



A GUIDE TO AWS CONSOLE-TO-CODE: RECORDING AND GENERATING INFRASTRUCTURE AS CODE

Mahendran Selvakumar
<https://devopstronaut.com/>



Step1: Record the creation of the EC2 instance

Go to the AWS Console and search for 'EC2'

The screenshot shows the AWS search interface with 'EC2' typed into the search bar. The 'Services' section is expanded, showing the 'EC2' card, which is highlighted with a blue border. Other cards include 'EC2 Image Builder' and 'Recycle Bin'. To the right, there's a sidebar for creating applications and selecting regions.

Click the Console-to-Code icon and then click 'Start Recording'

The screenshot shows the EC2 Dashboard with various resource counts. On the right, a 'Console-to-Code' overlay is open. It contains instructions for generating code, three steps for recording actions, and a 'Start recording' button. The 'Start recording' button is highlighted with a yellow box.

Click "Launch Instance"

The screenshot shows the EC2 Dashboard with resource counts. On the right, the 'Console-to-Code' overlay is active, showing a red dot next to 'Recording in progress'. The 'Recorded actions (0)' section is visible, along with a 'Launch instance' button on the dashboard.



Provide the EC2 instance name

EC2 > ... > Launch an instance

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name

consoletocode Add additional tags

Select an AMI image (Ex: Ubuntu)

▼ Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Recents **Quick Start**

Amazon Linux 	macOS 	Ubuntu 	Windows 	Red Hat 	SUSE Linux Enterprise Server
------------------	-----------	------------	-------------	-------------	----------------------------------

🔍 [Browse more AMIs](#)
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type ami-0a422d70f727fe93e (64-bit (x86)) / ami-00f117fe174f83c56 (64-bit (Arm)) Virtualization: hvm ENA enabled: true Root device type: ebs	Free tier eligible
--	--------------------

Description
Ubuntu Server 22.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Architecture <input style="width: 100%;" type="button" value="64-bit (x86)"/>	AMI ID ami-0a422d70f727fe93e	Username i ubuntu	Verified provider
--	---------------------------------	---	-------------------



Choose the instance type and keypair

▼ **Instance type** [Info](#) | [Get advice](#)

Instance type

t2.micro	Free tier eligible		
Family: t2	1 vCPU	1 GiB Memory	Current generation: true
On-Demand RHEL base pricing: 0.027 USD per Hour			
On-Demand Linux base pricing: 0.0126 USD per Hour			
On-Demand SUSE base pricing: 0.0126 USD per Hour			
On-Demand Windows base pricing: 0.0172 USD per Hour			

Additional costs apply for AMIs with pre-installed software

All generations

[Compare instance types](#)

▼ **Key pair (login)** [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

kubernetes

[Create new key pair](#)

Leave the values as defaults because we are creating a test EC2 instance to validate Console-to-Code

▼ **Network settings** [Info](#) [Edit](#)

Network [Info](#)
vpc-01c128bb0e2b77712

Subnet [Info](#)
No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)
Enable

Additional charges apply when outside of **free tier allowance**

Firewall (security groups) [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

[Create security group](#) [Select existing security group](#)

We'll create a new security group called '**launch-wizard-5**' with the following rules:

Allow SSH traffic from
Helps you connect to your instance

Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server

Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

⚠️ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. X



Leave the root volume details as defaults

▼ **Configure storage** [Info](#) [Advanced](#)

1x GiB [▼](#) Root volume (Not encrypted)

ⓘ Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage [X](#)

[Add new volume](#)

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

ⓘ Click refresh to view backup information [⟳](#)
The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

0 x File systems [Edit](#)

Click “Preview code” to view the code

▼ **Summary**

Number of instances [Info](#)

Software Image (AMI)
Canonical, Ubuntu, 24.04, amd6...[read more](#)
ami-0d64bb532e0502c46

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

ⓘ Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet. [X](#)

[Cancel](#) [Launch instance](#) [Preview code](#)



Here you can see the preview code that was created based on the recording

Console-to-Code

Console-to-Code > Preview code

Preview code
Preview code generates CLI commands for your current in-progress action.

EC2 / Launch Instances
Last updated Thu Oct 10 2024 22:24:30 GMT+0100 (British Summer Time)

CreateSecurityGroup
 aws ec2 create-security-group --group-name "launch-wizard-7" --description "launch-wizard-7 created 2024-10-10T21:24:04.246Z" --vpc-id "vpc-01c128bb0e2b77712"

AuthorizeSecurityGroupIngress
 aws ec2 authorize-security-group-ingress --group-id "sg-preview-1" --ip-permissions '[{"IpProtocol":"tcp","FromPort":22,"ToPort":22,"IpRanges":[{"CidrIp":"0.0.0.0/0"}]}'

RunInstances
 aws ec2 run-instances --image-id "ami-0d64bb532e0502c46" --instance-type "t2.micro" --key-name "kubernetes" --block-device-mappings '[{"DeviceName":"/dev/sda1","Ebs":{"Encrypted":false,"DeleteOnTermination":true,"Iops":3000,"SnapshotId": "snap-0e966f459c12e6315","VolumeSize":8,"VolumeType":"gp3","Throughput":125}}]' --network-interfaces '[{"AssociatePublicIpAddress":true,"DeviceIndex":0,"Groups":["sg-preview-1"]}' --credit-specification '{"CpuCredits":"standard"}' --tag-specifications '[{"ResourceType":"instance","Tags":[{"Key":"Name","Value":"console-to-code"}]}]' --metadata-options '{"HttpEndpoint": "enabled","HttpPutResponseHopLimit": 2,"HttpTokens": "required"}' --private-dns-name-options '{"HostnameType": "ip-name","EnableResourceNameDnsAAAARecord": true,"EnableResourceNameDnsAAAARecord": false}' --count "1"

ⓘ You can generate code in your desired Infrastructure as Code (IaC) formats using Amazon Q.

Stop the recording and click Generate CFN YAML

Console-to-Code

Recording in progress
Only actions in this tab are recorded.

Recorded actions (71)

Code generation by Amazon Q

Type

Type	Operations	Type	Creation time
<input type="checkbox"/>	EC2 / Launch Instan...		
<input type="checkbox"/>	RunInstances	Write	Thu Oct 10 2024 21:50:36 GMT+0100 (British Summer Time)
<input type="checkbox"/>	AuthorizeSearc...	Write	Thu Oct 10 2024 21:50:36 GMT+0100 (British Summer Time)
<input type="checkbox"/>	CreateSecurityG...	Write	Thu Oct 10 2024 21:50:35 GMT+0100 (British Summer Time)



Here you can see the CLI commands

Console-to-Code

Back to the list Reset recording

Review code

CLI commands

CreateSecurityGroup

```
 aws ec2 create-security-group --group-name "launch-wizard-5" --description "launch-wizard-5 created 2024-10-10T20:47:51.418Z" --vpc-id "vpc-01c128bb0e2b77712"
```

AuthorizeSecurityGroupIngress

```
 aws ec2 authorize-security-group-ingress --group-id "sg-036843f4ea19b4798" --ip-permissions '[{"IpProtocol":"tcp","FromPort":22,"ToPort":22,"IpRanges":[{"CidrIp":"0.0.0.0/0"}]} {"IpProtocol":"tcp","FromPort":443,"ToPort":443,"IpRanges":[{"CidrIp":"0.0.0.0/0"}]}]
```

RunInstances

```
 aws ec2 run-instances --image-id "ami-0a422d70f727fe93e" --instance-type "t2.micro" --key-name "kubernetes" --network-interfaces '{"AssociatePublicIpAddress":true,"DeviceIndex":0,"Groups":["sg-036843f4ea19b4798"]}' --credit-specification '{"CpuCredits":"standard"}' --tag-specifications '{"ResourceType":"instance","Tags":[{"Key":"Name","Value":"consoletocode"}]}' --meta-data-options '{"HttpEndpoint":"enabled","HttpPutResponseHopLimit":2,"HttpTokens":"required"}' --private-dns-name-options '{"HostnameType":"ip-name","EnableResourceNameDnsARecord":true,"EnableResourceNameDnsAAAARecord":false}' --count "1"
```

Select the operations needed to generate the CFN YAML

Console-to-Code

Recording stopped Reset Resume

Recorded actions (3/71)

Code generation by Amazon Q

Type

Write

<input checked="" type="checkbox"/>	Operations	Type	Creation time
<input type="checkbox"/>	EC2 / Launch Instan...		
<input checked="" type="checkbox"/>	RunInstances	Write	Thu Oct 10 2024 21:50:36 GMT+0100 (British Summer Time)
<input checked="" type="checkbox"/>	AuthorizeSecurit...	Write	Thu Oct 10 2024 21:50:36 GMT+0100 (British Summer Time)
<input checked="" type="checkbox"/>	CreateSecurityG...	Write	Thu Oct 10 2024 21:50:35 GMT+0100 (British Summer Time)

Here you can see the generated CloudFormation YAML template. Click 'Download' to download the template locally

<https://www.linkedin.com/in/mahendran-selvakumar-36444a77/>



CloudFormation YAML

ⓘ Amazon Q Developer uses generative AI to create code. Use the code as a starting point for your development. See the AWS Responsible AI Policy. [See the AWS Responsible AI Policy.](#)

Resources:

WebServerSecurityGroup:

Type: AWS::EC2::SecurityGroup

Properties:

GroupDescription: launch-wizard-5 created 2024-10-10T20:47:51.418Z

GroupName: launch-wizard-5

SecurityGroupIngress:

- IpProtocol: tcp

- FromPort: 22

- ToPort: 22

- CidrIp: 0.0.0.0/0

- IpProtocol: tcp

- FromPort: 443

- ToPort: 443

- CidrIp: 0.0.0.0/0

VpcId: vpc-01c128bb0e2b77712

WebServerInstance:

Type: AWS::EC2::Instance

Properties:

ImageId: ami-0a422d70f727fe93e

InstanceType: t2.micro

KeyName: kubernetes

NetworkInterfaces:

- AssociatePublicIpAddress: true

- DeviceIndex: 0

- GroupSet:

- !Ref WebServerSecurityGroup

CreditSpecification:

CPUCredits: standard

Tags:

- Key: Name

- Value: consoletocode

MetadataOptions:

HttpEndpoint: enabled

HttpPutResponseHopLimit: 2

HttpTokens: required

PrivateDnsNameOptions:

HostnameType: ip-name

EnableResourceNameDnsARecord: true

EnableResourceNameDnsAAAARecord: false

HttpTokens: required

PrivateDnsNameOptions:

HostnameType: ip-name

EnableResourceNameDnsARecord: true

EnableResourceNameDnsAAAARecord: false

Reasoning: {The provided AWS CLI commands create a security group with ingress rules for ports 22 (SSH) and 443 (HTTPS) from any IP address (0.0.0.0/0), and launch an EC2 instance with the specified AMI, instance type, key pair, security group, and other configurations. The generated CloudFormation YAML code creates the corresponding resources: a security group with the specified ingress rules and an EC2 instance with the specified properties. The reasoning is based on mapping the CLI commands to the appropriate CloudFormation resource types and properties.}

[Provide feedback](#)

Copy

Download



Step2: Create an EC2 instance using the generated CloudFormation YAML template

Go to CloudFormation and click 'Create Stack'

The screenshot shows the AWS CloudFormation Stacks page. At the top, there are navigation links: 'CloudFormation > Stacks'. Below this is a search bar labeled 'Filter by stack name' and a dropdown menu set to 'Active'. To the right are buttons for 'Delete', 'Update', 'Stack actions', and 'Create stack'. A large orange 'Create stack' button is prominently displayed in the center. Below the buttons, there's a message 'No stacks' and 'No stacks to display'. At the bottom of the page is a link 'View getting started guide'.

Choose an existing template, upload the downloaded Console-to-Code template file, and click 'Next'

The screenshot shows the 'Create stack' wizard at Step 1: 'Specify template'. The 'Prerequisite – Prepare template' step is completed. In the 'Prepare template' section, it says 'Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.' There are three options: 'Choose an existing template' (selected), 'Use a sample template', and 'Build from Application Composer'. In the 'Specify template' section, it says 'A template is a JSON or YAML file that describes your stack's resources and properties.' There are three options: 'Amazon S3 URL' (disabled), 'Upload a template file' (selected), and 'Sync from Git'. The 'Upload a template file' section shows a file named 'console-to-code.yaml' selected. At the bottom, there's a 'S3 URL' field with the value 'https://s3.eu-west-1.amazonaws.com/cf-templates-va661h129afg-eu-west-1/2024-10-10T205912.789Zyhp-console-to-code.yaml', a 'View in Application Composer' button, and 'Cancel' and 'Next' buttons.

If you receive the error below, remove the reasoning part in the code and upload it again

The screenshot shows the 'Specify template' page with an error message: 'Invalid template property or properties [Reasoning]'. The error is highlighted in a red box. At the bottom, there are 'Cancel' and 'Next' buttons.



Provide the stack name

CloudFormation > Stacks > Create stack

Step 1
[Create stack](#)

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review and create

Provide a stack name

Stack name
console-code-stack

Stack name must be 1 to 128 characters, start with a letter, and only contain alphanumeric characters. Character count: 18/128.

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

No parameters

There are no parameters defined in your template

[Cancel](#) [Previous](#) [Next](#)

Leave the other configurations as defaults because we are just creating the instance. In real-time, you can provide all the values based on your requirements

CloudFormation > Stacks > Create stack

Step 1
[Create stack](#)

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review and create

Tags - optional

Tags (key-value pairs) are used to apply metadata to AWS resources, which can help in organising, identifying and categorising those resources. You can add up to 50 unique tags for each stack.

No tags associated with the stack.

[Add new tag](#)

You can add 50 more tag(s)

Permissions - optional

Specify an existing AWS Identity and Access Management (IAM) service role that CloudFormation can assume.

IAM role - optional

Choose the IAM role for CloudFormation to use for all operations performed on the stack.

[Remove](#) [Edit](#)

Stack failure options

Behaviour on provisioning failure

Specify the roll-back behaviour for a stack failure. [Learn more](#)

Roll back all stack resources
Roll back the stack to the last known stable state.

Preserve successfully provisioned resources
Preserves the state of successfully provisioned resources, while rolling back failed resources to the last known stable state. Resources without a last known stable state will be deleted upon the next stack operation.

Delete newly created resources during a rollback

Specify whether resources that were created during a failed operation should be deleted regardless of their deletion policy. [Learn more](#)

Use deletion policy
Retains or deletes created resources according to their attached deletion policy.

Delete all newly created resources
Deletes created resources during a rollback regardless of their attached deletion policy.



Click “Next”

Advanced options
You can set additional options for your stack, like notification options and a stack policy. [Learn more](#)

► **Stack policy - optional**
Defines the resources that you want to protect from unintentional updates during a stack update.

► **Rollback configuration - optional**
Specify alarms for CloudFormation to monitor when creating and updating the stack. If the operation breaches an alarm threshold, CloudFormation rolls it back.

► **Notification options - optional**
Specify a new or existing Amazon Simple Notification Service topic where notifications about stack events are sent.

► **Stack creation options - optional**
Specify the timeout and termination protection options for stack creation.

[Cancel](#) [Previous](#) [Next](#)

Click “Submit” to create the stack

Stack creation options

Timeout
-

Termination protection
Deactivated

Quick-create link

Use quick-create links to get stacks up and running quickly from the AWS CloudFormation console with the same basic configuration as this stack. Copy the URL on the link to share. [Learn more](#)

[Open quick-create link](#)

[Create changeset](#)

[Cancel](#) [Previous](#) [Submit](#)

If you encounter the error below, remove the `MetadataOptions` and `PrivateDnsNameOptions` sections from the template

Events (4)					Detect root cause	
<input type="text"/> Search events						
Timestamp	▼	Logical ID	Status	Detailed status	Status reason	
2024-10-10 22:10:39 UTC+0100		console-to-code	ROLLBACK_COMPLETE	-	-	
2024-10-10 22:10:37 UTC+0100		console-to-code	ROLLBACK_IN_PROGRESS	-	Validation failed for following resources: [EC2Instance]. Rollback requested by user.	
2024-10-10 22:10:37 UTC+0100		EC2Instance	CREATE_FAILED	VALIDATION_FAILED	Properties validation failed for resource EC2Instance with message: [#: extraneous key [PrivateDnsNameOptions] is not permitted, #: extraneous key [MetadataOptions] is not permitted]	
2024-10-10 22:10:35 UTC+0100		console-to-code	CREATE_IN_PROGRESS	-	User Initiated	



Create the stack with the modified template, and now you can create the EC2 instance using the template developed by Console-to-Code

console-to-code

Events (5)

Timestamp	Logical ID	Status	Detailed status	Status reason
2024-10-10 22:12:13 UTC+0100	console-to-code	CREATE_COMPLETE	-	-
2024-10-10 22:12:13 UTC+0100	EC2Instance	CREATE_COMPLETE	-	-
2024-10-10 22:12:02 UTC+0100	EC2Instance	CREATE_IN_PROGRESS	-	Resource creation Initiated
2024-10-10 22:12:00 UTC+0100	EC2Instance	CREATE_IN_PROGRESS	-	-
2024-10-10 22:11:58 UTC+0100	console-to-code	CREATE_IN_PROGRESS	-	User Initiated

Verify the resource details in the Resources section

console-to-code

Resources (1)

Logical ID	Physical ID	Type	Status	Module
EC2Instance	i-0f7a0cf2ff2cd1d6f	AWS::EC2::Instance	CREATE_COMPLETE	-

Verify the EC2 instance created by CloudFormation

Instances (2) Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
consoletocode	i-0f7a0cf2ff2cd1d6f	Running	t2.micro	Initializing	View alarms +	eu-west-1a	ec2-52-31-136-51.eu-w...	52.31.136.51
consoletocode	i-0aa133ee577826f0	Running	t2.micro	2/2 checks passed	View alarms +	eu-west-1a	ec2-34-243-47-207.eu...	34.243.47.207



Step 3: Record the termination of the EC2 instance

Select the EC2 instance and click 'Terminate (Delete) Instance'

The screenshot shows the AWS EC2 Instances page. A single instance, named "console-to-code" with the ID i-065bfda99a2d33279, is selected. In the Actions dropdown menu, the "Terminate (delete) instance" option is highlighted with a blue border. The page includes filters for Name and Instance ID, and a search bar at the top.

Now the instance termination has been initiated

The screenshot shows the AWS EC2 Instances page after the instance termination was initiated. The instance "console-to-code" (ID i-065bfda99a2d33279) is now in a "Shutting-down" state, indicated by a circular arrow icon next to its name. The status check column shows a dash (-). The Actions dropdown menu still displays the "Terminate (delete) instance" option.

Stop the recording in Console-to-Code

The screenshot shows the AWS Lambda Console-to-Code tab. A red dot indicates "Recording in progress". Below it, a message says "Only actions in this tab are recorded." To the right are "Reset" and "Stop" buttons. A "Recorded actions (116)" section lists actions: "TerminateInstan..." (Write, Thu Oct 10 2024 22:28:39 GMT+0100), "RunInstances" (Write, Thu Oct 10 2024 22:25:45 GMT+0100), "AuthorizeSectur..." (Write, Thu Oct 10 2024 22:25:44 GMT+0100), and "CreateSecurityG..." (Write, Thu Oct 10 2024 22:25:44 GMT+0100). Buttons for "Generate CFN YAML" and "Copy CLI" are also present.



Select the 'TerminateInstances' operation and click 'Generate CFN YAML'

Recorded actions (116)		
Type	Type	Creation time
<input type="checkbox"/> Operations		
<input type="checkbox"/> EC2 / Instances		
<input type="checkbox"/> TerminateInstances	Write	Thu Oct 10 2024 22:28:39 GMT+0100 (British Summer Time)
<input type="checkbox"/> EC2 / Launch Instances		
<input type="checkbox"/> RunInstances	Write	Thu Oct 10 2024 22:25:45 GMT+0100 (British Summer Time)
<input type="checkbox"/> AuthorizeSecurityGroupIngress	Write	Thu Oct 10 2024 22:25:44 GMT+0100 (British Summer Time)
<input type="checkbox"/> CreateSecurityGroup	Write	Thu Oct 10 2024 22:25:44 GMT+0100 (British Summer Time)

You can view the CLI command and CloudFormation code for EC2 instance termination

[Back to the list](#) [Reset recording](#)

Review code

CLI commands

TerminateInstances

```
 aws ec2 terminate-instances --instance-ids "i-065bfda99a2d33279"
```

[Copy](#) [Download](#)

CloudFormation YAML

ⓘ Amazon Q Developer uses generative AI to create code. Use the code as a starting point for your development. [See the AWS Responsible AI Policy.](#)

```
Resources:  
EC2TerminateInstance:  
  Type: "AWS::EC2::Instance"  
  Properties:  
    InstanceId: i-065bfda99a2d33279  
    TerminationPolicy:  
      TerminateInstanceOnFailure: true
```

Reasoning: {The provided AWS CLI command is to terminate an EC2 instance with the ID "i-065bfda99a2d33279". In CloudFormation, we can represent this by creating an "AWS::EC2::Instance" resource with the InstanceId property set to the provided instance ID, and the TerminationPolicy set to terminate the instance on failure. This will ensure that the specified EC2 instance is terminated when the CloudFormation stack is deleted or updated.}

[Provide feedback](#) [Copy](#) [Download](#)



Step4: Record the creation of the RDS

Go to RDS (Relational Database Service) and click 'Create Database'

The screenshot shows the 'Amazon RDS' interface. On the left, there's a sidebar with links for Dashboard, Databases, Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Zero-ETL integrations, Events, Event subscriptions, Recommendations (0), and Certificate update.

The main area is titled 'Resources' and displays usage statistics for the Europe (Ireland) region. It includes sections for DB Instances (0/40), DB Clusters (0/40), Reserved instances (0/40), Snapshots (0), and Event subscriptions (0/20). To the right, there are sections for Parameter groups (0), Option groups (0), Subnet groups (0/50), and Supported platforms (VPC).

At the bottom, there's a 'Create database' section with a large orange 'Create database' button and a 'Restore from S3' button.

Choose the 'Easy Create' option and select 'MySQL'

The screenshot shows the 'Create database' wizard. At the top, it says 'RDS > Create database'. The first step is 'Choose a database creation method', where 'Easy create' is selected (indicated by a blue circle). The 'Standard create' option is also shown with a description: 'You set all of the configuration options, including ones for availability, security, backups, and maintenance.' Below this, there's a 'Configuration' section with a 'Engine type' dropdown set to 'MySQL'. Other options include Aurora (MySQL Compatible), Aurora (PostgreSQL Compatible), MariaDB, PostgreSQL, Oracle, and Microsoft SQL Server.



Choose 'Free Tier,' provide the DB instance identifier name and master username, and select 'Auto-generate password'

Edition

MySQL Community

DB instance size

Production

db.r6g.xlarge

4 vCPUs

32 GiB RAM

500 GiB

1.122 USD/hour

Dev/Test

db.r6g.large

2 vCPUs

16 GiB RAM

100 GiB

0.254 USD/hour

Free tier

db.t4g.micro

2 vCPUs

1 GiB RAM

20 GiB

0.020 USD/hour

DB instance identifier

Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

console-to-code

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

Master username [Info](#)

Type a login ID for the master user of your DB instance.

admin

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management

You can use AWS Secrets Manager or manage your master user credentials.

Managed in AWS Secrets Manager - *most secure*

RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

Self managed

Create your own password or have RDS create a password that you manage.

Auto generate password

Amazon RDS can generate a password for you, or you can specify your own password.

i You can view your credentials after you create your database. Click the 'View credential details' in the database creation banner to view the password.

Click “Create Database”



► Set up EC2 connection - optional

You can also set up a connection to an EC2 instance after creating the database. Go to the database list page or the database details page, choose **Actions**, and then choose **Set up to EC2 connection**.

► View default settings for Easy create

Easy create sets the following configurations to their default values, some of which can be changed later. If you want to change any of these settings now, use **Standard create**.

i You are responsible for ensuring that you have all of the necessary rights for any third-party products or services that you use with AWS services.

Cancel

Create database

The database will be created after some time

Successfully created database **console-to-code**

RDS has generated your database master password during the database creation and it will be displayed in the connection details. The only way to view your master password is to choose **View connection details** during database creation. You can modify your DB instance to create a new password at any time.

You can use settings from console-to-code to simplify configuration of **suggested** database add-ons while we finish creating your DB for you.

RDS > Databases

Consider creating a Blue/Green Deployment to minimize downtime during upgrades

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. [RDS User Guide](#) [Aurora User Guide](#)

DB Identifier	Status	Role	Engine	Region ...	Size	Recommendations	CPU
console-to-code	Available	Instance	MySQL Co...	eu-west-1a	db.t4g.mi...		19

In Console-to-Code, you can see the recorded operations



Console-to-Code

● Recording in progress
Only actions in this tab are recorded.

Reset Stop

Recorded actions (324)

Type Write ▾ Find actions by operation

Operations Type Creation time

- deleteDBInstance Write Thu Oct 10 2024 23:11:45 GMT+0100 (British Summer Time)
- createDBInstance Write Thu Oct 10 2024 22:36:50 GMT+0100 (British Summer Time)
- createDBSubnet... Write Thu Oct 10 2024 22:36:41 GMT+0100 (British Summer Time)
- attachRolePolicy Write Thu Oct 10 2024 22:36:40 GMT+0100 (British Summer Time)
- createRole Write Thu Oct 10 2024 22:36:40 GMT+0100 (British Summer Time)
- TerminateInstan... Write Thu Oct 10 2024 22:28:39 GMT+0100 (British Summer Time)
- RunInstances Write Thu Oct 10 2024 22:25:45 GMT+0100 (British Summer Time)
- AuthorizeSearc... Write Thu Oct 10 2024 22:25:44 GMT+0100 (British Summer Time)
- CreateSecurityG... Write Thu Oct 10 2024 22:25:44 GMT+0100 (British Summer Time)

Generate CFN YAML Copy CLI
Code generation by Amazon Q

You can generate the CFN YAML template

Step 5: Record the termination of the RDS instance

Select the database and click 'Delete'

RDS > Databases

Consider creating a Blue/Green Deployment to minimize downtime during upgrades
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. [RDS User Guide](#) [Aurora User Guide](#)

Databases (1)		Group resources	Modify	Actions ▾	Restore from S3	Create database	
<input type="text"/> Filter by databases							
DB identifier		Status					
console-to-code		Available	<ul style="list-style-type: none">Stop temporarilyRebootDeleteSet up EC2 connectionSet up Lambda connectionCreate read replicaCreate Aurora read replicaCreate Blue/Green Deployment - newPromoteTake snapshotRestore to point in timeMigrate snapshotCreate zero-ETL integrationCreate RDS ProxyCreate ElastiCache cluster - new				



Check the box 'I acknowledge that upon instance deletion' and click 'Delete'

Delete console-to-code instance

X

Permanently delete **console-to-code** DB instance. You can't undo this action.

⚠️ Proceeding with this action will delete the instance with all its content and can affect related resources. [Learn more](#) ↗

Create final snapshot

Determines whether a final DB Snapshot is created before the DB instance is deleted.

Retain automated backups

Determines whether retaining automated backups for 7 days after deletion

I acknowledge that upon instance deletion, automated backups, including system snapshots and point-in-time recovery, will no longer be available.

To avoid accidental deletion provide additional written consent.

To confirm deletion, type *delete me* into the field.

delete me

⚠️ We strongly recommend taking a final snapshot before instance deletion since after your instance is deleted, automated backups will no longer be available.

Cancel

Delete



The RDS instance deletion is also recorded in Console-to-Code. Click 'Stop' to stop the recording

Console-to-Code

● Recording in progress
Only actions in this tab are recorded.

Reset Stop

Recorded actions (324)

Type Write Find actions by operation

Operations Type Creation time

	Type	Creation time
<input type="checkbox"/>	RDS / Databases	
<input type="checkbox"/>	deleteDBInstance	Write Thu Oct 10 2024 23:11:45 GMT+0100 (British Summer Time)
<input type="checkbox"/>	createDBInstance	Write Thu Oct 10 2024 22:36:50 GMT+0100 (British Summer Time)
<input type="checkbox"/>	createDBSubnet...	Write Thu Oct 10 2024 22:36:41 GMT+0100 (British Summer Time)
<input type="checkbox"/>	attachRolePolicy	Write Thu Oct 10 2024 22:36:40 GMT+0100 (British Summer Time)
<input type="checkbox"/>	createRole	Write Thu Oct 10 2024 22:36:40 GMT+0100 (British Summer Time)
<input type="checkbox"/>	EC2 / Instances	
<input type="checkbox"/>	TerminateInstan...	Write Thu Oct 10 2024 22:28:39 GMT+0100 (British Summer Time)
<input type="checkbox"/>	EC2 / Launch Instan...	
<input type="checkbox"/>	RunInstances	Write Thu Oct 10 2024 22:25:45 GMT+0100 (British Summer Time)
<input type="checkbox"/>	AuthorizeSearc...	Write Thu Oct 10 2024 22:25:44 GMT+0100 (British Summer Time)
<input type="checkbox"/>	CreateSecurityG...	Write Thu Oct 10 2024 22:25:44 GMT+0100 (British Summer Time)

Generate CFN YAML Copy CLI

Code generation by Amazon Q

Select the RDS deletion operations and generate the CFN YAML or copy the CLI command

Console-to-Code

■ Recording stopped

Reset Resume

Recorded actions (1/324)

Type Write Find actions by operation

Operations Type Creation time

	Type	Creation time
<input checked="" type="checkbox"/>	RDS / Databases	
<input checked="" type="checkbox"/>	deleteDBInstance	Write Thu Oct 10 2024 23:11:45 GMT+0100 (British Summer Time)
<input type="checkbox"/>	EC2 / Instances	
<input type="checkbox"/>	TerminateInstan...	Write Thu Oct 10 2024 22:28:39 GMT+0100 (British Summer Time)
<input type="checkbox"/>	EC2 / Launch Instan...	
<input type="checkbox"/>	RunInstances	Write Thu Oct 10 2024 22:25:45 GMT+0100 (British Summer Time)
<input type="checkbox"/>	AuthorizeSearc...	Write Thu Oct 10 2024 22:25:44 GMT+0100 (British Summer Time)
<input type="checkbox"/>	CreateSecurityG...	Write Thu Oct 10 2024 22:25:44 GMT+0100 (British Summer Time)

Generate CFN YAML Copy CLI

Code generation by Amazon Q



You can see the CLI command to delete the RDS instance

Console-to-Code

[Back to the list](#) [Reset recording](#)

Review code

CLI commands

deleteDBinstance

```
aws rds delete-db-instance --db-instance-identifier "console-to-code"
```

[Copy](#) [Download](#)

Conclusion:

In this guide, we explored the Console-to-Code feature within AWS, which allows for seamless recording and generation of infrastructure as code. By creating EC2 and RDS instances, we demonstrated how to capture operations and convert them into CloudFormation (CFN) YAML templates. This functionality not only simplifies the process of resource management but also enables users to replicate their infrastructure easily. Console-to-Code empowers you to customize and automate your cloud deployments, ensuring efficient and consistent resource management based on your specific requirements.

Keep Learning, Keep Deploying!!!

Feel free to reach out to me, if you have any other queries or suggestions

Stay connected on Linkedin: <https://www.linkedin.com/in/mahendran-selvakumar-36444a77/>