

# Setup guide for Google Cloud Platform

This guide explains how to set up Google Cloud Platform (GCP) to use PyTorch 1.0.0 and fastai 1.0.2. At the end of this tutorial you will be able to use both in a GPU-enabled Jupyter Notebook environment.

## 1. Sign up on Google Cloud Platform Free Tier.

<https://cloud.google.com/free/>

## 2. Upgrade to a paid account (credit card required).

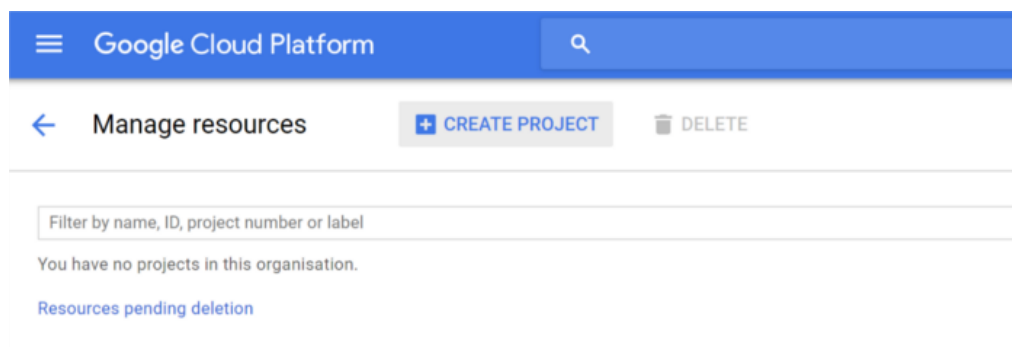
How do I upgrade to a paid account?

You can upgrade from the free trial to a paid account through the Google Cloud Platform Console. Click the **Upgrade** button at the top of the page. If you do not see **Upgrade**, click **Free trial status** in the upper-right of the page and **Upgrade** will appear. You must be a Billing Administrator on the account to make this change.

**Caution:** With an upgraded account, you will be automatically charged after your free credits are fully used or after your credits expire, whichever comes first.

If you are unable to find an Upgrade button at this point, just continue with the next steps. The Upgrade button should appear at Step 5 below when you request for an increase in quota.

## 3. Create a new project for the deep learning course course



Click on "Create Project".

#### 4. Request for increase in quota for GPU

IAM & Admin → Quotas

**Kommentar [DHN1]:** If you can't increase your quota, skip this part first and continue with step 7: create new instance. You won't be able to add a GPU, but now you can request for an increase in quota for GPU .

#### 5. In filter type, select metric to be GPUs (all regions) and Location as Global.

Kontingenttyp	Dienst	Messwert	Zone	
Alle Kontingente	Alle Dienste	GPUs (all regions)	Global	Löschen

Dienst	Zone	Aktuelle Nutzung	Spitzenzeiten in den letzten 7 Tagen	Limit
Compute Engine API GPUs (all regions)	Global	0	—	1

Click edit quotas and select the quota to edit (GPUs All Regions). Set the new quota limit to 1 or more. Your request may require confirmation, which Google claims typically takes two business days to get.

#### 6. Receive email approval of quota increase

**Google Compute Engine Quota Support** <esupport@google.com>  
an mich ▾

🌐 Englisch ▾ > Deutsch ▾ [Nachricht übersetzen](#)

Hello,

Your **quota** request for project '534733051528' has been approved and your **quota** has been adjusted accordingly.

Changed **Quota**:

```
+-----+-----+
| GLOBAL Attribute | GPUS_ALL_REGIONS |
+-----+-----+
| Changes | 0 -> 1 |
+-----+-----+
```

Please visit  
<https://console.cloud.google.com/iam-admin/quotas?project=534733051528&service=compute.googleapis.com>  
to review your updated **quota**.

Happy Computing!

Google Cloud Platform Support  
<http://support.google.com/enterprisehelp/>

## 7. Navigate to the Compute Engine -> VM-instances

Create new instance

Give a name for your instance

Region: us-central1, Zone: us-central1-c

Machine type: n1-highmem-8

GPU type: NVIDIA Tesla P4

**Kommentar [DHN2]:** Image should show the Test P4

Name <sup>?</sup>

instance-1

Region <sup>?</sup> Zone <sup>?</sup>

us-central1 (Iowa) us-central1-c

Machine configuration

Machine family

General-purpose Compute-optimized

Machine types for common workloads, optimized for cost and flexibility


Generation

First

Powered by Skylake CPU platform or one of its predecessors

Machine type

n1-highmem-8 (8 vCPU, 52 GB memory)

 vCPU Memory

8 52 GB

CPU platform <sup>?</sup>

Automatic

GPU type Number of GPUs

NVIDIA Tesla K80 1 X

☐ Enable Virtual Workstation (NVIDIA GRID)

Boot disk: Deep Learning Image: PyTorch 1.1.0 and fastai m30

Boot disk type: Standard persistent disk with 200GB

## Boot disk

Select an image or snapshot to create a boot disk; or attach an existing disk

[OS images](#)   Application images   Custom images   Snapshots   Existing disks

- ☐ built on 2019-06-17
- ☐ SUSE Linux Enterprise Server 15 SP1For SAP x86\_64  
built on 2019-06-25
- ☐ Windows Server version 1803 Datacenter Core for Containers  
Server Core, x64 built on 20190620
- ☐ Windows Server version 1803 Datacenter Core  
Server Core, x64 built on 20190620
- ☐ Windows Server version 1809 Datacenter Core for Containers  
Server Core, x64 built on 20190620
- ☐ Windows Server version 1809 Datacenter Core  
Server Core, x64 built on 20190620
- ☐ Windows Server 2008 R2 Datacenter  
Server with Desktop Experience, x64 built on 20190620
- ☐ Windows Server 2012 R2 Datacenter Core  
Server Core, x64 built on 20190620
- ☐ Windows Server 2012 R2 Datacenter  
Server with Desktop Experience, x64 built on 20190620
- ☐ Windows Server 2016 Datacenter Core  
Server Core, x64 built on 20190620
- ☐ Windows Server 2016 Datacenter  
Server with Desktop Experience, x64 built on 20190620
- ☐ Windows Server 2019 Datacenter Core for Containers  
Server Core, x64 built on 20190620
- ☐ Windows Server 2019 Datacenter Core  
Server Core, x64 built on 20190620
- ☐ Windows Server 2019 Datacenter for Containers  
x64 built on 20190620
- ☐ Windows Server 2019 Datacenter  
Server with Desktop Experience, x64 built on 20190620
- ☐ Deep Learning Image: Base m30 (with CUDA 10.0)  
A Debian based image with CUDA 10.0.
- ☒ Deep Learning Image: PyTorch 1.1.0 and fastai m30  
PyTorch 1.1.0 (and fastai) with CUDA 10.0 and Intel® MKL-DNN, Intel® MKL.
- ☐ Deep Learning Image: TensorFlow 1.14.0 m30  
TensorFlow 1.14.0 with CUDA 10.0 and Intel® MKL-DNN, Intel® MKL.
- ☐ Debian GNU/Linux 9 Stretch + TF 1-11  
A Debian linux image with Tensorflow Version 1-11 pre-installed and optimized for Cloud TPUs.
- ☐ Debian GNU/Linux 9 Stretch + TF 1-12  
A Debian linux image with Tensorflow Version 1-12 pre-installed and optimized for Cloud TPUs.
- ☐ Debian GNU/Linux 9 Stretch + TF 1-13  
A Debian linux image with Tensorflow Version 1-13 pre-installed and optimized for Cloud TPUs.
- ☐ Debian GNU/Linux 9 Stretch + TF 1-14-1-dev20190508  
A Debian linux image with Tensorflow Version 1-14-1-dev20190508 pre-installed and optimized for Cloud TPUs.
- ☐ Debian GNU/Linux 9 Stretch + TF 1-14-1-dev20190518  
A Debian linux image with Tensorflow Version 1-14-1-dev20190518 pre-installed and optimized for Cloud TPUs.
- ☐ Debian GNU/Linux 9 Stretch + TF 1-14-1-dev20190619  
A Debian linux image with Tensorflow Version 1-14-1-dev20190619 pre-installed and optimized for Cloud TPUs.
- ☐ Debian GNU/Linux 9 Stretch + TF 1-14  
A Debian linux image with Tensorflow Version 1-14 pre-installed and optimized for Cloud TPUs.
- ☐ Debian GNU/Linux 9 Stretch + TF nightly  
A Debian linux image with Tensorflow Version nightly pre-installed and optimized for Cloud TPUs.
- ☐ Debian GNU/Linux 9 Stretch + PyTorch/XLA  
A Debian linux image with PyTorch/XLA pre-installed and optimized for Cloud TPUs.

Can't find what you're looking for? Explore hundreds of VM solutions in [Marketplace](#)

Boot disk type ?

Size (GB) ?

Standard persistent disk ▼

200

Firewall: Allow HTTP and HTTPS traffic

Preemptibility: On

#### Firewall

Add tags and firewall rules to allow specific network traffic from the Internet

- ☒ Allow HTTP traffic
- ☒ Allow HTTPS traffic

Management   Security   Disks   Networking   Sole Tenancy

#### Description (Optional)

#### Labels (Optional)

#### Deletion protection

- ☐ Enable deletion protection  
When deletion protection is enabled, instance cannot be deleted. [Learn more](#)

#### Reservations

Use an existing reservation when creating this VM instance

#### Automation

##### Startup script (Optional)

You can choose to specify a startup script that will run when your instance boots up or restarts. Startup scripts can be used to install software and updates, and to ensure that services are running within the virtual machine. [Learn more](#)

#### Metadata (Optional)

You can set custom metadata for an instance or project outside of the server-defined metadata. This is useful for passing in arbitrary values to your project or instance that can be queried by your code on the instance. [Learn more](#)

Key	Value	
-----	-------	---

#### Availability policy

##### Preemptibility

A preemptible VM costs much less, but lasts only 24 hours. It can be terminated sooner due to system demands. [Learn more](#)

Price of your VM instance should be about \$0.327 hourly

## 8. Create a static IP address

Click on the three dots besides your created VM instance and select 'View network details'


Navigate to external IP addresses and select reserve a static address



### Reserve a static address

Name 

test123

Description (Optional)

Network Service Tier 


- ☒ Premium (Current project-level tier, [change](#)) 
- ☐ Standard 

IP version

- ☒ IPv4
- ☐ IPv6

Type

- ☒ Regional
- ☐ Global (to be used with Global forwarding rules [Learn more](#))


Region 

us-central1 (Iowa)

Attached to 

Some of the instances may be disabled due to the 'External IPs for VM instances' organization policy. [Learn more](#)

None

 Static IP addresses not attached to an instance or load balancer are billed at an hourly rate [Pricing details](#)

Reserve

Cancel

Equivalent [REST](#) or [command line](#)

Reserve and click on change to attach IP address to your instance

## Attach IP address

**IP address**

test1234 (34.68.228.225)

**Attach to** ?

Some of the instances may be disabled due to the 'External IPs for VM instances' organization policy. [Learn more](#)

instance-1test (Zone us-central1-c)

⚠ The static IP address currently attached to instance instance-1test (Zone us-central1-c) will be detached. Static IP addresses not attached to an instance or load balancer are billed at an hourly rate [Pricing details](#)

[CANCEL](#)
[OK](#)

### 9. Connect to the CM instance through SSH from the browser:

<input type="checkbox"/> Name ^	Zone	Recommendation	Internal IP	External IP	Connect
<input checked="" type="checkbox"/> fastai-instance-1	asia-east1-a		10.140.0.2	35.185.140.217 <a href="#">↗</a>	<div> SSH <div> Open in browser window Open in browser window on custom port View gcloud command Use another SSH client </div> </div>

### 10. Install Nvidia driver

### 11. Configure Jupyter Notebook Server

Type in terminal:

```
jupyter notebook --generate-config
```

```
cd .jupyter
```

open your jupyter notebook config with editor of your chose, e.g. vim jupyter\_notebook\_config.py

```
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.

This VM requires Nvidia drivers to function correctly. Installation takes ~1 minute.  
Would you like to install the Nvidia driver? [y/n] y  
Installing Nvidia driver.  
Downloading driver from GCS location gs://nvidia-drivers-us-public/tesla/410.104/NVIDIA-Linux-x86\_64-410.104.run  
Verifying archive integrity... OK  
Uncompressing NVIDIA Accelerated Graphics Driver for Linux-x86\_64 410.104.....  
.....  
.....  
.....  
.....  
  
WARNING: The nvidia-drm module will not be installed. As a result, DRM-KMS will not function with this  
installation of the NVIDIA driver.

WARNING: nvidia-installer was forced to guess the X library path '/usr/lib' and X module path  
'/usr/lib/xorg/modules'; these paths were not queryable from the system. If X fails to find the NVIDIA X  
driver module, please install the 'pkg-config' utility and the X.Org SDK/development package for your  
distribution and reinstall the driver.

WARNING: Unable to find a suitable destination to install 32-bit compatibility libraries. Your system may not be  
set up for 32-bit compatibility. 32-bit compatibility files will not be installed; if you wish to install  
them, re-run the installation and set a valid directory with the --compat32-libdir option.

Nvidia driver installed.  
name@instance-test12:~\$ jupyter notebook --generate-config  
Writing default config to: /home/name/.jupyter/jupyter\_notebook\_config.py  
name@instance-test12:~\$ cd ~/.jupyter  
-bash: cd: ~/.jupyter: No such file or directory  
name@instance-test12:~\$ cd .jupyter  
name@instance-test12:~/.jupyter\$ vim jupyter\_notebook\_config.py  
name@instance-test12:~/.jupyter\$ █

add somewhere following lines:

```
c.NotebookApp.ip = '0.0.0.0'
```

```
c.NotebookApp.port = 8888
```

```
# Configuration file for jupyter-notebook.
#-----
# Application(SingletonConfigurable) configuration
#-----

## This is an application.

## The date format used by logging formatters for %(asctime)s
#c.Application.log_datefmt = '%Y-%m-%d %H:%M:%S'

## The Logging format template
#c.Application.log_format = '[%(name)s]%(highlevel)s %(message)s'

## Set the log level by value or name.
#c.Application.log_level = 30

#-----
# JupyterApp(Application) configuration
#-----

c.NotebookApp.ip = '0.0.0.0'
c.NotebookApp.port = 8888

## Base class for Jupyter applications

## Answer yes to any prompts.
#c.JupyterApp.answer_yes = False

## Full path of a config file.
#c.JupyterApp.config_file = ''

## Specify a config file to load.
#c.JupyterApp.config_file_name = ''

## Generate default config file.
#c.JupyterApp.generate_config = False

#-----
# NotebookApp(JupyterApp) configuration
#-----

"jupyter notebook config.py" 769L, 29412C

24,0-1
```



## 12. Start Jupyter Notebook

Type in terminal jupyter notebook and copy your token

Type in your browser address bar "external-ip-of-your\_instance:8888"

Paste your Token and create new Password

Password or token:

### Token authentication is enabled

If no password has been configured, you need to open the notebook server with its login token in the URL, or paste it above. This requirement will be lifted if you [enable a password](#).

The command:

```
jupyter notebook list
```

will show you the URLs of running servers with their tokens, which you can copy and paste into your browser. For example:

```
Currently running servers:  
http://localhost:8888/?token=c8de56fa... :: /Users/you/notebooks
```

or you can paste just the token value into the password field on this page.

See [the documentation on how to enable a password](#) in place of token authentication, if you would like to avoid dealing with random tokens.

Cookies are required for authenticated access to notebooks.

### Setup a Password

You can also setup a password by entering your token and a new password on the fields below:

Token

New Password

**13. Important: Remember to stop your VM instance when you are done with your notebooks or you will continue to incur charges.**

VM instances

 CREATE INSTANCE



 IMPORT VM



Filter VM instances



Name ^

Zone

Recommendation

Internal IP

External IP

Connect



 fastai-instance-1

asia-east1-a

10.140.0.2

35.185.140.217 

SSH



- Start
- Stop
- Reset
- Delete
- New instance group
- View logs