D387 – Advanced Java

Student ID: 006813879

Joseph Ruiter

Deploy a Docker Container to AWS Cloud Service

Step 1. Log into AWS

- Log into your existing AWS account as the root user. If you do not have an account, create one.
- Once logged in, navigate to the Console Home page.
- In the search bar, type in “IAM” and click on “IAM” under Services. Usually, this is the first result. Click on the link and it will navigate to the IAM dashboard.
- In the left column in IAM dashboard, under Access Management, click on Roles. Once on the Roles page, click on the button “Create role.”
- The only thing you will need to do now is to click on EC2 under “Use case” heading. Click on “Next.”
- Under “Permissions policies” search for “SSM” and check the box for the Policy name “AmazonEC2RoleforSSM.” Click on “Next.”
- Under Role details, give the Role name a unique name (i.e. SSMDemo), and then provide a description for the role.
- Scroll to the bottom of the page and click on “Create role.” AWS will navigate back to Roles, and your newly created role will appear.

Step 2. Launch an EC2 Linux Instance

- Navigate back to the Console Home page.
- In the search bar, type in “EC2” and click on “EC2” under Services. Click on the link and it will navigate to the EC2 dashboard.
- In the left column in EC2 dashboard, under Instances, click on Instances. Once on the Instances page, click on the button “Launch instances.”
- Give the new instance a name.

- Scroll down and ensure that “Amazon Linux” under Application and OS Images (Amazon Machine Image) is highlighted.
- Keep the Instance type as Free tier eligible.
- Under Key pair (login), click on the drop-down menu and then click on launch. This uses RSA for launching.
- Under Network settings, check the boxes for “Allow HTTPS traffic from the internet” and “Allow HTTP traffic from the internet.” Due to our application being a website, these are important to enable http and https traffic.
- Under Advanced details, click on the drop-down menu for IAM instance profile, and then click on the newly created IAM role that was created in step 1. Finally, launch the instance.
- Click on “View all instances” and then check the box of the newly created instance.
- Copy the instance IP Address, then paste it into a new tab in your browser. Don’t worry, it will not work right now.

Step 3. Start a Session

- In the search bar, type “Systems Manager” and click on “Systems Manager” link under Services. Click on the link and it will navigate to the AWS Systems Manager dashboard.
- In the left column, scroll down to “Node Management” and click on “Session Manager.” Once on the Session Manage page, click on “Start session.”
- Under “Target instances” click on the radio button for your created instance. Click on “Start session.” This will load your Linux instance.
- In your Linux instance, run the following commands:
  - `$sudo sh`
  - `$yum install -y docker`
  - `$service docker start`

- Once Docker is installed and running, we can finally move onto Step 4.

Step 4. Pull Instance from Docker and Deploy

- In the running instance, log into Docker hub using the following command:
  - `$docker login`
- Type in your username and use a personal access token as the password, created in settings in your Docker account.
- Pull the Docker Container from Docker Hub by using the following command:
  - `$docker pull nameofrepository/nameofinstance`

  ***Note: replace* `nameofrepository` *with the repository name and* `nameofinstance` *with the name of the instance in the repository*.
- Wait for the container to pull and install, then run the following command:
  - `$docker images`
- Finally, run the following command:

```
$docker run –name awsapplication -d –p 80:8080 nameofrepository/nameofinstance
```

***Note: replace* `awsapplication` with a name of your choosing.

- Navigate back to the tab with your pasted instance IP Address and refresh. The website should populate. It may take a few minutes.
- Once the website is loaded, congratulations! The application is successfully deployed to the AWS Cloud.