**Introduction**

Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud. Before launching actual RDS instances, you need to configure a DB Subnet Group.

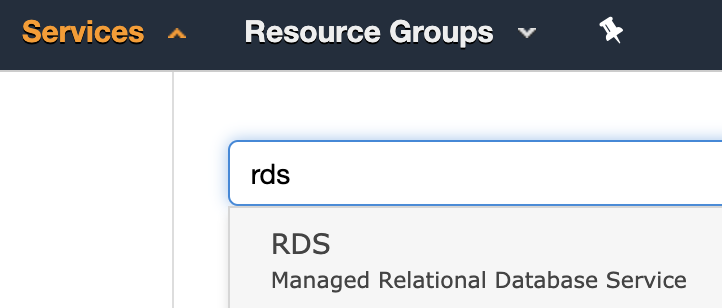
Subnets are segments of a VPC's IP address range that allow you to group your resources based on security and operational needs. A DB Subnet Group is a collection of subnets (typically private) that you create in a VPC and designate for your DB instances. Each DB subnet group should have subnets in at least two Availability Zones in a given region. Note that SQL Server Mirroring with a SQL Server DB instance requires at least 3 subnets in distinct Availability Zones.

When creating a DB instance in a VPC, you must select a DB subnet group. Amazon RDS uses that DB subnet group and your preferred Availability Zone to select a subnet and an IP address within that subnet to associate with your DB instance. When Amazon RDS creates a DB instance in a VPC, it assigns a network interface to your DB instance by using an IP address selected from your DB Subnet Group. If the primary DB instance of a Multi-AZ deployment fails, Amazon RDS can promote the corresponding standby and subsequently create a new standby using an IP address from an assigned subnet in one of the other Availability Zones.

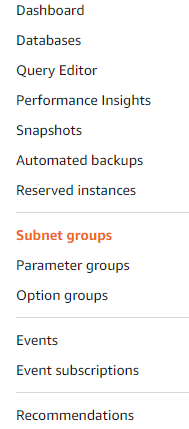
You can create an RDS Subnet Group using the RDS launch wizard.

**Instructions**

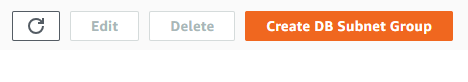
1. Click **Services** at the top of the AWS Management Console, type *rds* into the search bar and click the **RDS** result that appears as a result:



2. From the RDS dashboard, click **Subnet Groups** from the left-hand menu:

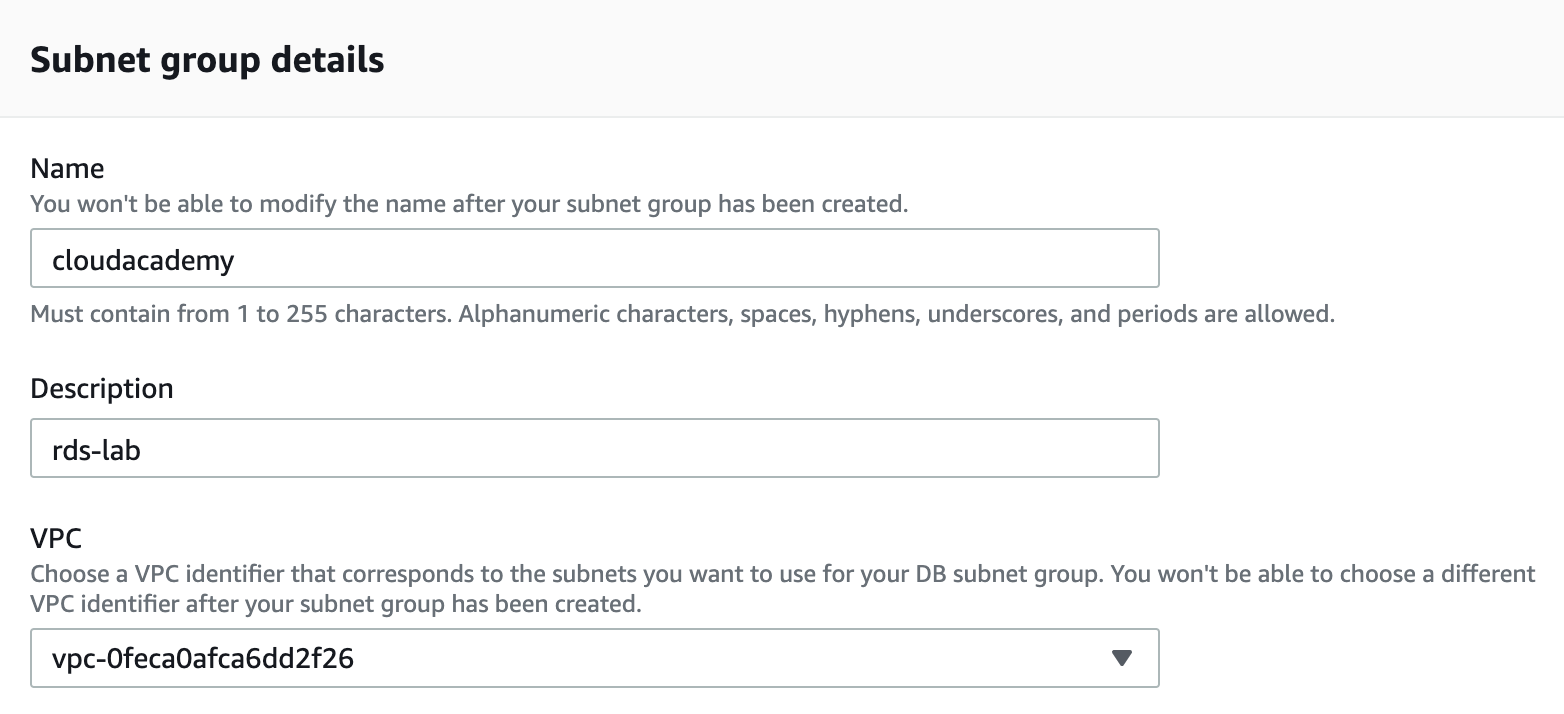


3. Click **Create DB Subnet Group**to open the creation wizard:

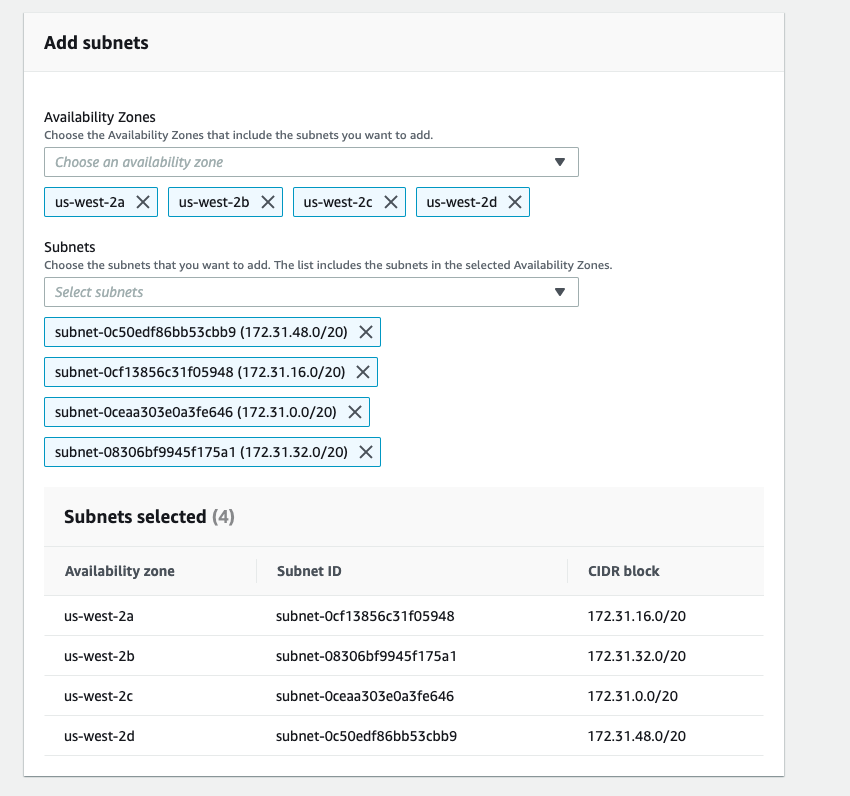


4. Fill out the form using the following data:

* **Name:***cloudacademy*
* **Description:***rds-lab*
* **VPC ID:**select the available one



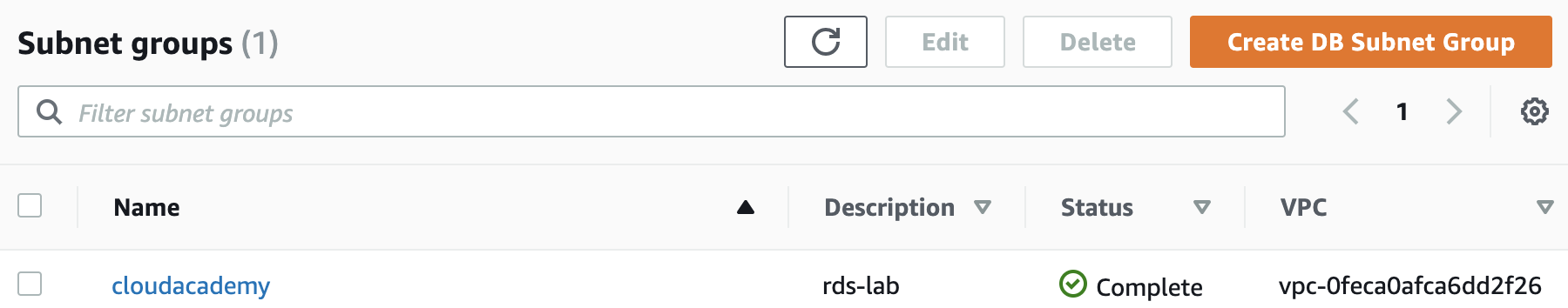
5. Select all the subnets available from the dropdown menu and then click **Create**:



*Note*: It could take a few seconds for the AZs to appear. If not, just manually select the correct VPC.

*Warning*: Be sure to be in the right region and be sure to have selected the correct VPC.

After a few seconds, your DB Subnet Group will be available and ready for use:



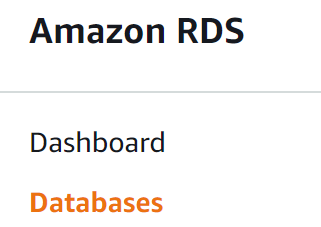
**Summary**

In this Lab Step, you used the AWS Management Console to create a DB Subnet Group.

# Creating a Database Using RDS

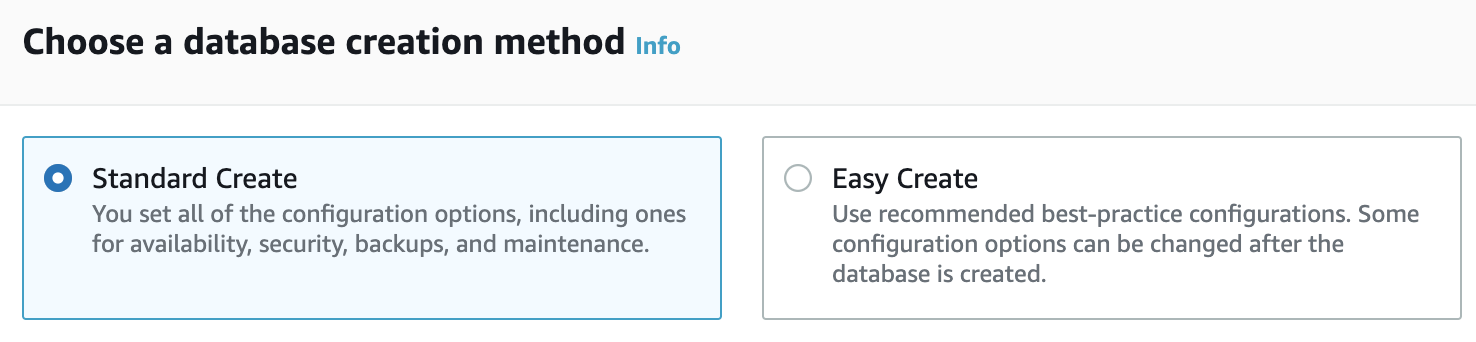
**Instructions**

1. Click **Databases** on the left menu followed by **Create database**:



The **Create database**wizard appears. You will only be instructed to set values that need to be changed from the default values. Leave any unspecified values at their defaults. Click on **info** to read more about any setting in the wizard.

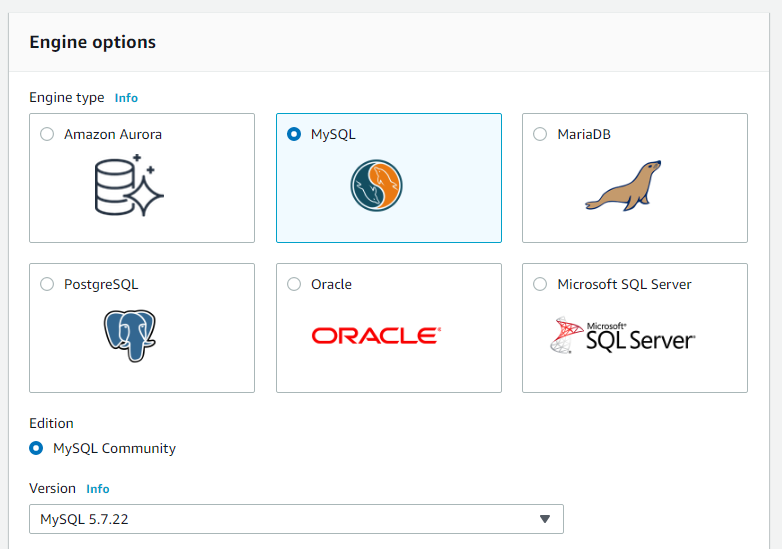
2. Under **Choose a database creation method**ensure that **Standard Create** is selected:



You will configure all of the database settings up-front using this method.

*Note*: If you don't see this option, you may be using the old database creation flow. Click **Switch to the new database creation flow** to change the flow  alt

3. Choose the **MySQL** database engine and leave the version as the default selected**:**



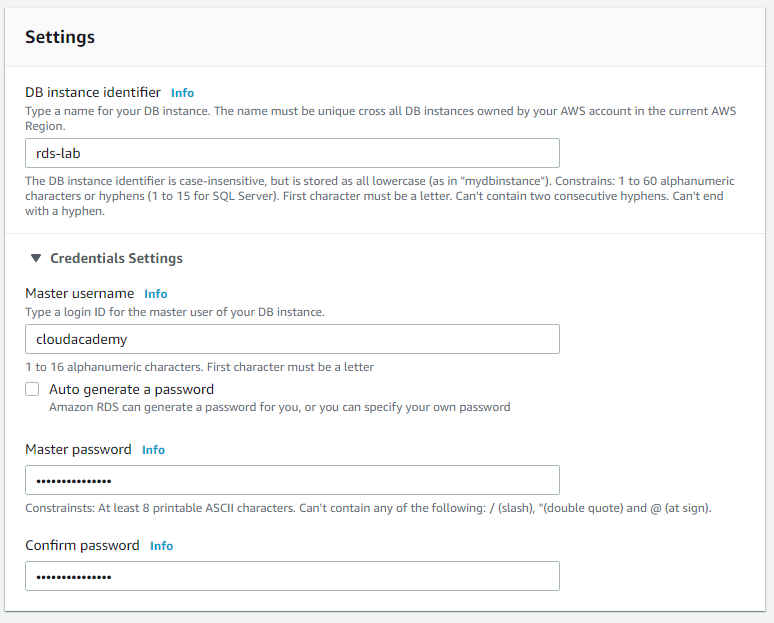
4. Select **Dev/Test** as the creation template:



You don't require the high performance gurarantees of a production database for this lab.

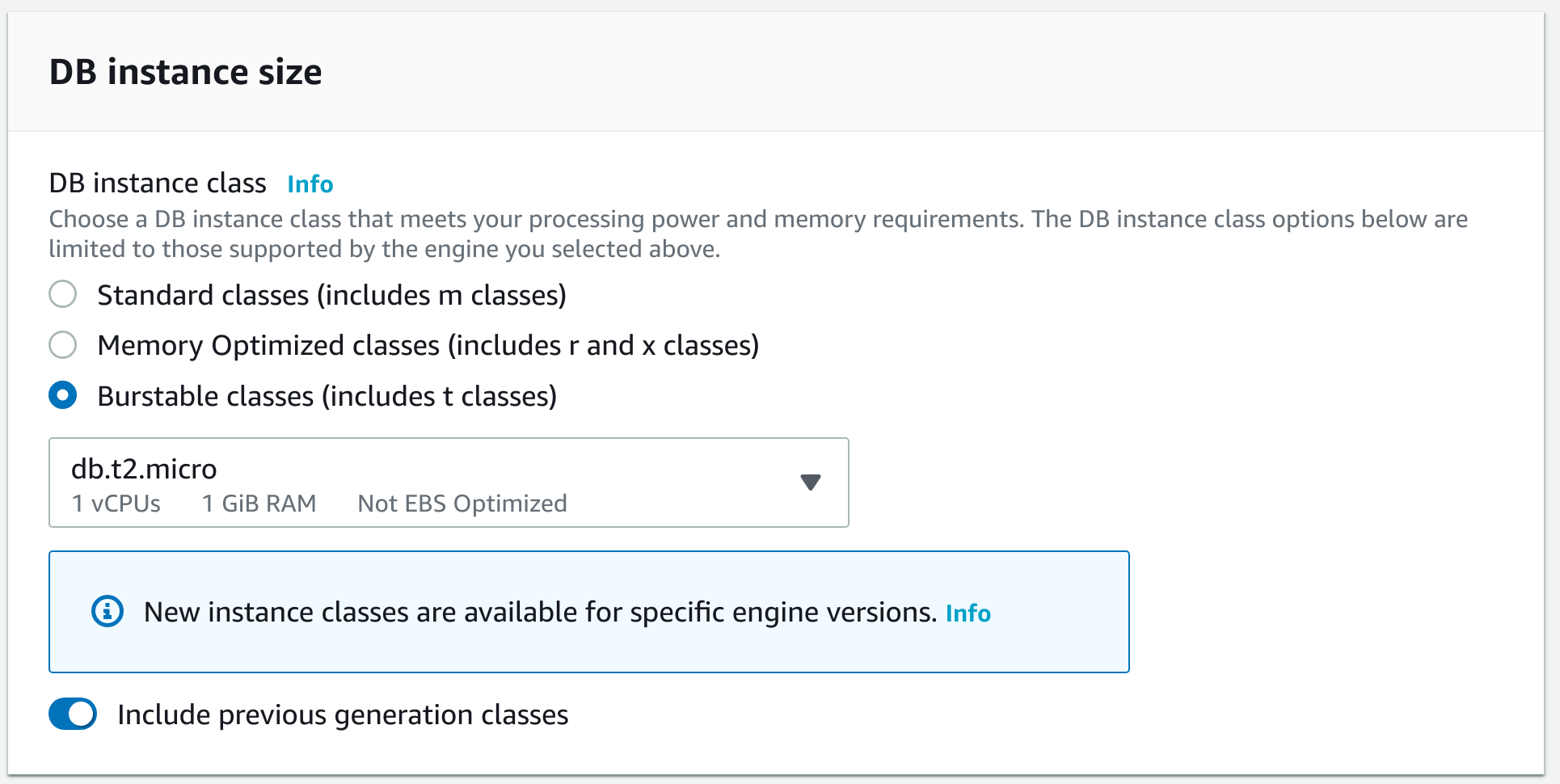
5. In the **Settings** section, set the following options:

* **DB instance identifier**: *rds-lab*
* **Master username**: *cloudacademy*
* **Master password**: *myStrongRDSpwd!*



6. In the **DB instance size** section, set the following options:

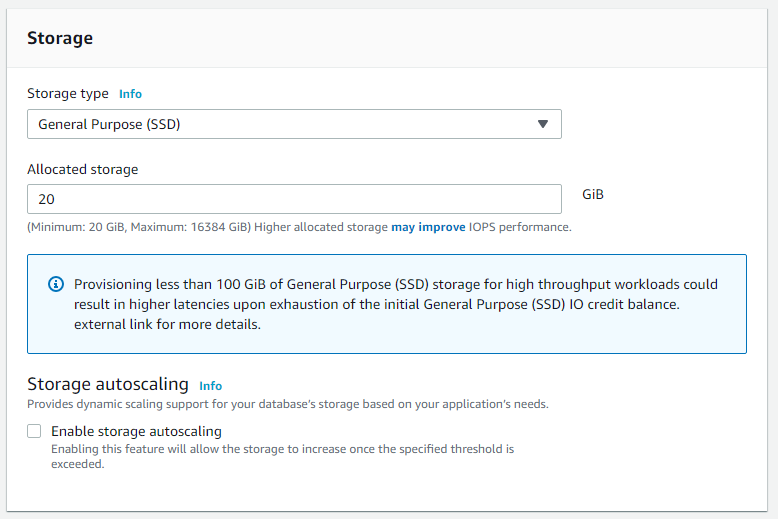
* Click the **Include previous generation classes** toggle
* **DB Instance Class**: Select **Burstable classes** and select **db.t2.micro** from the drop-down



The **db.t2.micro** has modest specs but is sufficient for this lab.

7. In the **Settings** section, set the following option:

* **Enable Storage Autoscaling**: Unchecked



For production workloads, you can consider using provisioned IOPS as the **Storage type** to guarantee consistent throughput levels. **Storage autoscaling** can also be useful in production to avoid manually scaling up the database's storage when it breaches a certain threshold. Neither is needed for this lab.

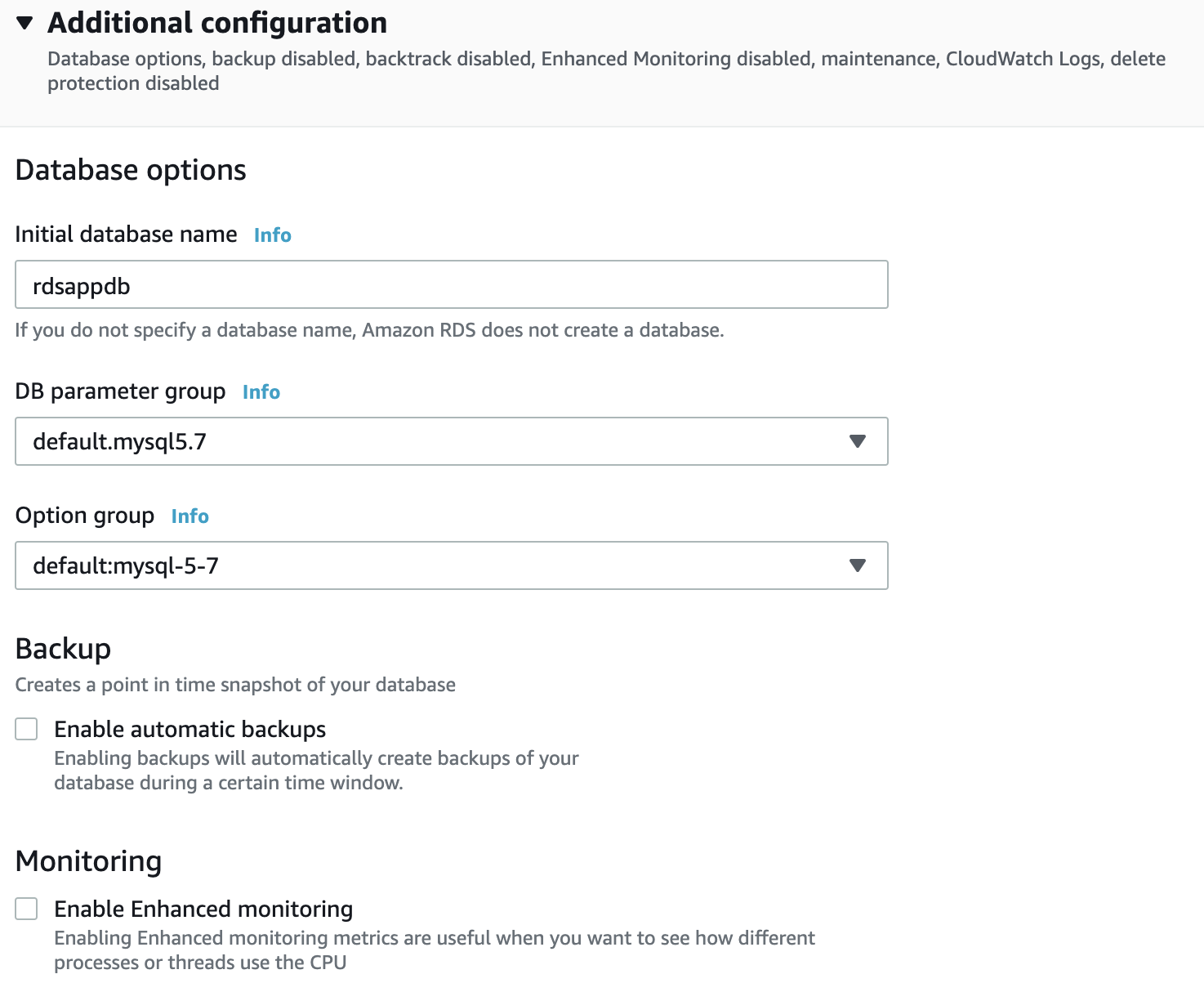
8. In the **Connectivity** section, expand the **Additional connectivity configuration**section and provide additional information that RDS needs to launch the MySQL DB instance:

* **VPC Security Group(s)**: Select the **rds-launch-wizard**

9. In the **Additional configuration** section, set the following values.

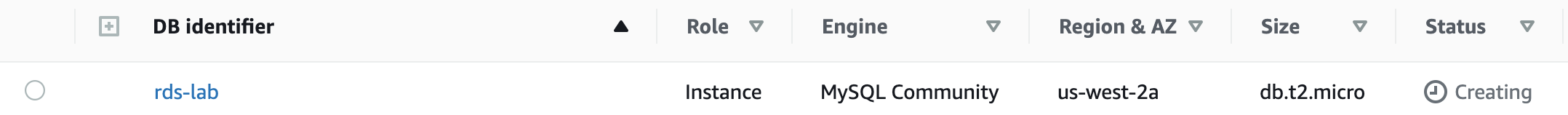
*Note:* You may see permissions errors while completing this section. They are expected and only mean that your access is limited to the scope needed for this Lab:

* **Database options**
  + **Initial database name**: *rdsappdb*
* **Backup**
  + **Enable automatic backups**: Unchecked
* **Monitoring**
  + **Enable Enhanced monitoring**: Unchecked



10. Click **Create database**.

The **Databases** view appears



*Note*: RDS instance creation requires up to 10 minutes for completion. Wait until its status becomes **Available**.

**Summary**

In this Lab Step, you used the AWS Management Console to create an RDS Database.

# Starting an AWS Systems Manager Session Manager Browser Shell Session

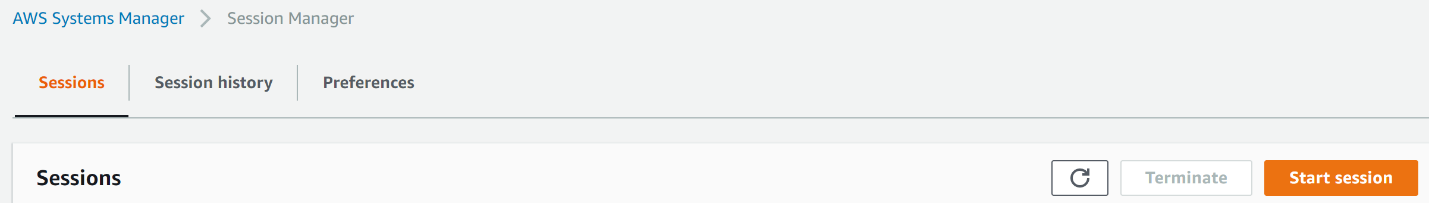
### Introduction

Session Manager is part of AWS Systems Manager suite of tools for gaining operational insights and taking action on AWS resources. Session Manager gives you browser-based shell access to EC2 instances running the Systems Manager agent. Both Windows and Linux instances are supported. Session manager provides secure access to instances without the need to distribute passwords or SSH keys. Session Manager also allows you to connect to instances without having to open any inbound ports. All communication is encrypted and IAM policies can restrict access to sessions running in Session Manager.

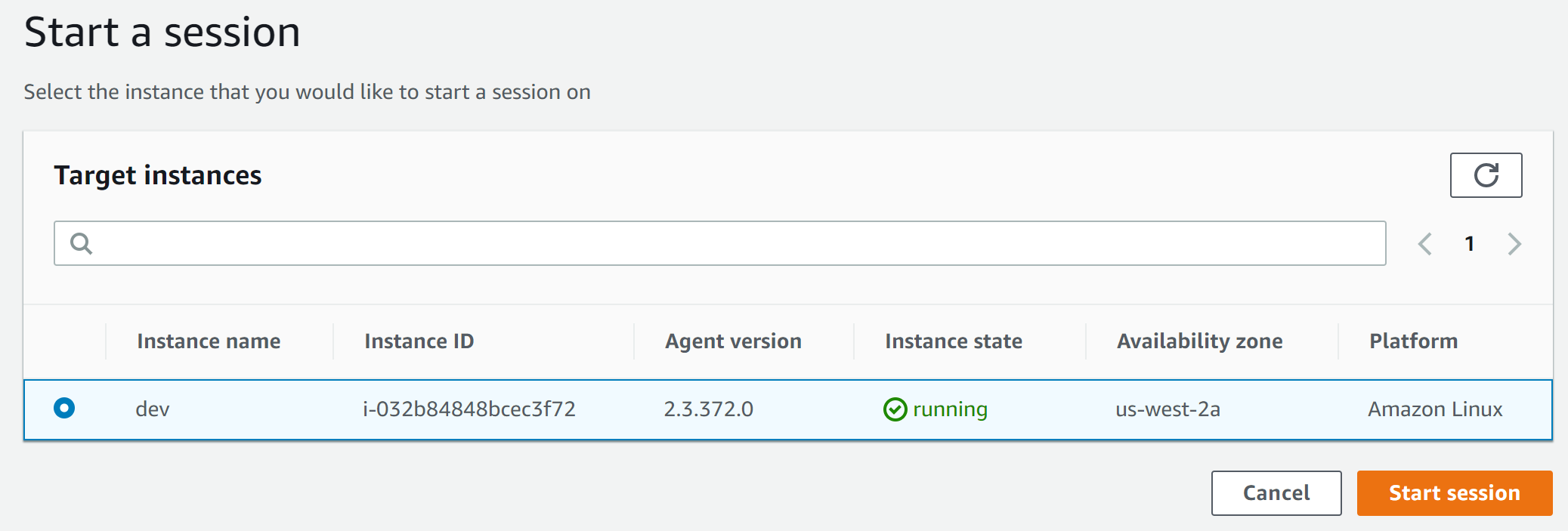
You will use Session Manager to start a session on an EC2 instance running in your Cloud Academy Lab environment in this Lab Step.

### Instructions

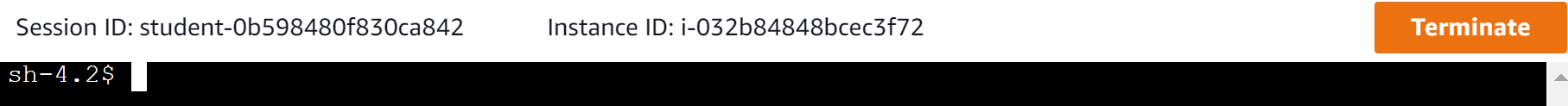
1. [Navigate to AWS Systems Manager > Session Manager](https://us-west-2.console.aws.amazon.com/systems-manager/session-manager/sessions?region=us-west-2) and click **Start session** to create a browser-based Linux shell session:



2. Select the **dev** instance as the **Target instance** and click **Start Session**:



The browser-based Linux shell session starts in a new browser tab:



### Summary

In this Lab Step, you created a session to open a browser-based shell on an EC2 instance using AWS Systems Manager Session Manager.

# Connecting to RDS and Creating a Database Table