

# AWS Developer: Designing and Developing

---

DESIGNING AND DEVELOPING ON AWS



**Ryan Lewis**

CLOUD ENGINEER

@ryanmurakami    [ryanlewis.dev](http://ryanlewis.dev)

We're here to level up your  
proficiency and  
understanding of AWS

# Previously on AWS Developer

## AWS Developer: Getting Started

by Ryan Lewis

Amazon Web Services is the largest cloud provider in the world. This course will teach you how to develop, deploy, and integrate web applications with AWS.

[Resume Course](#) [Bookmark](#) [Add to Channel](#) [Download Course](#)

Table of contents Description Transcript Exercise files Discussion Related Courses

This course is part of:  AWS Certified Developer - Associate Path [Expand All](#)

▶ Course Overview	✓	Bookmark	1m 23s	▼
▶ Welcome to AWS	✓	Bookmark	27m 1s	▼
▶ Sounding the Alarm with IAM and Cloudwatch	✓	Bookmark	32m 3s	▼
▶ Getting Inside the Virtual Machine with EC2 and VPC	✓	Bookmark	56m 29s	▼
▶ Putting All the Things with S3	✓	Bookmark	75m 7s	▼

Course author

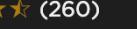


Ryan Lewis

Ryan Lewis is a Software Engineer who specializes in ambitious single page web applications. He started building websites over 15 years ago to promote his bands and record label. After traveling...

Course info

Level Beginner

Rating  (260)

My rating 

Duration 4h 6m

Updated 20 Sep 2019

Share course

[f](#) [t](#) [in](#)

## AWS Developer: Getting Started

Familiarity with AWS Services

AWS Developer: Designing  
and Developing  
Proficient with AWS Services



Super cool  
web app

# Our Tools



**AWS CLI**

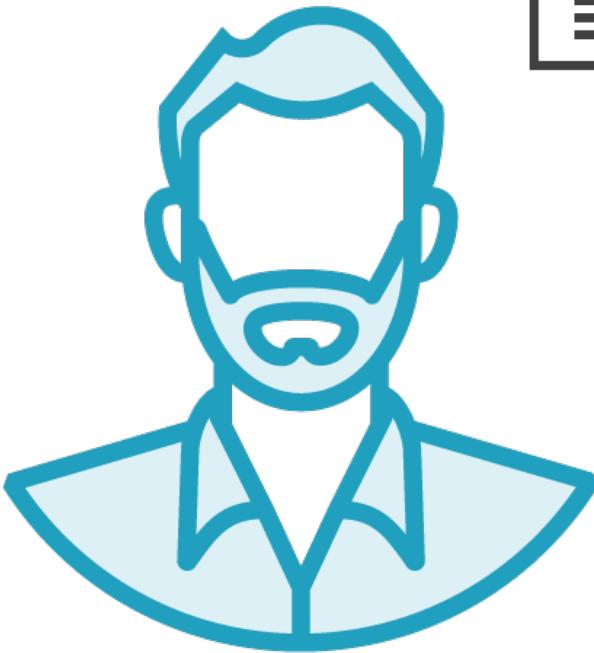


**AWS SDKs**

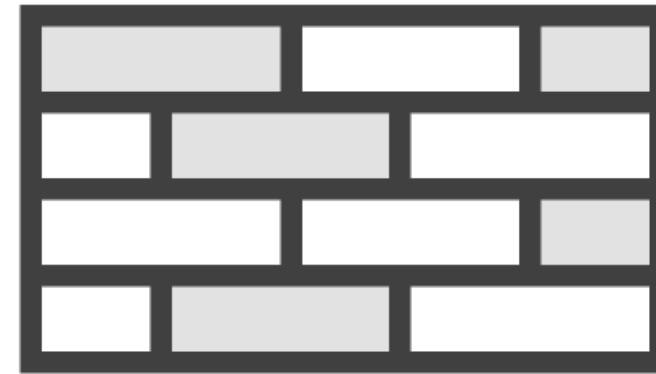
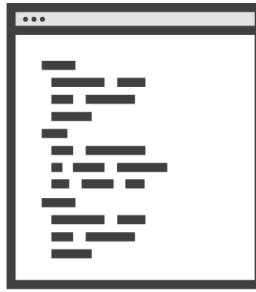
# Approaching Cloud Development on AWS

---

# The Dark Ages of Deployment



Development



Operations

# The Many Hats of a Developer

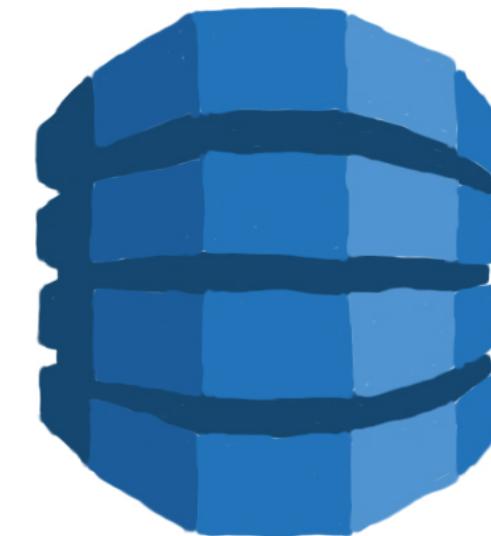
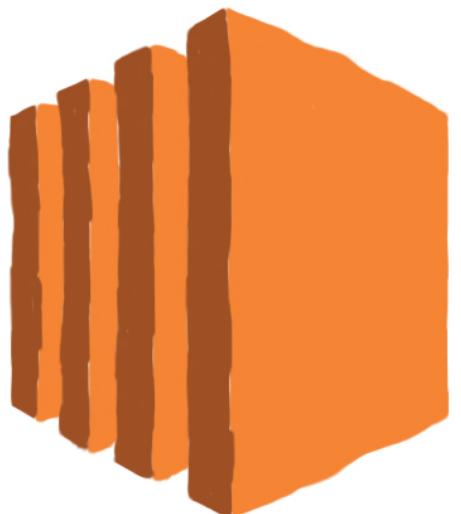
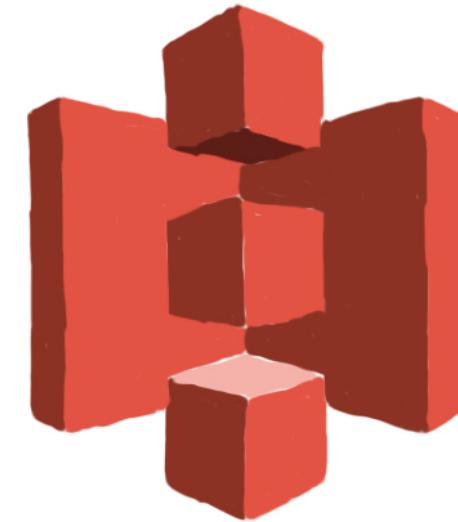
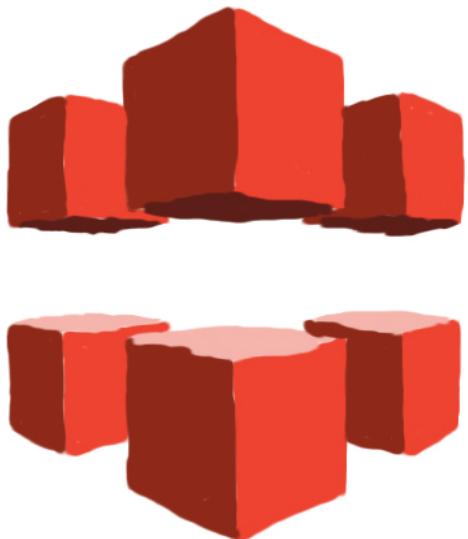
Coding

Debugging

Testing

Operations

# Your Brain on AWS



# Cloud Development 101

Start with the Infrastructure

# Cloud Development Decisions

**Computing**

EC2 or Lambda?

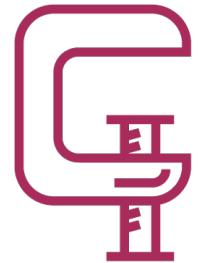
**Persistence**

S3, Elasticache, RDS, or  
DynamoDB?

**Routing**

API Gateway or Route 53  
DNS?

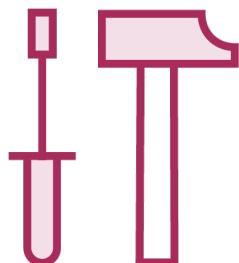
# Cloud Development Toolkit



**Elastic Cloud Compute**



**Simple Storage Service**



**DynamoDB**



**Simple Queue Service**

**Distributed and  
scalable combination  
of code and services**

**Resilient and  
Maintainable**

A Better Application

# Service Domains in AWS

---

# AWS Service Catalog by Domain

**AWS services**

**Find Services**  
You can enter names, keywords or acronyms.

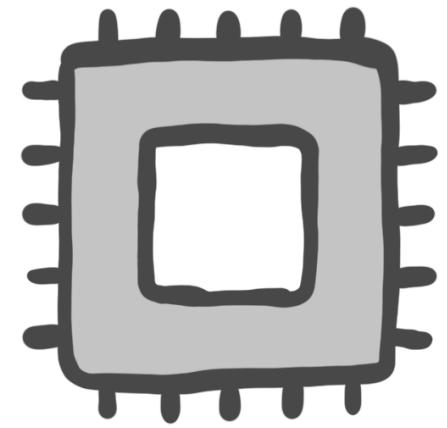
 Example: Relational Database Service, database, RDS

▶ Recently visited services

▼ All services

-  **Compute**
  - EC2
  - Lightsail 
  - ECR
  - ECS
  - EKS
  - Lambda
  - Batch
  - Elastic Beanstalk
  - Serverless Application Repository
  - AWS Outposts
  - EC2 Image Builder
-  **Storage**
  - S3
  - EFS
  - FSx
  - S3 Glacier
  - Storage Gateway
  - AWS Backup
-  **Satellite**
  - Ground Station
-  **Quantum Technologies**
  - Amazon Braket 
-  **Management & Governance**
  - AWS Organizations
  - CloudWatch
  - AWS Auto Scaling
  - CloudFormation
  - CloudTrail
  - Config
  - OpsWorks
  - Service Catalog
  - Systems Manager
  - Trusted Advisor
  - Control Tower
  - AWS License Manager
  - AWS Well-Architected Tool
  - Personal Health Dashboard 
-  **Security, Identity, & Compliance**
  - IAM
  - Resource Access Manager
  - Cognito
  - Secrets Manager
  - GuardDuty
  - Inspector
  - Amazon Macie 
  - AWS Single Sign-On
  - Certificate Manager
  - Key Management Service
  - CloudHSM
  - Directory Service
  - WAF & Shield
  - Artifact
  - Security Hub
  - Detective
-  **AWS Cost Management**
  - AWS Cost Explorer
  - AWS Budgets

AWS Marketplace Subscriptions



Compute

**Runs code that you provide**

**Services:**

- Elastic Cloud Compute (EC2)
- Lambda
- EC2 Container Service
- Elastic Beanstalk
- Batch



Storage

**Stores data**

**Services:**

- Simple Storage Service (S3)
- Elastic File System (EFS)
- Glacier
- Storage Gateway



Database

**Provides databases**

**Services:**

- Relational Database Service (RDS)
- DynamoDB
- ElastiCache
- Redshift

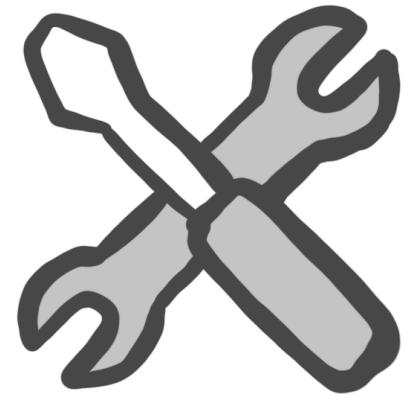


## Networking and Content Delivery

**Controls networking and routing**

**Services:**

- Virtual Private Cloud (VPC)
- CloudFront
- API Gateway
- Route 53



## Developer Tools

**Facilitates development lifecycle**

**Services:**

- CodeStar
- CodeCommit
- CodeBuild
- CodeDeploy
- CodePipeline
- X-Ray



Management  
and Governance

## **Service management and orchestration**

### **Services:**

- CloudWatch
- CloudFormation
- CloudTrail
- OpsWorks



Security, Identity  
and Compliance

## **Manages service access**

### **Services:**

- Identity and Access Management (IAM)
- Inspector
- Certificate Manager
- Directory Service
- WAF and Shield
- Artifact

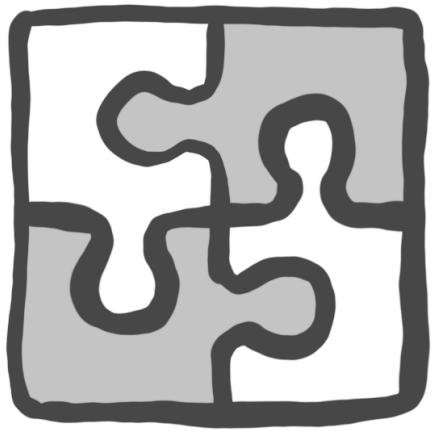


## Analytics

**Consumes and processes data**

**Services:**

- Athena
- Elastic Map Reduce (EMR)
- CloudSearch
- ElasticSearch
- Kinesis
- Data Pipeline



## Application Integration

**Integrates AWS services**

**Services:**

- Simple Notification Service
- Simple Queue Service
- Simple Workflow (SWF)
- Step Functions

Use AWS Domain types to  
give you hints to a service's  
function

# SDKs for AWS

---

# AWS Console

The screenshot shows the AWS Management Console homepage. At the top, there's a navigation bar with the AWS logo, 'Services' dropdown, 'Resource Groups' dropdown, user info (Ryan H Lewis, N. Virginia), and 'Support' dropdown. Below the header, the main title 'AWS Management Console' is displayed. On the left, there's a sidebar with 'AWS services' sections for 'Compute', 'Storage', and 'Analytics'. The 'Compute' section lists services like EC2, Lightsail, ECR, ECS, EKS, Lambda, Batch, Elastic Beanstalk, Serverless Application Repository, AWS Outposts, and EC2 Image Builder. The 'Storage' section lists S3, EFS, FSx, S3 Glacier, Storage Gateway, and AWS Backup. The 'Analytics' section lists Machine Learning, Amazon SageMaker, Amazon CodeGuru, Amazon Comprehend, Amazon Forecast, Amazon Fraud Detector, Amazon Kendra, Amazon Lex, Amazon Machine Learning, Amazon Personalize, Amazon Polly, Amazon Rekognition, Amazon Textract, Amazon Transcribe, Amazon Translate, AWS DeepLens, and AWS DeepRacer. A search bar at the top says 'Example: Relational Database Service, database, RDS'. To the right, there are three main boxes: 'Access resources on the go' (with a mobile phone icon), 'Explore AWS' (with sections for Amazon DynamoDB, Amazon SageMaker Studio, AWS Security Hub, and Free Digital Training), and 'User Feedback'.

# AWS CLI

The screenshot shows a terminal window titled '1. mrsaturn@Mr-Saturns-MacBook-Pro: ~/Code/hbfl (zsh)'. It displays the output of the 'aws configure' command. The terminal shows:

```
$ aws configure
AWS Access Key ID [*****Cm7A]:
AWS Secret Access Key [*****IvAB]:
Default region name [us-east-1]:
Default output format [json]
```

The terminal also shows the path '/Code/hbfl', version 'v8.3.0', and the current time '15:25:09'.

# AWS JavaScript SDK Documentation

**Class List**  
[Classes](#) | [Methods](#) | [Properties](#) | [Files](#)  
Search:

**Top Level Namespace**

▼ **Services**

- ▶ [AccessAnalyzer](#) < Service
- ▶ [ACM](#) < Service
- ▶ [ACMPCA](#) < Service
- ▶ [AlexaForBusiness](#) < Service
- ▶ [Amplify](#) < Service
- ▶ [APIGateway](#) < Service
- ▶ [ApiGatewayManagementApi](#) < Service
- ▶ [ApiGatewayV2](#) < Service
- ▶ [AppConfig](#) < Service
- ▶ [ApplicationAutoScaling](#) < Service
- ▶ [ApplicationInsights](#) < Service
- ▶ [AppMesh](#) < Service
- ▶ [AppStream](#) < Service
- ▶ [AppSync](#) < Service
- ▶ [Athena](#) < Service
- ▶ [AugmentedAIRuntime](#) < Service
- ▶ [AutoScaling](#) < Service
- ▶ [AutoScalingPlans](#) < Service
- ▶ [Backup](#) < Service
- ▶ [Batch](#) < Service
- ▶ [Budgets](#) < Service
- ▶ [Chime](#) < Service
- ▶ [Cloud9](#) < Service
- ▶ [CloudDirectory](#) < Service
- ▶ [CloudFormation](#) < Service
- ▶ [CloudFront](#) < Service
- ▶ [CloudHSM](#) < Service
- ▶ [CloudHSMV2](#) < Service

Index » File: README

## AWS SDK for JavaScript

npm v2.596.0 downloads 14M/month chat on gitter

build passing codecov 97% vulnerabilities 0

The official AWS SDK for JavaScript, available for browsers and mobile devices, or Node.js backends

For release notes, see the [CHANGELOG](#). Prior to v2.4.8, release notes can be found at <https://aws.amazon.com/releasenotes/?tag=releasenotes%23keywords%23javascript>

If you are upgrading from 1.x to 2.0 of the SDK, please see the [upgrading notes](#) for information on how to migrate existing code to work with the new major version.

### Installing

#### In the Browser

To use the SDK in the browser, simply add the following script tag to your HTML pages:

```
<script src="https://sdk.amazonaws.com/js/aws-sdk-2.596.0.min.js"></script>
```

You can also build a custom browser SDK with your specified set of AWS services. This can allow you to reduce the SDK's size, specify different API versions of services, or use AWS services that don't currently support CORS if you are working in an environment that does not enforce CORS. To get started: <http://docs.aws.amazon.com/sdk-for-javascript/v2/developer-guide/building-sdk-for-browsers.html>

The AWS SDK is also compatible with [browserify](#).

For browser-based web, mobile and hybrid apps, you can use [AWS Amplify Library](#) which extends the AWS SDK and provides an easier and declarative interface.

#### In Node.js

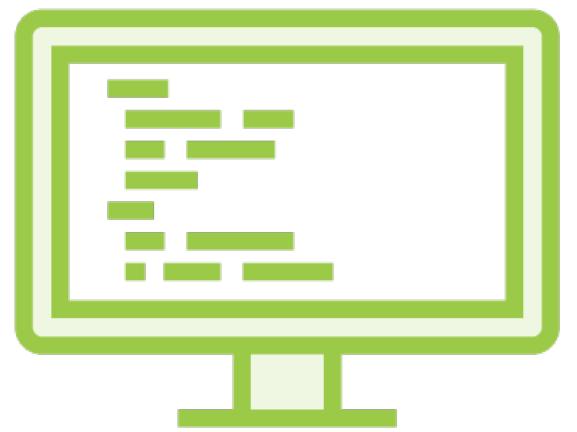
The preferred way to install the AWS SDK for Node.js is to use the [npm](#) package manager for Node.js. Simply type the following into a terminal window:

```
npm install aws-sdk
```

#### In React Native

**Table of Contents**

- [1. Installing](#)
  - [1. In the Browser](#)
  - [2. In Node.js](#)
  - [3. In React Native](#)
  - [4. Using Bower](#)
- [2. Usage and Getting Started](#)
- [3. Usage with TypeScript](#)
  - [1. Pre-requisites](#)
  - [2. In the Browser](#)
  - [3. In Node.js](#)
  - [4. With React](#)
  - [5. With Angular](#)
  - [6. Known Limitations](#)
- [4. Getting Help](#)
- [5. Opening Issues](#)
- [6. Supported Services](#)
- [7. License](#)



There are officially supported  
SDKs from AWS and non-  
supported, open source SDKs  
from third parties

# Officially Supported Server-side AWS SDKs

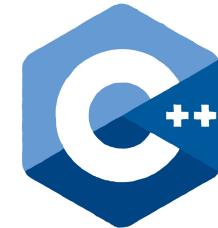
Java



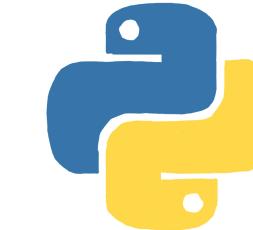
.NET



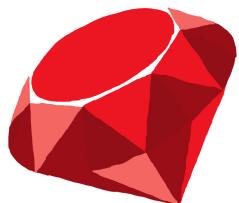
C++



Python



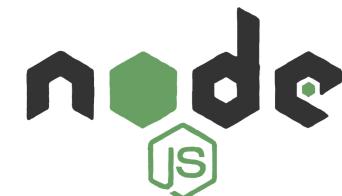
Ruby



Go

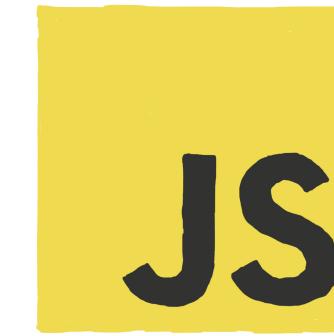


Node.js



# Officially Supported Client-side AWS SDKs

JavaScript



# Officially Supported Mobile AWS SDKs

Android



iOS

iOS

Xamarin



Unity



# We'll be using the JavaScript SDK

<https://docs.aws.amazon.com/AWSJavaScriptSDK/latest/index.html>

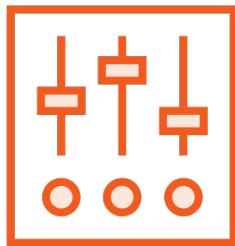
# Preparing for the AWS Developer Exam

---

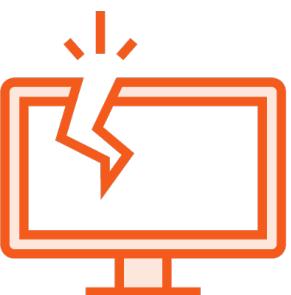
# AWS Developer Associate Certification



**Knowledge of AWS services**



**Options and details of each service**



**Different problems and scenarios when using each service**

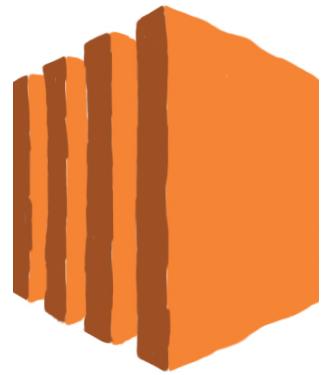
# Certification Specifics

**Multiple choice  
questions**

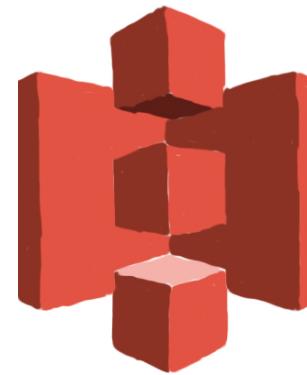
**Service specific  
topic, detail, or  
limit**

**Debugging and  
architecture**

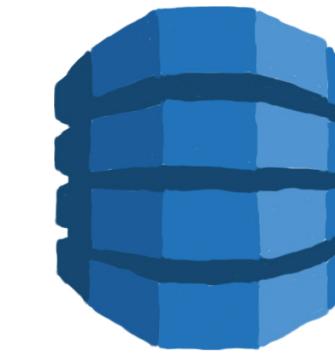
# Previously Sighted on the Exam



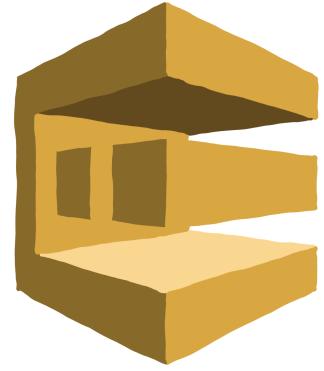
**EC2**



**S3**



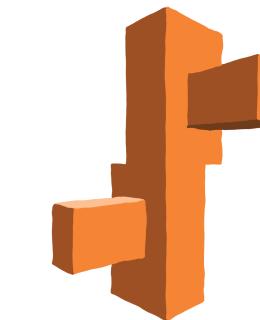
**DynamoDB**



**SQS**



**Route 53**



**Elastic Beanstalk**

Not familiar with it?  
Use it!

# Taking the AWS Certified Developer Exam?

# Demystifying the AWS Certified Developer: Associate Exam

by Ryan Lewis

This course will guide you through preparing to ace the AWS Certified Developer - Associate exam. You'll get tips on how to level up your skills, what to study, and what to expect with the exam.

[Resume Course](#) [Bookmark](#) [Add to Channel](#) [Download Course](#)

**Table of contents** Description Transcript Exercise files Discussion Learning Check Related Courses

This course is part of:  AWS Certified Developer - Associate Path [Expand All](#)

- Course Overview
- Leveling up Your AWS Skills
- Studying for the Exam
- Taking the Exam

The trademarks and trade names of third parties mentioned in this course are the property of their respective owners, and Pluralsight is not affiliated with or endorsed by these parties.

Course author



Ryan Lewis

Ryan Lewis is a Software Engineer who specializes in ambitious single page web applications. He started building websites over 15 years ago to promote his bands and record label. After traveling...

Course info

Level Intermediate

Rating ★★★★☆

My rating ★★★★☆

Duration 0h 57m

Released 4 Jun 2019

Share course

f t in

# Taking a Hamster to the Races

---

# Creating an AWS Billing Alarm

---

Create a new account or use  
an existing one

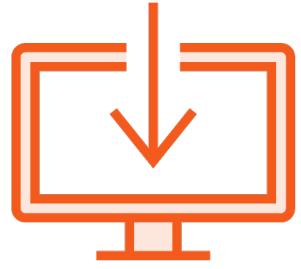
## Objective

Create a Billing Alarm

# Before We Start

---

# AWS CLI



**Download and install AWS CLI**



**Configure with a user's access key and secret**

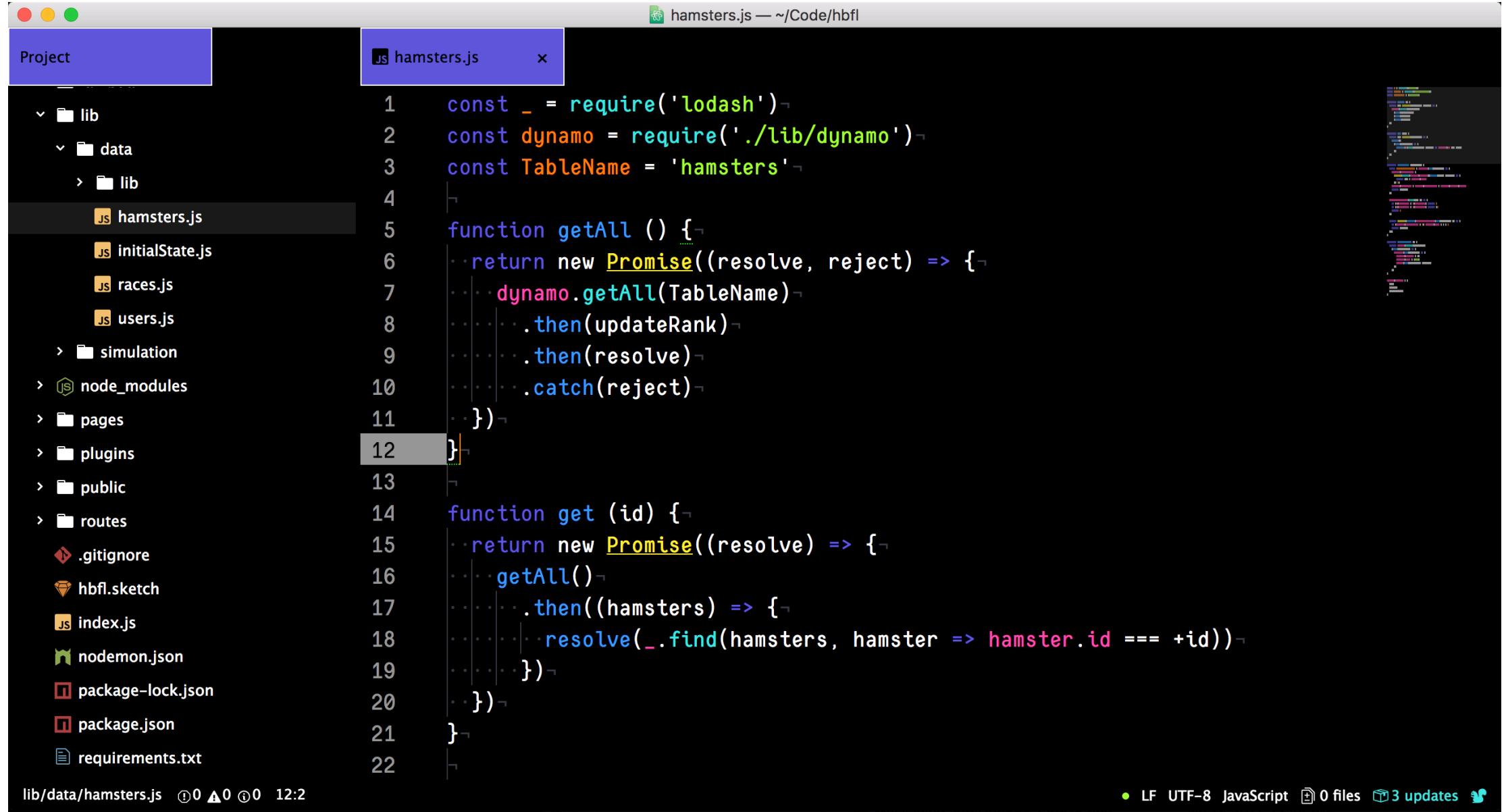


**Ensure permissions for all services used in course**

# Code Editor

Try Atom!

Download at  
[atom.io](https://atom.io)



The screenshot shows the Atom code editor interface. On the left is a sidebar titled "Project" containing a tree view of a codebase. The tree includes directories like "lib", "data", and "node\_modules", and files such as "hamsters.js", "initialState.js", "races.js", "users.js", "index.js", "nodemon.json", "package-lock.json", "package.json", and "requirements.txt". The main editor area has a dark background and displays the contents of the "hamsters.js" file. The file contains JavaScript code using promises and lodash. The status bar at the bottom shows "lib/data/hamsters.js" and "12:2". The bottom right corner of the status bar includes icons for file type (JavaScript), file count (0 files), update count (3 updates), and a refresh icon.

```
1  const _ = require('lodash')
2  const dynamo = require('../lib/dynamo')
3  const TableName = 'hamsters'
4
5  function getAll () {
6    return new Promise((resolve, reject) => {
7      dynamo.getAll(TableName)
8        .then(updateRank)
9        .then(resolve)
10       .catch(reject)
11    })
12  }
13
14  function get (id) {
15    return new Promise((resolve) => {
16      getAll()
17        .then((hamsters) => {
18          resolve(_.find(hamsters, hamster => hamster.id === +id))
19        })
20    })
21  }
22
```

● LF UTF-8 JavaScript 0 files 3 updates

# Node.js and npm



Download at [nodejs.org](https://nodejs.org)

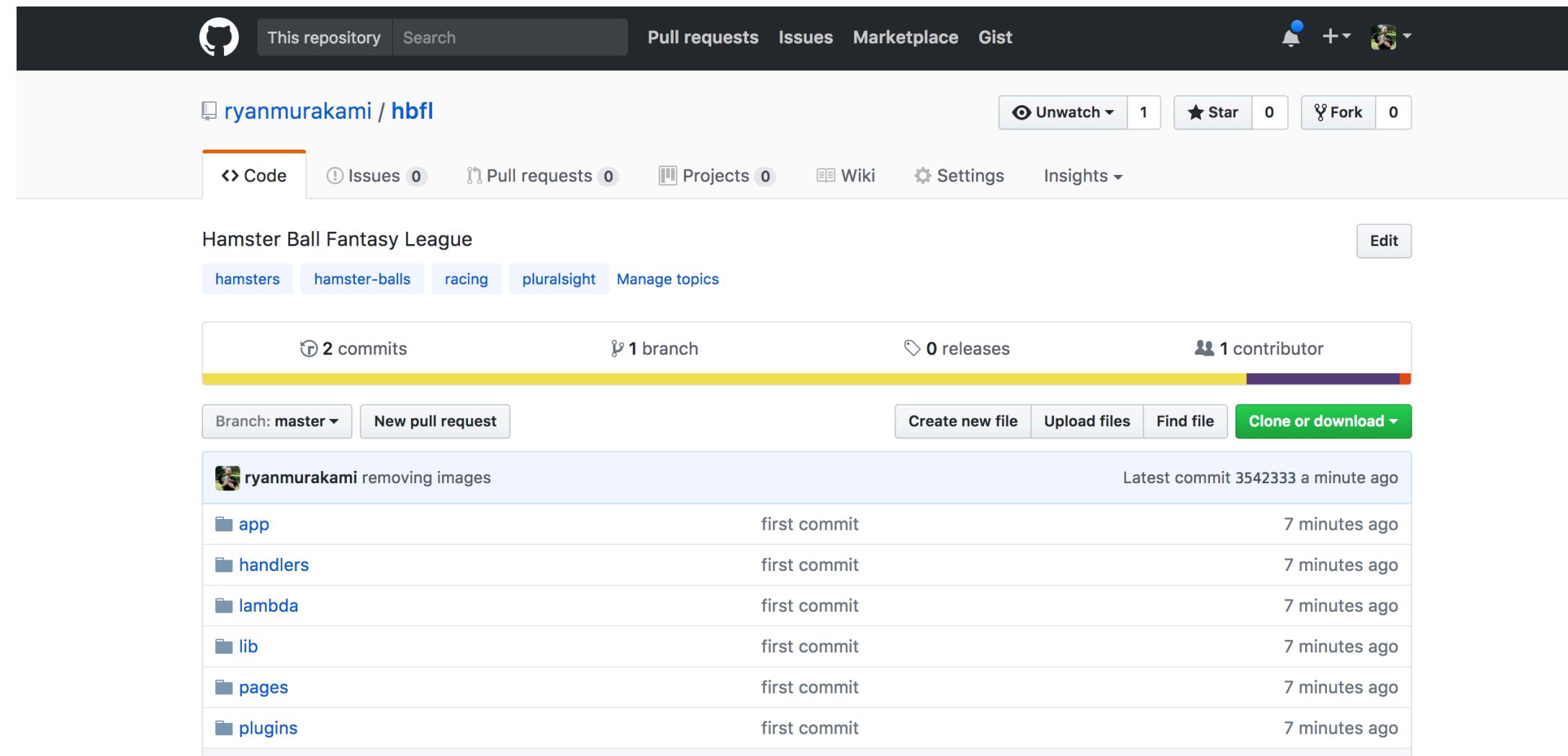
They come packaged together

# Download the Demo Project

The screenshot shows a dark-themed user interface for a video course. At the top, there's an orange button labeled "Resume Course" with a play icon. To its right are four icons: "Bookmark" (a bookmark icon), "Add to Channel" (a channel icon), and "Live mentoring" (a person icon). Below this is a navigation bar with tabs: "Table of contents", "Description", "Transcript", "Exercise files" (which is highlighted in orange), "Discussion", and "Recommended". A horizontal line indicates the active tab. The main content area contains text explaining the purpose of exercise files: "These exercise files are intended to provide you with the assets you need to create a video-based hands-on experience. With the exercise files, you can follow along with the author and re-create the same solution on your computer. We find this to be even more effective than written lab exercises." At the bottom left of this area is an orange button labeled "Download exercise files".

Exercise Files in Pluralsight

# Download the Demo Project



Github Repository: [github.com/ryanmurakami/hbfl](https://github.com/ryanmurakami/hbfl)

Let's dive deeper into the  
world of Amazon Web  
Services