VPC Setup

Task 1

1. Go to AWS Management Console
2. Click on Create VPC
3. Give VPC a name
4. Give it an IPv4 CIDR Block such as 10.0.0.0/16. CIDR Block specifies the number of IP Addresses that you can have in your VPC.
5. You can also give it an IPv6 CIDR Block if required. Next tenancy will indicate whether you want to put it on shared hardware or dedicated hardware. Default will specify shared hardware.
6. Click on Create
7. Click on Subnets. Subnets are the sub networks of your VPC in which you will place your resources.
8. Click on Create Subnet
9. Give the subnet a name tag to identify it.
10. We will call it Public Subnet 1
11. Put it in the VPC that you created right now.
12. Put it in availability Zone a
13. Give it a CIDR Block such as 10.0.1.0/27.
14. Click on Create Subnet
15. Name this subnet as Private Subnet 1
16. Put it in your VPC
17. Again put in AZ one
18. Give it a CIDR Block as 10.0.2.0/24
19. Click on Create.

Task 2

Now that we have the subnets set up let us actually make the subnets public and private. How do you set that up?

1. Go to Internet Gateways
2. Create a new Internet Gateway
3. Attach it to the VPC that you created before
4. Now, go to Route Tables
5. Select the main route table for your VPC
6. Go to Routes in the Description Tab
7. Click on Edit Routes
8. Add a route specifying destination as 0.0.0.0/0 and target as Internet Gateway which you created right now.
9. Then click on Subnet Associations
10. Click on Edit and select the public subnet

This is how you make the subnet public.

11. Create an instance in the public subnet and update the instance to show it has internet access

Next, for making the subnet private,

1. Click on Create route table
2. Give it a name of Private Route Table and select your own VPC.
3. Click on Create
4. Click on Routes.
5. Observe the route for internal traffic.
6. Go to subnet associations and associate it with the private subnet.

   This is how you actually set up private access for the subnet.

   1. Create an EC2 Instance in the private subnet.
   2. SSH into it using the public subnet and show that the instance does not have any internet access.

      For SSH into a private subnet,

      https://aws.amazon.com/blogs/security/securely-connect-to-linux-instances-running-in-a-private-amazon-vpc/