

AWS LAB 3 | Monarch Nigam

EXERCISE 3.1

Create a New S3 Bucket and Upload a File

- From the AWS S3 dashboard, create a new bucket and use either the dashboard or the AWS CLI (`aws s3 cp mylocalfile.txt s3://mybucketname/`) to upload one or more files, giving public read access to the object.

The screenshot shows the AWS S3 dashboard. At the top, there is a green success message: "Successfully created bucket 'monarchs3bucket'. To upload files and folders, or to configure additional bucket settings, choose View details." Below this, there is an "Account snapshot - updated every 24 hours" section with a link to "View Storage Lens dashboard". The main area shows "General purpose buckets" and "Directory buckets". Under "General purpose buckets", there is a table with one item: "monarchs3bucket" (Name), "US East (Ohio) us-east-2" (AWS Region), and "March 2, 2025, 15:53:01 (UTC-06:00)" (Creation date). There are buttons for "Copy ARN", "Empty", "Delete", and "Create bucket".

The screenshot shows the AWS S3 dashboard. At the top, there is a green success message: "Upload succeeded. For more information, see the Files and folders table." Below this, there is an "Upload: status" section with a note: "After you navigate away from this page, the following information is no longer available." The "Files and folders" tab is selected, showing a table with one item: "MonarchS3file.txt" (Name), "-" (Folder), "text/plain" (Type), "519.0 B" (Size), "Succeeded" (Status), and "-" (Error). There is also a "Configuration" tab.

- Use the AWS CLI to list the contents of your new bucket to confirm that the new file or files are there:

```
[loaner@20-10369 ~ % aws s3 ls s3://monarchs3bucket
2025-03-02 16:02:15      519 MonarchS3file.txt
loaner@20-10369 ~ %
```

EXERCISE 3.2

Enable Versioning and Life Cycle Management for an S3 Bucket

- Select your bucket and edit its properties to enable versioning.

Successfully edited Bucket Versioning
To transition, archive, or delete older object versions, [configure lifecycle rules](#) for this bucket.

monarchs3bucket [Info](#)

[Objects](#) | [Metadata](#) | **Properties** | [Permissions](#) | [Metrics](#) | [Management](#) | [Access Points](#)

Bucket overview

AWS Region US East (Ohio) us-east-2	Amazon Resource Name (ARN) arn:aws:s3:::monarchs3bucket	Creation date March 2, 2025, 15:53:01 (UTC-06:00)
--	--	--

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning
Enabled [Edit](#)

- Upload a file to that bucket, edit the copy on your local computer, and upload the new copy (keeping the filename the same). Make sure you give the new file any access permissions you might need.

SecondFileMonarchS3file.txt — Edited

Name: Monarch University: SMU (Southern Methodist University)
Batch: Fall 2024

Testing AWS S3 Bucket Upload

This is a sample file created for testing AWS S3 bucket uploads and downloads. The file contains basic details about Monarch, including university affiliation and batch details.

Additional Details:

- * Purpose: Testing AWS S3 functionality
- * File Type: Text
- * Created On: MARCH 2nd
- * Test Scenario: Testing S3 bucket file upload

End of File.

The screenshot shows the AWS S3 console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and various icons. Below it, a green success message box displays: "Upload succeeded. For more information, see the Files and folders table." To the right of the message is a close button. Underneath, a section titled "Upload: status" shows a note: "After you navigate away from this page, the following information is no longer available." A "Close" button is located to the right of this note. The main content area is titled "Summary" and includes a table comparing "Succeeded" (1 file, 465.0 B) and "Failed" (0 files, 0 B). Below this, tabs for "Files and folders" and "Configuration" are visible, with "Files and folders" currently selected. The "Files and folders" table lists one item: "SecondFileMonarchS3file.txt" (Type: text/plain, Size: 465.0 B, Status: Succeeded). The table has columns for Name, Folder, Type, Size, Status, and Error.

Name	Folder	Type	Size	Status	Error
SecondFileMonarchS3file.txt	-	text/plain	465.0 B	Succeeded	-

SecondFileMonarchS3file.txt

Name: Monarch University: SMU (Southern Methodist University)
 Batch: Fall 2024

Testing AWS S3 Bucket Upload

This is a sample file created for testing AWS S3 bucket uploads and downloads. The file contains basic details about Monarch, including university affiliation and batch details.

Additional Details:

End of File.

aws | Search [Option+S] | United States (Ohio) | monarch.smuclass.2025 | [Close](#)

Upload succeeded
 For more information, see the [Files and folders](#) table.

Upload: status

After you navigate away from this page, the following information is no longer available.

Summary		Failed
Destination s3://monarchs3bucket	Succeeded <small>1 file, 332.0 B (100.00%)</small>	Failed <small>0 files, 0 B (0%)</small>

[Files and folders](#) [Configuration](#)

Files and folders (1 total, 332.0 B)

Name	Folder	Type	Size	Status	Error
SecondFileMonarchS3file.txt	-	text/plain	332.0 B	Succeeded	-

- With the contents of the bucket displayed in the dashboard, select Show Versions. You should now see two versions of your file.

Properties | **Permissions** | **Versions**

Versions (2)

Version ID	Type	Last modified	Size	Storage class
jaEDeqqZE1.MyiOlohSw3GUi...	txt	March 2, 2025, 16:48:40 (UTC-06:00)	332.0 B	Standard
Kn1zxjx7IC05i02TmLttMMJY...	txt	March 2, 2025, 16:43:43 (UTC-06:00)	465.0 B	Standard

- Add a couple of directories with files to your bucket.

Successfully created folder "s3logs".

monarchs3bucket [Info](#)

Objects | **Metadata** | **Properties** | **Permissions** | **Metrics** | **Management** | **Access Points**

Objects (3)

Name	Type	Last modified	Size	Storage class
MonarchS3file.txt	txt	March 2, 2025, 16:02:15 (UTC-06:00)	519.0 B	Standard
s3logs/	Folder	-	-	-
SecondFileMonarchS3file.txt	txt	March 2, 2025, 16:48:40 (UTC-06:00)	332.0 B	Standard

Upload: status

After you navigate away from this page, the following information is no longer available.

Summary

Destination	Succeeded	Failed
s3://monarchs3bucket/s3logs/	2 files, 255.0 B (100.00%)	0 files, 0 B (0%)

Files and folders (2 total, 255.0 B)

Name	Folder	Type	Size	Status	Error
FourthFileMonarchS3file.txt	-	text/plain	129.0 B	Succeeded	-
ThirdFileMonarchS3file.txt	-	text/plain	126.0 B	Succeeded	-

- On the bucket's Management tab, create a life cycle rule and specify a prefix/tag filter that matches the directory name of one of the directories you uploaded.

Screenshot of the AWS S3 Lifecycle Configuration creation page for the 'monarchs3bucket' bucket.

Object tags
You can limit the scope of this rule to the key/value pairs added below.
[Add tag](#)

Object size
You can limit the scope of this rule to apply to objects based on their size. [Learn more](#)

Specify minimum object size
 Specify maximum object size

Lifecycle rule actions
Choose the actions you want this rule to perform.

Transition current versions of objects between storage classes
This action will move current versions.
 Transition noncurrent versions of objects between storage classes
This action will move noncurrent versions.
 Expire current versions of objects
 Permanently delete noncurrent versions of objects
 Delete expired object delete markers or incomplete multipart uploads
These actions are not supported when filtering by object tags or object size.

Review transition and expiration actions

Current version actions	Noncurrent versions actions
Day 0 No actions defined.	Day 0 No actions defined.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Configure a life cycle rule by adding transitions and configuring the transition timing (in days) and target for each one.

Screenshot of the AWS S3 Lifecycle Configuration page for the 'monarchs3bucket' bucket, showing a success message after rule creation.

Lifecycle configuration
To manage your objects so that they are stored cost effectively throughout their lifecycle, configure their lifecycle. A lifecycle configuration is a set of rules that define actions that Amazon S3 applies to a group of objects. Lifecycle rules run once per day.

Default minimum object size for transitions
All storage classes 128K

Lifecycle rules (1)
Use lifecycle rules to define actions you want Amazon S3 to take during an object's lifetime such as transitioning objects to another storage class, archiving them, or deleting them after a specified period of time. [Learn more](#)

[View details](#) [Edit](#) [Delete](#) [Actions](#) [Create lifecycle rule](#)

Lifecycle rule name	Status	Scope	Current version ac...	Noncurrent versio...	Expired object del...	Incomplete multip...
monarchlifecyclerule	Enabled	Filtered	Transition to Standard-IA, 1	Transition to Standard-IA, 1	-	-

By default, objects less than 128KB will not transition across any storage class. We don't recommend transitioning objects less than 128 KB because the transition costs can outweigh the storage savings. If your use case requires transitioning objects less than 128 KB, specify a minimum object size filter for each applicable lifecycle rule with a transition action.

Transition current versions of objects between storage classes

Choose transitions to move current versions of objects between storage classes based on your use case scenario and performance access requirements. These transitions start from when the objects are created and are consecutively applied. [Learn more](#)

Choose storage class transitions

Standard-IA

Days after object creation

30

Remove

Add transition

Transition noncurrent versions of objects between storage classes

Choose transitions to move noncurrent versions of objects between storage classes based on your use case scenario and performance access requirements. These transitions start from when the objects become noncurrent and are consecutively applied. [Learn more](#)

Choose storage class transitions

Standard-IA

Days after objects become noncurrent

30

Number of newer versions to retain - Optional

Number of versions

Remove

Can be 1 to 100 versions. All other noncurrent versions will be moved.

Add transition

Upload succeeded
For more information, see the Files and folders table.

Upload: status

After you navigate away from this page, the following information is no longer available.

Summary		Failed
Destination	s3://monarchs3bucket/s3logs/	0 files, 0 B (0%)
Succeeded	2 files, 851.0 B (100.00%)	

Files and folders (2 total, 851.0 B)

Name	Folder	Type	Size	Status	Error
MonarchS3file.txt	-	text/plain	519.0 B	Succeeded	-
SecondFileMonarchS3file.txt	-	text/plain	332.0 B	Succeeded	-

EXERCISE 3.3

Generate and Use a Presigned URL

- Use the complete URL of a private object in an S3 bucket belonging to you to generate a presigned URL using a variation of this command:
`aws s3 presign s3://MyBucketName/PrivateObject --expires-in 600`

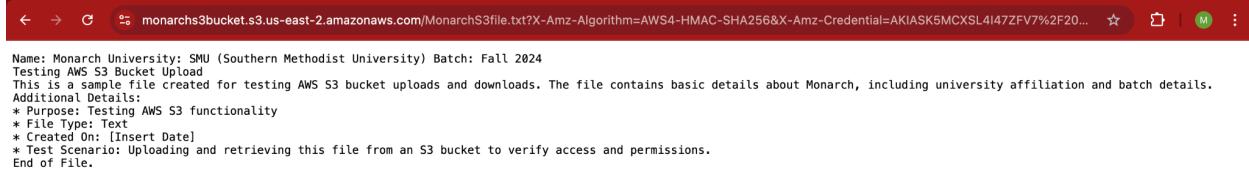
```

[loaner@20-10369 ~ % aws s3 ls s3://monarchs3bucket
2025-03-02 16:02:15      519 MonarchS3file.txt
[loaner@20-10369 ~ % aws s3 ls s3://monarchs3bucket
                           PRE s3logs/
2025-03-02 16:02:15      519 MonarchS3file.txt
2025-03-02 16:48:40      332 SecondFileMonarchS3file.txt
[loaner@20-10369 ~ % aws s3 ls s3://monarchs3bucket/s3logs/
2025-03-02 16:56:41      0
2025-03-02 17:02:54      129 FourthFileMonarchS3file.txt
2025-03-02 17:02:55      126 ThirdFileMonarchS3file.txt
[loaner@20-10369 ~ % aws s3 presign s3://monarchs3bucket/MonarchS3file.txt --expires-in 600
https://monarchs3bucket.s3.us-east-1.amazonaws.com/MonarchS3file.txt?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIASK5MCXSL4I47ZFV7%2F2025030303%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20250303T002048Z&X-Amz-Expires=600&X-Amz-SignedHeaders=host&X-Amz-Signature=3fc8b8d56d3d2f7d5913fa970f2581ab0928e6035be900ee062c4c8075b614c32
[loaner@20-10369 ~ % aws s3 presign s3://monarchs3bucket/MonarchS3file.txt --region us-east-2 --expires-in 600
https://monarchs3bucket.s3.us-east-2.amazonaws.com/MonarchS3file.txt?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIASK5MCXSL4I47ZFV7%2F2025030303%2Fus-east-2%2Fs3%2Faws4_request&X-Amz-Date=20250303T002401Z&X-Amz-Expires=600&X-Amz-SignedHeaders=host&X-Amz-Signature=6162b6781f68f55cbcb23acecb63f806ffb15e23109d306b44a59d3de299cbc
loaner@20-10369 ~ %

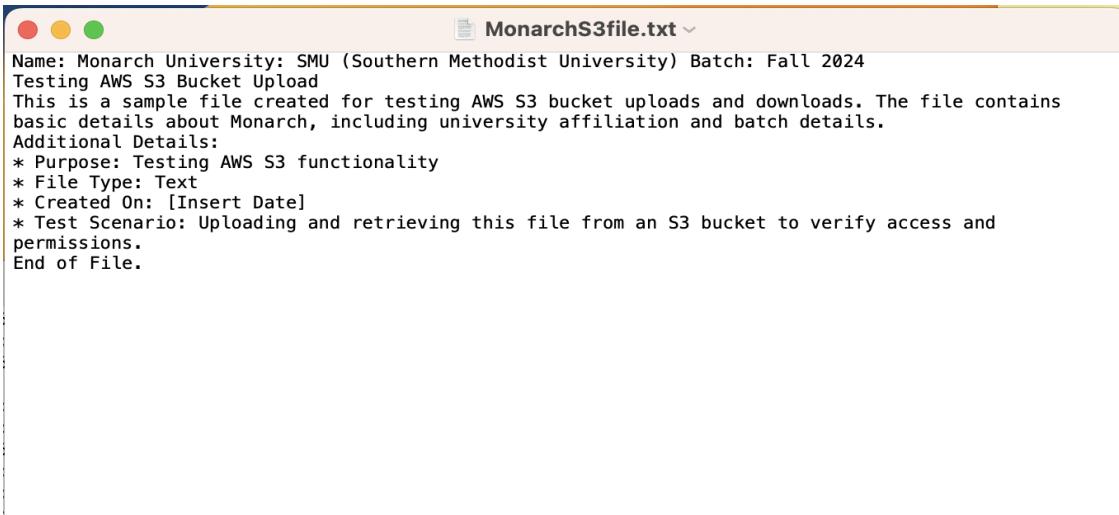
```

- Copy the full URL from the command output, and from a browser that's not logged into your AWS account, try to open the file.

In the below screenshot we can see the above presign url is accessible and the file contents are seen



As my expectation the above contents of the file match with the offline version of the file



- Wait for the URL to expire and try again. This time, it should not work.

After waiting for 10 minutes (600 seconds) we can see its not accessible



EXERCISE 3.4

Enable Static Website Hosting for an S3 Bucket

- From the S3 dashboard, select (or create) an S3 bucket that contains at least one file with some simple text named index.html. Any files you want to be accessible should be readable by the public. You may need to change the Block Public Access settings for the bucket to allow access.

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access

Off

► Individual Block Public Access settings for this bucket

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

```
{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": "*", "Action": "s3:GetObject", "Resource": "arn:aws:s3:::monarchs3bucket/*" } ] }
```

The screenshot shows the AWS S3 console interface. At the top, there are several tabs: "Delete objects - S3 bucket", "Upload objects - S3 bucket", "<Error> <Code>PermanentRe", and "monarchs3bucket.s3.us-east-2". The main URL in the address bar is "us-east-2.console.aws.amazon.com/s3/upload/monarchs3bucket?region=us-east-2&bucketType=general". The top navigation bar includes "Search" and "[Option+S]" buttons, along with account information for "United States (Ohio)" and the bucket name "monarch.smiclass.2025". A green success message box at the top left says "Upload succeeded" and "For more information, see the Files and folders table." Below this, a summary table shows "Destination s3://monarchs3bucket" with "Succeeded" (1 file, 143.0 B (100.00%)) and "Failed" (0 files, 0 B (0%)). The "Files and folders" tab is selected, showing a table with one item: "index.html" (text/html, 143.0 B, Status: Succeeded). A note below the table says "After you navigate away from this page, the following information is no longer available."

- On the bucket's Properties tab, enable static website hosting and specify your index.html file as your index document.

The screenshot shows the "Edit static website hosting" page for the "monarchs3bucket". The top navigation bar includes "Search" and "[Option+S]" buttons, along with account information for "United States (Ohio)" and the bucket name "monarch.smiclass.2025". The page title is "Edit static website hosting" with an "Info" link. The "Static website hosting" section is active, with the "Enable" radio button selected. The "Hosting type" section has the "Host a static website" radio button selected. A note below the hosting type says "For your customers to access content at the website endpoint, you must make all your content publicly readable. To do so, you can edit the S3 Block Public Access settings for the bucket. For more information, see Using Amazon S3 Block Public Access". The "Index document" section specifies "index.html" as the home or default page. The "Error document - optional" section specifies "error.html" as the page returned for errors.

- Paste the static website endpoint into the URL field of a browser that's not logged into your AWS account and confirm that you can access the website.



Welcome to My Static Website on S3!

EXERCISE 3.5

Calculate the Total Life Cycle Costs for Your Data

Estimate the following scenario at the beginning of this section's overall monthly cost with the AWS Simple Monthly Calculator

Even better, use your own company's scenario that will suit. Attempt to give an overall usage scenario, with requests, scans, and data retrieval. Notice that you access the S3 section of the calculator by clicking on the Amazon S3 tab on the left, and you can keep track of your itemized estimate via the Estimate Of Your Monthly Bill tab on top.

I created a small scenario for the Monarch local business where we will be needing following S3 Services -

A screenshot of the AWS Pricing Calculator interface. The URL is 'calculator.aww/#/createCalculator/S3'. The page title is 'aws pricing calculator'. It shows a step-by-step process for creating an estimate for Amazon Simple Storage Service (S3). Step 1: 'Create estimate: Configure Amazon Simple Storage Service (S3)'. It asks for a 'Description' (with placeholder 'Enter a description for your estimate') and to 'Choose a location type' (Region dropdown set to 'US East (Ohio)'). Step 2: 'Configure service'. It asks to 'Select S3 Storage classes and other features'. A grid of options is shown with some selected: S3 Standard, S3 Intelligent - Tiering, S3 Standard - Infrequent Access, S3 One Zone - Infrequent Access, S3 Glacier Flexible Retrieval, S3 Glacier Deep Archive, S3 Management and Insights, S3 Object Lambda, S3 Express One Zone, S3 Data Transfer, S3 Access Grants, and S3 Glacier Instant Retrieval. At the bottom, it says 'S3 Standard feature'.

Then by using aws calculator <https://calculator.aww/#/> I estimated the price and it is shown below-

The screenshot shows the AWS Pricing Calculator interface. In the 'Estimate summary' section, it displays the following costs:

- Upfront cost: 0.00 USD
- Monthly cost: 105.22 USD
- Total 12 months cost: 1,262.64 USD (Includes upfront cost)

In the 'My Estimate' section, there is a table with one row for an 'Amazon Simple Sto...'. The columns include Service Name, Status, Upfront cost, Monthly cost, Description, Region, and Config Summary.

Service Name	Status	Upfront cost	Monthly cost	Description	Region	Config Summary
Amazon Simple Sto...	-	0.00 USD	105.22 USD	Monarch local business	US East (Ohio)	S3 Standard storag...

EXERCISE 5.1

Create an RDS Database Instance

In this exercise, we will create an RDS database instance using MariaDB as the database engine.

- Click the RDS Dashboard link in the RDS service console.
- Click Create Database.
- Under Choose A Database Creation Method, select Standard Create.

The screenshot shows the 'Create database' page in the AWS RDS service console. The 'MariaDB' engine is selected. On the right, there is a detailed description of the MariaDB engine:

MariaDB

MariaDB Community Edition is a MySQL-compatible database with strong support from the open source community, and extra features and performance optimizations.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.
- Supports global transaction ID (GTID) and thread pooling.
- Developed and supported by the MariaDB open source community.

Engine version [Info](#)
View the engine versions that support the following database features.

Hide filters

Show only versions that support the Amazon RDS Optimized Writes [Info](#)
Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.

Engine version
MariaDB 11.4.4

- Under Engine Options, select MariaDB. Keep the default version.
- Under Templates, select Free Tier.
- Enter a master username and master password of your choice.

The screenshot shows the 'Create database' wizard on the AWS RDS console. The current step is 'Settings'. On the left, there's a sidebar with 'DB instance identifier' set to 'database-1'. Under 'Credentials Settings', 'Master username' is 'admin' and 'Self managed' is selected for the password. There are also options for 'Managed in AWS Secrets Manager' and 'Auto generate password'. Below that, 'Master password' is set to '*****' and 'Password strength' is 'Strong'. On the right, a 'MariaDB' section provides a brief overview of the engine, mentioning its MySQL compatibility and various features like support for up to 15 Read Replicas and global transaction ID (GTID).

Instance configuration

- Scroll down to the Additional Configuration section and expand it.
- For Initial Database Name, enter a database name of your choice

The screenshot shows the 'Create database' wizard on the AWS RDS console, with the 'Additional configuration' section expanded. Under 'Database options', 'Initial database name' is 'monarchdb'. Other settings include 'DB parameter group' (set to 'default.mariadb11.4'), 'Option group' (set to 'default:mariadb-11-4'), and 'Backup' (with 'Enable automated backups' checked). Under 'Backup retention period', it shows '1 day'. At the bottom, there's a checkbox for 'Copy tags to snapshots'. On the right, the 'MariaDB' section reiterates its features and support for GTID.

- Click Create Database.

EXERCISE 5.2

Create a Read Replica

- In the navigation pane of the RDS service console, click Databases.
- Select the instance you created earlier, click Actions, and then click Create Read Replica.
- Under the Settings section, in the DB Instance Identifier field, enter a name for your read replica.

- Click the Create Read Replicas button.

In the below snapshot we can see the replica is been created

EXERCISE 5.3

Promote the Read Replica to a Master

In this exercise, we will promote the read replica you just created to master. Before beginning, wait for the replica to enter the Available state.

- In the RDS service console, click Databases.
- Select the read replica you created.
- Click Actions and then select Promote.

- Click Continue.
- Click Promote Read Replica.

After clicking in the promote button we can observe in the below screenshot that the replica database “monarchdbreplica” is no longer under the database-1.

The screenshot shows the AWS RDS console with the URL us-east-2.console.aws.amazon.com/rds/home?region=us-east-2#databases. The left sidebar is titled 'Amazon RDS' and includes links for Dashboard, Databases (selected), Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, and Option groups. The main content area is titled 'Databases (2)' and lists two entries:

DB identifier	Status	Role	Engine	Region ...	Size	Recommendations
database-1	Available	Instance	MariaDB	us-east-2a	db.t4g.micro	View
monarchdbreplica	Reboot...	Instance	MariaDB	us-east-2b	db.t4g.micro	View

A blue banner at the top right provides a tip: 'Consider creating a Blue/Green Deployment to minimize downtime during upgrades'. It explains that using Amazon RDS Blue/Green Deployments can minimize downtime during upgrades. A link to the 'RDS User Guide' and 'Aurora User Guide' is provided.

EXERCISE 5.4

Create a Table in DynamoDB Using Provisioned Mode

1. Using the following command to create a table named Authors with a partition key named LastName and sort key named FirstName. Both keys should use the string data type. Provision the table with a WCU and an RCU of 1.

```
aws dynamodb create-table --table-name Authors --attribute-definitions
AttributeType=S
AttributeName=LastName,AttributeName=FirstName
--keyschema AttributeName=LastName,KeyType=HASH
AttributeName=FirstName,KeyType=RANGE
--provisioned-throughput ReadCapacityUnits=1,WriteCapacityUnits=1
```

In the below snapshot we can see that the above command has been executed in AWS CLI

```

loaner@20-10369 ~ % aws dynamodb create-table --table-name Authors --attribute-definitions AttributeName=LastName,Att..
--attribute-definitions \
    AttributeName=LastName,AttributeType=S \
    AttributeName=FirstName,AttributeType=S \
--key-schema \
    AttributeName=LastName,KeyType=HASH \
    AttributeName=FirstName,KeyType=RANGE \
--provisioned-throughput ReadCapacityUnits=1,WriteCapacityUnits=1

{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "FirstName",
                "AttributeType": "S"
            },
            {
                "AttributeName": "LastName",
                "AttributeType": "S"
            }
        ],
        "TableName": "Authors",
        "KeySchema": [
            {
                "AttributeName": "LastName",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "FirstName",
                ...
skipping...
{
        "TableDescription": {
            "AttributeDefinitions": [
                {
                    "AttributeName": "FirstName",
                    "AttributeType": "S"
                },
                {
                    "AttributeName": "LastName",
                    "AttributeType": "S"
                }
            ],
            "TableName": "Authors",
            "KeySchema": [
                {
                    "AttributeName": "LastName",
                    "KeyType": "HASH"
                },
                {
                    "AttributeName": "FirstName",
                    "KeyType": "RANGE"
                }
            ]
        }
    }
}

```

2. Go to the DynamoDB service console to view the table you just created.
- As we can see in the below screenshot the Authors table has been created

	Name	Status	Partition key	Sort key	Indexes	Replication Regions	Deletion protection	Favorite	Read capacity mode	Write capacity mode
<input type="checkbox"/>	Authors	Active	LastName (\$)	FirstName (\$)	0	0	Off		On-demand	Off