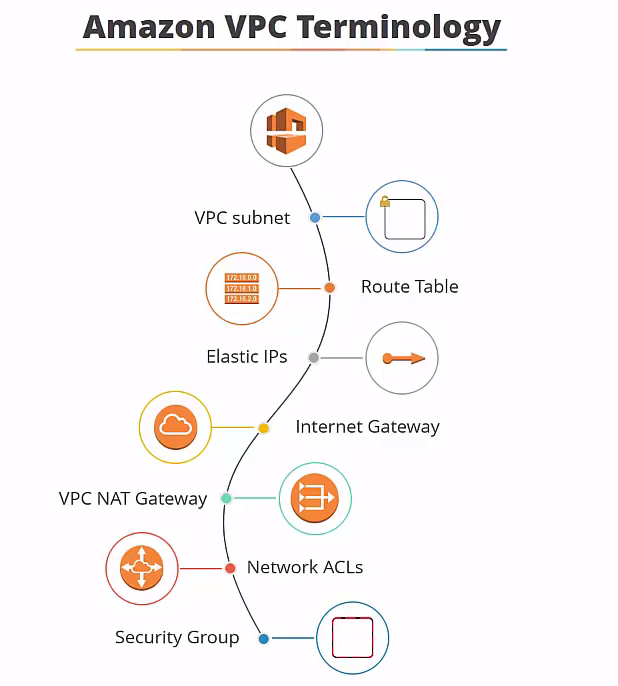
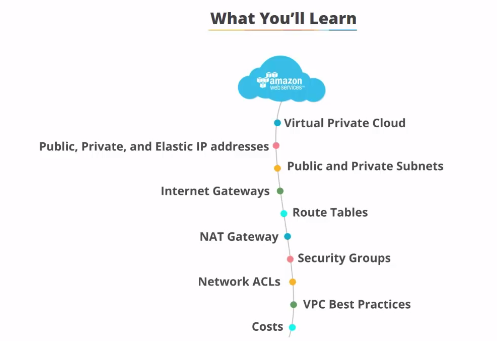
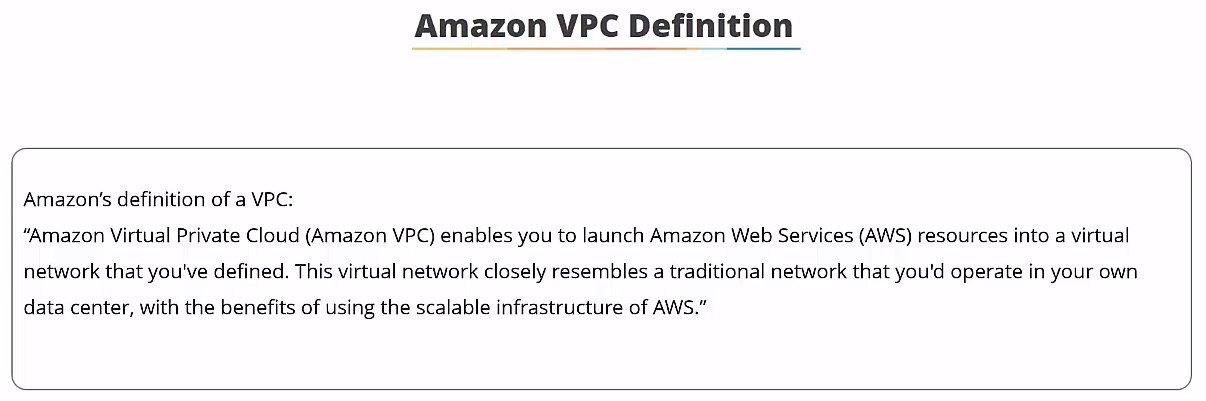
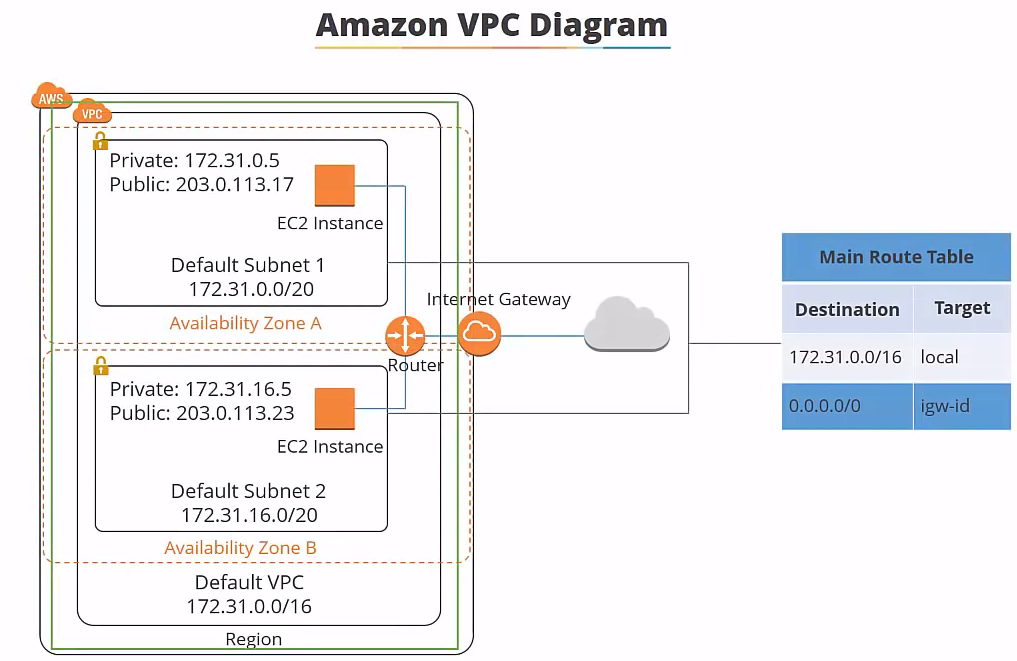
Migrating to the cloud doesn't mean that resources become completely separated from the local infrastructure.

In fact, running applications in the cloud will be completely transparent to your end-users.

AWS offers many services to fully and seamlessly integrate your local resources with the cloud one such service is the Amazon virtual private cloud.

Creating virtual networks that closely resemble the ones that operate in your own data centers but with the added benefit have been able to take full advantage of AWS.



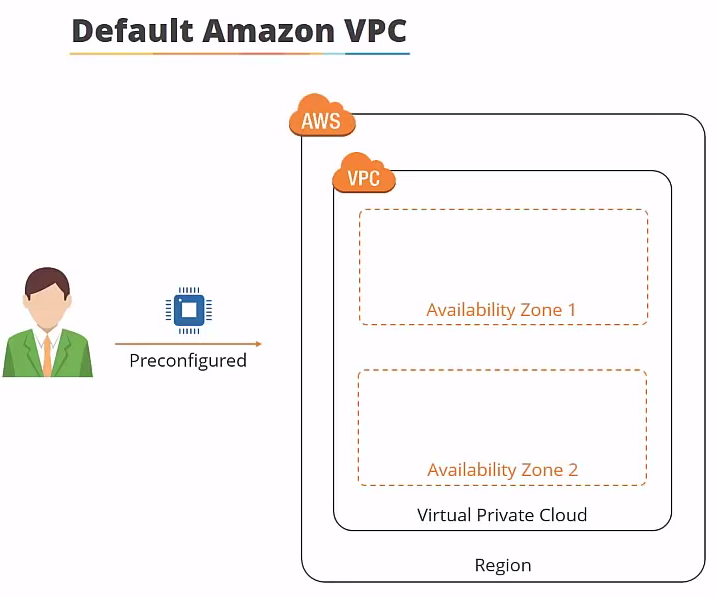
 

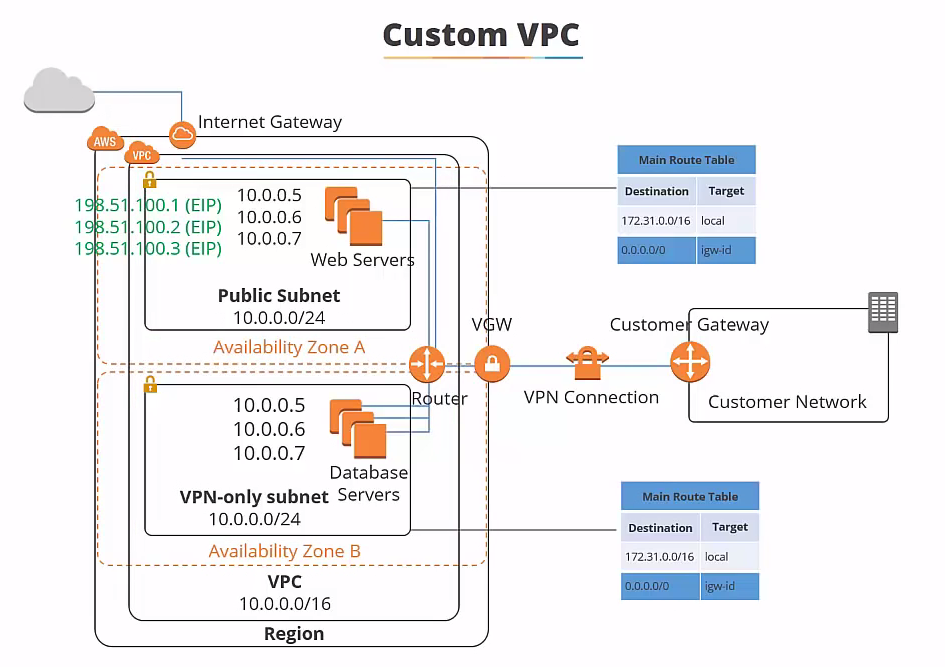
VPC is your own virtual network in the Amazon Cloud which is used as the network layer for your ec2 resources and this is a diagram of default VPC

each VPC that you create is logically isolated from other virtual networks in the AWS cloud. **It's fully customizable you can select the IP address range, create subnets, configure route tables, setup network gateways, define security settings using security groups and network access control lists.**

So each Amazon account comes with a default V PC that's pre-configured for you to start using straight away so you can launch your ec2 instances without having to think about anything.

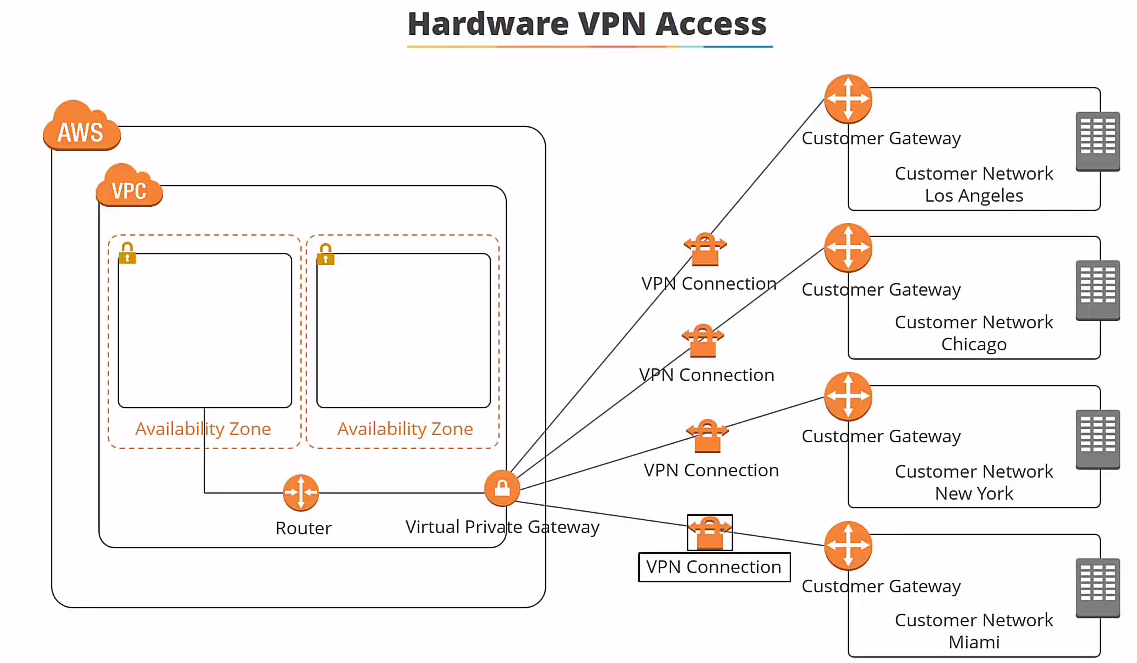
**A VPC can span multiple availability zones in a region and here's a very basic diagram of a VPC it isn't this simple in reality.**

****

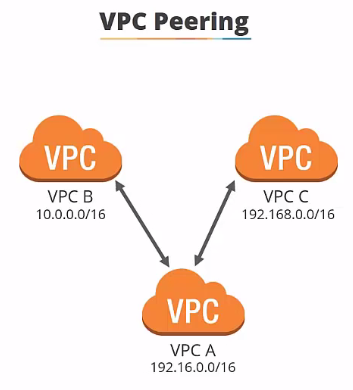
why wouldn't you just use the default VPC??? The default V PC is great for launching new instances when you're testing AWS but creating a custom VPC allows you to make things more secure and you can customize your virtual network as you can define your own our IP address range, you can create your own subnets that are both private and public and you can tighten new security settings. 

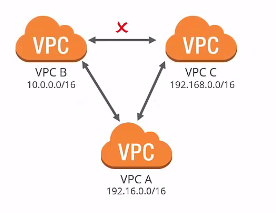
**hardware VPN access**

By default, instances that you launch into a VPC can't communicate with your own network. So you can connect your VPC to your existing data center using something called hardware VPN access so that you can effect of the extend your data centerinto the cloud and create a hybrid environment.



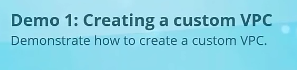
**VPC peering :** 1.) connection could be made between your own V pcs or with a V PC in another AWS account as long as it's in the same region. So what that means is, if you have instances in VPC A they wouldn't be able to communicate with instant is in VPC B or C unless you set up appear in connection ...



2.) Peering is a one-to-one relationship.. a VPC can have multiple peering connections to other V pcs but and this is important transitive peering is not supported... In other words VPC a can connect to B and C in this diagram but C wouldn't be able to communicate with B unless they were directly paired..

3.) Also vpcs with overlapping CI DRS cannot be paired,... So in this diagram you can see they all have different IP ranges which is fine... but if they have the same IP arranges they wouldn't be able to be paired..

-🡪if you delete the default V PC you have to contact AWS support to get it back again so be careful with it and only delete it if you have good reason to do so ...

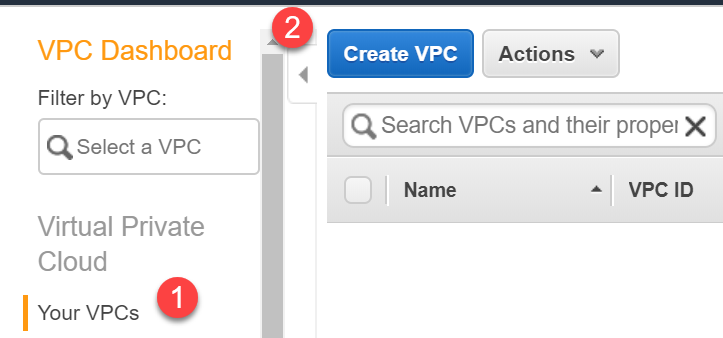


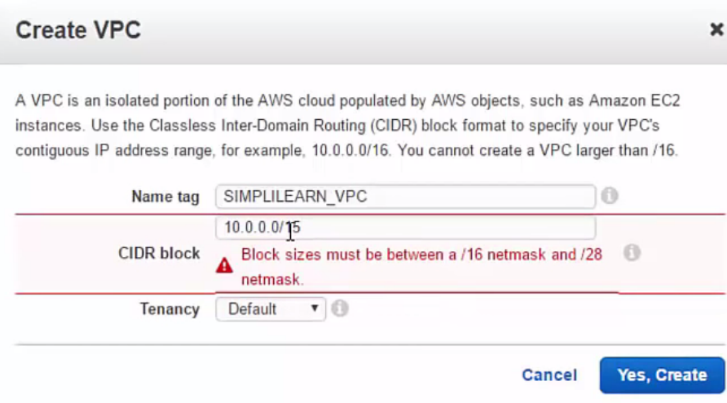
There's a couple of ways you can create a custom V PC there's something called the V PC wizard which will build Vpcs on your behalf from a selection of different configurations

for example: a V PC with a single public subnet

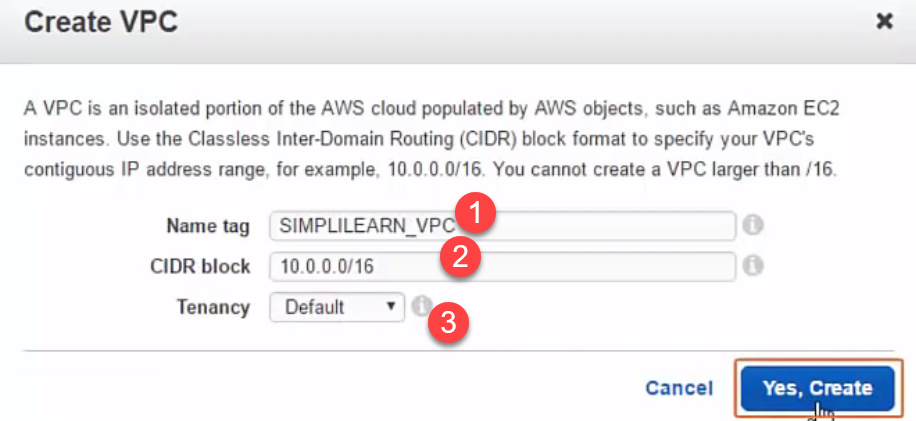
a V PC with public and private subnets ...

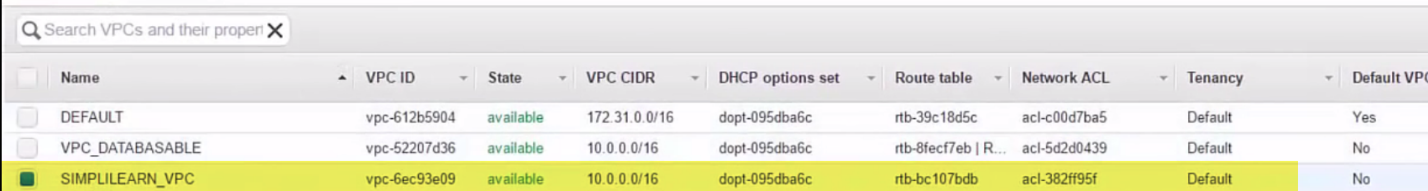
Now this is great because you click about and type in a few details and it does the work for you however you're not going to learn much or pass the exam if this is how you do it so we'll cancel that and we'll go to your V pcs



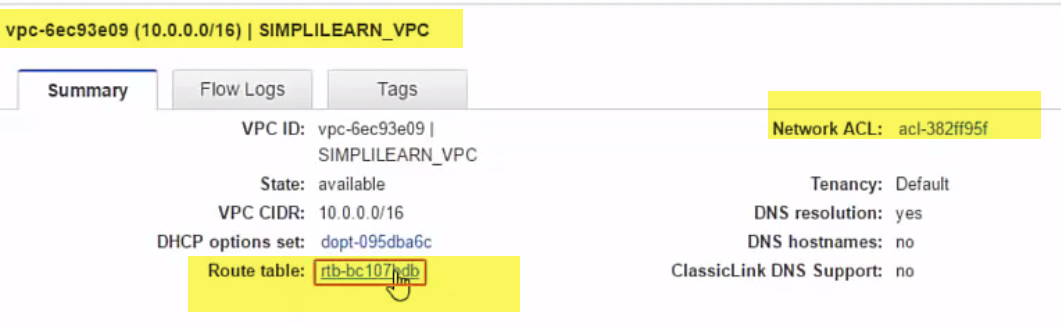
Goto Your VPC---> create a V PC

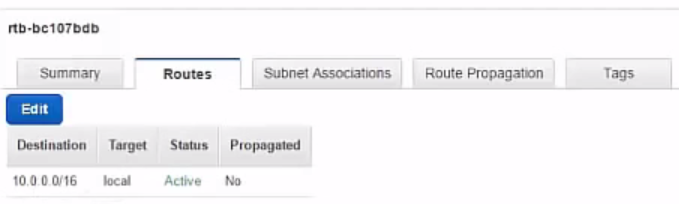
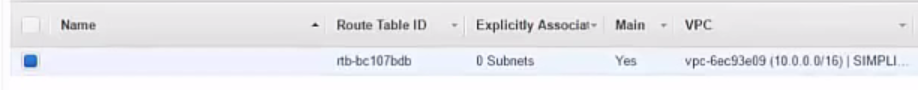
**CIDR:** give it a very simple one 10.0.0.0 and then we need to give it the subnet mask so you're not allowed to go larger than 15 so if I try to put 15 in it says no not going to happen ...(subnet mask of 15 would give you around 130 1000 IP addresses and subnet 16 will give you 65,536 which is probably more than enough for what we're going to do)..

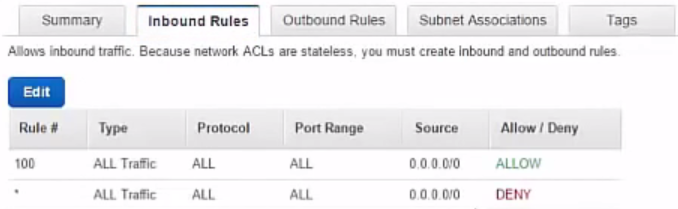
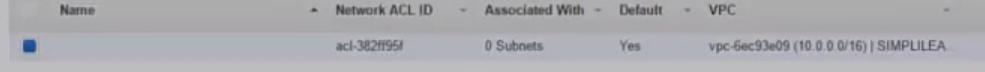
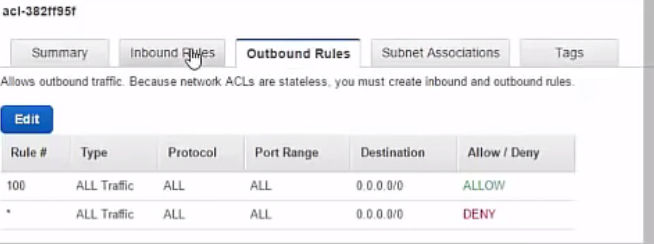
**Tenancy:** there's two options default and dedicated.. if you select dedicated then your ec2 instances will reside on hardware there's dedicated to you see performance is going to be great.. But your cost is going to be significantly higher. 

In our dashboard we can see our simply learned vpc has been created. 

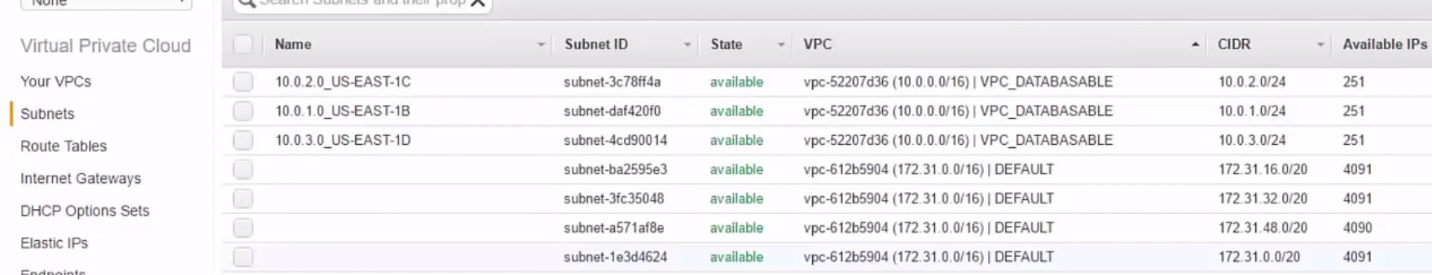
Now if we go down to the bottom , we can see it has a root table associated with it which is our default route table.

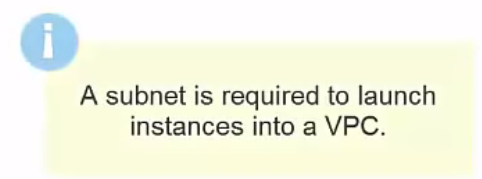


**Click on Routing table:** we can see that it's only allowing local traffic at the moment.

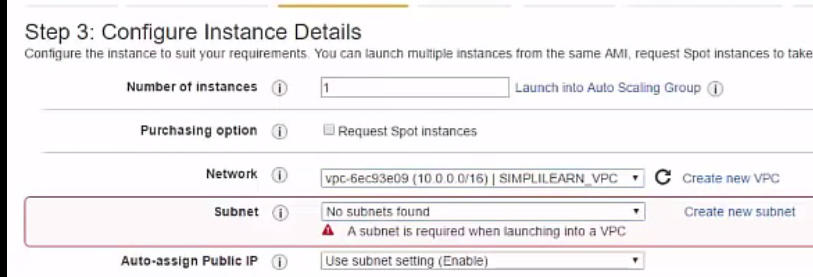
**Click on NACL** 

it's allowing all traffic from all sources inbound and outbound..

Go to the subnets section and just widen the VPC area here... you can see there's no subnets associated with the vpc which is created..

so that means we won't better launch any instances into our vpc

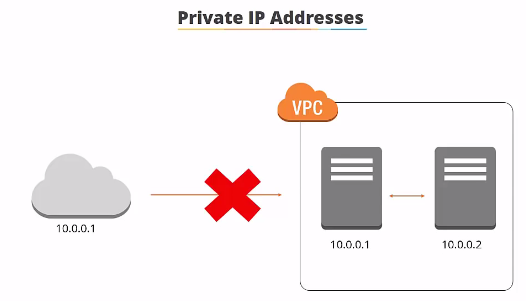
To prove it I'll just show you we'll go to the ec2 section

network section---> if I try and select simply learned VPC --->is saying no subnets found.

This is not going to work so we basically need to create some subnet in our V PC and that is what we're going to look at in the next lesson.

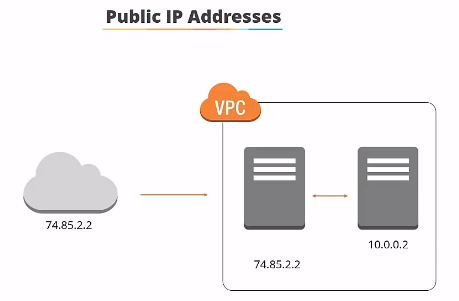
**private IP addresses** are IP addresses that are not reachable over the internet and they used for communication between instances in the same network.

When you launch a new instance is given a private IP address and an internal DNS hostname that resolves to the private IP address of the instance… but if you one connect to this from the Internet it's not going to work.

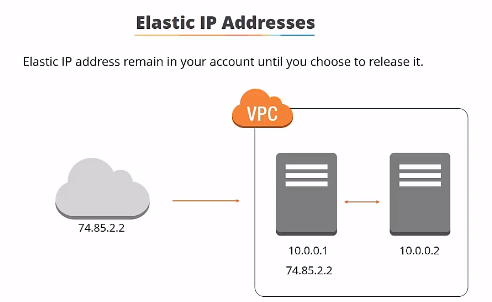


So then you'd need a **public IP address** which is reachable from the internet you can use public IP addresses for communication between your instances and the Internet.

Each instance that receives a public IP address is also given an external DNS hostname.

Public IP addresses are associated with your instances from the Amazon pool of public IP addresses. when you stop will terminate your instance the public IP address is released and a new one is associated when the instance starts.

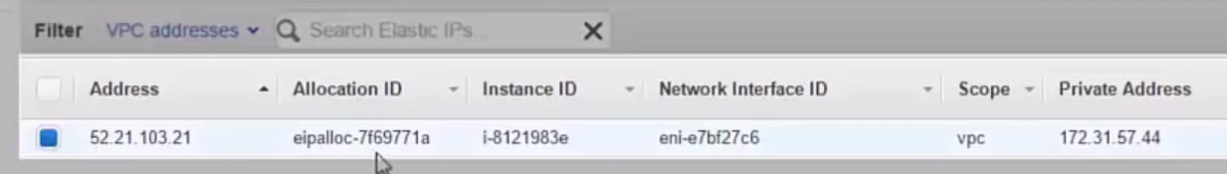
So if you want your instance to retain this public IP address you need to use something called an **elastic IP address**. An elastic IP address is a static or persistent public IP address there's allocated your account and can be associated to and from your instances as required an elastic IP address remains in your account until you choose to release it .There is a charge associated with an elastic IP address if it's in your account but not actually allocated to an instance.



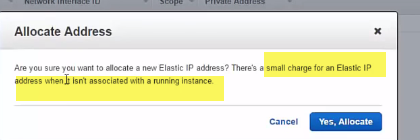
how to create an elastic IP address:

Under VPC dashboard on the left hand side will click on elastic IPS.

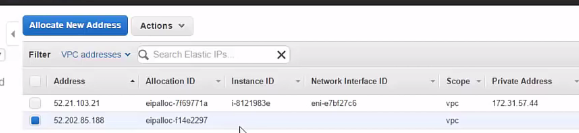
Now you'll see a list of any elastic IPS that you have associated in your account and remember any the elastic IP address that you're using that isn't allocated to something you'll be charged for..

 so I have one available and that is allocated to an instance currently .

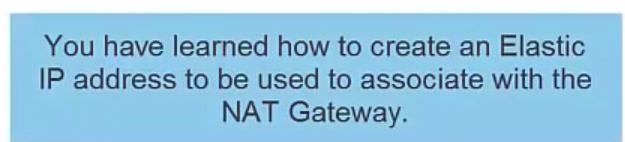
so we want to allocate a new address and **it reminds you that there's a charge if you're not using..**

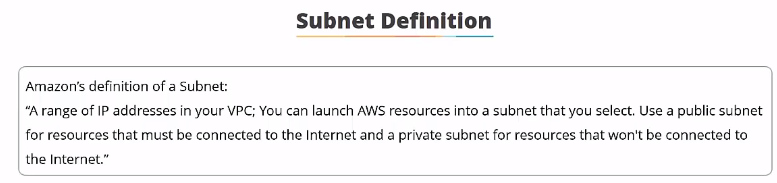


it I'm saying yes allocate and it takes a couple of seconds and there's our new elastic IP address now..



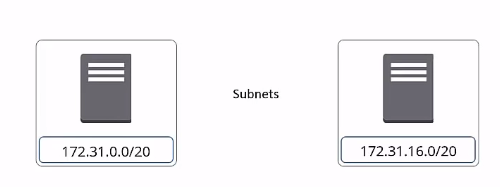
we’ll using this IP address to associate with than that gateway when we build that .



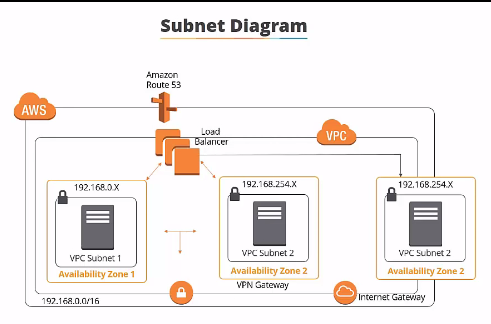
**Subnet:** AWS defines subnet asa range of IP addresses in your VPC. You can launch AWS resources into a subnet that you select…

you can use a public subnet for resources that must be connected to the Internet and

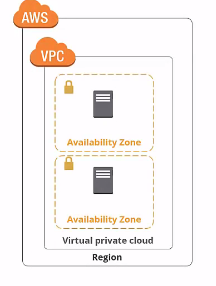
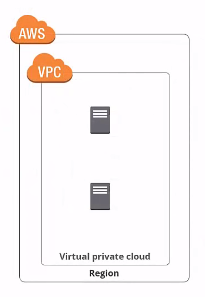
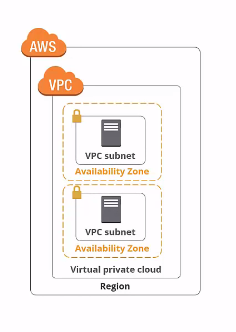
a private subnet for resources that won't be connected to the Internet.

Net mask for the default subnet in your V PC is always 20 which provides up to 4096 addresses per subnet and a few of them are reserved for AWS use .

**VPC can span multiple availability zones but the subnet is always mapped to a single availability zone this is important to know**

****

So here's our basic diagram which we're now going to start adding to so..

we can see the virtual private cloud and you can see the availability zones and now inside each availability zone we've rated a subnet..

**Now you won't be able to launch any instances unless there are subnets in your VPC so it's good to spread them across availability zones for redundancy and failover purposes ..**

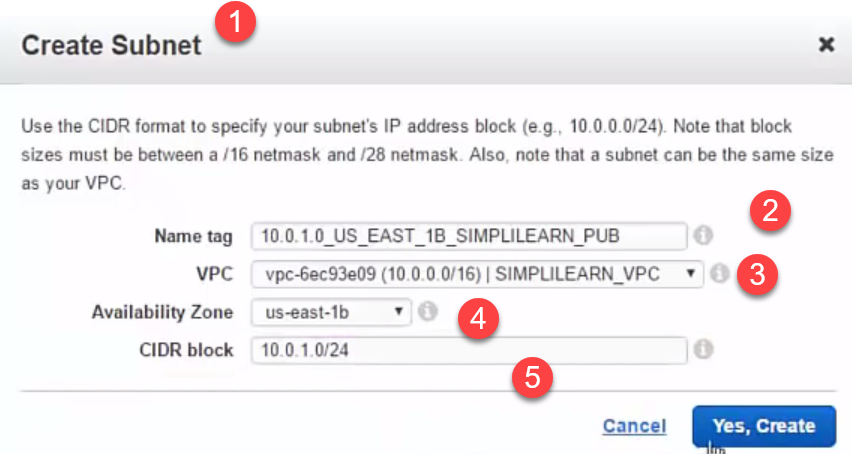
there's two **different types of subnet** public and private you use a public subnet for resources that must be connected to the Internet.

**For example:** web servers a public subnet is made public because the main route table sends two subnets traffic that is destined for the Internet to the internet gateway.

**private subnets** are for resources that don't need an internet connection or that you want to protect from the internet **for example** database instances .



So in this demonstration we're going to create some subnets a public and a private subnet and we're going to put them in our custom VPC in different availability zones ..

VPC Dashboard🡪click on subnets 🡪create subnet 

24 🡪give us two hundred and fifty-one addresses in this range

there's our new subnet and I'll just widen this you can see so that's the IP range that's the availability..

I'm going to give the private the IP address block that I'm going to put this one in u.s. the east one see and it's going to be the private sub now obviously I wanted to be in the same be PC by the bit is using of us the East 1c and we're going to give it 10.0 2.0 - 24 and will click yes create again it takes a few seconds okay let me sort by name so we can see now we've got our private subnet and our public seven air right let me just type in simply learnt that we are so now you can see them both there and you can see they're both in the same V PC simply learn V PC now if we go down to the bottom you can see the root table associated with these V pcs and you can see that they can communicate with each other internally but there's no internet access so that's what we need to do next in the next lesson you're going to learn about Internet gateways and how we can make these subnets have internet access you welcome to the networking section in this section we're going to take a look at Internet gateways route tables and NAT devices and we'll have a demonstration on how to create each of these AWS VPC items you so to allow your VPC the ability to connect to the internet you need to attach an Internet gateway and you can only attach one Internet gateway pervy pc so attaching an Internet gateway is the first stage in permitting Internet access to instances in your V PC now here's our diagram again and now we've added the Internet gateway which is providing the connection to the Internet to your V PC but before you can configure internet correctly there's a couple more steps for an ec2 instance to be internet connected you have to adhere to the following rules firstly you have to an attach an Internet gateway to your V PC which we just discussed then you need to ensure that your instances have public IP addresses or elastic IP addresses so they're able to connect to the internet then you need to ensure that your subnets root table points to the Internet gateway and you need to ensure that your network access control and security group rules allow relevant traffic to flow to and from your instance so you need to allow the rules to let in the traffic you want for example HTTP traffic after the demonstration for this section we're going to look at how route tables access control lists and security groups are used in this demonstration we're going to create an Internet gateway and attach it to our custom V PC you you so let's get a networking EPC bring up the VP see dashboard and on the left hand side we click on Internet gateways so here's a couple of Internet gateways I have already but I need to create a new one so create