var x={‘id’:20,’name’}; //you are creating an object

var y=”{‘id’:20,’name’}”; //string json string

JSON.parse(y); //y is a string

alert(x); //you will get [object:object]

alert(JSON.stringify(x));

var obj=JSON.parse(y);

for(var i=0;i<arr.length;i++)

{

}

<input type=”text” style=”color:red;background-color:blue” id=”txt” />

var obj=document.getElementById(“txt”);

obj.style.backgroundColor=”blue”;

EC2

Elastic Compute Cloud

In aws management console, we can start an EC2 instance (of any type (linux, windows, etc)

1. Go to aws.amazon.com management console
2. Search for EC2
3. In the EC2 dashboard, there are 0 instances running. Click that
4. Click Launch instance button
5. Enter the name of server
6. Choose OS. We have chosen “Windows”
7. Instance type: by default it is “t2.micro” you can choose any of the type based on the RAM size
8. Key Pair (login):

Create new key pair button

Give name of the key pair

Choose .pem

Click button to download key pair file

Now check downloads folder. There is jag5.pem file

1. Network settings (click edit)

Auto-assign public ip (it is disabled. So enable them)

Security group: select existing security group (and choose vpc)

Create instance.

1. Now in the dashboard, you can see 1 instance

First status is pending. Then changes to “Running”

1. Click the instance id hyperlink

Connect using RDP Remote desktop app.

For password, upload .pem file and generate password for administrator.

ECS

**ECS**This is the maestro of the system. The Elastic Container Service which conducts the orchestra.

**Service/Task — Application container**This is the dockerized application that you want running — to basically use the application.

**ECS Agent side car**This ECS Agent would be deployed alongside your application container. This would get instructions from ECS to spawn up or bring down the application container.

Task is where the focus is, atleast for the application owners. So everyone is watching the state of the task. The states of the tasks are named from the perspective of ECS.

**PROVISIONING**This means the infrastructure is being set up in which to run the task. ECS takes care of this part.

**PENDING**Now, ECS has provisioned compute in which to run the tasks, and is waiting for the ECS Container agent to bring up the task.

**ACTIVATING**At this point, the task is started. However, it is not yet ready to be exposed to the rest of the world. ECS has other stuff to do related to the task before the task can be called running. For e.g., registering the target groups.

**RUNNING**This means that the application task is actually running, and available to use.

**DEACTIVATING**ECS has taken charge of the task and is getting ready to stop the task.

**DEPROVISIONING**ECS is now deprovisioning the compute that it had provisoned earlier for running the task.

**STOPPED**Now, the task has been stopped fully by ECS.

**Why is my task stuck in PROVISIONING state?**  
If your task is stuck in PROVISIONING state for a while, that means ECS was unable to provision compute for the task. For example, this could mean that ECS was unable to bring up EC2 instances to run the task on.

This has happened to me when I had UserData section in the Auto Scaling Group tied to the Capacity provider for the ECS cluster where I was trying to start the task. This had happened because I was trying to mount FSX into EC2, and FSX did not have the right security group/ subnet configurations.