AWS practice tasks:

Create new IAM Group

In this exercise, you will create a new IAM group for the users with Admin privileges.

- ✓ Login to the Management console
- ✓ Choose IAM service
- ✓ Choose Groups -> Create new group
- ✓ Call your group Administrators
- ✓ On the next step attach AdministratorAccess policy to the Group
- ✓ Create group

Create an IAM User

In this exercise, you will create a test IAM user who can perform all administrative IAM functions. Then you will log in as that user so that you no longer need to use the root user login. Using the root user login only when explicitly required is a recommended security practice (along with adding MFA to your root user).

- ✓ While logged in as the root user, create a new IAM user called Administrator-test.
- ✓ Add your new user to the Administrators group.
- ✓ On the Details page for the administrator user, create a password.
- ✓ Log out as the root user.

Use the customized sign-in link to sign in as Administrator.

Create an Amazon Simple Storage Service (Amazon S3) Bucket and make it public

In this exercise, you will create a new Amazon S3 bucket in us-east-1 region. You will use this bucket in the following exercises.

- ✓ Log in to the AWS Management Console.
- ✓ Choose an appropriate region, such as US East (N. Virginia).
- ✓ Navigate to the Amazon S3 console. Notice that the region indicator now says Global. Remember that Amazon S3 buckets form a global namespace, even though each bucket is created in a specific region.
- ✓ Start the create bucket process.
- ✓ When prompted for Bucket Name, use mynewbucket.
- ✓ Choose a region, such as US East (N. Virginia).
- ✓ Try to create the bucket. You almost surely will get a message that the requested bucket name is not available. Remember that a bucket name must be unique globally.
- ✓ Try again using your surname followed by a hyphen and then today's date in a sixdigit format as the bucket name (a bucket name that is not likely to exist already). You should now have a new Amazon S3 bucket.

Make your bucket public

- ✓ Select your bucket
- ✓ Go to Permissions tab
- ✓ Edit Block public access and uncheck Block all public access option
- ✓ Save changes

Upload, Make Public, Rename, and Delete Objects in Your Bucket

Upload an Object

- ✓ Load your new bucket in the Amazon S3 console.
- ✓ Select Upload, then Add Files.
- ✓ Locate a file on your PC that you are okay with uploading to Amazon S3 and making public to the Internet.
- ✓ Select a suitable file, then Start Upload. You will see the status of your file in the Transfers section.
- ✓ After your file is uploaded, the status should change to Done.

Open the Amazon S3 URL

- ✓ Select your object
- ✓ In the end of Overview section, you can find the object's URL
- ✓ Copy the Amazon S3 URL for the object.
- ✓ Paste the URL in the address bar of a new browser window or tab. You should get a message with an XML error code AccessDenied. Even though the object has a URL, it is private by default, so it cannot be accessed by a web browser.

Make the Object Public

- ✓ Go back to the Amazon S3 Console and select your objet.
- ✓ Select make public
- ✓ Copy the Amazon S3 URL again and try to open it in a browser or tab. Your public image file should now display in the browser.
- ✓ In the Amazon S3 console, select Rename.
- ✓ Rename the object, but keep the same file extension.
- ✓ Copy the new Amazon S3 URL and try to open it in a browser or tab. You should see the same image file.

Delete the Object

- ✓ In the Amazon S3 console, select Delete. Select OK when prompted if you want to delete the object.
- ✓ The object has now been deleted.

To verify, try to reload the deleted object's Amazon S3 URL. You should once again get the XML AccessDenied error message.

Launch and Connect to a Linux EC2 Instance

- ✓ Launch an instance in the Amazon EC2 console.
- ✓ Choose the Amazon Linux AMI.
- ✓ Choose the t2.micro instance type.
- ✓ Launch the instance in either the default VPC or EC2-Classic.
- ✓ Check that instance will be launched with a public IP address (auto-assign public IP should be enabled).
- ✓ Add a tag to the instance of Key: Name, Value: Exercise 1.
- ✓ Create a new security group called Cert_Book.
- ✓ Add a rule to Cert Book allowing SSH access from the IP address of your workstation (www.whatsmyip.org is a good way to determine your IP address).
- ✓ Launch the instance.
- ✓ When prompted for a key pair, choose a key pair you already have or create a new one and download the private portion.
- ✓ SSH into the instance using the public IP address, the user name ec2-user, and the keyname file.
- ✓ From the command-line prompt, run sudo yum update -y.
- ✓ Close the SSH window and terminate the instance.

Create an Amazon EBS Volume and Show That It Remains After the Instance Is Terminated

- ✓ Launch an instance in the Amazon EC2 console.
- ✓ Choose the Amazon Linux AMI.
- ✓ Choose the t2.micro instance type.
- ✓ Launch the instance in either the default VPC or EC2-Classic.
- ✓ Assign the instance a public IP address.
- ✓ Add a second Amazon EBS volume of size 8 GB. Note that the Root Volume is set to Delete on Termination.
- ✓ Add a tag to the instance of Key: Name, Value: Exercise 3
- ✓ Use the Cert Book security group from earlier exercises.
- ✓ Launch the instance.
- ✓ Find the two Amazon EBS volumes on the Amazon EBS console. Name them both Exercise 3
- ✓ Terminate the instance. Notice that the boot drive is destroyed, but the additional Amazon EBS volume remains and now says Available.
- ✓ Delete the Available volume.

Create a Custom Amazon VPC

- ✓ Sign into the AWS Management Console as an administrator or power user.
- ✓ Select the Amazon VPC icon to launch the Amazon VPC Dashboard.
- ✓ Create an Amazon VPC with a CIDR block equal to 192.168.0.0/16, a name tag of My First VPC, and default tenancy. You have created your first custom VPC.

Create Two Subnets for Your Custom Amazon VPC

- ✓ Create a subnet with a CIDR block equal to 192.168.1.0/24 and a name tag of My First Public Subnet. Create the subnet in the Amazon VPC from the previous exercise and specify an Availability Zone for the subnet (for example, us-east-1a).
  - Change "Auto-assign public IP" option to Enable through the Actions->Modify Auto-assign IP settings
- ✓ Create a subnet with a CIDR block equal to 192.168.2.0/24 and a name tag of My First Private Subnet. Create the subnet in the Amazon VPC from the previous exercise and specify a different Availability Zone for the subnet than previously specified (for example, us-east-1b).

Connect Your Custom Amazon VPC to the Internet and Establish Routing

- ✓ Create an IGW with a name tag of My First IGW and attach it to your custom Amazon VPC.
- ✓ Add a route to the main route table for your custom Amazon VPC that directs Internet traffic (0.0.0.0/0) to the IGW.

Launch an Amazon EC2 Instance and Test the Connection to the Internet

- ✓ Launch a t2.micro Amazon Linux AMI as an Amazon EC2 instance into the public subnet of your custom Amazon VPC, give it a name tag of My First Public Instance, and select the key pairs for secure access to the instance (Note Use a default security group with opened port 22).
- ✓ Securely access the Amazon EC2 instance in the public subnet via SSH with the key pair.
- ✓ Execute an update to the operating system instance libraries by executing the following command: # sudo yum update -y
  - You should see output showing the instance downloading software from the Internet and installing it.

Delete everything you've created.