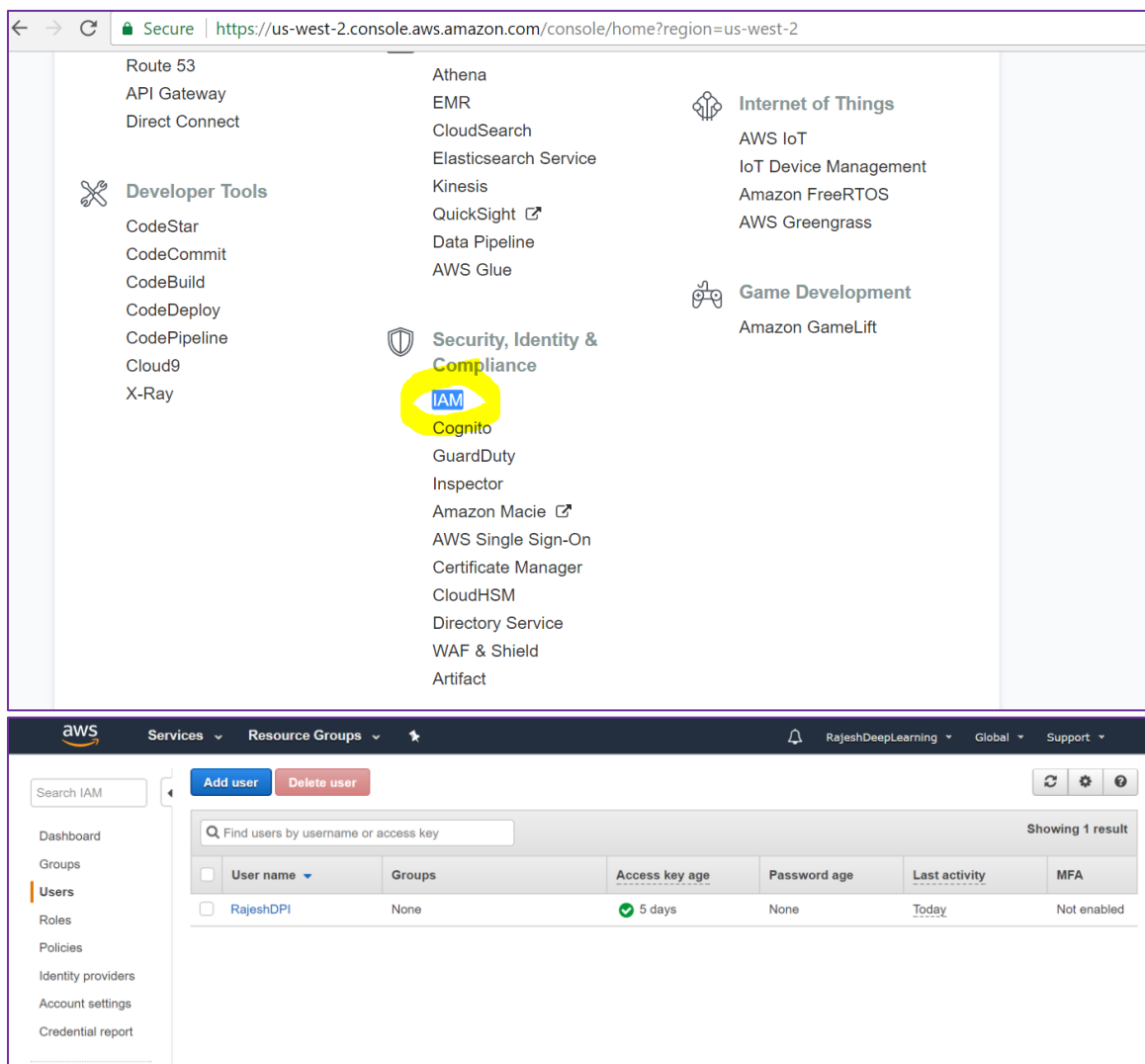


# How to setup a FREE t2.micro AWS instance for fast.ai lessons.

One of the pre-requisite for fast.ai course is set up an AWS deep learning capable machine. The website hosts a nice video on how to set up a P2 instance which has a powerful GPU. However, it cost \$0.9 per hour. For those who are cost conscious and want to set up a free instance, the option is go for a t2.micro with storage size of 30 GB.

Getting this up and running has been a tough task with lots of effort being spent on the discussion forums. This article is an effort towards simplifying the task of setting up a free t2.micro instance.

1. Sign up in AWS
2. Create a user with Admin Access in AWS->IAM->Users-> Add User



**Set user details**

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name\*

[Add another user](#)

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Access type\* ☒ **Programmatic access**  
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

☐ **AWS Management Console access**  
Enables a **password** that allows users to sign-in to the AWS Management Console.

\* Required

[Cancel](#) [Next: Permissions](#)

Attach **AdministratorAccess** to the user

Add user to group
 Copy permissions from existing user
 Attach existing policies directly

Attach one or more existing policies directly to the users or create a new policy. [Learn more](#)

[Create policy](#) [Refresh](#)

Filter: Policy type  Showing 309 results

	Policy name	Type	Attachments	Description
<input checked="" type="checkbox"/>	AdministratorAccess	Job function	1	Provides full access to AWS services and resources.

Keep the Access key ID and Secret access Key handy as its required during the AWSCLI configuration

Details Permissions Review Complete

**Success**

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://494167118960.signin.aws.amazon.com/console>

[Download .csv](#)

User	Access key ID	Secret access key
RajDeep2	AKIAISRPOLEVNR5FNKQ	pyjybQCz+vyqFgFtazGmP9AF+f5i2hcWI2jkGg <a href="#">Hide</a>

3. Install a bash shell. Cygwin is preferred one for fast.ai course. Install Cywgin from <https://cygwin.com/>
  - a. Key point to notice during installation – During Select Packages step -> View “Full” and ensure following components are installed

- i. `wget`
  - ii. `openssh`: The OpenSSH server and client programs
  - iii. Python(`python2`: Python 2 language interpreter)
  - iv. Pip (`python2-pip`: Python package installation tool)
4. Open `cygwin` and Install AWS Command line tools through `pip2.7 install awscli`
5. Configure AWS with user created in step#2. Use `aws configure`
  - a. Use the Access Key ID and Secret Access Key
  - b. Default region Name : `us-west-2`
  - c. Default output format: `text`

```
rajunnik@RAJUNNIK-6AL06 ~
$ aws
usage: aws [options] <command> [<subcommand> ...] [parameters]
To see help text, you can run:

    aws help
    aws <command> help
    aws <command> <subcommand> help
aws: error: too few arguments

rajunnik@RAJUNNIK-6AL06 ~
$ aws configure
AWS Access Key ID [None]: AKIAISRPOLEVNRAF5NKQ
AWS Secret Access Key [None]: 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
Default region name [None]: us-west-2
Default output format [None]: text

^a
```

6. Download the following files from GitHub ( Original files are modified to create a `t2.micro` instance with 30 GB)
  - i. [setup\\_t2.sh](#)
  - ii. [Setup\\_instance.sh](#)
  - iii. [Aws-alias.sh](#)

```
rajunnik@RAJUNNIK-6AL06 ~
--Fast.ai/master/setup_instance.sh content.com/rajeshunnikrishnan/DeepLearning-through-
--2017-12-22 01:27:33-- https://raw.githubusercontent.com/rajeshunnikrishnan/DeepLearning-through-Fast.ai/master/setup_instance.sh
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.0.133, 151.101.64.133, 151.101.128.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)[151.101.0.133]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 6781 (6.6K) [text/plain]
Saving to: 'setup_instance.sh'

setup_instance.sh                               100%[=====]
2017-12-22 01:27:35 (405 KB/s) - 'setup_instance.sh' saved [6781/6781]

rajunnik@RAJUNNIK-6AL06 ~
--Fast.ai/master/aws-alias.sh content.com/rajeshunnikrishnan/DeepLearning-through-
--2017-12-22 01:28:27-- https://raw.githubusercontent.com/rajeshunnikrishnan/DeepLearning-through-Fast.ai/master/aws-alias.sh
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.64.133, 151.101.128.133, 151.101.192.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)[151.101.64.133]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1627 (1.6K) [text/plain]
Saving to: 'aws-alias.sh'

aws-alias.sh                                   100%[=====]
```

7. Create AWS `t2.micro` free instance using `bash setup2.sh`
8. Keep the ipaddress handy to be used at a later point for accessing Jupiter notebook
  - a. `source aws-alias.sh`
  - b. `aws-ip` (to get the ipaddress)
9. Connect to the server using `ssh` command that was visible during the `setup2.sh` execution

```
rajesh.unnikrishnan@BDC7-L-6523564 ~
$ ssh -i /home/rajesh.unnikrishnan/.ssh/aws-key-fast-ai.pem ubuntu@ec2-35-163-27-247.us-west-2.compute.amazonaws.com
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-1013-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

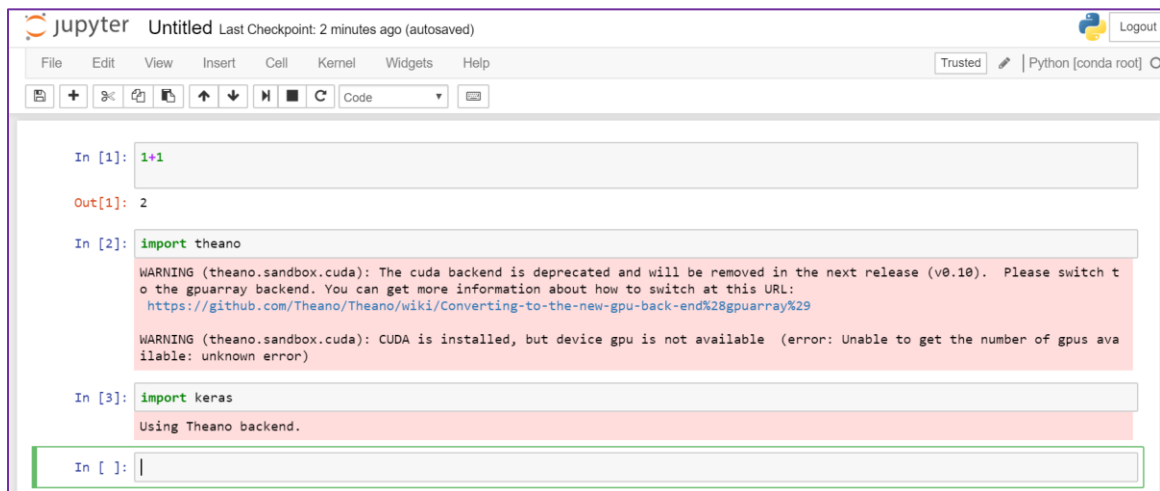
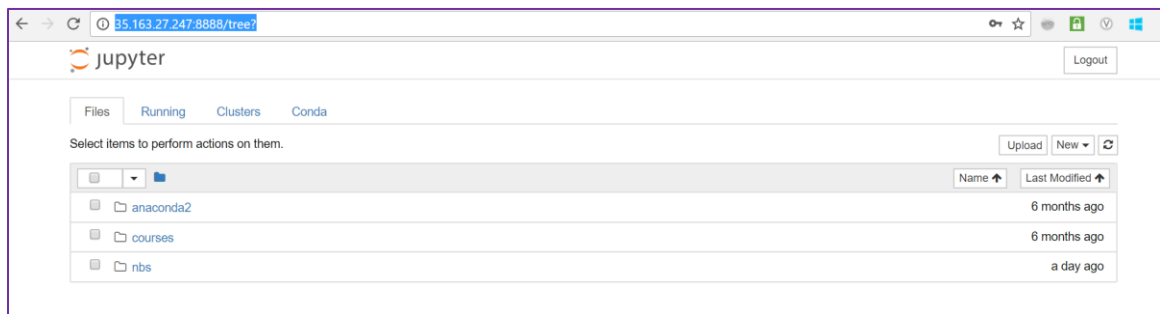
248 packages can be updated.
9 updates are security updates.

*** System restart required ***
Last login: Thu Dec 21 02:09:07 2017 from 122.172.81.143
ubuntu@ip-10-0-0-6:~$
```

## 10. Run Jupiter notebook in the aws machine by *jupyter notebook*

```
ubuntu@ip-10-0-0-6:~$ jupyter notebook
[I 20:18:14.326 NotebookApp] [nb_conda_kernels] enabled, 2 kernels found
[I 20:18:14.330 NotebookApp] Writing notebook server cookie secret to /run/user/1000/jupyter/notebook_cookie_secret
[W 20:18:15.116 NotebookApp] WARNING: The notebook server is listening on all IP addresses and not using encryption. This is not recommended.
[I 20:18:15.816 NotebookApp] [nb_anacondacloud] enabled
[I 20:18:15.816 NotebookApp] [nb_conda] enabled
[I 20:18:15.906 NotebookApp] ✓ nbpresent HTML export ENABLED
[W 20:18:15.906 NotebookApp] x nbpresent PDF export DISABLED: No module named nbbrowserpdf.exporters.pdf
[I 20:18:15.909 NotebookApp] Serving notebooks from local directory: /home/ubuntu
[I 20:18:15.909 NotebookApp] 0 active kernels
[I 20:18:15.909 NotebookApp] The Jupyter Notebook is running at: http://[all ip addresses on your system]:8888/
[I 20:18:15.909 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
```

## 11. Open the Jupiter notebook using [http://\[ip address\]:8888](http://[ip address]:8888) . use ip address fetched from step#8. Use dl\_course as the password for accessing the notebook



**YOU ARE ALL SET TO START THE DEEPLARNING COURSE. HAPPY LEARNING**