

# Setting up Deep Learning Environment on AWS & GCP

CSCI 599  
2019-02-12

**AWS & GCP are not free!**

**GPU Instances are expensive!**

**You are responsible for all the billings!**

**Even if you are not running processes, they will charge you if your machine is running.**

**Remember to shutdown/terminate your machines when not using them.**

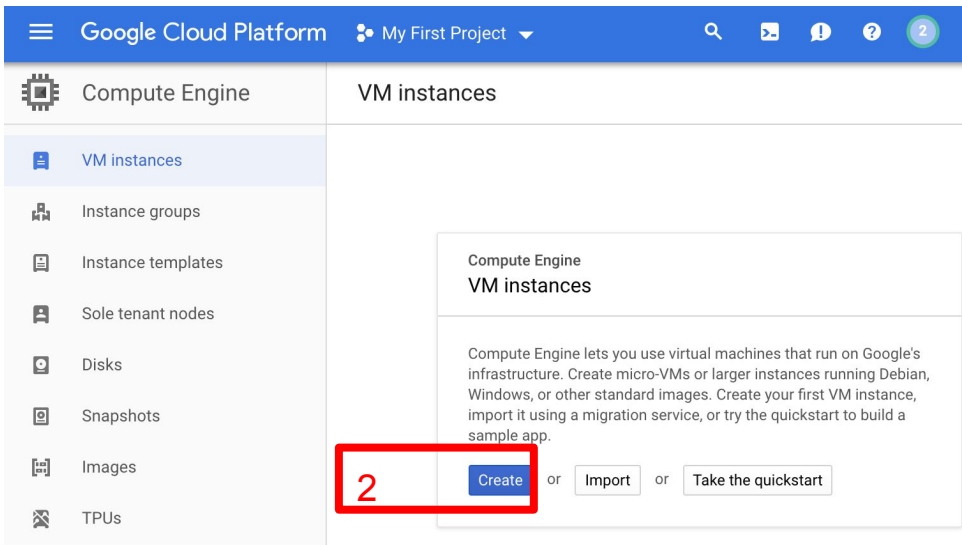
# In case you cannot create a GPU instance

- By default, AWS/GCP don't allow you to create GPU instances (GPU limits = 0)
- You need to increase the GPU limits
- The instructions are

[https://docs.google.com/presentation/d/1iZQ\\_KuwdYDdZkpBjmWRahP1NzIPFS8he-qOySqRPumQ/edit#slide=id.p](https://docs.google.com/presentation/d/1iZQ_KuwdYDdZkpBjmWRahP1NzIPFS8he-qOySqRPumQ/edit#slide=id.p)

# GCP

- Create a project if you haven't
- Go to **"Compute Engine"**
- Click **"Create"** to create a VM



Google Cloud Platform My First Project

Compute Engine VM instances

VM instances

Instance groups

Instance templates

Sole tenant nodes

Disks

Snapshots

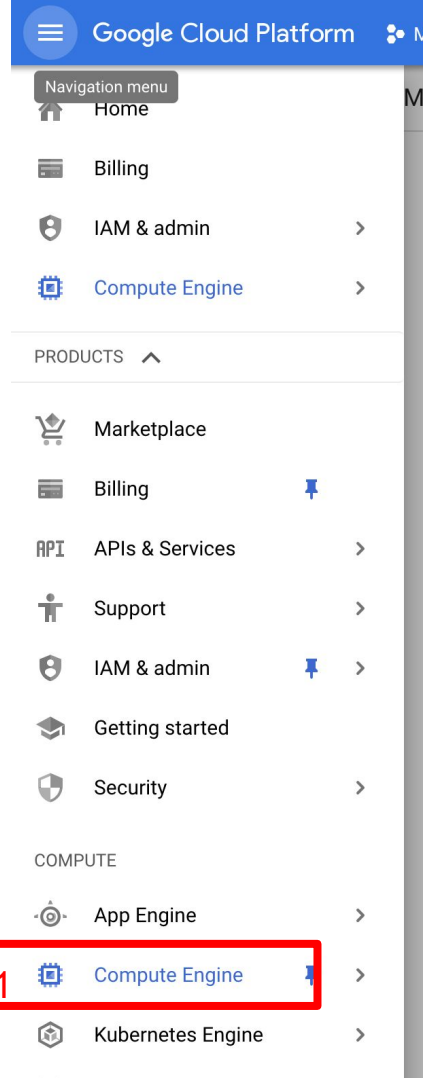
Images

TPUs

Compute Engine VM instances

Compute Engine lets you use virtual machines that run on Google's infrastructure. Create micro-VMs or larger instances running Debian, Windows, or other standard images. Create your first VM instance, import it using a migration service, or try the quickstart to build a sample app.

2 Create or Import or Take the quickstart



Google Cloud Platform

Navigation menu

Home

Billing

IAM & admin

Compute Engine

PRODUCTS

Marketplace

Billing

APIs & Services

Support

IAM & admin

Getting started

Security

COMPUTE

App Engine

1 Compute Engine

Kubernetes Engine

# GCP

- Click “Marketplace”
- search “deep learning”
- choose “Deep Learning VM”

The screenshot displays the Google Cloud Platform (GCP) Marketplace interface. The top navigation bar shows the GCP logo and a search bar containing the text "deep learning". The left sidebar, titled "Create an instance", lists three options: "New VM instance", "New VM instance from template", and "Marketplace". The "Marketplace" option is highlighted with a red box and a red arrow. The main content area shows the search results for "deep learning", with a red box around the search bar and another red box around the "Deep Learning VM" result. The "Deep Learning VM" result is the first item in the list, featuring the NVIDIA logo and the text "Deep Learning VM" and "Google Click to Deploy".

Google Cloud Platform

deep learning

Create an instance

To create a VM instance, select one of the options:

- New VM instance  
Create a single VM instance from scratch
- New VM instance from template  
Create a single VM instance from an existing template
- Marketplace**  
Deploy a ready-to-go solution onto a VM instance

Marketplace > "deep learning"

### Virtual machines

Filter by

TYPE

Virtual machines

CATEGORY

- Analytics (7)
- Big data (1)
- Compute (10)
- Databases (1)
- Developer stacks (1)
- Developer tools (1)
- Machine learning (13)

PRICE


- Free (15)
- Paid (1)
- BYOL (1)

17 results

- 3** **Deep Learning VM**  
Google Click to Deploy  
Intel® optimized and GPU-ready machine learning frameworks
- NVIDIA GPU Cloud Image for Deep Learning and HPC**  
NVIDIA  
Optimized for GPU-Accelerated Containers
- MXNet 1 Python 3.6 CPU Production**  
Jetware  
MXNet, an open-source deep learning framework
- MXNet 1 Python 3.6 NVidia GPU Production**  
Jetware  
Fully integrated software stack with MXNet for NVidia GPU
- AISE PyTorch CPU Notebook**  
Jetware  
Minimal web lab with PyTorch and Jupyter Notebook
- AISE TensorFlow CPU Notebook**  
Jetware  
Minimal web lab with TensorFlow and Jupyter Notebook

# GCP

- click “LAUNCH ON COMPUTE ENGINE”



## Deep Learning VM

[Deep Learning VM \(Google Click to Deploy\)](#)

Estimated costs: \$294.45/month | 1,000+ recent deployments

Intel® optimized and GPU-ready machine learning frameworks

[LAUNCH ON COMPUTE ENGINE](#)

### Runs on

Google Compute Engine

### Type

[Virtual machines](#)

Single VM

### Last updated

2/12/19, 11:07 AM

### Category

[Compute](#)

[Developer tools](#)

### Overview

Deploy a Compute Engine instance with your favorite machine learning framework configured to support common GPU workloads out of the box. This deployment setting up a high-performance computing environment: the latest NVIDIA and latest Intel® libraries (Intel® MKL-DNN/MKL) are all ready to go, along with also includes support for both python2 and python3 with key packages for pandas, and nltk. Currently, Intel® optimized TensorFlow 1.12.0, PyTorch 1.1.0, TensorFlow 2.0, Chainer 5.0.0, XGBoost 0.81, and MXNet 1.3 are supported (dependent on usage). Other frameworks can be installed on top of the Ubuntu base images, which include the common set of NVIDIA and python libraries and packages.

[Learn more](#)

# GCP

- if you want, you can
  - update the name
  - update the zone
  - update number of CPUs
  - change disk size
  - change network
- you can also
  - change the number of GPUs and the GPU type
  - change a different framework
- **remember to select**
  - Beta. Enable access via URL...
  - Install NVIDIA GPU Driver...
- review the terms and then click **“Deploy”** at the bottom of the page

Google Cloud Platform My First Project

## New Deep Learning VM deployment

**Deployment name**  
tensorflow-1

**Zone** ⓘ  
GPU availability is limited to certain zones. [Learn more](#) ⓘ  
us-central1-c

**Machine type** ⓘ  
2 vCPUs 13 GB memory [Customize](#)

**GPUs**  
The number of GPU dies is linked to the number of CPU cores and memory selected for this instance. For this machine type, you can select no fewer than 1 GPU die. [Learn more](#)  
**Number of GPUs** 1 **GPU type** NVIDIA Tesla K80  
Machines with GPUs can't migrate on host maintenance

**Framework**  
Choose the primary machine learning framework you will be using. If the library you would like to use is not listed, choose the base image, which provides core packages.  
Intel® optimized TensorFlow 1.12 (with Intel® MKL-DNN/MKL and CUDA 1...

**Access to the Jupyter Lab**  
☒ **Beta. Enable access via URL instead of SSH** ⓘ  
Enabling this Beta feature allows you to access your JupyterLab instance using a URL. Anyone who is in the Editor or Owner role in your GCP project can access this URL. This feature is available only in the US, EU and Asia.

**GPU**  
☒ **Install NVIDIA GPU driver automatically on first startup?** ⓘ  
I want to use NVIDIA GPUs with this image. Please fetch NVIDIA GPU drivers from a third-party location and install them on my behalf (requires internet access on the VM).

# GCP

your server is being deployed (it takes time)

Google Cloud PlatformMy First Project

Deployment Manager

DeploymentsType registry

← tensorflow-1STOPDELETE

tensorflow-1 is being deployedView details

Overview - tensorflow-1

tensorflow tensorflow.jinja

tensorflow-vm-tmpl vm\_instance.py

tensorflow-1-vm vm instance

software-status software\_status.py

tensorflow-1-config config

tensorflow-1-software config walter

tensorflow

tensorflow has resource warnings  
tensorflow-1-vm: Disk size: '100 GB' is larger than image size: '30 GB'. You might need to resize the root repartition manually if the operating system does not support automatic resizing. See <https://cloud.google.com/compute/docs/disks/persistent-disks#repartitionrootpd> for details.

Deep Learning VM

Solution provided by Google Click to Deploy

Instance	Pending
Instance zone	Pending
Instance machine type	Pending

More about the software

Get started with Deep Learning VM

You will be able to use Deep Learning VM after the deployment is completed.

Documentation

Official Documentation

StackOverflow: Deep Learning VM

Google Group: Deep Learning VM

Support

If you have non-framework related issues, you can bring them up at the Deep Learning VM [Stack Overflow](#).

Template properties

More

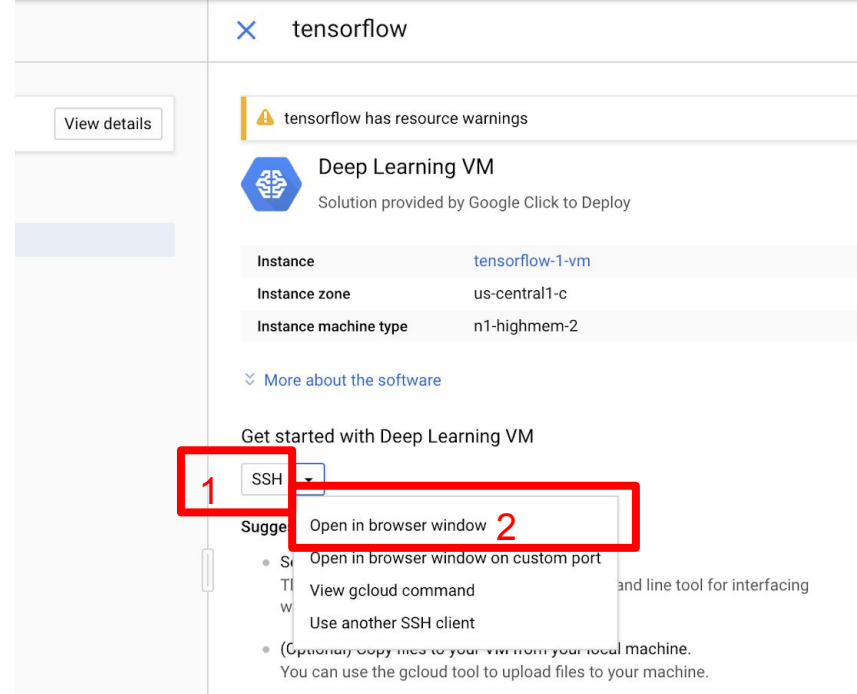


# GCP

Once it's done, you can connect to your server by

- click the “**SSH**” button under “Get started with Deep Learning VM”
- click “**Open in browser window**”

You can also find the SSH button in the “**VM instance detail**” page



# GCP

After you connect successfully,

- type “nvidia-smi” if you are using a GPU instance and you can see
- type “python3 -V” to check the python version
- then you can upload your code and run your scripts.

```
please use the binaries that are pre-built for this image. You can find the binaries at
/opt/deeplearning/binaries/tensorflow/
If you need to install a different version of Tensorflow manually, use the command
n Deep Learning image with the
right version of CUDA

Linux tensorflow-1-vm 4.9.0-8-amd64 #1 SMP Debian 4.9.130-2 (2018-10-27) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
hanpeng_liu_cs@tensorflow-1-vm:~$ nvidia-smi
Tue Feb 12 23:59:56 2019

+-----+
| NVIDIA-SMI 410.72                Driver Version: 410.72                CUDA Version: 10.0                |
+-----+
| GPU   Name           Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan   Temp  Perf    Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |
|=====+-----+
|    0   Tesla K80           Off    | 00000000:00:04:0 Off |                    0 |
| N/A   33C    P0       62W / 149W |  0MiB / 11441MiB |   100%    Default   |
+-----+
+-----+
| Processes:                       GPU Memory |
|   GPU   PID     Type    Process name      Usage   |
|=====+-----+
|   No running processes found               |
+-----+
hanpeng_liu_cs@tensorflow-1-vm:~$ python3 -V
Python 3.5.3
```

# Tips

- It's highly recommended to run your code in **tmux** or **screen**, so that you can detach the window and reattach to the terminal window later.
- It's highly recommended to save checkpoints when you train your model, so that they can be resumed after unexpected program halts.
- To upload files to GCP, you can check the tutorial given by Google <https://cloud.google.com/compute/docs/instances/transfer-files>

# GCP Jupyter Notebook

- The VM instance has set up a jupyter notebook,
- in order to access it, you need to
  - [install Google Cloud SDK](#)
  - once installed, run “gcloud auth login”
  - run the corresponding command shown in the deployment page
  - it will give you a link
  - open the link in browser
    - if it shows login error, you can try open it in a private window

× tensorflow

⚠ tensorflow has resource warnings

tensorflow-1-vm: Disk size: '100 GB' is larger than image size: '30 GB'. You might not support automatic resizing. See <https://cloud.google.com/compute/docs/di>



## Deep Learning VM

Solution provided by Google Click to Deploy

Instance	tensorflow-1-vm
Instance zone	us-central1-c
Instance machine type	n1-highmem-2

🔽 [More about the software](#)

Get started with Deep Learning VM

SSH

### Suggested next steps

1

- Set up the Cloud SDK.  
The Cloud SDK (gcloud) is the preferred command line tool for interfacing with your instance. [Download it here.](#)
- (Optional) Copy files to your VM from your local machine.  
You can use the gcloud tool to upload files to your machine.

```
$ gcloud compute scp --project organic-area-231418 --zone us-ce
```

2

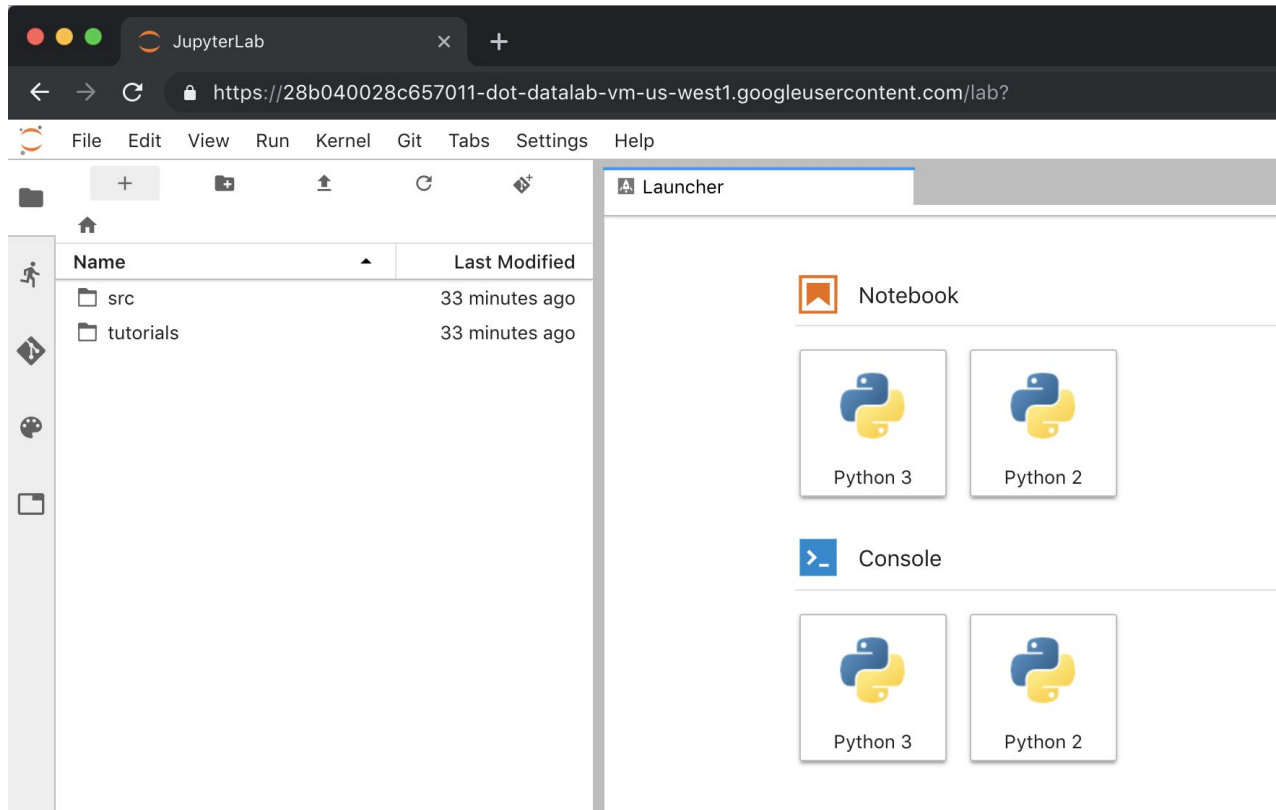
- Access the running Jupyter notebook.  
We've already started a Jupyter notebook instance on the VM for your convenience. In order to get link that can be used to access Jupyter Lab run the following command.

```
$ gcloud compute instances describe --project organic-area-2314
```

- Assign a static external IP address to your VM instance.  
An ephemeral external IP address has been assigned to the VM instance. If you require a static external IP address, you may promote the address to static. [Learn more](#)

# GCP Jupyter Notebook

You should be able to open the JupyterLab website,



**AWS & GCP are not free!**

**GPU Instances are expensive!**

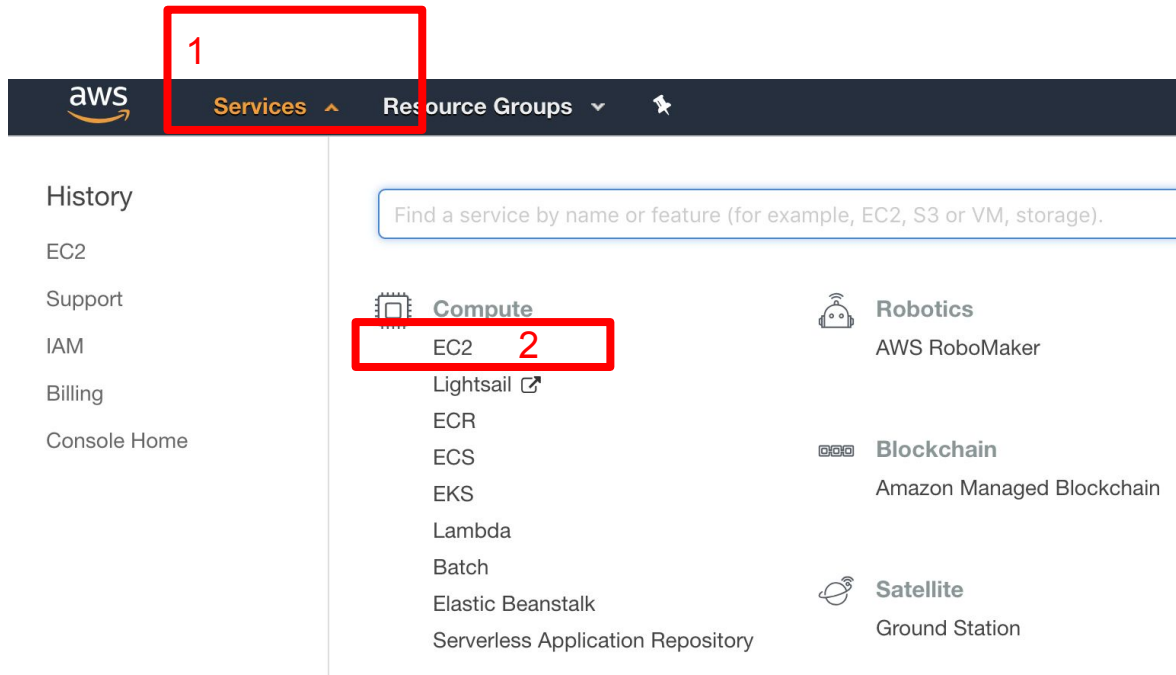
**You are responsible for all the billings!**

**Even if you are not running processes, they will charge you if your machine is running.**

**Remember to shutdown/terminate your machines when not using them.**

# AWS

- login to AWS console
- click “**Services**”
- then click “**EC2**”



# AWS

- hover the left button to “Support” in the banner,
- click the desired region, for example, Oregon

The screenshot shows the AWS Management Console interface. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and a user profile 'hanpeng-liu-c'. The 'Oregon' region is selected in the top right, and a 'Support' button is next to it. A red box labeled '1' highlights the 'Support' button. The left sidebar shows the 'EC2 Dashboard' with links to 'Events', 'Tags', 'Reports', 'Limits', and 'INSTANCES'. The main content area is titled 'Resources' and displays a list of EC2 resources in the 'US West (Oregon)' region: 0 Running Instances, 0 Elastic IPs, 0 Dedicated Hosts, 0 Snapshots, 0 Volumes, 0 Load Balancers, 1 Key Pairs, and 2 Security Groups. A red box labeled '2' highlights the 'US West (Oregon)' region in the dropdown menu.

aws Services Resource Groups

hanpeng-liu-c Oregon Support

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Resources

You are using the following Amazon EC2 resources in the US West (Oregon) region:

- 0 Running Instances
- 0 Elastic IPs
- 0 Dedicated Hosts
- 0 Snapshots
- 0 Volumes
- 0 Load Balancers
- 1 Key Pairs
- 2 Security Groups

US East (N. Virginia)

US East (Ohio)

US West (N. California)

**US West (Oregon)**

Asia Pacific (Mumbai)

Asia Pacific (Seoul)



# AWS

- click “Launch Instance”

The screenshot shows the AWS Management Console interface. On the left is a navigation sidebar with the following items: EC2 Dashboard (selected), Events, Tags, Reports, Limits, INSTANCES (with a sub-menu: Instances, Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations), IMAGES (with a sub-menu: AMIs, Bundle Tasks). The main content area is titled 'Resources' and displays a summary of EC2 resources in the US West (Oregon) region: 0 Running Instances, 0 Elastic IPs, 0 Dedicated Hosts, 0 Snapshots, 0 Volumes, 0 Load Balancers, 1 Key Pairs, 2 Security Groups, and 0 Placement Groups. Below this is a light blue box with a link to 'EC2 Videos'. The 'Create Instance' section follows, with the text 'To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.' A red box with the number '1' highlights the 'Launch Instance' button. At the bottom, a note states: 'Note: Your instances will launch in the US West (Oregon) region'.

EC2 Dashboard

- Events
- Tags
- Reports
- Limits
- INSTANCES
  - Instances
  - Launch Templates
  - Spot Requests
  - Reserved Instances
  - Dedicated Hosts
  - Scheduled Instances
  - Capacity Reservations
- IMAGES
  - AMIs
  - Bundle Tasks

## Resources

You are using the following Amazon EC2 resources in the US West (Oregon) region:

0 Running Instances	0 Elastic IPs
0 Dedicated Hosts	0 Snapshots
0 Volumes	0 Load Balancers
1 Key Pairs	2 Security Groups
0 Placement Groups	

Learn more about the latest in AWS Compute from AWS re:Invent by viewing the [EC2 Videos](#).

## Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

**1** Launch Instance

Note: Your instances will launch in the US West (Oregon) region

# AWS

- type “**gpu**” in the search bar and press the enter key
- select the “**Deep Learning Base AMI (Ubuntu) Version 15.0**”

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

## Step 1: Choose an Amazon Machine Image (AMI)

[Cancel and Exit](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

1 gpu

Quick Start (2)

My AMIs (0)

AWS Marketplace (97)

Community AMIs (236)

☐ Free tier only ⓘ

Deep Learning Base AMI (Ubuntu) Version 15.0 - ami-0eca2297483cebc64

Comes with foundational platform of Nvidia CUDA, cuDNN, NCCL, GPU Drivers, Intel MKL-DNN and other system libraries to deploy your own custom deep learning environment. For a fully managed experience, check: <https://aws.amazon.com/sagemaker>

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

2 Select

64-bit (x86)

Deep Learning Base AMI (Amazon Linux) Version 16.1 - ami-0549811cd38764ef9

Comes with foundational platform of Nvidia CUDA, cuDNN, NCCL, GPU Drivers, Intel MKL-DNN and other system libraries to deploy your own custom deep learning environment. For a fully managed experience, check: <https://aws.amazon.com/sagemaker>

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Amazon Linux

Select

64-bit (x86)

# AWS

- if (1) you have GPU limits updated and (2) you want to create a GPU machine, select **p2.xlarge** or p2.8xlarge or p2.16xlarge
  - Otherwise, you can choose other instances such as “t2.small”
- Click **“Review and Launch”**

<input checked="" type="checkbox"/>	<a href="#">GPU instances</a>	p2.xlarge	4	61	EBS only	Yes	Hi
<input type="checkbox"/>	<a href="#">GPU instances</a>	p2.8xlarge	32	488	EBS only	Yes	10 Gi
<input type="checkbox"/>	<a href="#">GPU instances</a>	p2.16xlarge	64	732	EBS only	Yes	25 Gi
<input type="checkbox"/>	<a href="#">GPU instances</a>	p3.2xlarge	8	61	EBS only	Yes	Up to 10
<input type="checkbox"/>	<a href="#">GPU instances</a>	p3.8xlarge	32	244	EBS only	Yes	10 Gi

[Cancel](#)

[Previous](#)

[Review and Launch](#)

# AWS

- review and if everything is fine, click “Launch”

## Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.



**Your instance configuration is not eligible for the free usage tier**

To launch an instance that's eligible for the free usage tier, check your AMI selection, instance type, configuration options, or storage devices. Learn more about [free usage tier](#) eligibility and usage restrictions.



[Don't show me this again](#)

### AMI Details

[Edit AMI](#)



#### Deep Learning Base AMI (Ubuntu) Version 15.0 - ami-0eca2297483cebc64

Comes with foundational platform of Nvidia CUDA, cuDNN, NCCL, GPU Drivers, Intel MKL-DNN and other system libraries to deploy your own custom deep learning environment. For a fully managed experience, check: <https://aws.amazon.com/sagemaker>

Root Device Type: ebs    Virtualization type: hvm

### Instance Type

[Edit instance type](#)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
p2.xlarge	11.75	4	61	EBS only	Yes	High

### Security Groups

[Edit security groups](#)

Security group name

launch-wizard-2

Description

launch-wizard-2 created 2019-02-12T15:08:11.912-08:00

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ
--------	------------	--------------	----------	---------------

*This security group has no rules*

### Instance Details

[Edit instance details](#)

[Cancel](#)

[Previous](#)

[Launch](#)

# AWS

- choose “**Create a new key pair**”
- type the key pair name, e.g. “key”
- click “**Download Key Pair**”
- click “**Launch Instances**”

Then you will see your instance is launching, click “**View Instances**”

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about managing existing key pairs from a public AMI.

Choose an existing key pair

✓ Create a new key pair

Proceed without a key pair

key

Download Key Pair

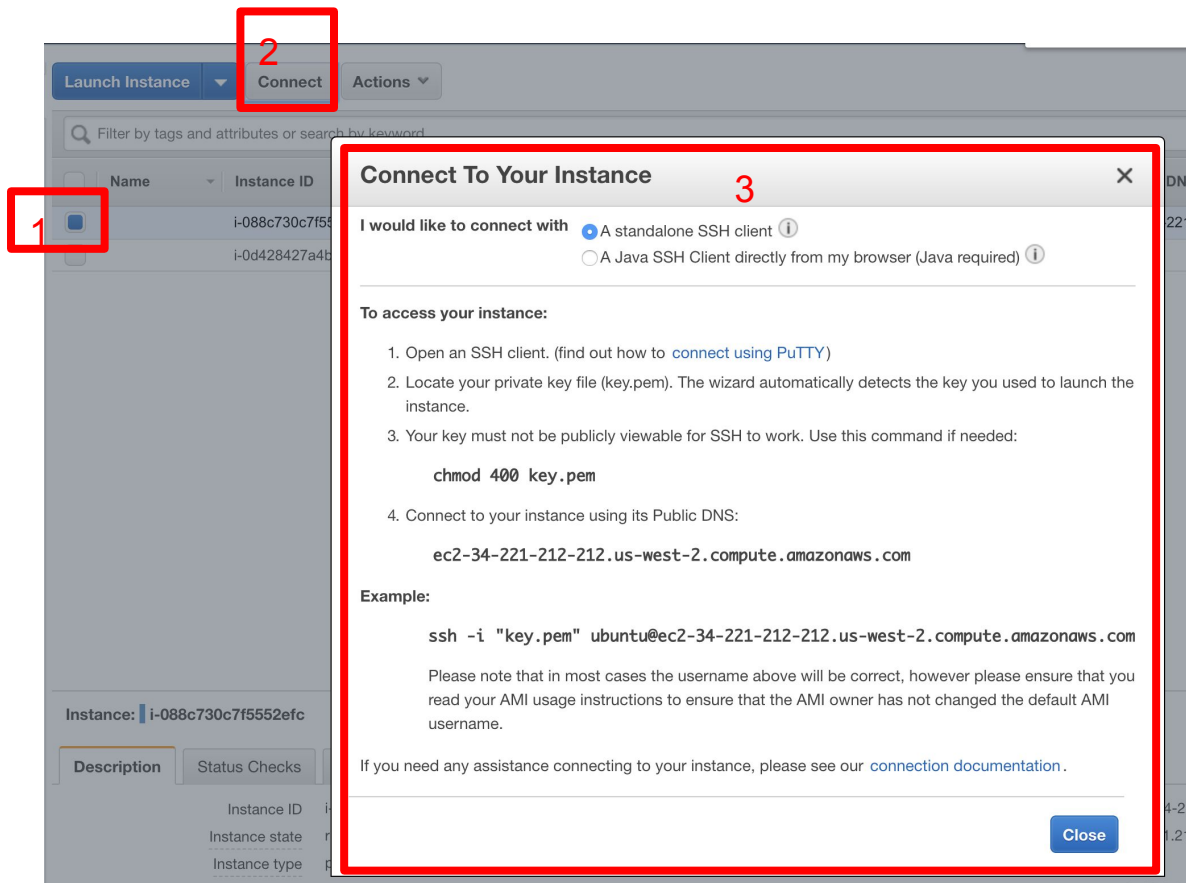
You have to download the **private key file** (\*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Cancel

Launch Instances

# AWS

- select the instance you just create
- click “**connect**”
- choose a way suggested to connect your server



# AWS

If you choose to use terminal SSH, assume your key file is stored in “~/Downloads/key.pem”

- first, go to the folder where your keys are,
  - e.g. “cd ~/Downloads”
- then, change the permission of the key file
  - e.g., “chmod 400 key.pem”
- then, connect through SSH
  - e.g., `ssh -i "key.pem" ubuntu@ec2-34-221-212-212.us-west-2.compute.amazonaws.com`
-

# AWS

After you connect successfully,

- type “nvidia-smi” if you are using a GPU instance and you can see
- type “python3 -V” to check the python version
- then you can upload your code and run your scripts.

```
ubuntu@ip-172-31-42-170:~$ nvidia-smi
```

```
Tue Feb 12 23:18:23 2019
```

-----											
NVIDIA-SMI 410.79			Driver Version: 410.79			CUDA Version: 10.0					
-----											
GPU	Name	Persistence-M	Bus-Id	Disp.A	Volatile	Uncorr.	ECC				
Fan	Temp	Perf	Pwr:Usage/Cap	Memory-Usage	GPU-Util	Compute M.					
=====											
0	Tesla K80	On	00000000:00:1E.0	Off			0				
N/A	43C	P8	27W / 149W	0MiB / 11441MiB	0%	Default					
-----											

-----				
Processes:				GPU Memory
GPU	PID	Type	Process name	Usage
=====				
No running processes found				
-----				

```
ubuntu@ip-172-31-42-170:~$ python3 -V
```

```
Python 3.5.2
```

```
—
```



# AWS

you can upload your files through “scp”, “rsync”, or “Filezilla”, or other programs.

Useful links:

- <https://angus.readthedocs.io/en/2014/amazon/transfer-files-between-instance.html>
- <https://stackoverflow.com/questions/18169455/uploading-file-to-aws-from-local-machine>
- <https://www.google.com/search?q=how+to+upload+files+to+aws+ec2&oq=how+to+upload+files+to+aws+ec2&aqs=chrome..69i57j0l2.4679j0j4&sourceid=chrome&ie=UTF-8>

# Tips

- It's highly recommended to run your code in **tmux** or **screen**, so that you can detach the window and reattach to the terminal window later.
- It's highly recommended to save checkpoints when you train your model, so that they can be resumed after unexpected program halts.
- If you want to setup Jupyter Notebook, you can check the AWS's tutorial <https://docs.aws.amazon.com/dlami/latest/devguide/setup-jupyter.html>

**AWS & GCP are not free!**

**GPU Instances are expensive!**

**You are responsible for all the billings!**

**Even if you are not running processes, they will charge you if your machine is running.**

**Remember to shutdown/terminate your machines when not using them.**