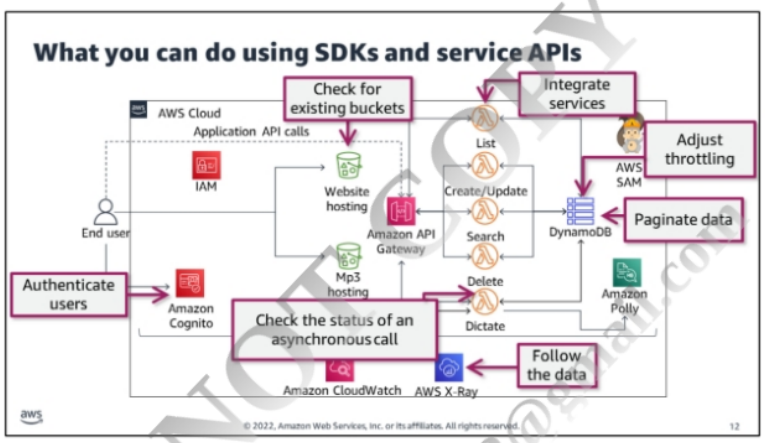


AWS Command Line Interface (AWS CLI) –You can use the AWS CLI to manage your AWS services directly from the command program in Linux/macOS or Windows. You can create scripts consistently and run them as needed. The AWS CLI uses the Python SDK.

You can use all modes of access interchangeably.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

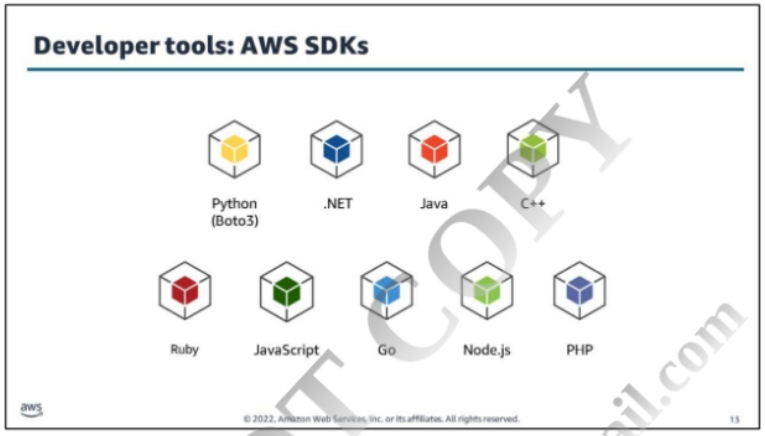


When you use AWS APIs
These are some typical usage patterns that you may encounter. You might find that you need to perform the following operations:
Grant access to your application
Adjust the throttling to accounts for limited bandwidth
Integrate several services

The AWS SDKs cover and simplify some of these patterns.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



AWS provides a robust set of tools for developers to develop applications that will run on AWS.

AWS SDK
Consider the example of a music-sharing application. The application must upload the user's music files to an S3 bucket. In this case, the application might use the AWS SDK to upload files to the S3 bucket programmatically.

With the AWS SDK, you can use AWS services in your application by using your preferred programming language. SDKs are available in a number of programming languages and technology platforms, such as Android, iOS, Go, Java, JavaScript, .NET, Node.js, PHP, Python, and Ruby.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

For more SDK information and installation instructions, see the following programming languages:

Python: https://boto3.readthedocs.org/en/latest/guide/quickstart.html

.NET:

https://docs.aws.amazon.com/AWSSdkDocsNET/latest/V3/DeveloperGuide/net-dg-install-assemblies.html

https://docs.aws.amazon.com/AWSSdkDocsNET/latest/V3/DeveloperGuide/net-dg-start-new-project.html

Java: https://docs.aws.amazon.com/AWSSdkDocsJava/latest/DeveloperGuide/java-dg-install-sdk.html

AWS Amplify: https://docs.amplify.aws/

AWS SDK for Ruby: https://aws.amazon.com/sdk-for-ruby/

AWS SDK for JavaScript in Node.js: https://aws.amazon.com/sdk-for-javascript/

AWS SDK for Go: https://docs.aws.amazon.com/sdk-for-go/v1/developer-guide/welcome.html

AWS SDK for JavaScript: https://docs.aws.amazon.com/AWSJavaScriptSDK/latest/

AWS SDK for PHP: https://docs.aws.amazon.com/sdk-for-php/v3/developer-guide/welcome.html

AWS SDK for C++: https://aws.amazon.com/sdk-for-cpp/

API references:

AWS CLI: https://docs.aws.amazon.com/pt_br/cli/latest/index.html

AWS SDK for Python (Boto3): https://boto3.amazonaws.com/v1/documentation/api/latest/index.html#

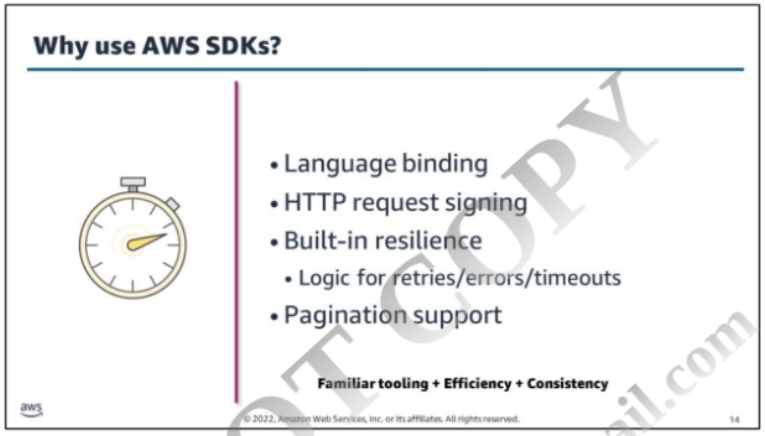
AWS SDK for Java: https://docs.aws.amazon.com/AWSJavaSDK/latest/javadoc/overview-summary.html

AWS SDK for .NET: https://docs.aws.amazon.com/sdkfornet/v3/apidocs

DO NOT COPY sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

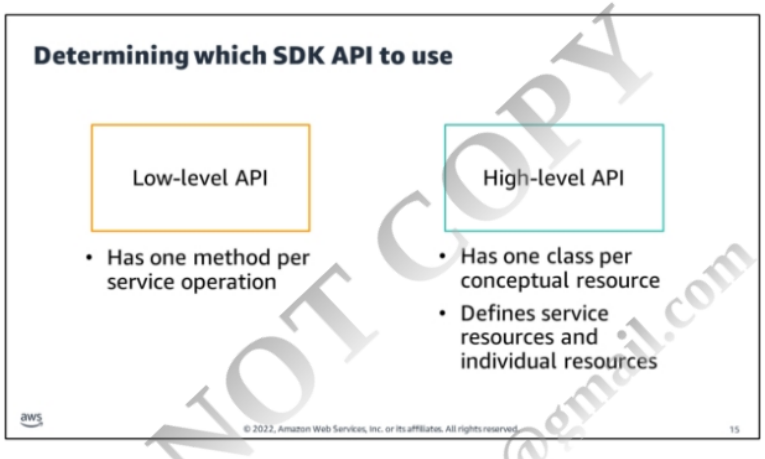
AWS Training and Certification Module 3: Getting Started with Development on AWS



When interacting with AWS services programmatically through AWS CLI or one of the AWS SDKs, you benefit from a number of efficiencies. The APIs can automatically calculate signatures, handle errors, and even request retries. You don't have to write the request-parsing logic or the retry logic in the same way you would if you used the API directly. For example, with a single line of code, you can use a paginator. By doing so, you avoid writing the logic of transferring the data using continuation tokens, loops, and multiple API calls.

DO NOT COPY sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



You can connect to an AWS service by using either high-level or low-level APIs. In general, if the high-level API provides methods you need to perform an operation, use the high-level API because of its simplicity. For example, when uploading a file to an Amazon S3 bucket, the high-level API uses the file size to determine whether to upload the file in a single operation or upload it in multiple parts. High-level APIs are easier to understand and simpler to use because they provide a higher-level abstraction than low-level APIs.

Low-level API (Service client API in Python)
AWS SDKs provide a set of client classes, each exposing a direct mapping of an AWS service's API. These client objects have a method for each operation that the service supports, with corresponding objects representing the request parameters and the response data. Using this low-level client API gives you full control over the requests that you make to the service. You can tightly control the behavior and performance of your calls to AWS services.

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS

High-level API (Resource API in Python)
Some AWS SDKs, such as the AWS SDK for Python (Boto3), provide higher-level APIs called the resource APIs. The resource APIs provide a higher-level abstraction than the low-level calls made by clients. The high-level APIs consist of classes that represent each of the conceptual resources that you interact with when using a service. These classes expose methods to do the following:
Retrieve data about the resource
Invoke actions that can be taken on the resource
Retrieve links to other related resources

High-level APIs are easier to understand and simpler to use because they provide a higher-level abstraction and built-in efficiencies than client APIs.

DO NOT COPY
sameersheik68@gmail.com

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Example: Connecting to a service using API

Low-level API - List objects in bucket

```
# Return type: dict / additional API calls needed to get objects
def listClient():
    s3client = boto3.client('s3')
    response = s3client.list_objects_v2(Bucket='mybucket')
    for content in response['Contents']:
        print(content['Key'], content['LastModified'])
```

Python

High-level API - List objects in bucket

```
# Resources represent an object-oriented interface to AWS. They provide a
# higher-level abstraction than the raw, low-level calls made by service clients
def listResource():
    s3resource = boto3.resource('s3')
    bucket = s3resource.Bucket('mybucket')
    for object in bucket.objects.all():
        print(object.key, object.last_modified)
```

aws © 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved. 16

DO NOT COPY sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS

Locking the API version

- AWS services have versioned APIs
- You can lock at the SDK level or service level

Example configuration

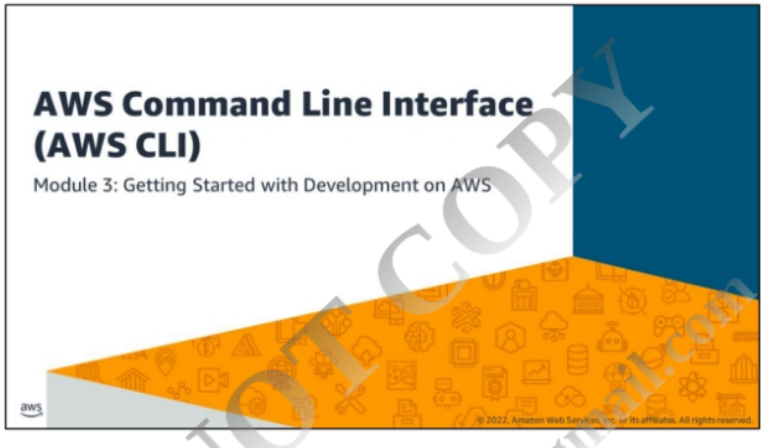
```
[profile development]
aws_access_key_id=foo
aws_secret_access_key=bar
api_versions =
  ec2 = 2015-03-01
  cloudfront = 2015-09-17
```

aws © 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved. 17

APIs can change. Therefore, if you rely on a particular version of the API in your code, AWS recommends that you lock the service to the version number that you are using. You can also lock to the SDK version. However, to optimize your application, pull in only the components you need instead of the entire SDK.

DO NOT COPY sameersheik68@gmail.com

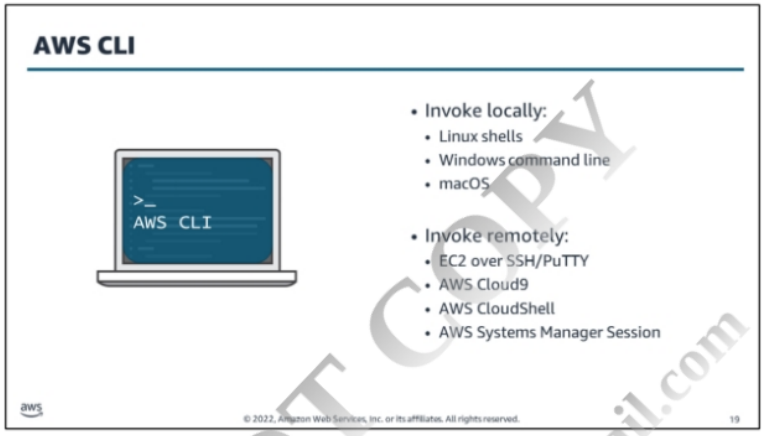
Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS



You can use AWS services by running the AWS Command Line Interface (AWS CLI) commands from your terminal program.
Linux shells – Run commands in Linux or macOS through common shell programs such as bash, zsh, and tcsh.
Windows command line – Run commands at the Windows command prompt or in PowerShell.
Remotely – Run commands on Amazon Elastic Compute Cloud (Amazon EC2) instances through a remote terminal program, such as PuTTY or Secure Shell (SSH), or with AWS Systems Manager.

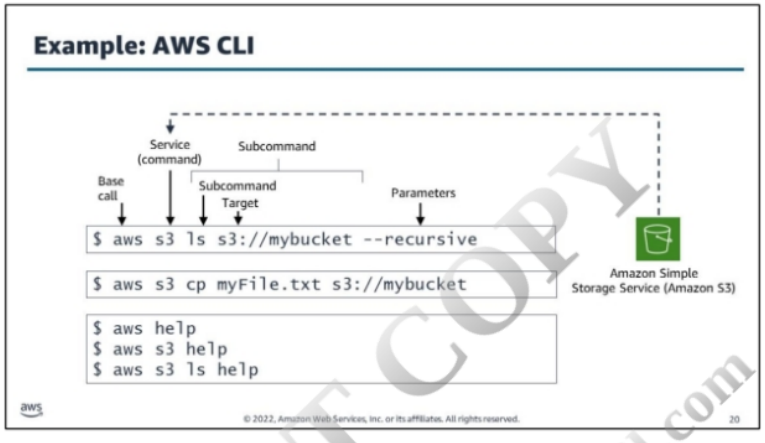
AWS Cloud9 EC2 environments come preinstalled with AWS CLI, which is authenticated with the permissions of the logged-in AWS user automatically.

The AWS CLI can be downloaded from the "AWS Command Line Interface" webpage (http://aws.amazon.com/cli/).

The AWS CLI is available for use on Linux, Windows, and macOS platforms. You can also install it on Linux and Windows EC2 instances. To set a number of defaults for all AWS CLI commands, use the aws configure command on machines that host the AWS CLI. By default, the access key ID and the secret access key are associated with the AWS Identity and Access Management (IAM) account that you use to access AWS resources. You can also specify a default AWS Region for all commands.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



The AWS CLI uses a Python library named Botocore. The AWS Python SDK Boto3 library is built on top of Botocore. The basic structure of an AWS command line call is the aws command, followed the following parts:

- Command – The top-level service that you are calling (for example, Amazon S3, Amazon EC2, Amazon CloudWatch). For a list of currently supported services, see the AWS CLI Command Reference (http://docs.aws.amazon.com/cli/latest/reference/).
- Subcommand – The operation to perform on that service (for example, run-instances, put-metric-data). In this Amazon S3 example, the list subcommand requests listing the content of the target bucket.
- Parameters – Any arguments required to perform the operation. Argument names are preceded by two dashes (--). For example, the recursive parameter will list all the content in a bucket in order.
- Options – The AWS CLI also provides options that you can specify when running the operation.

Here is a CLI example:

```
$ aws s3 ls s3://mybucket --recursive
```

The base call is `aws`, the service or command is `s3`, the subcommand is ls with a target of `s3://mybuket`. And the optional parameter is `--recursive`.

You can get help on commands at different levels. For example:

```
$ aws help
$ aws s3 help
$ aws s3 ls help
```

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.

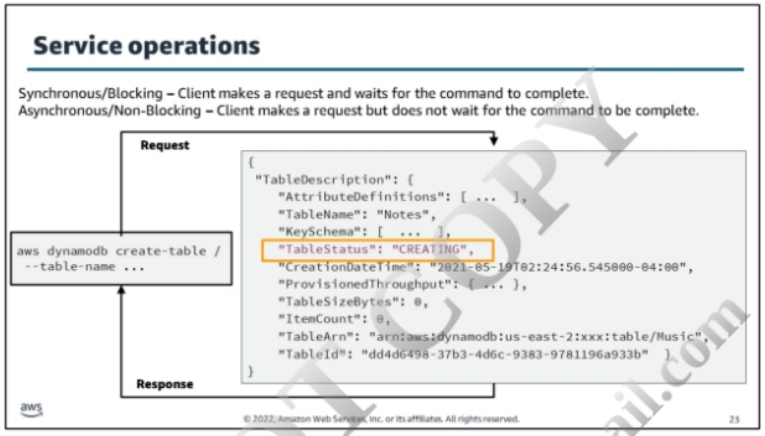
Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS

DO NOT COPY
sameersheik68@gmail.com

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Operation calls on AWS services can be synchronous or asynchronous. Lambda can be invoked synchronously or asynchronously. Amazon S3 invokes functions asynchronously. The CreateTable operation for DynamoDB is an asynchronous operation. When making an asynchronous call, it is assumed that some resilience is built on the client side to handle errors and retries.

Synchronous operation
A synchronous call blocks and prevents the client from continuing with the program until the client receives a response from the service.

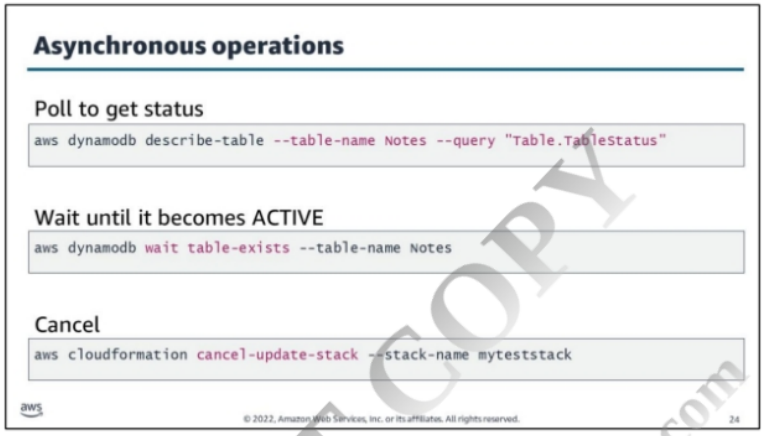
Asynchronous operation
An asynchronous call does not stop the client thread from continuing to run. Asynchronous calls are not dependent on each other and do not block the client from running. An asynchronous call returns control to the client before the service returns a response. This is an effective way to call long-latency operations. The application receives a response that indicates the request has been received. The application will not know whether the request will be successful—not right away. It can continue running and not slow down while waiting for a response.

An asynchronous call allows the client to continue running before a response is available. Consequently, you need a way to get the response when it's ready or determine whether the request failed.

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS



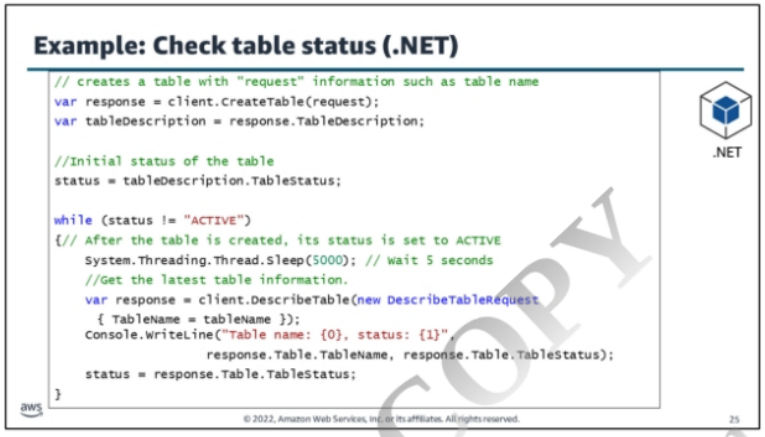
Waiters are an abstraction for polling a resource until a desired state is reached or an error indicates that the state cannot be reached.

With asynchronous programming, polling to ensure that a state of a service is necessary in some use cases. For example, starting the infrastructure before you can use it may require you to "wait" and poll the resources for their state. You must ensure that a resource is ready before you can use it.

The waiter utility provides an API for handling the polling task. This means that you don't have to write the polling logic for each AWS resource that you are using and its specific polling APIs and success states. You can configure waiters with a maximum number of attempts and the back-off strategy between each attempt.

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

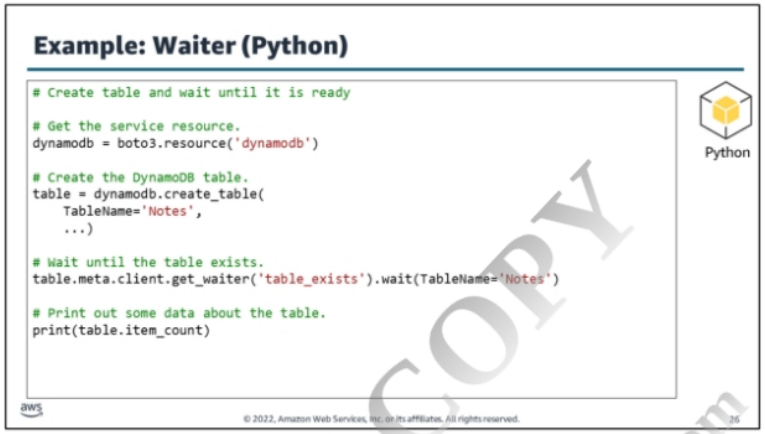


The CreateTable response includes the TableDescription property that provides the initial table information. You can also call the DescribeTable method of the client. at any time, to get latest table information.
In the example, the operation loops and polls DynamoDB, until it creates the table and sets its status to ACTIVE. The CreateTable response includes the TableDescription property that provides the necessary table information.

For more information about the example, see "Working with DynamoDB Tables in .NET" in the Amazon DynamoDB Developer Guide (https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/LowLevelDotNetWorkingWithTables.html).

DO NOT COPY
sameersheik68@gmail.com

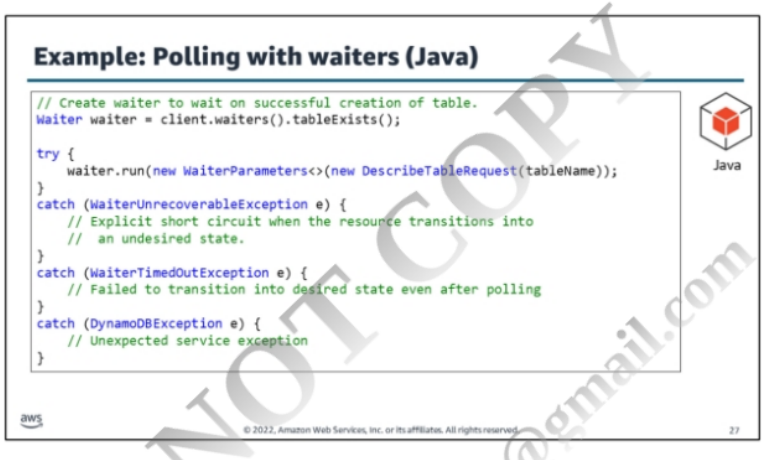
Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



For more information about this example, see "Amazon DynamoDB" in the Boto3 documentation (https://boto3.amazonaws.com/v1/documentation/api/latest/guide/dynamodb.html)

DO NOT COPY
sameersheik68@gmail.com

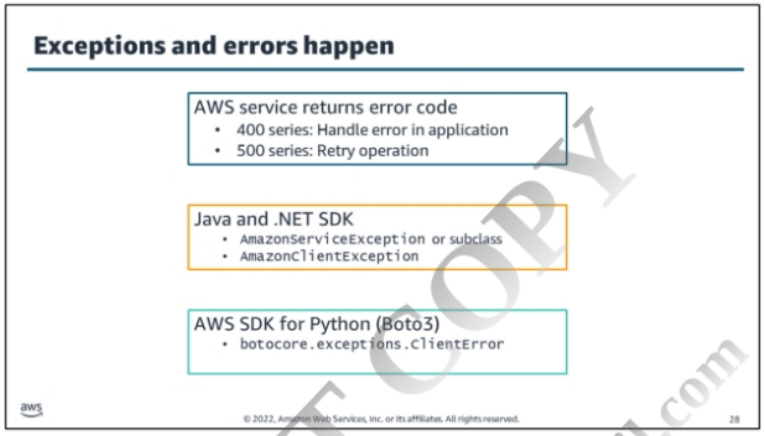
Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



For more information about the example, see "Waiters in the AWS SDK for Java" in the AWS Developer blog (https://aws.amazon.com/blogs/developer/waiters-in-the-aws-sdk-for-java/).

DO NOT COPY sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



When an error occurs, an AWS service returns an error code that you can use to determine how to handle the error. For example, Amazon S3 returns a 404 error code if the bucket you are trying to access does not exist. Your application can handle this error by first creating the bucket and then performing operations on it.

Sometimes an AWS service can return an error code, such as 500, that indicates an internal server error. In this case, you could retry the operation. Each AWS SDK implements automatic retry logic. You can configure the maximum number of retry attempts. For more information, see "Error Handling with DynamoDB" in the Amazon DynamoDB Developer Guide (http://docs.aws.amazon.com/amazondynamodb/latest/developerguide/ErrorHandling.html#APIRetries).

Different AWS SDKs communicate error situations in different ways. Examples from some SDKs are described.

DO NOT COPY sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS SDK for Java

The AWS SDK for Java throws the following unchecked (runtime) exceptions when errors occur:
com.amazonaws.AmazonServiceException – This exception (or its subclass) indicates that the request was correctly transmitted to the service. However, the service was not able to process it and returned an error response instead. The exception has the following information that you can use to determine how your application should handle the error:

- HTTP status code returned by the service
- AWS error code returned by the service
- Detailed error message from the service
- AWS request ID for the failed request

com.amazonaws.AmazonClientException – This exception indicates that a problem occurred inside the Java client code, either while trying to send a request to AWS or while trying to parse a response from AWS. For example, if no network connection is available when you call an operation on one of the clients, the AWS SDK for Java throws an AmazonClientException.
java.lang.IllegalArgumentException – This exception is thrown if you pass an illegal argument when performing an operation on a service.

For more information, see "Developer guide - AWS SDK for Java 2.x" in the AWS SDK for Java Developer Guide (https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/handling-exceptions.html).

AWS SDK for .NET

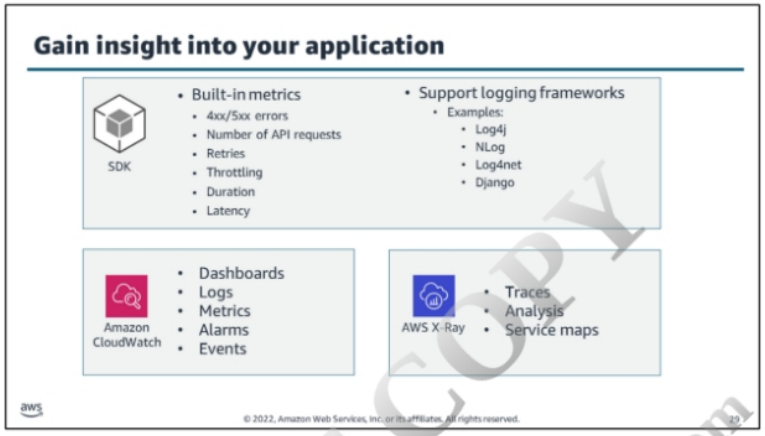
The AWS SDK for .NET throws Amazon.Runtime.AmazonServiceException and Amazon.Runtime.AmazonClientException, similar to the AWS SDK for Java.

AWS SDK for Python (Boto3)
Boto3 SDK throws botocore.exceptions.ClientError on errors.

DO NOT COPY sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS



Make your system observable. For example, you can enable metrics for the SDKs or services, and logs. CloudWatch collects, aggregates, and summarizes compute utilization information like CPU, memory, disk, and network data, as well as diagnostic information. AWS X-Ray provides an end-to-end, cross-service view of requests made to your application. It aggregates data gathered from individual services in your application into a single unit called a trace (path of a request as it passes through each service or tier in your application), creates a map of services with trace data, allows you to drill into this data for analysis.

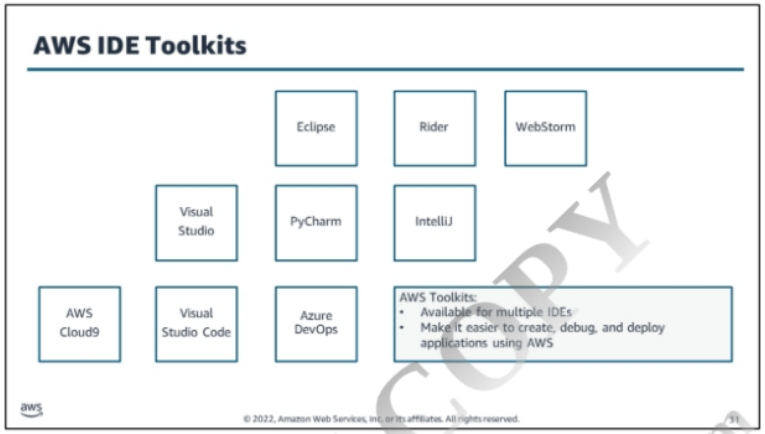
DO NOT COPY sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS



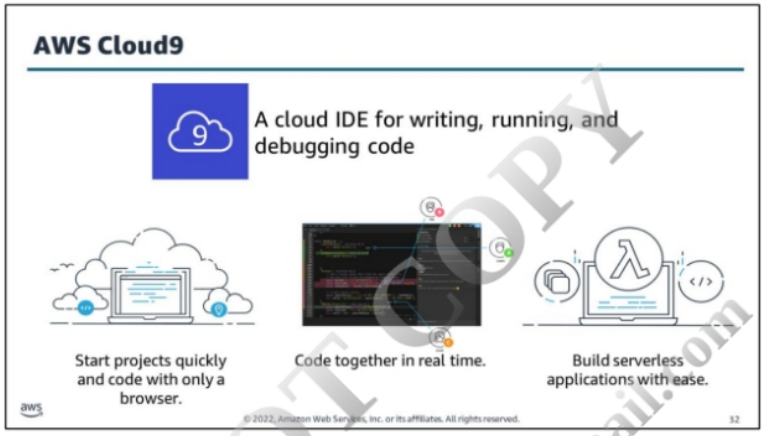
You can use an integrated development environment (IDE) to write, run, and debug code for your applications. Because the tools you need as developer can be accessed from within a single application, IDEs improve your productivity. For example, editing code in an IDE increases the readability of code with syntax highlighting, autocomplete/hints for syntax. IDEs help you gain a deep understanding of code with compilers or interpreters and debugging features.

IDEs can be locally installed or cloud-based. Cloud-based IDEs offer many benefits, such as the following:

- You can access the IDE from any computer or OS through a web browser.
- IDEs are supported by computing power that may be superior to your local machine.
- You can quickly bring on preconfigured development environments for new projects or new developers to the team.
- They are code-collaborative and "break the silos" by working with multiple developers on the same code, at the same time.
- Use platform-specific SDKs.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



AWS Cloud9 is an IDE that supports developers who develop for the cloud. With AWS Cloud9 you can write, run, and debug code using only a browser.

You can run your development environment on a managed EC2 instance or an existing Linux server that supports Secure Shell (SSH).

By using Amazon EC2 environments, AWS Cloud9 moves your development to the cloud, and you can perform the following functions:

- Use powerful infrastructure for your development through your browser. If necessary, you can use the AWS Cloud9 terminal to manage the EC2 instance that is hosting your development environment.
- Isolate your project's resources by maintaining multiple development environments. In the cloud, the process to create an environment and use it when needed is streamlined. The environment remains dormant when not in use.
- Programmers can simultaneously edit the same code because it has features that support pair programming.

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS

AWS Cloud9 comes prepackaged with tools for popular programming languages, including Java, JavaScript, Python, PHP, .NET, and more. It is preconfigured with SDKs, libraries, and plugins to support your serverless development efforts. Use the existing setup or create an Amazon Machine Image (AMI) that is specific to your needs. Others can use the AMI for faster onboarding of new developers on the project.

AWS Cloud9 works for the application you are building in this course. The reason is that it preconfigures the development environment with all the SDKs, libraries, and plugins needed for serverless development. AWS Cloud9 provides an environment where an AWS Lambda function can be built and tested from within the IDE. The AWS Cloud9 terminal is also a pre-authenticated AWS CLI so that you use it to build with AWS services.

AWS Cloud9 best practices

We recommend the following best practices for using your AWS Cloud9 environment:

- Use **source control**, and **back up** your environment frequently. AWS Cloud9 does not perform automatic backups.
- Perform regular **updates of software** on your environment. AWS Cloud9 does not perform automatic updates.
- **Turn on AWS CloudTrail in your AWS account** to track activity in your environment.
- Share your environment with only **trusted users**. Sharing your environment might put your AWS access credentials at risk.

DO NOT COPY
sameersheik68@gmail.com

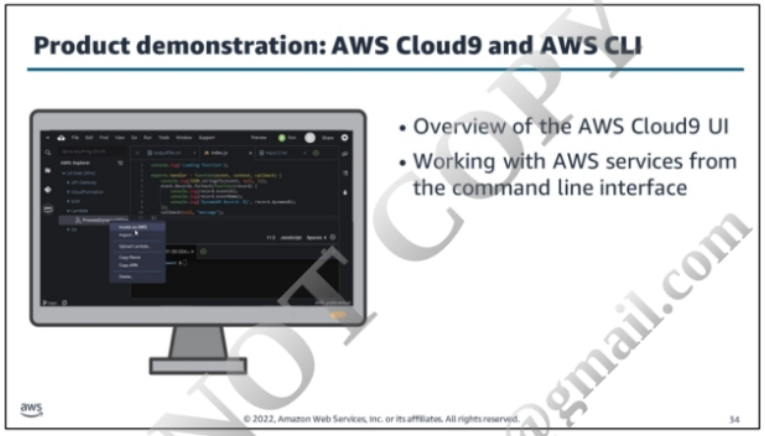
© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification
Module 3: Getting Started with Development on AWS



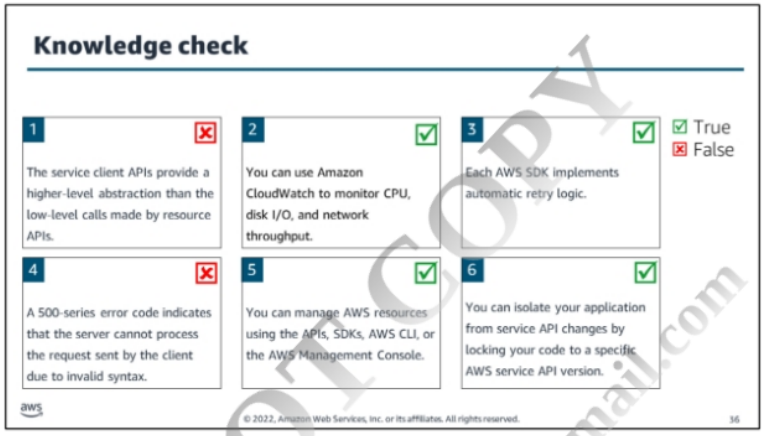
DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS



1. (False) The resource APIs provide a higher-level abstraction than the low-level calls made by clients.
2. (True)
3. (True) Each AWS SDK implements automatic retry logic, along with exponential backoff.
4. (False) When an error occurs, an AWS service returns an error code that you can use to determine how to handle the error. A 400-series error code indicates to handle the error in the application. Sometimes an AWS service can return an error code such as 500, indicating an internal server error.
5. (True)
6. (True)

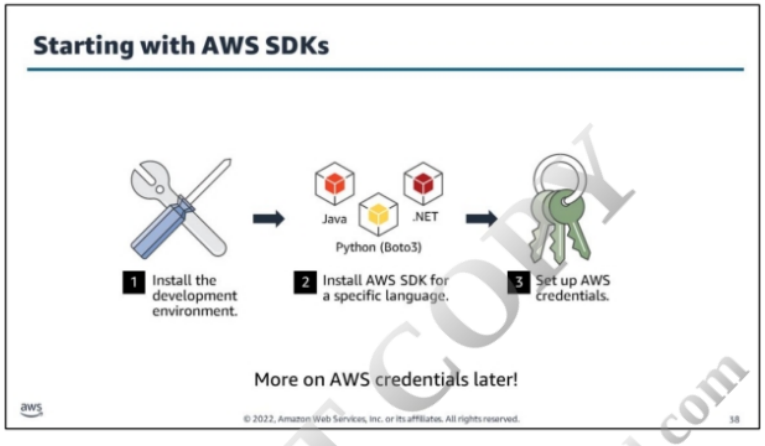
DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS



The diagram shows the steps that are required to set up your development environment to work with the AWS SDKs.

To use the AWS SDK for Java to access AWS, you need an AWS account and AWS credentials. To increase the security of your AWS account, we recommend that you use an IAM user to provide access credentials. Use your IAM user instead of your AWS account root user credentials.

DO NOT COPY
sameersheik68@gmail.com

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Module summary

You are now able to:

- Describe how to access AWS services programmatically
- List programmatic patterns within AWS SDKs
- Explain the value of AWS Cloud9

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved. 19

DO NOT COPY sameersheik68@gmail.com

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 3: Getting Started with Development on AWS

Thank you

Corrections, feedback, or other questions?
Contact us at https://support.aws.amazon.com/#/contacts/aws-training.
All trademarks are the property of their owners.

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

DO NOT COPY sameersheik68@gmail.com

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.