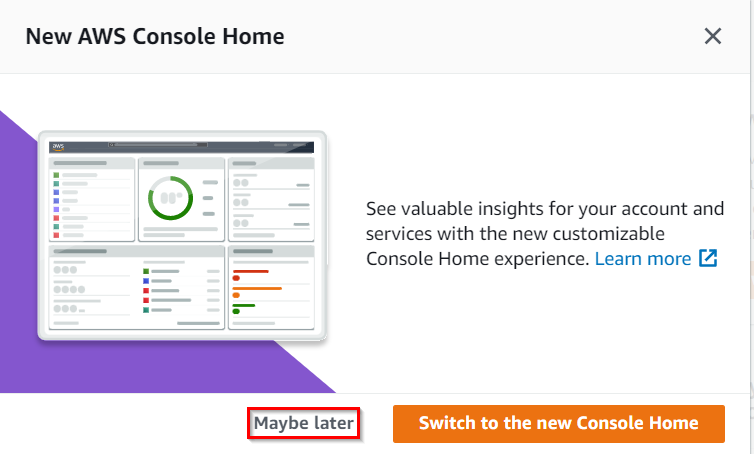
**Lab Steps**

Task 1: Sign in to AWS Management Console

1. Click on the  button, and you will get redirected to AWS Console in a new browser tab.
2. On the AWS sign-in page,
   * Leave the Account ID as default. Never edit/remove the 12 digit Account ID present in the AWS Console. otherwise, you cannot proceed with the lab.
   * Now copy your **User Name** and **Password** in the Lab Console to the **IAM Username and Password** in AWS Console and click on the **Sign in** button.
3. Once Signed In to the AWS Management Console, Make the default AWS Region as **US East (N. Virginia) us-east-1.**
4. Select Maybe later in New AWS Console Home page pop-up



Task 2: Setup Visual studio code

1. Create a folder using File Explorer on your local machine's desktop with the name task\_10000\_vpc.
2. Open the Visual Studio Code application. If you haven't installed the VSC software, download and install using the software using the link provided in the Prerequisite section above.
3. If you have already installed and using Visual studio code, Open a New Window by clicking **File** from the menu bar present on the top and select **New Window**.
4. And, now select **Open** present under the **File** menu and navigate to the desktop location. Select the Folder task\_10000\_vpc.
5. A new window will open a new file and release notes page (only if you have installed or updated Visual Studio Code recently). **Close the Release notes tab**.
6. Visual Studio Code is now ready to use.

Task 3: Create a variables file

In this task, you will create variables files where you will declare all the global variables with a short description and a default value.

1. To create variables files, click on the **File** from the menu bar and choose **New file**
2. Press Ctrl + S to save the new file as **variables.tf**and click on the **Save** button after entering the file name.
3. **Note:** Don't change the location of the new file, keep it default, i.e. inside the **task\_10000\_vpc** folder**.**
4. Paste the below contents in **variables.tf**file:

|  |
| --- |
| variable "access\_key" {  description = "Access key to AWS console"  }  variable "secret\_key" {  description = "Secret key to AWS console"  }  variable "region" {  description = "Region of AWS VPC"  } |

1. In the above content, you are declaring a variable called, access\_key, secret\_key, and region with a short description of all 3.
2. After pasting the above contents, save the file by pressing Ctrl + S.
3. Now create the **terraform.tfvars** file by selecting **New file** present under **File** in the menu bar.
4. Name the file by pressing Ctrl + S and enter **terraform.tfvars**
5. Paste the below content into **terraform.tfvars** file

|  |
| --- |
| region = "us-east-1"  access\_key = "<YOUR AWS CONSOLE ACCESS ID>"  secret\_key = "<YOUR AWS CONSOLE SECRET KEY>" |

1. In the above code, you are defining the dynamic values of variables declared earlier.
2. Replace the values of **access\_key** and **secret\_key** by copying from the lab page.
3. After replacing the values of access\_key and secret\_key, save the file by pressing Ctrl + S.

Task 4: Create VPC and its components in main.tf file  
  
In this task, you will create **main.tf** file where you will add details of the provider and resources.

1. To create **main.tf** file, click on the **File** from the menu bar and choose **New file**
2. Press Ctrl + S to save the new file as **main.tf**and click on the **Save** button after entering the file name.
3. Paste the below content into **main.tf** file.

|  |
| --- |
| provider "aws" {  region = var.region  access\_key = var.access\_key  secret\_key = var.secret\_key  }  resource "aws\_vpc" "vpc" {  cidr\_block = "10.0.0.0/16"  } |

1. In the above code, you are defining provider as ***aws*** and defining resource block using ***aws\_vpc***  for creating an Amazon VPC having CIDR Block as 10.0.0.0/24.
2. Define Internet gateway, route table, and subnets in all availability zone. Add the below contents after aws\_vpc block.

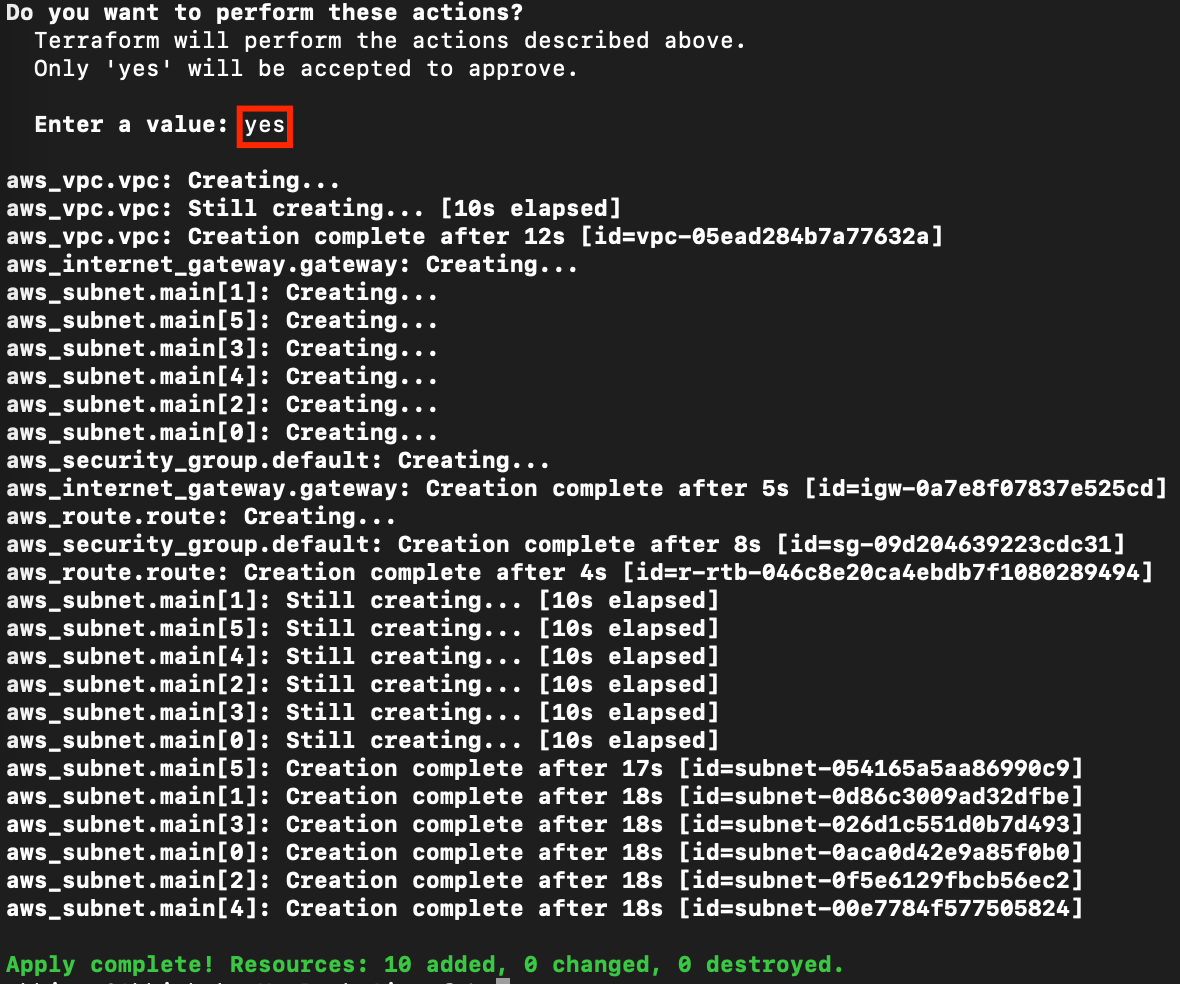
|  |
| --- |
| resource "aws\_internet\_gateway" "gateway" {  vpc\_id = aws\_vpc.vpc.id  }  resource "aws\_route" "route" {  route\_table\_id = aws\_vpc.vpc.main\_route\_table\_id  destination\_cidr\_block = "0.0.0.0/0"  gateway\_id = aws\_internet\_gateway.gateway.id  }  data "aws\_availability\_zones" "available" {}  resource "aws\_subnet" "main" {  count = length(data.aws\_availability\_zones.available.names)  vpc\_id = aws\_vpc.vpc.id  cidr\_block = "10.0.${count.index}.0/24"  map\_public\_ip\_on\_launch = true  availability\_zone = element(data.aws\_availability\_zones.available.names, count.index)  } |

1. In the above code, you are performing the following tasks:
   1. Creating an Internet gateway and attaching it with the VPC.
   2. Adding route as 0.0.0.0/0 and destination as an Internet gateway.
   3. Checking all the availability zones.
   4. Creating subnets in all availability zones.
2. Save the file by pressing Ctrl + S.

Task 5: Confirm the installation of Terraform by checking the version

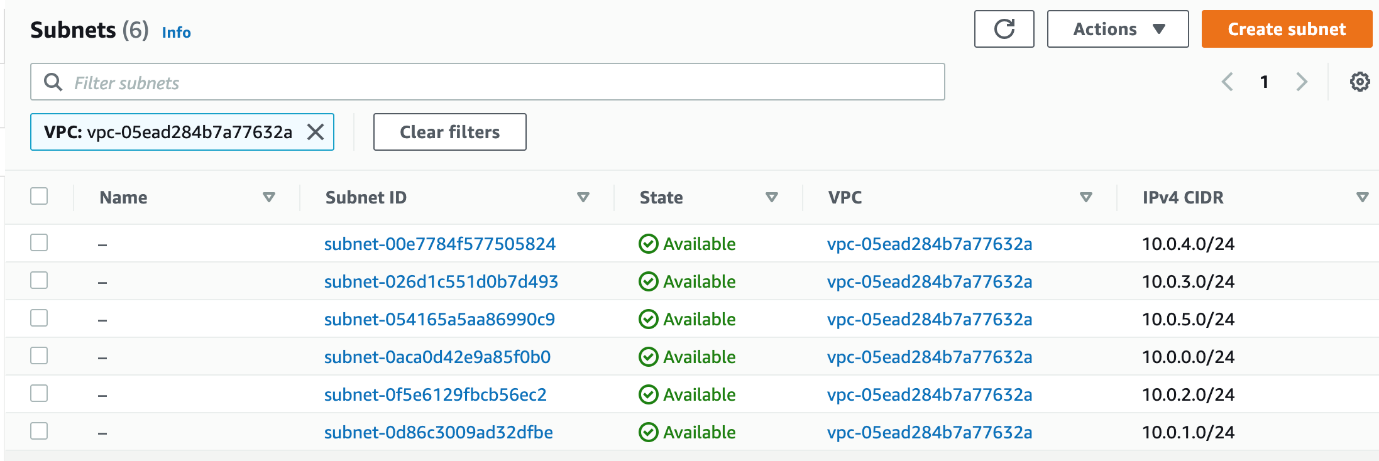
1. Open the Terminal by clicking on the **View** from the menu bar and choosing **Terminal**
2. Navigate to the folder using the cd command, example, to navigate to the desktop, use cd Desktop
3. To confirm the installation of Terraform, run the below command to check the version:
   * terraform version
4. If you are getting output as **command not found: terraform**, this means that terraform is not installed on your system, To install terraform follow the official guide link provided in the **Prerequisite** section above.

Task 6: Apply terraform configurations

1. Make sure you are in the same location where all these 3 files are present, you can check the same using the ls command.
2. Initialize Terraform by running the below command:
   * terraform init
3. **Note:**terraform init will check for all the plugin dependencies and download them if required, this will be used for creating a deployment plan.
4. To generate the action plans run the below command:
   * terraform plan
5. Review the whole generated plan.
6. To create all the resources declared in **main.tf** configuration file, run the below command:
   * terraform apply
7. You will be able to see the resources which will be created, approve the creation of all the resources by entering **yes.**
8. It may take up to 2 minutes for the **terraform apply** command to create the resources.
9. Ids of all the resources created by terraform will be visible there.  
     
   
10. Optionally, you can note down the IDs of all the resouces.

Task 7: Check the resources in AWS Console

1. Click on the  button, AWS Management Console will open in a new tab.
2. On the AWS sign-in page, the 12 digit Account ID will be present by default.
   * Leave the Account ID as default. Do not remove or change the Account ID otherwise, you cannot proceed with the lab.
3. Copy and paste the***IAM user name*** and ***Password*** into AWS Console. Click on **Sign in** to log into the AWS Console.
4. Make sure you are inthe **US East (N. Virginia) us-east-1**Region.
5. Navigate to **VPC** by clicking on **Services** under the Networking **& Content Delivery** section.Click on **VPC** (under **Networking & Content Delivery** section) or you can also search for VPC.
6. Click on **Your VPCs**from the left menu.
7. Custom VPC is present.  
   Graphical user interface, text, application

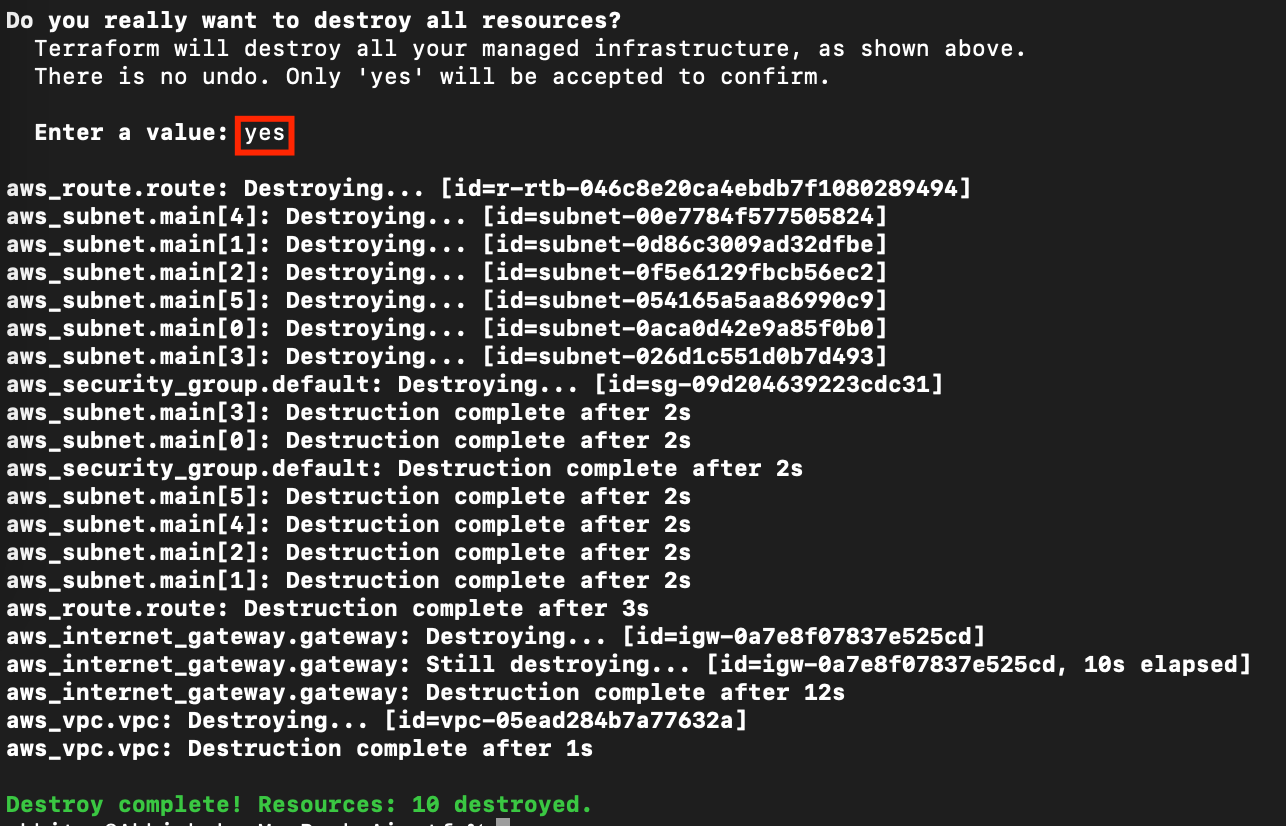
   Description automatically generated
8. To check Subnets, click on the Subnets option from the left sidebar.
9. All the present subnets are of both the VPC i.e. Default VPC and Custom VPC.
10. Apply the filter of Custom VPC by clicking on the Search bar and select **VPC** as Properties and **VPC Id of custom VPC** as VPC values.  
    
11. **Similarly, you can check Route Tables, Internet Gateways, and Security Groups.**

Task 8: Validation of the lab

1. Once the lab steps are completed, please click on the  button on the left side panel.
2. This will validate the resources in the AWS account and displays whether you have completed this lab successfully or not.
3. Sample output :   
           Graphical user interface, text, application, email

   Description automatically generated

Task 9: Delete AWS Resources

1. To delete the resources, open Terminal again.
2. Run the below command to delete all the resources.
   * **terraform destroy**
3. Enter ***yes***to confirm the deletion.  
     
   

Resources

* To download the Terraform configuration for this lab, click [**here**](https://labresources.whizlabs.com/aaa459ca98c8c0f2523311d32193cf8f/task_10002_vpc_subnet.zip).

**Completion and Conclusion**

* You have successfully done the setup of the Visual Studio Code editor.
* You have successfully created **variables.tf** and **terraform.tfvars** file
* You have successfully created a **main.tf** file.
* You have successfully executed the terraform configuration commands to create the resources.
* You have successfully checked all the resources are created by opening the Console.
* You have successfully deleted all the resources.

**End Lab**

1. Sign out of AWS Account.
2. You have successfully completed the lab.
3. Once you have completed the steps, click on  from your whizlabs dashboard.