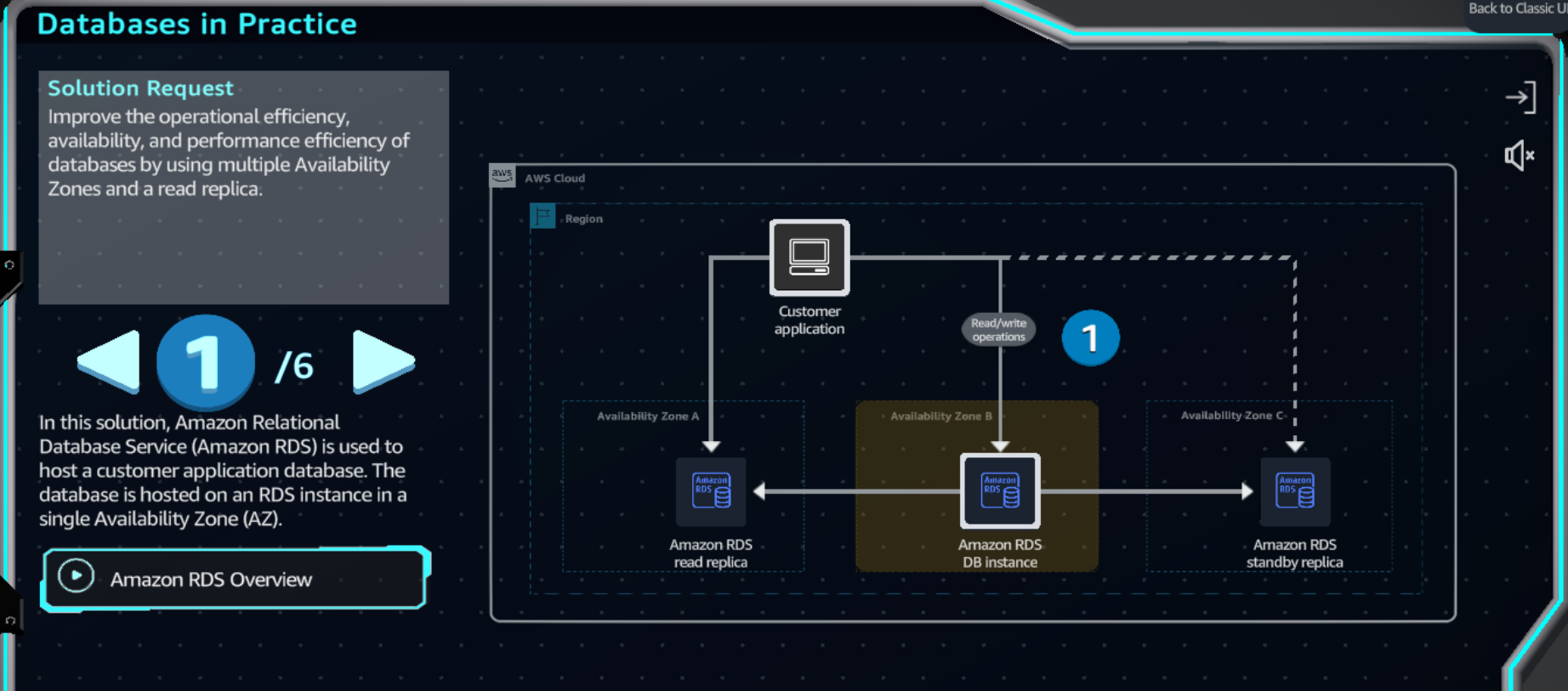
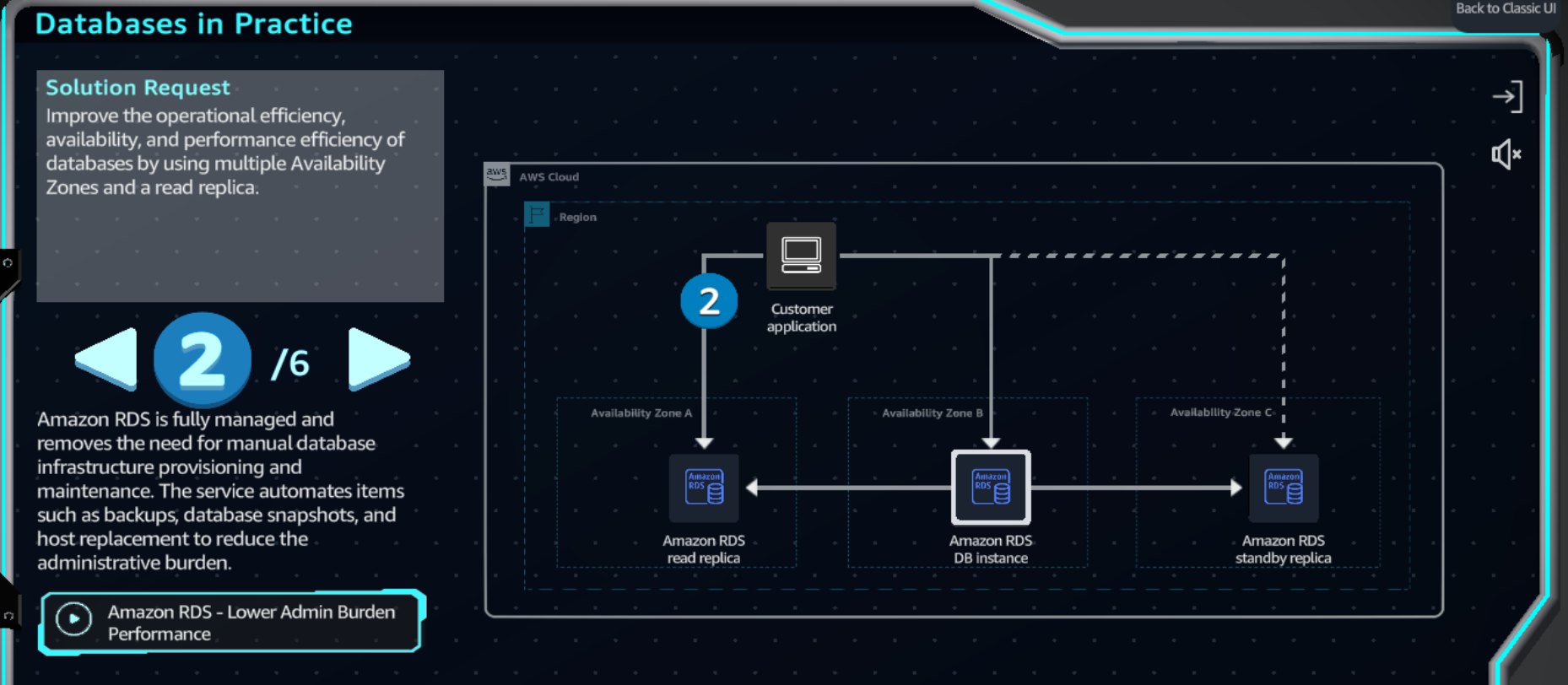
AWS Cloud Quest  
**Task 5. Database in Practice**  
Improve the operational efficiency, availability, and performance efficiency of databases by using multiple AZs and a read replica.

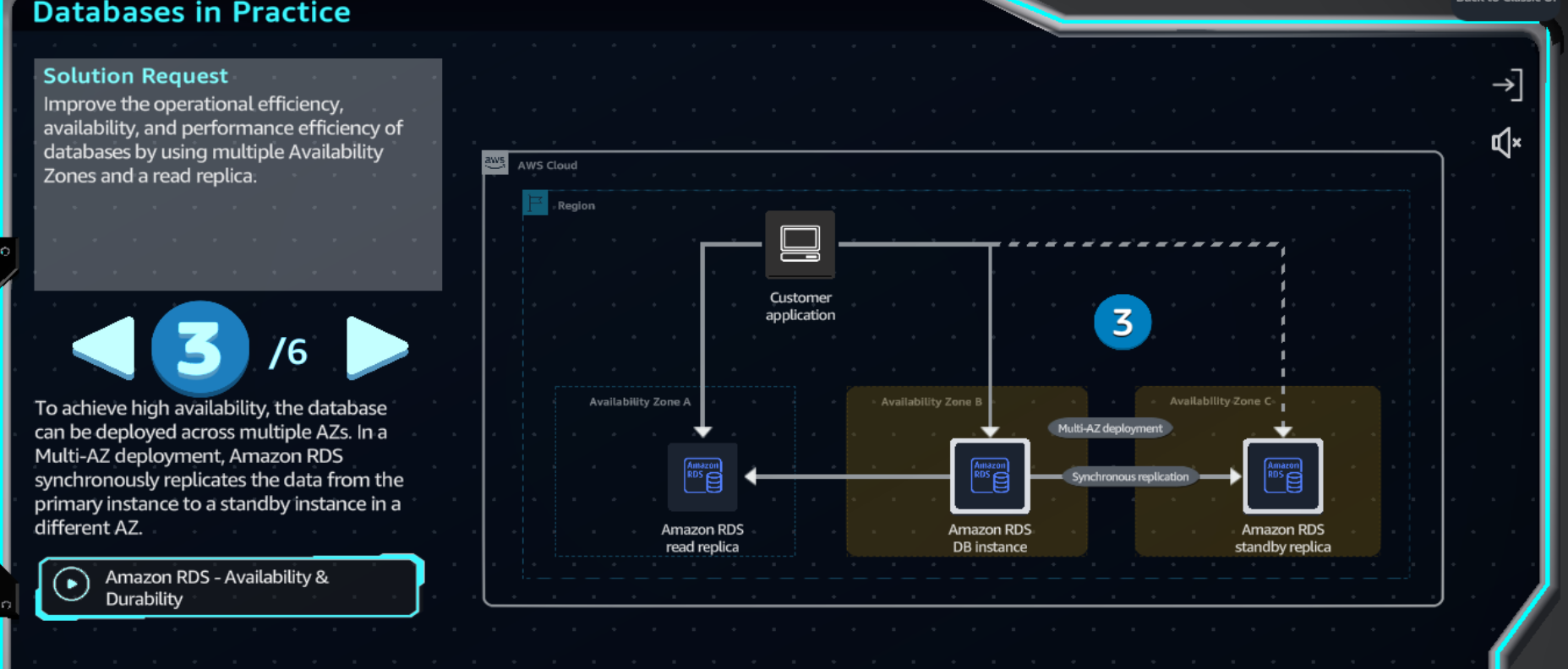
|  |  |
| --- | --- |
| Person | Hi, and welcome to the headquarters of our insurance company. Thanks for answering our call. |
| Me | No, problem. How can I help you? |
| Person | Our database administrators have recently complained that they spend too much time with operational tasks, such as patching and managing database infrastructure, instead of innovating for our customers. We heard that AWS might have a solution. |
| Me | That’s common pain point. AWS offers many database services to fit your needs, along with self-hosting options.  From description, it sounds like Amazon RDS, Amazon’s managed relational database service is what your administrators need. Amazon Relational Database Service, commonly known as Amazon RDS, removes the operational burden of patching and managing underlying infrastructure. |
| Person | That sounds like what we want. I can have our database admis experiment with Amazon RDS in our dev test environments. Can Amazon RDS also protect us from data loss. |
| Me | Yes! Amazon RDS can perform routine backups on your database. You can choose how long you’d like to store those backups, which is known as the retention period. |
| Person | Okay, but how else can we make sure that our systems are resilient in case of a disaster? |
| Me | Using Amazon RDS you can deploy your database instances in multiple AZs. Should a disaster impact one AZ, an automatic failover to the other AZ will occur. Your data is replicated synchronously, so your customers wont see any downtime. |
| Person | This is exactly what we want. Will deploying our database in multiple AZs mean that we have twice the database performance? |
| Me | Well, not quite. Using multiple AZs is intended for high availability and disaster recovery, not increased performance. |
| Person | Okay, so what can I do to improve performance? Our data analytics team is constantly running real-time queries and big data analytics. Data performance is impacted with each large read load. |
| Me | If you’re doing real-intensive workloads, you can deploy something called read replica. Read replicas have the same data as your primary database, but they allow only read operations. You can direct queries to read replica so that your primary database is free to do more write operations. |
| Person | This all sounds great! If we do decide to move our production database to Amazon RDS, will our administration have a difficult time migrating? |
| Me | Not at all. AWS Database Migration Service, or AWS DMS, provides secure and user-friendly database migrations while keeping source databases fully operational during the migration process. |
| Person | Nice! Cant wait to see how well Amazon RDS performs.  Can you help improve our database infrastructure with Amazon RDS? |
| Me | Accepted! |
| Person | Awesome! Let’s get to it. |

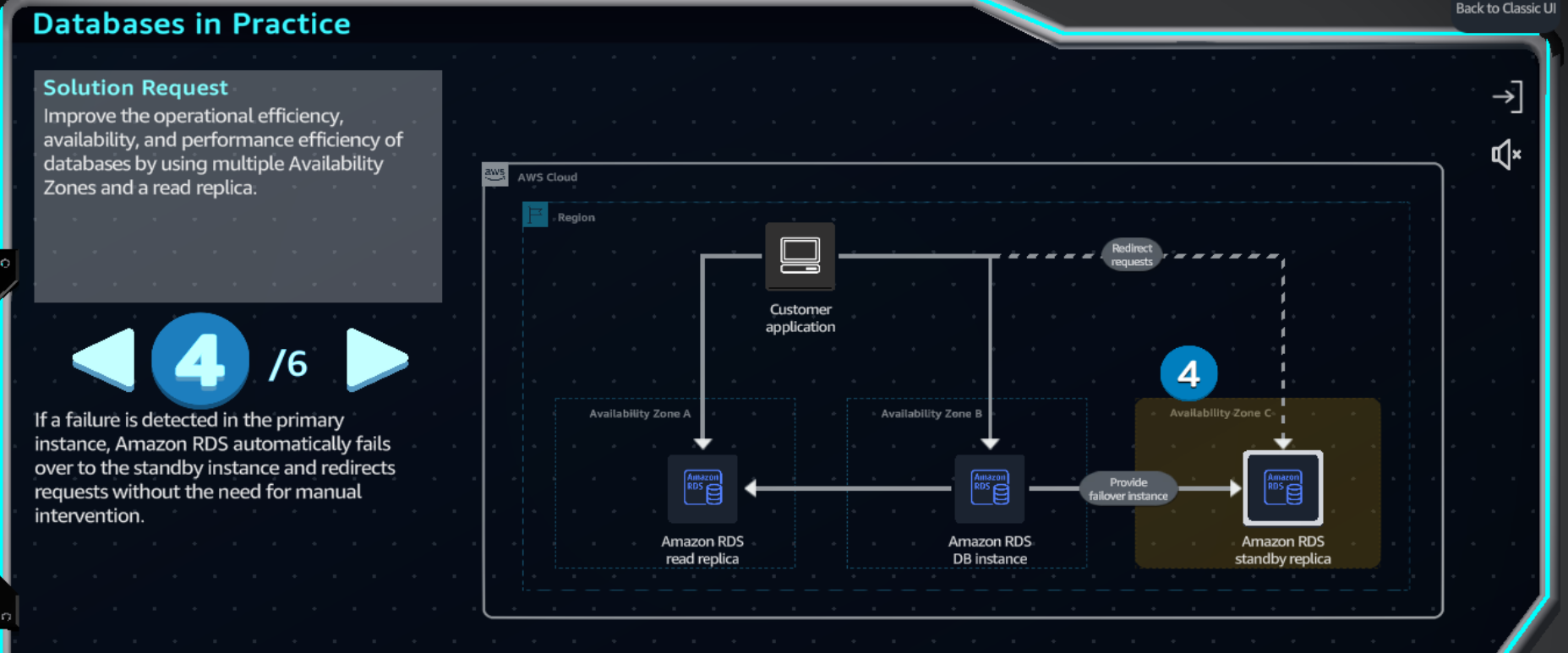
**Learning Objectives**

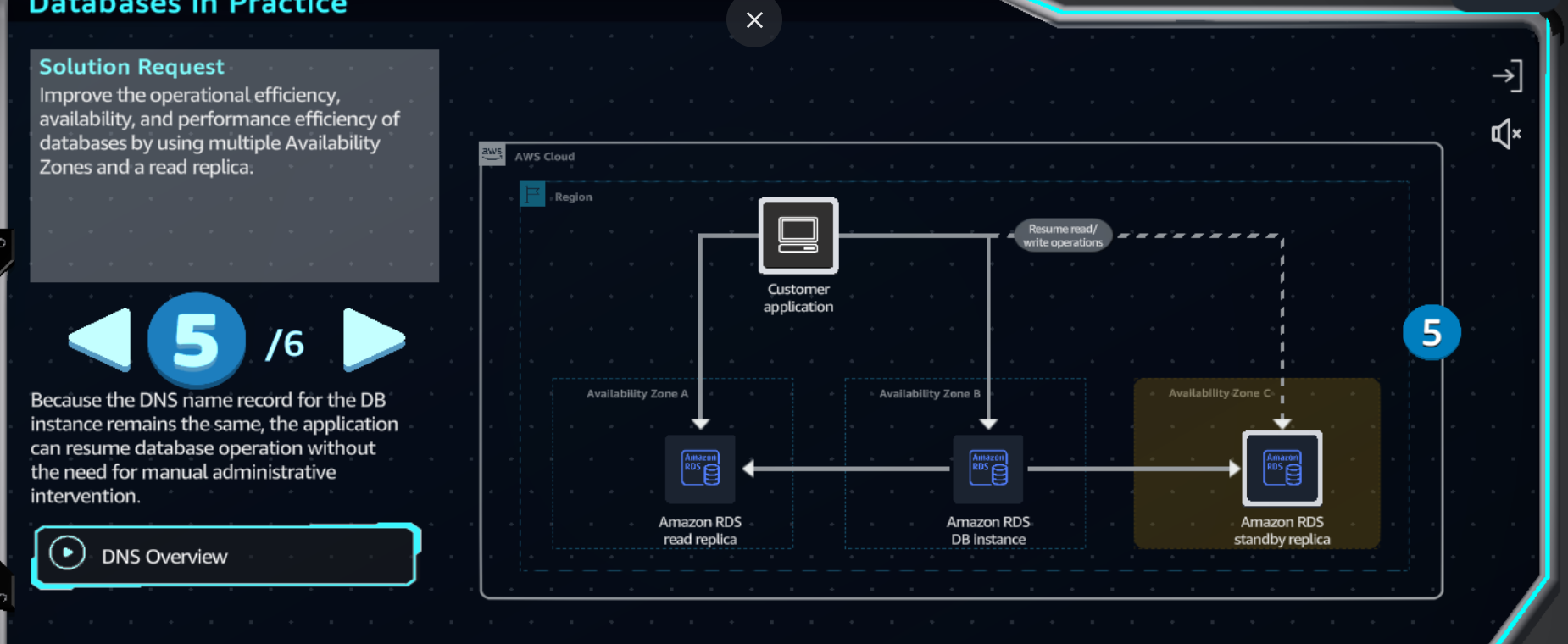
* Create an Amazon RDS DB instance.
* Enable backups to your database.
* Enable multiple AZs for your Amazon RDS deployment.
* Create an Amazon RDS read replica.

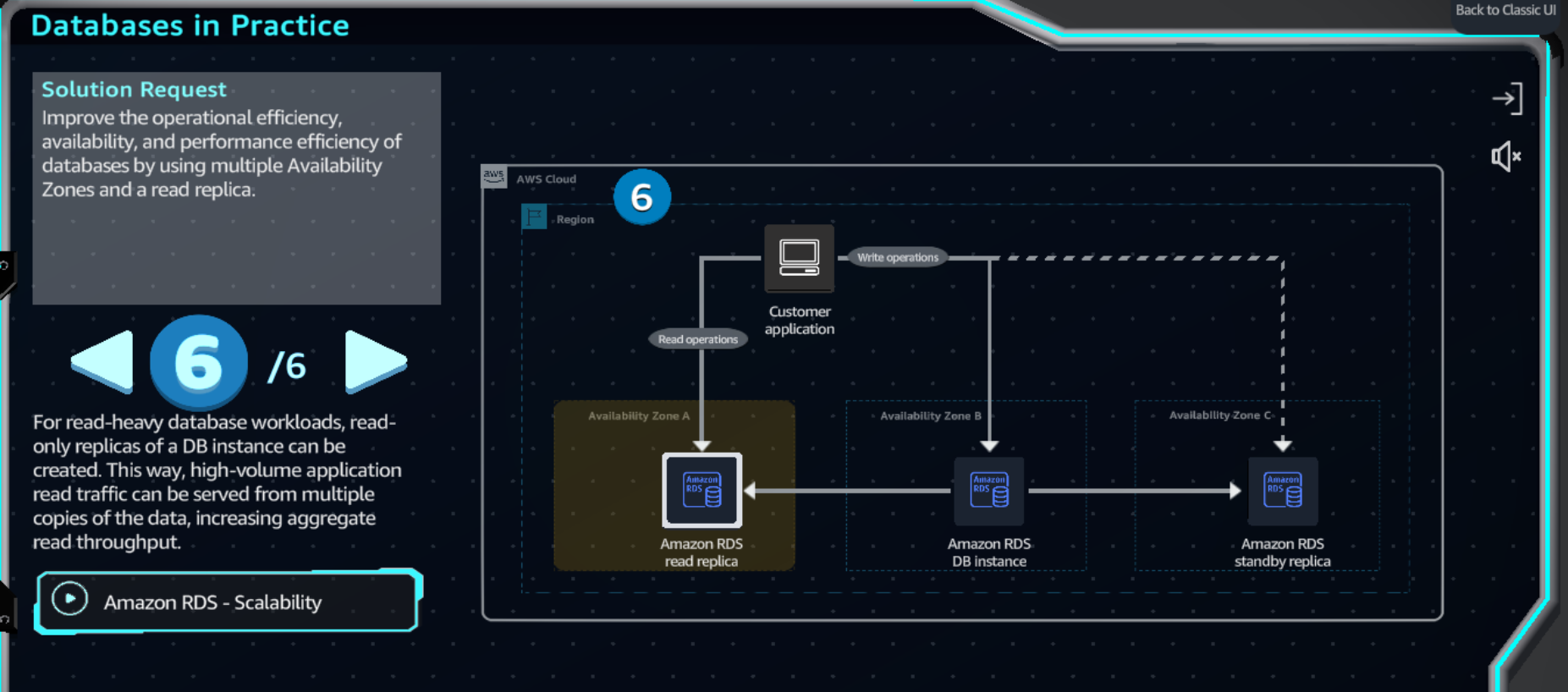












**Step 2:**

1. In the top navigation bar search box, type: ec2
2. In the search results, under Services, click EC2.
3. Go to the next step.

**Amazon Elastic Compute Cloud** (EC2) is a web service that provides secure and scalable computing capacity in the AWS Cloud.

**Step 3:**

1. In the left navigation pane, click AMI Catalog.
2. Go to the next step.

An **Amazon Machine Image** (AMI) is an image that provides the software that is required to set up and boot an EC2 instance. AMIs might also include software packages such as database servers, offering a hosted database option in addition to the managed options provided by other AWS services.

**Step 4:**

1. In the AMIs search box, type: sql and press Enter.
2. On the Quick Start AMIs tab, review the available AMIs on Amazon EC2.
   1. Depending on your requirements, you can find AMIs provided directly by AWS, through trusted third-parties in the AWS Marketplace, or through the AWS community.
3. Go to the next step

When hosting a database on an EC2 instance, AWS handles the physical infrastructure, hardware, and operating system installation, but you remain responsible for managing the instance, database management, query optimization, and customer data. Managed solutions such as Amazon Relational Database Service (Amazon RDS) aim to remove these tasks, the trade-off being customization options.

**Step 5:**

1. In the top navigation bar search box, type: database
2. In the search results, click Show more.
3. Go to the next step.

**Step 6:**

1. Review the list of available database options.
2. Click Aurora and RDS.
3. Go to the next step.

AWS provides a comprehensive portfolio of database services for different use cases. **Relational databases** such as **Amazon RDS** provide structured, SQL-based data management that is ideal for applications that require complex queries and transactions. **Non-relational databases**, such as **Amazon DynamoDB**, offer flexible, schema-less storage suitable for rapidly changing data and large-scale applications. **Memory databases**, such as **Amazon MemoryDB**, deliver ultra-fast data access by storing data in-memory, optimizing performance for real- time applications."

**Step 7:**

1. In the left navigation pane, click Databases.
2. In the Databases section, click Create database.
3. Go to the next step.

**Amazon RDS** is a web service that helps you set up, operate, and scale relational databases in the cloud. It provides cost- efficient and resizable capacity while automating time-consuming administration tasks such as hardware provisioning, database setup, patching, and backups.

**Step 8:**

1. For Choose a database creation method, choose Standard create.
2. For Engine type, choose MariaDB.
3. In the right side panel, review the MariaDB description.
4. Go to the next step.

Amazon RDS offers several different open source and commercial database (DB) engines.

**Step 9:**

1. For Engine version, keep the provided default MariaDB version.
   1. The default version in your practice lab might be different from what is displayed in the screenshot example.
2. For Templates, choose Dev/Test.
3. For DB instance identifier, type: my-database
4. Go to the next step.

Engine versions come in two types: **major versions** and **minor versions**. **Major versions** are supported for at least 3 years after their initial offering by Amazon RDS. **Minor versions** are typically supported for at least 1 year after initial offering. These versions are periodically deprecated when they reach community end of life or when they no longer receive software fixes or security updates.

**Step 10:**

1. Under Credential settings, for Master username, keep the default choice of admin.
2. For Credentials management, choose Self-managed.
3. For Master password, type: ILoveLearning!123
4. For Confirm master password, type the password again.
5. Go to the next step.

For production workloads, we recommend using **AWS Secrets Manager** for **credential management**. With Secrets Manager, you can store and manage various types of secrets, including database credentials, passwords, third-party API keys, and arbitrary text.

**Step 11:**

1. In the Instance configuration section, for DB instance class, choose Burstable classes.
2. Below that, on the dropdown menu list, choose db.t3.xlarge.
   1. Only t3 db classes are supported in this practice lab.
3. For Storage type, choose General Purpose SSD (gp3).
4. For Allocated storage, type: 20
5. Go to the next step.

Amazon RDS provides a selection of instance types optimized to fit different relational database use cases. Instance types comprise varying combinations of CPU, memory, storage, and networking capacity and give you the flexibility to choose the appropriate mix of resources for your database.

**Step 12:**

1. Click to expand Additional storage configuration.
2. Review to confirm that the default option, Enable storage autoscaling, is selected.
3. For Maximum storage threshold, review to confirm that the default threshold, 1000 GiB, is selected.
4. For multi-AZ deployment, choose Create a standby instance.
5. Go to the next step.

A standby instance is a synchronously maintained copy of the primary RDS DB instance that is automatically provisioned and maintained by Amazon RDS in a different Availability Zone (AZ). This replica serves as high availability and failover support for the primary RDS instance.

**Step 13:**

1. For Virtual private cloud (VPC), keep the default choice of Default VPC.
2. For DB subnet group, keep the default setting.
3. For Public access, keep the default choice of No.
4. For VPC security group (firewall), keep the default choice of Choose existing.
5. Go to the next step.

Amazon RDS helps you control network access to your database. You can also run your RDS DB instances in a virtual private cloud (VPC). This way, you can isolate your database instances and connect to your existing IT infrastructure through an industry standard encrypted IPsec VPN.

**Step 14:**

1. In the Monitoring section, clear the checkbox to deselect Enable Performance Insights.
2. Click to expand Additional configuration.
3. Clear the checkbox to deselect Enable Enhanced Monitoring.
   1. If either Performance Insights or Enhanced Monitoring are enabled, you'll get a permissions error when you try to create the database.
4. Click to expand the Additional configuration section.
5. Go to the next step.

**Step 15:**

1. For Initial database name, type: my\_database
2. For DB parameter group and Option group, review the default options.
3. Under Backup, review the default options.
4. Go to the next step.

For AWS to successfully provision an RDS DB instance for you, you must first specify an initial database name. If you fail to specify an initial database, your instance can still be provisioned, but it might not work properly.

**Step 16:**

1. Under Encryption, review the default encryption options.
2. Go to the next step.

Amazon RDS provides encryption at rest capabilities to protect data stored in database instances. When encryption is enabled, the data stored at rest, disk I/O, and snapshots are all encrypted using AES- 256-bit encryption.

**Step 17:**

1. For Maintenance, clear the checkbox to deselect Enable auto minor version upgrade.
2. For Maintenance window, review the default choice of No preference.
3. Scroll down to the bottom of the page, and then click Create database (not shown).
4. Go to the next step.

When AWS determines a new engine minor version contains significant bug fixes, they schedule automatic upgrades for instances that have auto minor version upgrade enabled. AWS also announces upgrades on the Amazon RDS Forum with customer email notifications at least 30 days in advance.

**Step 18:**

* 1. If any pop-up boxes appear (offering add-ons), close them.
  2. The RDS DB instance takes 5-10 minutes to be created.
  3. After the database is created, the status shows Modifying.

1. Wait 5-10 additional minutes after the database is created, and then click the Databases refresh icon.
2. Under Status, review to confirm that the status is Available.
   1. If the status hasn't changed to Available, continue to click the refresh icon every few minutes until it does.
3. Click my-database.
4. Go to the next step.

**Step 19:**

1. In the Summary section, under DB identifier, review the identifier.
2. Click Actions to expand the dropdown list.
3. Review the different options.
   1. You must return to these options, such as Create read replica, in the upcoming DIY section of this solution.
4. Go to the next step.

An Amazon RDS read replica is a read-only copy of a source database instance that uses asynchronous replication to stream database changes. You can create one or more replicas of a given source DB instance and serve high-volume application read traffic to your replicas, thereby increasing aggregate read throughput.

**Step 20:**

1. In the top navigation bar search box, type: dms
2. In the search results, under Services, click Database Migration Service.
3. Go to the next step.

AWS Database Migration Service (AWS DMS) is a web service that helps you migrate data between different data stores. The service provides secure and user-friendly database migrations, keeping source databases fully operational during the migration process to minimize application downtime.

**Step 21:**

1. In the left navigation pane, click Recommendations.
2. Review the How it works section.
3. In the left navigation pane, scroll down and click Getting started (not shown).
4. Go to the next step.

AWS DMS can handle migrations between data stores by using the same or different database engines. The service supports a wide range of database types, including relational databases, data warehouses, and NoSQL databases.

**Step 22:**

1. For Choose a starting point ..., choose Discover and assess.
2. Below that, review the description.
3. Repeat for the Convert and Migrate options.
4. Go to the next step.

