- **Task 1: Create VPC**

1. Go to **AWS Management Console** -> **Services** -> **VPC**
2. Click **Launch VPC wizard**
3. In the navigation pane, click **VPC with Public and Private subnet**
4. Click **select**
5. Configure the following settings:

- **IPv4 CIDR block Type:** 10.0.0.0/16
- **VPC Name:** My Lab VPC
- **Public Subnet IPv4 CIDR:** 10.0.1.0/24
- **Availability Zone**: select the first one
- **Public Subnet name:** Public Subnet 1
- **Private subnet IPv4 CIDR:** 10.0.3.0/24
- **Availability Zone** select the same which you selected for Public Subnet 1
- **Private Subnet name:** Private Subnet 1
- Select **Use a NAT instance instead**
- **Key pair name:** Select the key which you used during EC2 Instance

6. Select **Create VPC**
7. In the success message, click **OK**

- **Task 2: Create Additional Subnets**

In this task you create two additional subnets in another Availability Zone and associate the subnets with existing route tables.

8. In the navigation pane, Click **Subnets**.
9. Click **Create Subnet**
10. In the **Create Subnet** dialog box configure the following settings:

- **Name tag:** Public Subnet 2
- **VPC:** Click **My Lab VPC**
- **Availability Zone:** Select the second one
- **IPv4 CIDR block:** Type 10.0.2.0/24

11. Click **yes Create**
12. Click **Create Subnet**
13. In the **Create Subnet** dialog box configure the following settings:

- **Name tag:** Private Subnet 2
- **VPC:** Click **My Lab VPC**
- **Availability Zone:** Select the same zone used for Public Subnet 2
- **CIDR block:** Type 10.0.4.0/24

14. Click **yes Create**
15. In the navigation pane, click **route tables**
16. Select the route table with the VPC **My Lab VPC** and **yes** under **Main**
17. Double click the empty **Name** for this route table, type Private Route table, and click the checkmark to save.
18. In the lower pane, click **Routes** and note that **destination 0.0.0.0/0** is set to **target eni-xxxxxxxx/i-xxxxxxxx.** This route table is used to route traffic from traffic subnets to the NAT instance, as identified by an Elastic Network Interface (ENI) and Instance ID.
19. Click **subnet associations** and the click **Edit**.

20. Select **Private Subnet 1** and **Private subnet 2**.
21. Click **save**.
22. Select the route table with the VPC **My Lab VPC** and **no** under **Main**
23. Double click the empty **Name** for this route table, type Public Route table, and click the checkmark to save.
24. In the lower pane, click **Routes** and note that **destination 0.0.0.0/0** is set to **target eni-xxxxxxxx/i-xxxxxxxx**. This route table is used by public subnets for communication.
25. Click **subnet associations** and the click **Edit**.
26. Select **Public Subnet 1** and **Public subnet 2**.
27. Click **save.**

- **Task 3: Create a VPC Security Group**

In this task, you create a VPC security group that permits access for web traffic.

28. In the navigation pane, click **Security Groups**.
29. Click **create Security group**
30. In the **Create security group** dialog box, configure the following settings

- **Name Tag:** WebSecurityGroup
- **Group Name:** Click **WebSecurityGroup**
- **Description:** Type Enable HTTP access
- **VPC:** Click **My Lab VPC**. This is the VPC you created in task 1

31. Click **Yes, create**
32. Select **WebSecurityGroup**
33. Click the **Inbound Rules** Tab
34. Click **Edit**
35. For **type** Click HTTP (80)
36. Click in the **source** box and type 0.0.0.0/0
37. Click **save.**

- **Task 4: Launch Your First Web Server instance**

In this task, you can launch EC2 instance into the VPC you created and bootstrap the instance to act as a web server.

38. On the **services** menu, Click **EC2**
39. Click **Launch Instance**
40. You will be asked to select **Amazon Machine Image(AMI)** click **Select**. If you receive a warning click **continue.**
41. Select **instance type t2.micro**
42. Click next configure instance details
43. Configure the following settings:

- **Network:** Click **My Lab VPC**. This is the VPC you created in task 1
- **Subnet**: Click the **Public subnet 2 (10.0.2.0/24)**. This is the subnet created in task 2.
- **Auto-assign Public IP:** Click **Enable.**

44. Go to **Advanced Details**
45. Paste the code:

```
#!/bin/bash -ex
```

```
yum -y install httpd php mysql php-mysql
chkconfig httpd on
service httpd start
if [ ! -f /var/www/html/lab2-app.tar.gz ]; then
cd /var/www/html
wget https://us-west-2-aws-training.s3.amazonaws.com/awsu-ilt/AWS-100-ESS/v4.2/lab-2-configure-website-datastore/scripts/lab2-app.tar.gz
tar xvfz lab2-app.tar.gz
chown apache:root /var/www/html/rds.conf.php
fi
```

46. Click next: **Add Storage**
47. Click next: **Add Tags**
48. Click **Add Tag**, Configure
    - **Key:** Name
    - **Value:** Web Server 1
49. Click **next: Configure Security Group**
50. Click **select an existing security group** and the select security group you created in task 3
51. Click **review and launch**. When prompted with a warning that you will not be able to connect to the instance through port 22, Click **continue.**
52. Click on **review and launch**
53. In the **key pair** dialog box leave it selected on **Choose an existing key pair**, select the acknowledgement box, then click **launch instances**.
54. Now you will get the instance ready scroll down click on **view instance.** You will see 2 instances- **Web Server 1** and NAT instance launched by the VPC wizard.
55. Wait until **Web Server 1** show 2/2 checks passed in the **status checks** column. This will take 3 to 5 minutes. Click the refresh icon in the upper right pane to check for updates.
56. Select Web Server 1 and copy the **Public DNS** value on the **description** tab.
57. Paste the **Public DNS** value in a new web browser window or tab and press **ENTER**. You will see a web page displaying the AWS logo and instance metadata values.