

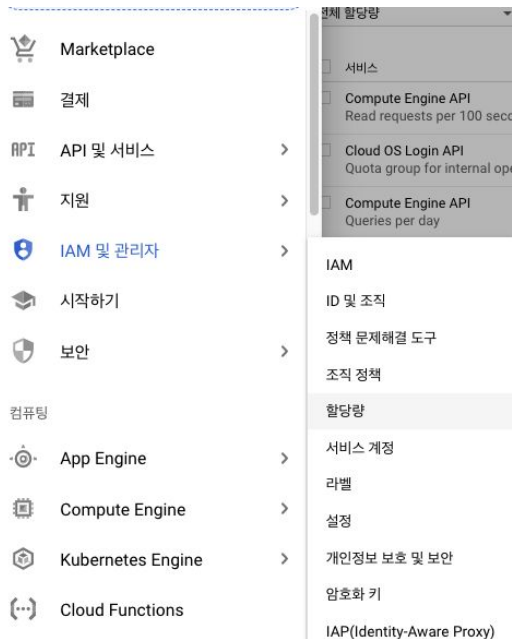
GCP 사용하귀이이

JOMINHO

가입은... explain useless...?



GPU 할당량 신청-1



할당량

+ 할당량 수정

할당량 유형

전체 할당량

서비스

서비스 0개

측정항목

전체 측정항목

지정된 옵션으로 제공되는 할당량이

compu

모두 선택 선택 해제

Compute Engine API

할당량 유형

전체 할당량

서비스

Compute Engine API

측정항목

측정항목 0개

지정된 옵션으로 제공되는 할당량이 없습니다.

gpu

모두 선택 선택 해제

Committed NVIDIA K80 GPUs

Committed NVIDIA P100 GPUs

Committed NVIDIA P4 GPUs

Committed NVIDIA T4 GPUs

Committed NVIDIA V100 GPUs

GPUs (all regions)

NVIDIA K80 GPUs

NVIDIA P100 GPUs

NVIDIA P100 Virtual Workstation GPUs

NVIDIA P4 GPUs

NVIDIA P4 Virtual Workstation GPUs

NVIDIA T4 GPUs

NVIDIA T4 Virtual Workstation GPUs

NVIDIA V100 GPUs

Preemptible NVIDIA K80 GPUs

Preemptible NVIDIA P100 GPUs

반드시 GPUs(all regions)체크
원하는 gpu 체크하고
할당량 수정

할당량
들어가귀

GPU 할당량 신청-1

✕ 할당량 1개 선택

Compute Engine API

할당량: GPUs (all regions)

새 할당량 한도
새 할당량 한도를 입력하세요. 승인을 위해 해당 요청이 서비스 제공업체에 전송됩니다. ?

요청 설명
필수 ?

giv it to me....

완료

취소

요청 제출

뒤로

quota 올려달라고 요청
(보통 2일 정도 걸린다 뵙)

인스턴스 만들기

홈

고정하면 여기에 표시됩니다.

Marketplace

결제

API 및 서비스

지원

IAM 및 관리자

시작하기

보안

컴퓨팅

App Engine

Compute Engine

VM 인스턴스

인스턴스 그룹

인스턴스 템플릿

단독 테넌트 노드

디스크

스냅샷

이미지

TPU

약정 사용 할인

메타데이터

상태 확인

영역

네트워크 엔드포인트 그룹

작업

보안 스캔

설정

VM 인스턴스

[인스턴스 만들기](#)[VM 가져오기](#)[새로고침](#)[시작](#)[중지](#)[재설정](#)

열

<input type="checkbox"/>	이름 ^	영역	권장사항	다음에서 사용	내부 IP	외부 IP	연결
<input type="checkbox"/>	instance-2	us-central1-a			10.128.0.5 (nic0)	35.188.15.99 ↗	SSH ▾ ⋮
<input type="checkbox"/>	instance-4	us-central1-a			10.128.0.7 (nic0)	34.68.228.247 ↗	SSH ▾ ⋮
	instance-5					없음	

인스턴스 만들기 클릭

인스턴스 만들기

리전은 us-central로 하는게 좋다.

화살표로 닫혀있는 GPU 할당을 열어서 원하는 gpu 할당

부팅 디스크는 deep learning 되어있는것 중에 필요한 library에 따라 선택

바하버 http https 트래픽 필요

- ☐ Server with Desktop Experience, x64 built on 20191210
- ☐ GPU Optimized Debian m32 (with CUDA 10.0)
A Debian 9 based image with CUDA/CuDNN/NCCL pre-installed
- ☐ Deep Learning Image: Base m38 (with CUDA 10.0)
A Debian based image with CUDA 10.0.
- ☐ Deep Learning Image: PyTorch 1.2.0 and fastai m36
PyTorch 1.1.0 (and fastai) with CUDA 10.0 and Intel® MKL-DNN, Intel® MKL.
- ☐ Deep Learning Image: PyTorch 1.3.0 and fastai m39
PyTorch 1.3.0 (and fastai) with CUDA 10.0 and Intel® MKL-DNN, Intel® MKL.
- ☐ Deep Learning Image: TensorFlow 1.15.0 m42
TensorFlow 1.15.0 with CUDA 10.0 and Intel® MKL-DNN, Intel® MKL.
- ☐ Debian GNU/Linux 9 Stretch - TF 1.11

이름은 영구적입니다.

instance-6

리전 [?] 리전은 영구적입니다.

us-central1(미아오와)

영역 [?] 영역은 영구적입니다.

us-central1-a

마신 구성 [?]

마신 계열

일반 용도 메모리 최적화

일반적인 작업 부하에 적합한 마신 유형이며 가격 및 유연성을 위해 최적화되었습니다.

시리즈

N1

Intel Skylake CPU 플랫폼 또는 이전 버전의 플랫폼에서 제공

마신 유형

n1-standard-1(vCPU 1개, 3.75GB 메모리)

 vCPU 1 메모리 3.75GB

CPU 플랫폼 [?] CPU 플랫폼 구성은 영구적입니다.

자동

GPU 유형 GPU 수

NVIDIA Tesla K80 1 X

☐ 가상 워크스테이션 사용 설정(NVIDIA GRID)

디스플레이 기기

스크린 캡처 및 녹화 도구를 사용하면 디스플레이 기기를 사용 설정하세요.

☐ 디스플레이 기기 사용 설정

⚡ CPU 플랫폼 및 GPU

컨테이너 [?]

☐ 이 VM 인스턴스에 컨테이너 이미지를 배포합니다. 자세히 알아보기

부팅 디스크 [?]

 새로운 30GB 표준 영구 디스크 이미지

gpu 활성화하기

<input type="checkbox"/> 이름 ^	영역	권장사항	다음에서 사용	내부 IP	외부 IP	연결	
<input type="checkbox"/>  instance-2	us-central1-a			10.128.0.5 (nic0)	35.188.15.99 ↗	SSH ▾	⋮
<input type="checkbox"/>  instance-4	us-central1-a			10.128.0.7 (nic0)	34.68.228.247 ↗	SSH ▾	⋮
 instance-5					없음		↺ 🗑
 instance-6					없음		↺ 🗑

이위 gpu가 깔려 있긴 한데 활성화를 해야해요. 오른쪽에 ssh 버튼 눌러서 instance terminal 열기!

열면 nvidia 깔꺼냐 물어보는데 yes만 해주면 됩니다아(이미 만들어서 정확한 문구를 모르겠.. 알려주시면 수정하겠습니다아)

gcloud 연결하기

(윈도우 컴퓨터가 없어서 윈도우 doc 참조해서 적긴 하는데 이슈 있으면
말해주세요)

윈도우 <https://cloud.google.com/sdk/docs/quickstart-windows?hl=ko>

맥 <https://cloud.google.com/sdk/docs/quickstart-macos?hl=ko>

위에 URL 들어가서 시작하기 전에 보고 따라하귀

gcloud 연결하기

윈도우든 mac이든 터미널 같이 생긴창이 떠야 함(윈도우는 확인 필요)

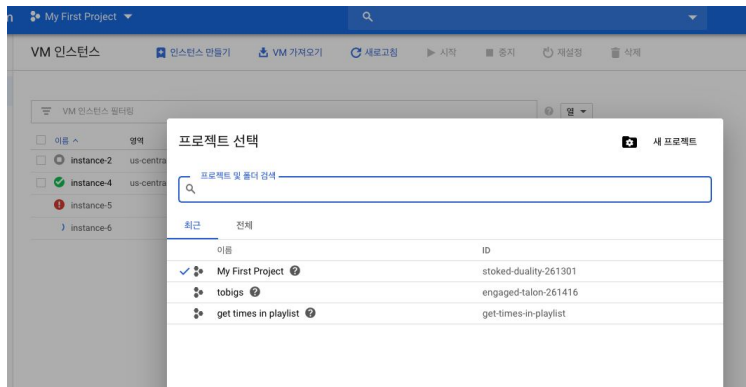
gcloud init

1. 로그인을 일단 하라고 할거예요. 로그인을 하기

2. 프로젝트 선택하기

(상단 메뉴바에 **project** 이름 누르면 ID가 보임
인스턴스 만든 프로젝트 선택)

3. wldur 선택하기(우리의 경우 us-central1)



gcloud로 인스턴스 열기

1. 일단 포트 8081로 뚫기(인스턴스 이름은 처음 만들때 쓴 이름) - 이걸 내 컴퓨터에서 실행

```
gcloud compute ssh instance-2(instance 이름)
--zone=us-central1-a --ssh-flag="-L
8081:localhost:8081"
```

2. 인스턴스에 들어가서 아래 command 실행

```
jupyter notebook --no-browser --allow-root
--NotebookApp.allow_origin='https://colab.research.google.co
m' --NotebookApp.token=''
--NotebookApp.disable_check_xsrf=True --port=8081
```

jupyter notebook 실행하기

내 컴퓨터 크롬에서(뭐 댄것도 노상관) localhost:8081 치면 짤 하고 뜹니다. 이젠 쓰면 됨.

여기서 jupyter notebook 만들고 git clone하고 파일 업로드 하고 하면 되겠죠??

