

Demo: Demo on creating Kubernetes cluster

Problem statement: Commands to create a Kubernetes cluster using GKE.

Solution:

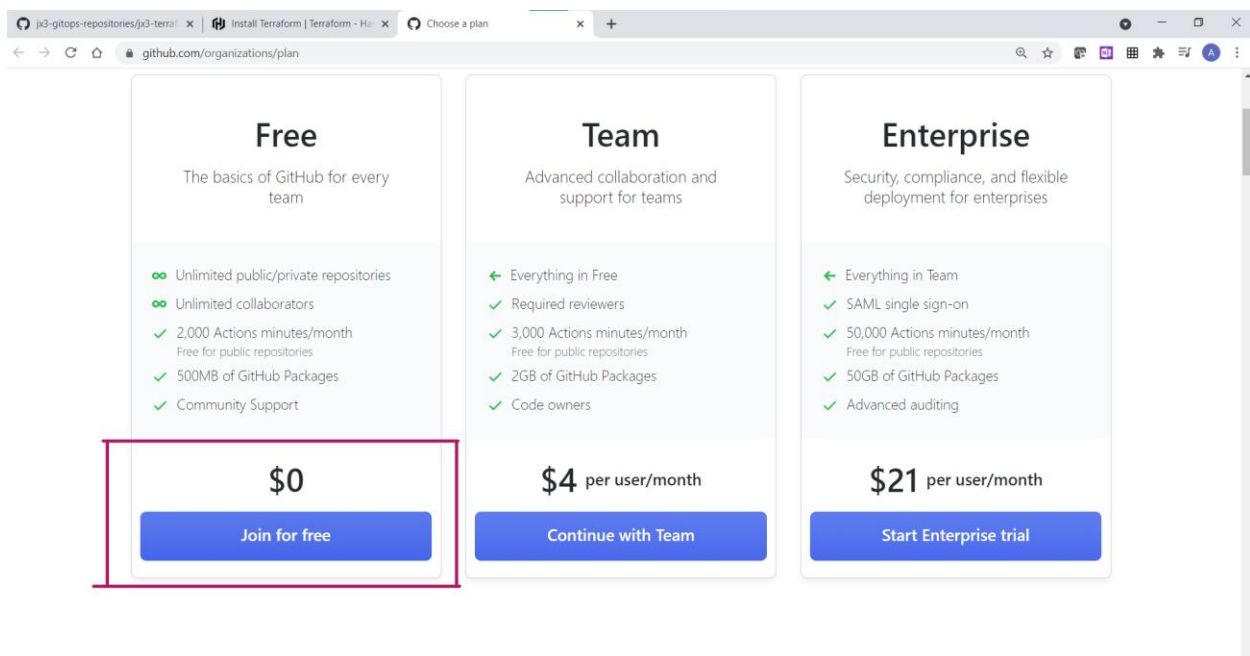
Step 1:

Please log in to github.com

URL to access: <https://github.com/organizations/plan>

You need to create git organization. This will be used by Jenkins X to create GitOps repositories.

Note: You also need to create another GitHub user (we will call it as a bot user). We will be adding this user to the organization.



CICD using Jenkins X – Part 1

jk3-github-repositories/jk3-terra x | Install Terraform | Terraform - H x | Set up your organization x +

github.com/account/organizations/new?coupon=&plan=team_free

Tell us about your organization

Set up your organization

Organization account name *

stepaheadinfo ✓

This will be the name of your account on GitHub.
Your URL will be: <https://github.com/stepaheadinfo>.

Contact email *

abhishek.ghode@gmail.com ✓

This organization belongs to: *

☒ My personal account
I.e., abhishekghode

☐ A business or institution
For example: GitHub, Inc., Example Institute, American Red Cross

Verify your account

GitHub - Where software is built x | learn x | Invite your team members x +

github.com/organizations/stepaheadinfo1/invite

Incognito

Search or jump to... Pull requests Issues Marketplace Explore

Start collaborating

Welcome to stepaheadinfo1

Add organization members

Organization members will be able to view repositories, organize into teams, review code, and tag other members using @mentions.

[Learn more about permissions for organizations →](#)

Search by username, full name or email address

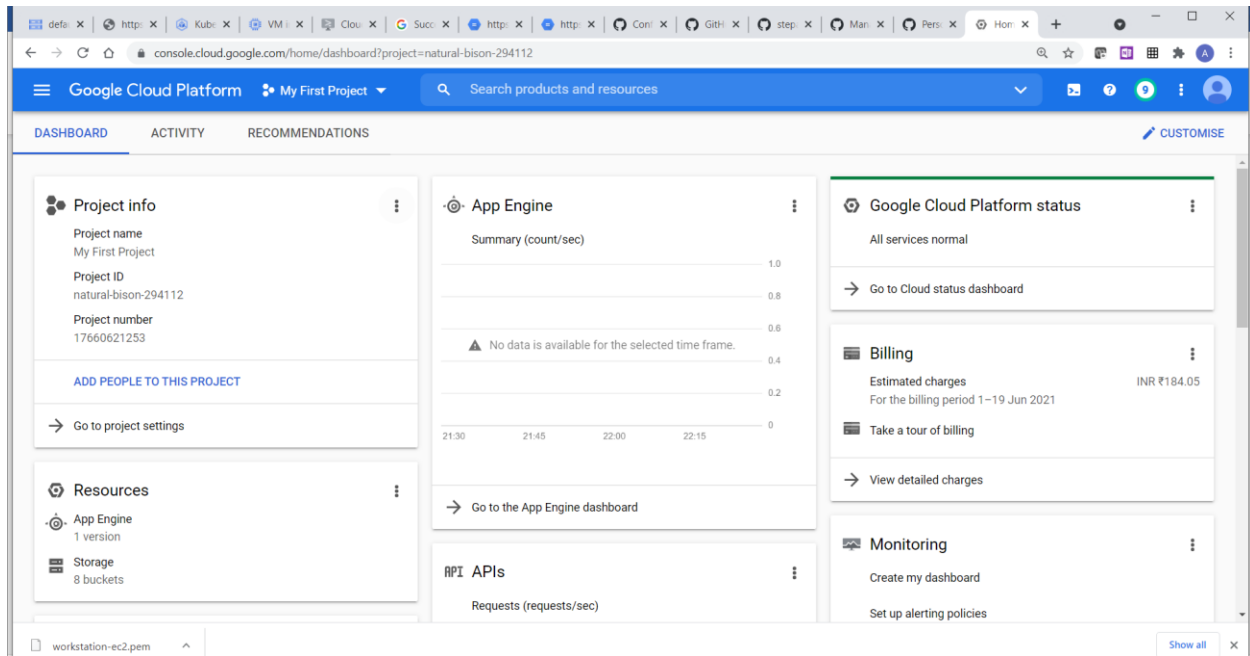
BOT-ABHI
BOT-ABHI

Complete setup

CICD using Jenkins X – Part 1

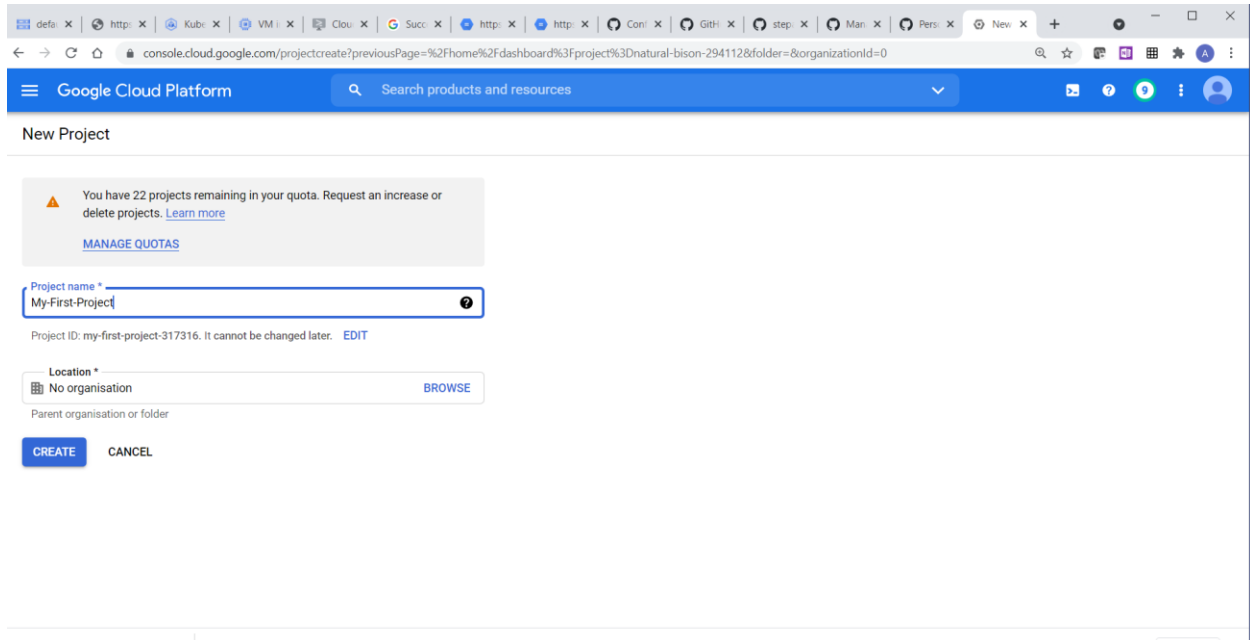
Step2: Create a GKE cluster.

URL to access: console.cloud.google.com



CICD using Jenkins X – Part 1

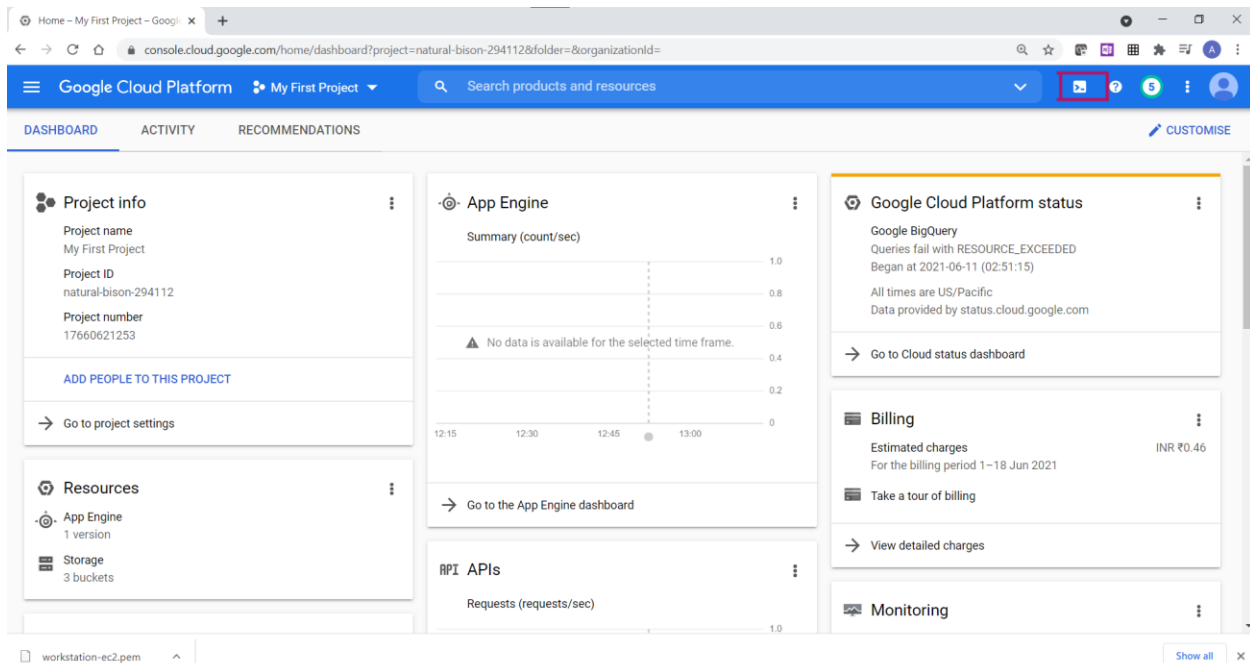
You would need to create a new project with billing enabled.



The screenshot shows the 'New Project' page in the Google Cloud Platform console. At the top, there's a navigation bar with the Google Cloud Platform logo and a search bar. Below the navigation bar, the page title is 'New Project'. A warning message states: 'You have 22 projects remaining in your quota. Request an increase or delete projects. [Learn more](#)'. Below this, there's a 'Project name' field with the value 'My-First-Project' and a 'Project ID' field with the value 'my-first-project-317316'. The 'Location' is set to 'No organisation'. At the bottom, there are 'CREATE' and 'CANCEL' buttons.

Click on 'Activate cloud shell'.

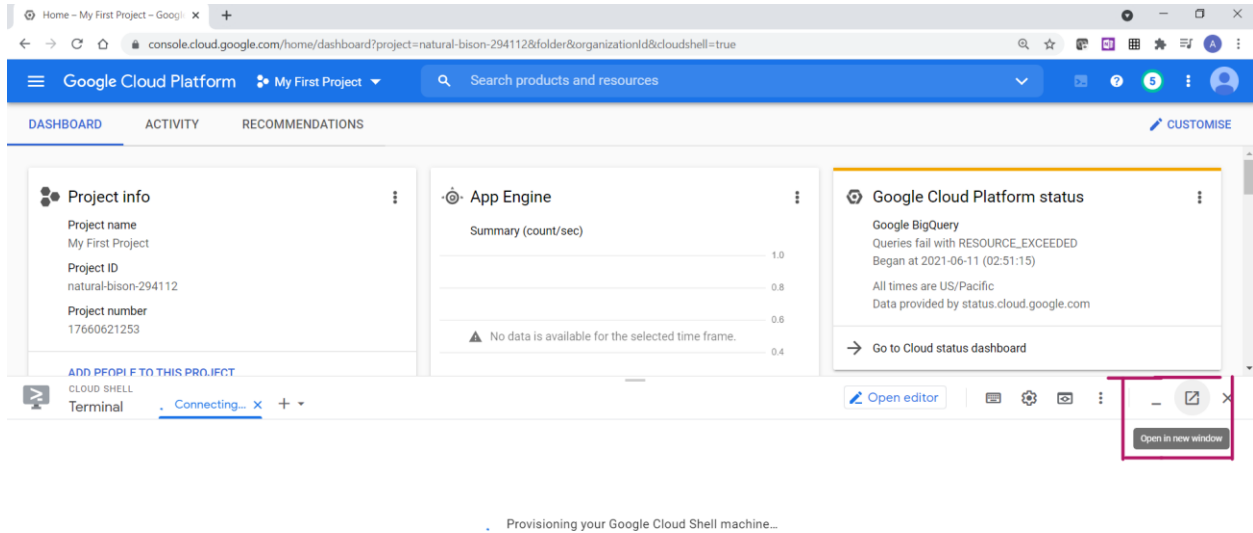
This will be our workstation on GCP platform.



The screenshot shows the Google Cloud Platform dashboard for the 'My First Project'. The dashboard is divided into several sections: 'Project info' (Project name: My First Project, Project ID: natural-bison-294112, Project number: 17660621253), 'Resources' (App Engine: 1 version, Storage: 3 buckets), 'App Engine' (Summary (count/sec) chart), 'Google Cloud Platform status' (Google BigQuery status), 'Billing' (Estimated charges: INR ₹0.46), and 'Monitoring' (RPI APIs: Requests (requests/sec) chart). A red box highlights the 'Activate cloud shell' button in the top right corner of the dashboard.

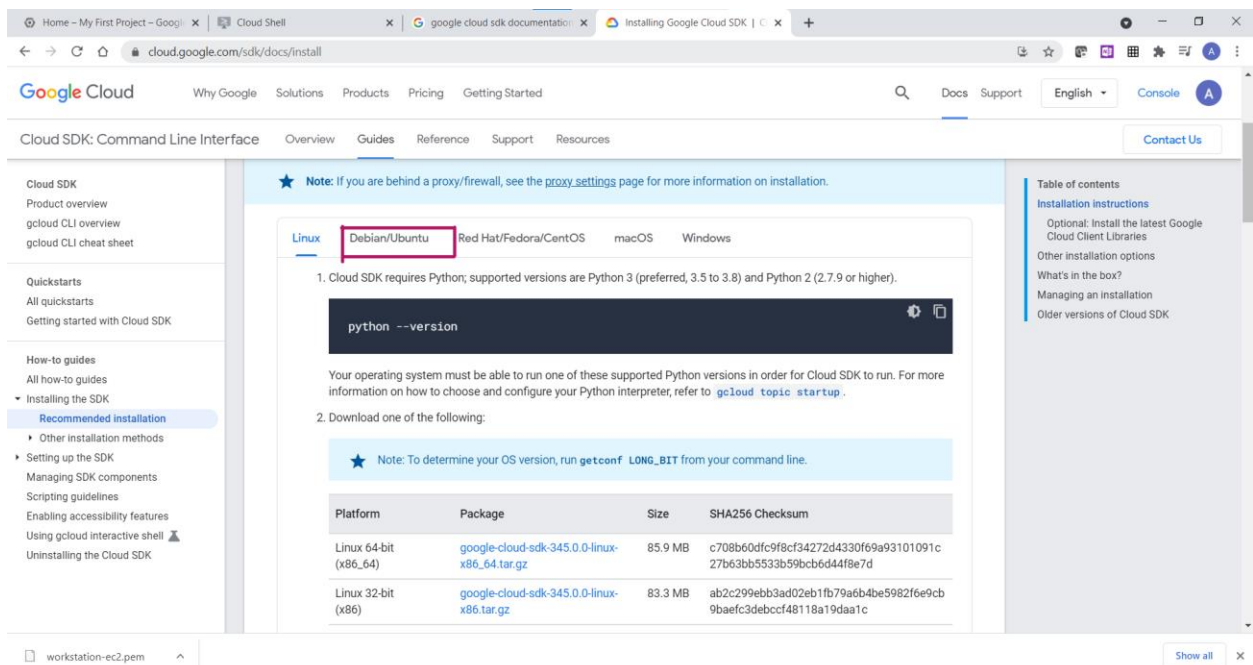
CICD using Jenkins X – Part 1

Click on 'Open in new window' so that cloud shell will be accessible in a separate tab on browser.



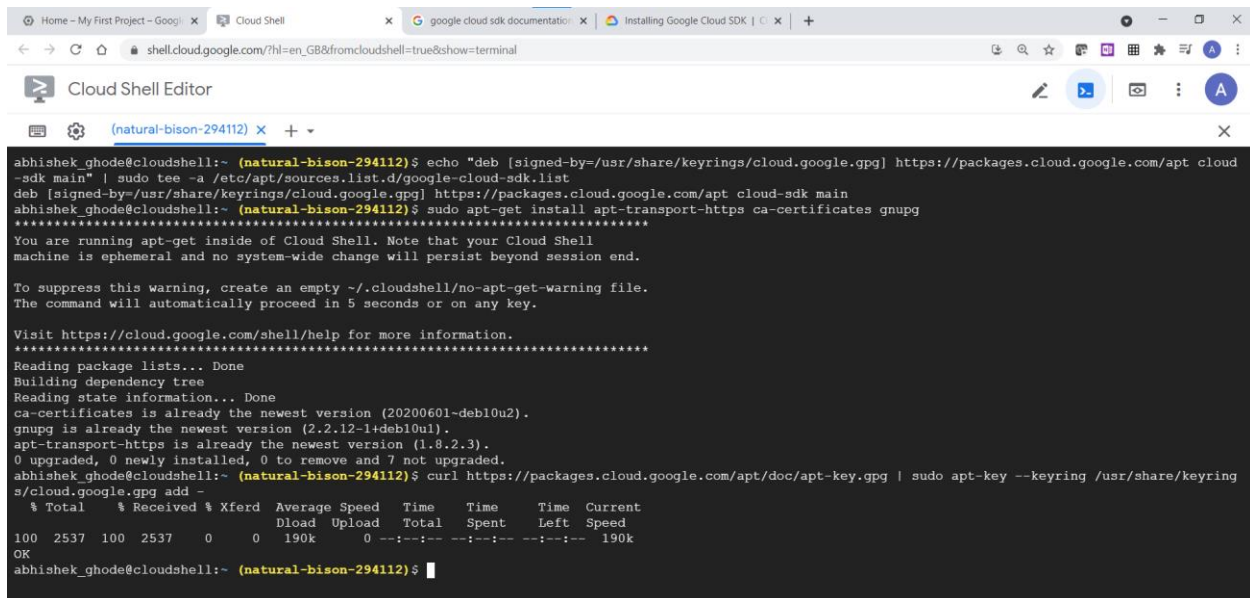
Step 3: Check GCP cloud SDK using below link.

URL to access: <https://cloud.google.com/sdk/docs/install#deb>



CICD using Jenkins X – Part 1

Step 4: We will try to SDK installation on Ubuntu machine.

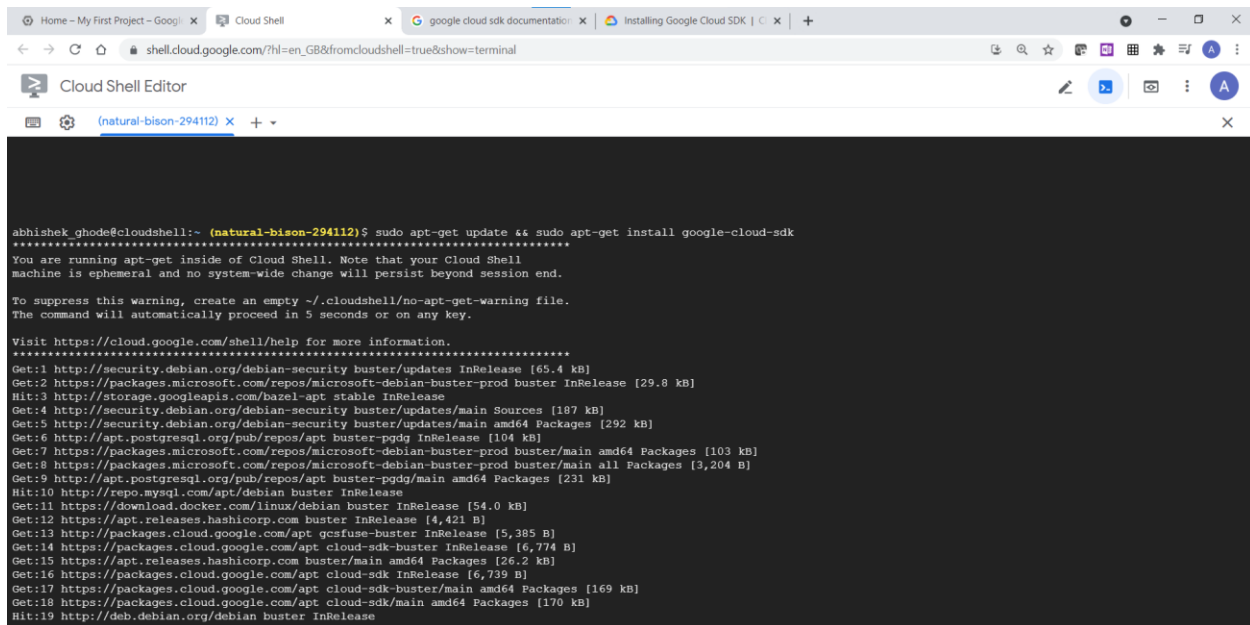


```
abhishek_ghode@cloudshell:~ (natural-bison-294112)$ echo "deb [signed-by=/usr/share/keyrings/cloud.google.gpg] https://packages.cloud.google.com/apt cloud-sdk main" | sudo tee -a /etc/apt/sources.list.d/google-cloud-sdk.list
deb [signed-by=/usr/share/keyrings/cloud.google.gpg] https://packages.cloud.google.com/apt cloud-sdk main
abhishek_ghode@cloudshell:~ (natural-bison-294112)$ sudo apt-get install apt-transport-https ca-certificates gnupg
*****
You are running apt-get inside of Cloud Shell. Note that your Cloud Shell
machine is ephemeral and no system-wide change will persist beyond session end.

To suppress this warning, create an empty ~/.cloudshell/no-apt-get-warning file.
The command will automatically proceed in 5 seconds or on any key.

Visit https://cloud.google.com/shell/help for more information.
*****
Reading package lists... Done
Building dependency tree
Reading state information... Done
ca-certificates is already the newest version (20200601-deb10u2).
gnupg is already the newest version (2.2.12-1+deb10u1).
apt-transport-https is already the newest version (1.8.2.3).
0 upgraded, 0 newly installed, 0 to remove and 7 not upgraded.
abhishek_ghode@cloudshell:~ (natural-bison-294112)$ curl https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key --keyring /usr/share/keyring
s/cloud.google.gpg add -
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 2537 100 2537    0     0  190k      0 --:--:-- --:--:-- --:--:-- 190k
OK
abhishek_ghode@cloudshell:~ (natural-bison-294112)$
```

Note: This step is redundant as Cloud shell will already have access to SDK. You can use these steps on Linux vm (Ubuntu) other than cloud shell. Please remember, cloud shell has ephemeral storage.



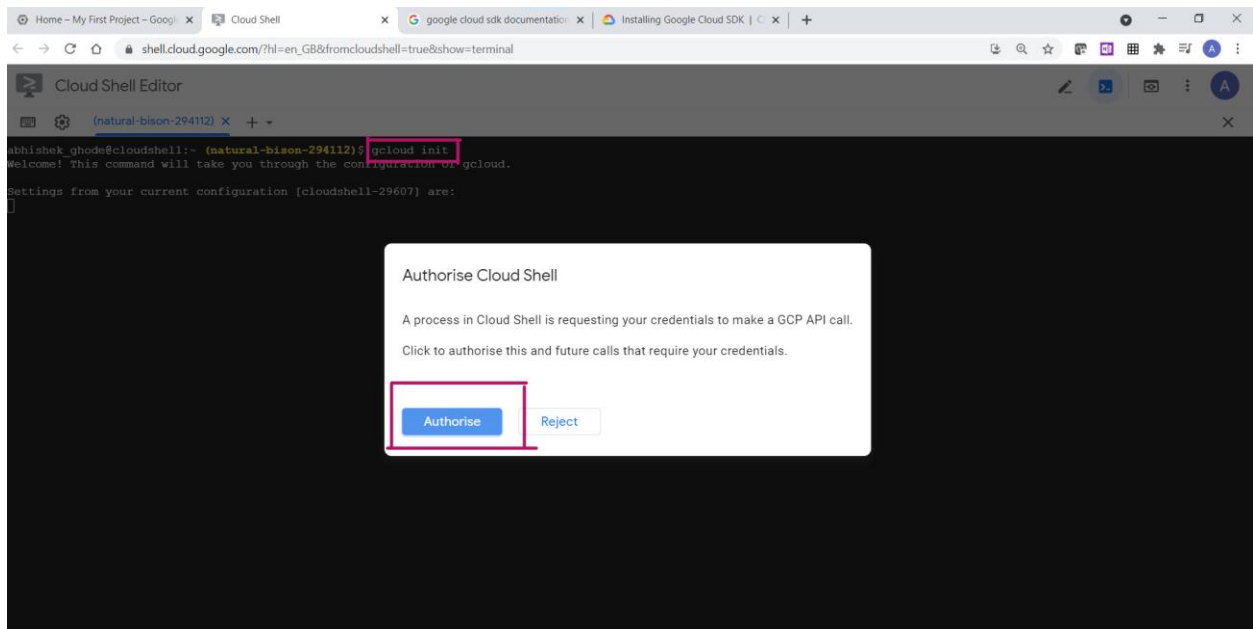
```
abhishek_ghode@cloudshell:~ (natural-bison-294112)$ sudo apt-get update && sudo apt-get install google-cloud-sdk
*****
You are running apt-get inside of Cloud Shell. Note that your Cloud Shell
machine is ephemeral and no system-wide change will persist beyond session end.

To suppress this warning, create an empty ~/.cloudshell/no-apt-get-warning file.
The command will automatically proceed in 5 seconds or on any key.

Visit https://cloud.google.com/shell/help for more information.
*****
Get:1 http://security.debian.org/debian-security buster/updates InRelease [65.4 kB]
Get:2 https://packages.microsoft.com/repos/microsoft-debian-buster-prod buster InRelease [29.8 kB]
Hit:3 http://storage.googleapis.com/bazel-apt stable InRelease
Get:4 http://security.debian.org/debian-security buster/updates/main Sources [187 kB]
Get:5 http://security.debian.org/debian-security buster/updates/main amd64 Packages [292 kB]
Get:6 http://apt.postgresql.org/pub/repos/apt buster-pgdg InRelease [104 kB]
Get:7 https://packages.microsoft.com/repos/microsoft-debian-buster-prod buster/main amd64 Packages [103 kB]
Get:8 https://packages.microsoft.com/repos/microsoft-debian-buster-prod buster/main all Packages [3,204 B]
Get:9 https://apt.postgresql.org/pub/repos/apt buster-pgdg/main amd64 Packages [231 kB]
Hit:10 http://repo.mysql.com/apt/debian buster InRelease
Get:11 https://download.docker.com/linux/debian buster InRelease [54.0 kB]
Get:12 https://apt.releases.hashicorp.com buster InRelease [4,421 B]
Get:13 http://packages.cloud.google.com/apt gcsfuse-buster InRelease [5,385 B]
Get:14 https://packages.cloud.google.com/apt cloud-sdk-buster InRelease [6,774 B]
Get:15 https://apt.releases.hashicorp.com buster/main amd64 Packages [26.2 kB]
Get:16 https://packages.cloud.google.com/apt cloud-sdk InRelease [6,739 B]
Get:17 https://packages.cloud.google.com/apt cloud-sdk-buster/main amd64 Packages [169 kB]
Get:18 https://packages.cloud.google.com/apt cloud-sdk/main amd64 Packages [170 kB]
Hit:19 http://deb.debian.org/debian buster InRelease
```

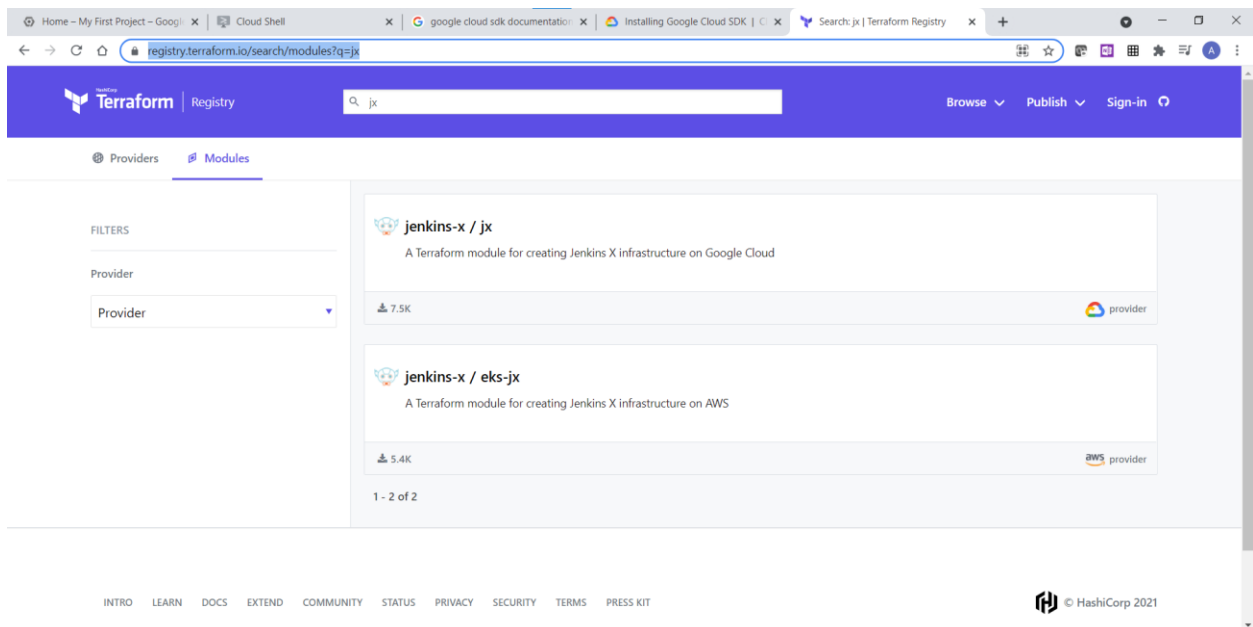
CICD using Jenkins X – Part 1

You may have to authorize access to gcloud utility using 'Authorize' option on pop up.



Step 4: Create a GKE cluster using gcloud.

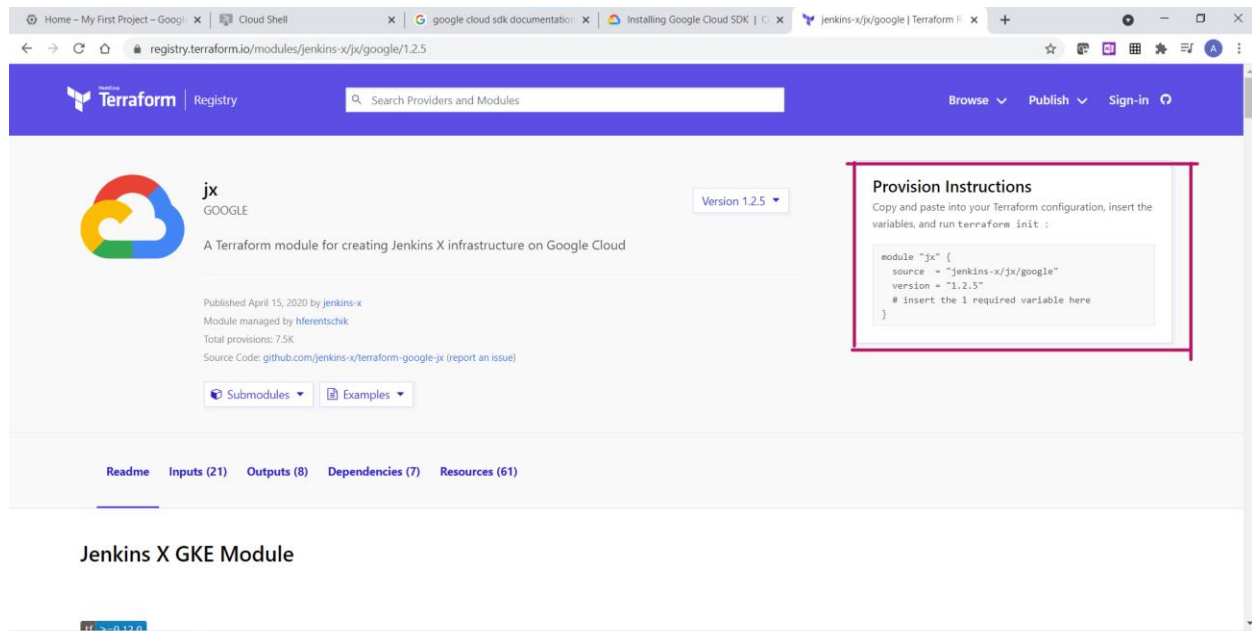
We would be using terraform to create Jenkins X infrastructure.



Copy module snippet for Provision instructions

CICD using Jenkins X – Part 1

URL to access: <https://registry.terraform.io/modules/jenkins-x/jx/google/1.10.0>



Go to cloud shell.

Commands to run:

```
mkdir jenkins-x-hello-cluster
```

```
cd jenkins-x-hello-cluster/
```

```
nano main.tf
```

Add below snippet to main.tf:

```
module "jx" {  
  source = "jenkins-x/jx/google"  
  version = "1.10.0"  
  gcp_project = "natural-bison-294112"  
  cluster_name = "hello-cluster"  
}
```

```
terraform {  
  backend "gcs" {
```


CICD using Jenkins X – Part 1

```
bucket = "terraform-state-44"

prefix = "terraform/state"

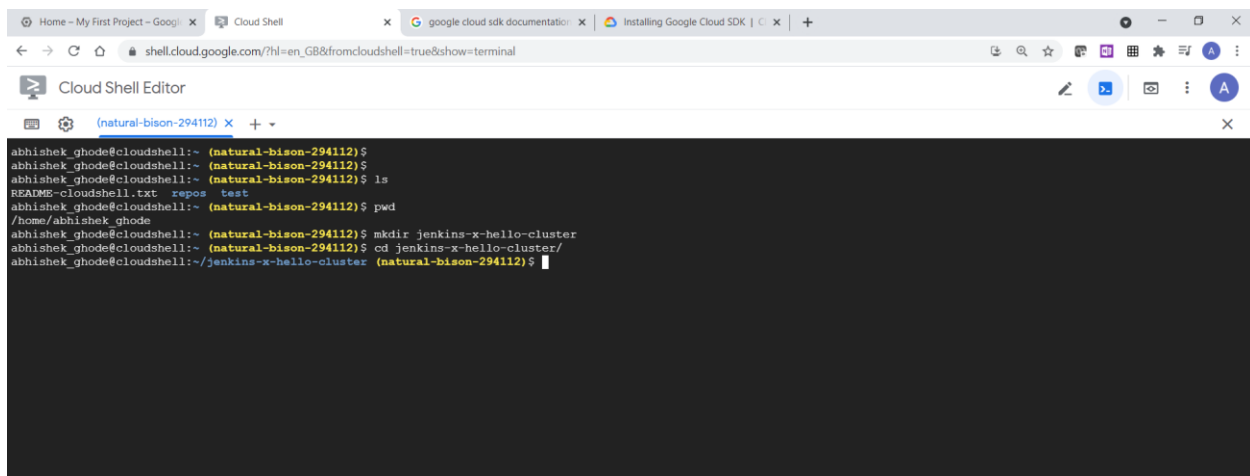
}

}
```

```
output "jx_requirements" {

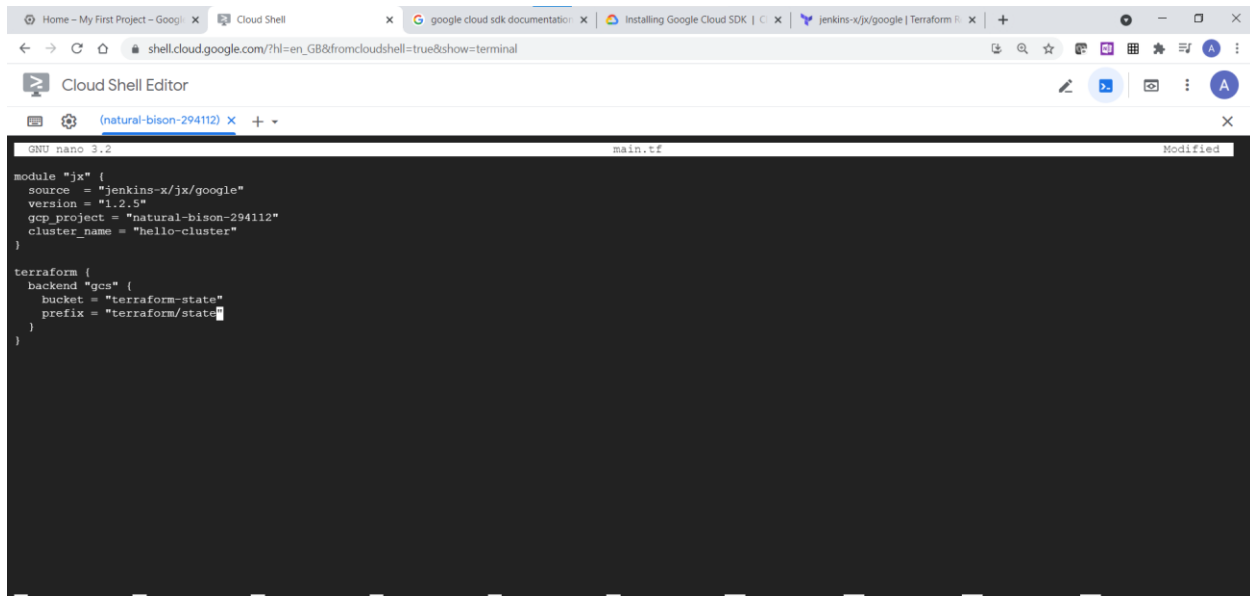
  value = module.jx.jx_requirements

}
```



The screenshot shows a Cloud Shell terminal window with the following commands and outputs:

```
abhishek_ghode@cloudshell:~ (natural-bison-294112)$
abhishek_ghode@cloudshell:~ (natural-bison-294112)$
abhishek_ghode@cloudshell:~ (natural-bison-294112)$ ls
README-cloudshell.txt  repos  test
abhishek_ghode@cloudshell:~ (natural-bison-294112)$ pwd
/home/abhishek_ghode
abhishek_ghode@cloudshell:~ (natural-bison-294112)$ mkdir jenkins-x-hello-cluster
abhishek_ghode@cloudshell:~ (natural-bison-294112)$ cd jenkins-x-hello-cluster/
abhishek_ghode@cloudshell:~/jenkins-x-hello-cluster (natural-bison-294112)$
```



The screenshot shows a Cloud Shell terminal window with the GNU nano 3.2 editor open, editing a file named main.tf. The content of the file is as follows:

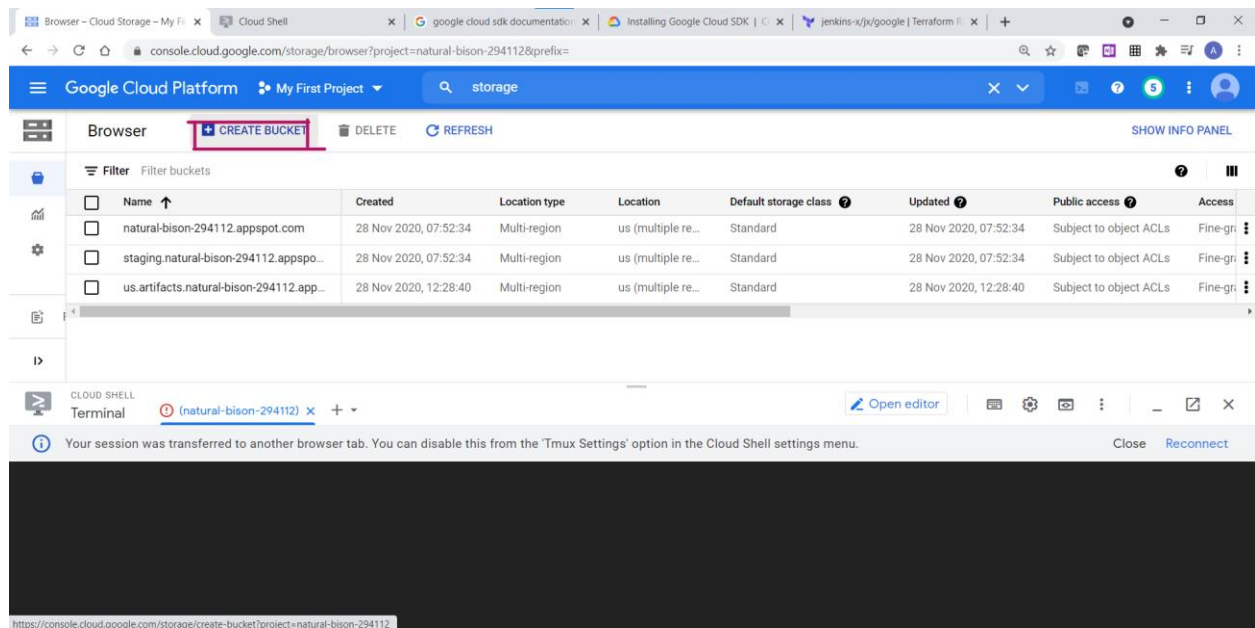
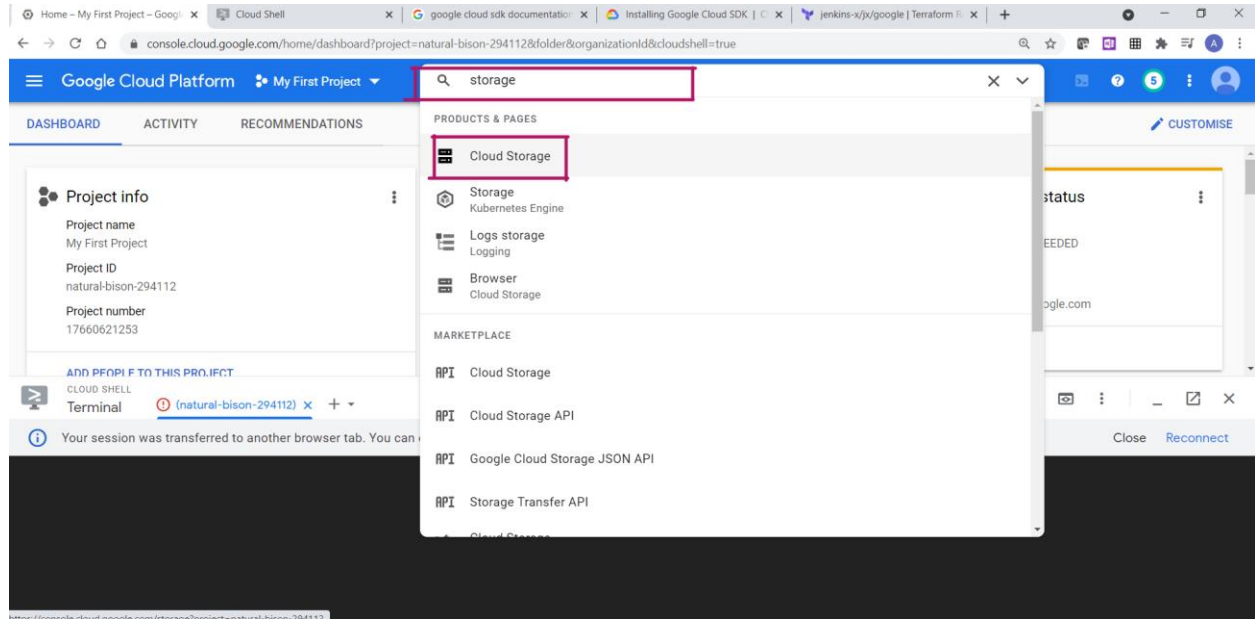
```
module "jx" {
  source = "jenkins-x/jx/google"
  version = "1.2.5"
  gcp_project = "natural-bison-294112"
  cluster_name = "hello-cluster"
}

terraform {
  backend "gcs" {
    bucket = "terraform-state"
    prefix = "terraform/state"
  }
}
```

CICD using Jenkins X – Part 1

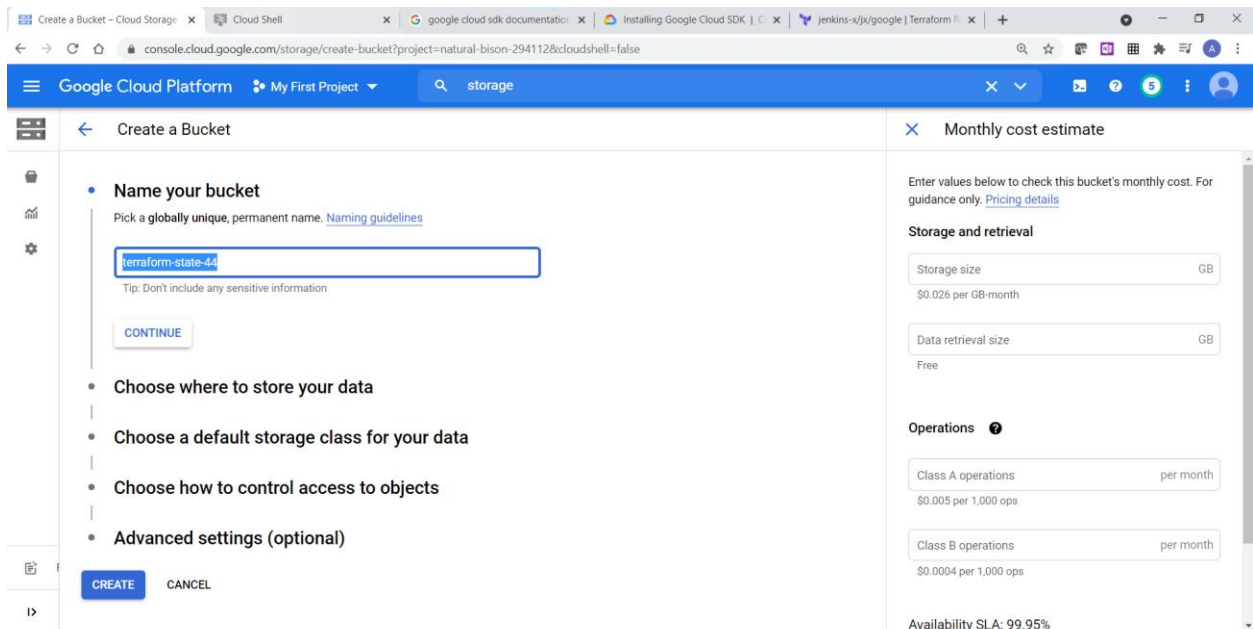
Step 5: Create a bucket mentioned in main.tf.

Note: This bucket must have unique name.



CICD using Jenkins X – Part 1

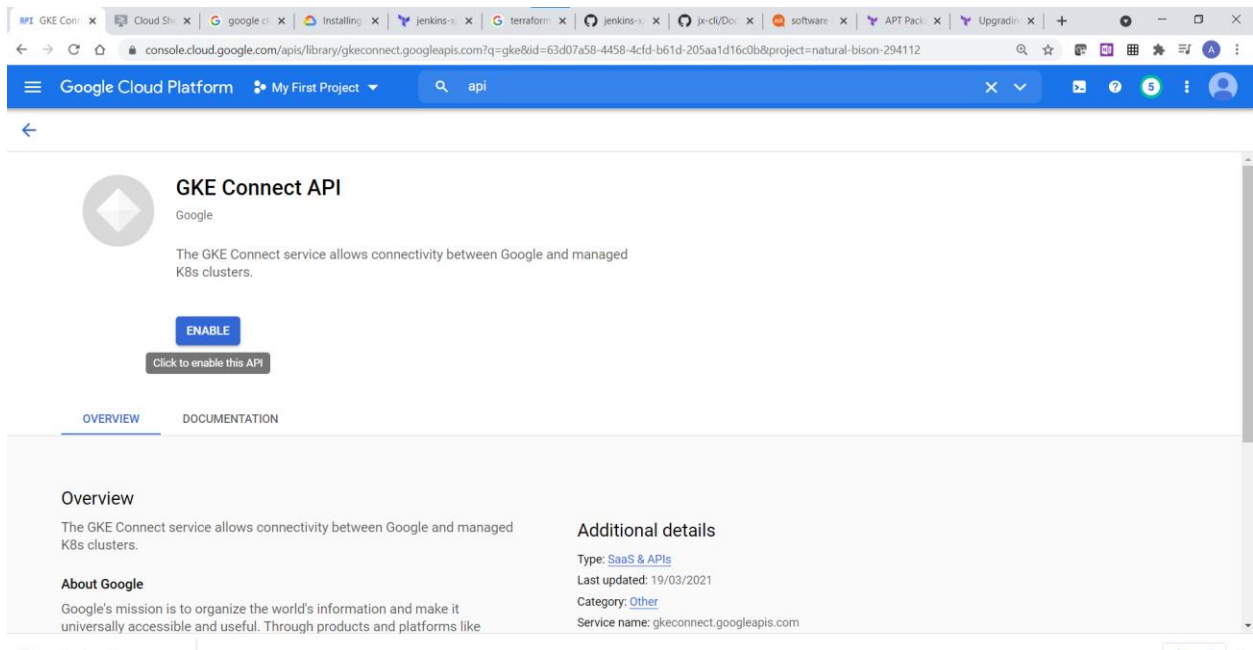
Note: Please select bucket name which is unique and update it in main.tf if required.



The screenshot shows the Google Cloud Platform console with the 'Create a Bucket' wizard. The 'Name your bucket' step is active, showing a text input field with the value 'terraform-state-44'. Below the input field, there is a tip: 'Tip: Don't include any sensitive information'. To the right, the 'Monthly cost estimate' sidebar is open, showing the 'Storage and retrieval' section with 'Storage size' and 'Data retrieval size' both set to 'GB'. The 'Operations' section shows 'Class A operations' and 'Class B operations' both set to 'per month'. The 'Availability SLA' is listed as '99.95%'.

Search 'api and services' in search bar.

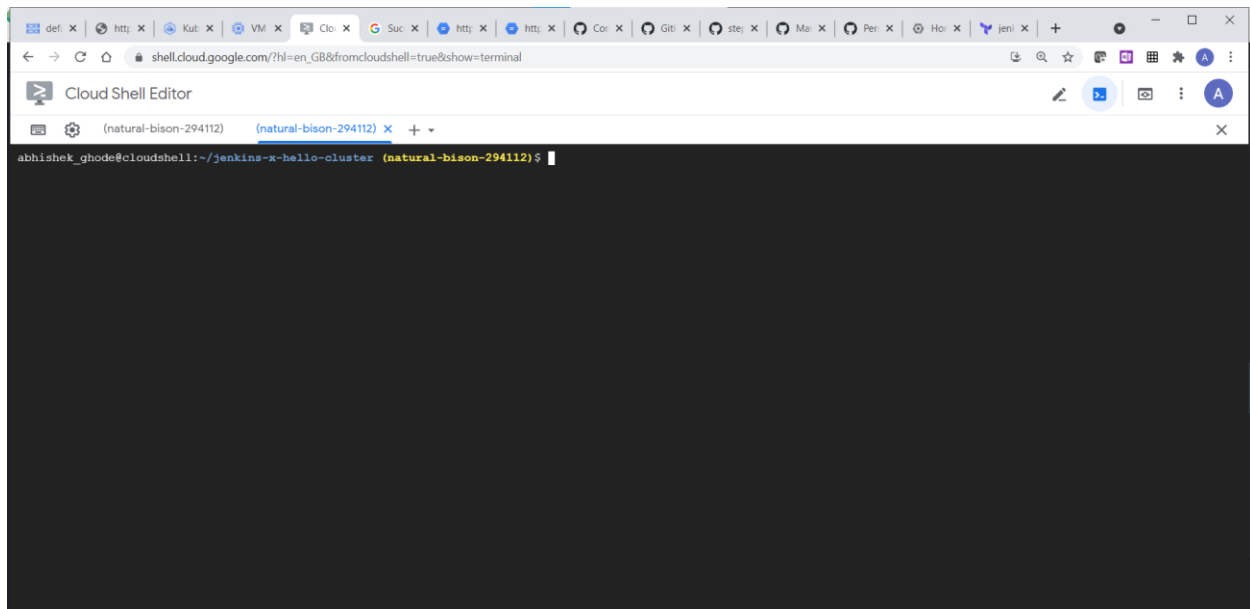
Please ensure to enable 'GKE connect API'



The screenshot shows the Google Cloud Platform console with the 'GKE Connect API' page. The 'ENABLE' button is visible, and the 'OVERVIEW' tab is selected. The 'Overview' section contains the text: 'The GKE Connect service allows connectivity between Google and managed K8s clusters.' The 'Additional details' section shows the 'Type' as 'SaaS & APIs', 'Last updated' as '19/03/2021', 'Category' as 'Other', and 'Service name' as 'gkeconnect.googleapis.com'.

CICD using Jenkins X – Part 1

Step 6: Install commands on cloud shell to create a cluster.



Commands to run:

```
cd ~
```

```
sudo apt install terraform=0.14.0
```

```
terraform -v
```

```
cd jenkins-x-hello-cluster/
```

```
terraform init
```

```
terraform plan
```

```
terraform apply
```

```
terraform output jx_requirement > ./jx-requirements.yml
```

Step 6: To view the cluster:

Commands to run on cloud shell

```
gcloud config set project <GCP project>
```

```
gcloud config set compute/zone <compute zone GCP>
```

```
gcloud container clusters get-credentials hello-cluster
```

CICD using Jenkins X – Part 1

You can also view cluster using below URL:

The screenshot shows the Google Cloud Platform console interface. The search bar at the top right contains the text 'gke'. Below the search bar, the 'PRODUCTS & PAGES' section is expanded, showing 'Kubernetes Engine' as the top result. The 'RESOURCES' section lists several Kubernetes Engine clusters, including 'gke-hello-cluster-3260-pvc-371c00f0-dc39-482c-8b89-55d70e238ce7' and 'gke-hello-cluster-3260-pvc-65c99632-a5b4-4efc-a79a-a950ac63db53'. The 'MARKETPLACE' section shows 'API' and 'GKE Connect API'.

Google Cloud Platform console showing the search results for 'gke'. The search results are displayed in a sidebar on the right, listing various Kubernetes Engine clusters and their associated resources. The main content area shows the details of the selected resource, including its type, size, creation time, and permissions.

Overview

Property	Value
Type	text/plain; charset=utf-8
Size	128.7 KB
Created	18 Jun 2021, 15:13:34
Last modified	18 Jun 2021, 15:13:34
Custom time	—
Public URL	Not applicable
Authenticated URL	https://storage.cloud.google.com/terraform-state-44/terraform/state/default.tfstate?project=natural-bison-294112
gsutil URI	gs://terraform-state-44/terraform/state/default.tfstate
Permission	Public access: Not public
Protection	Hold status: None
Retention policy	None
Encryption type	Google-managed key

RESOURCES

Resource Name	Resource Type
gke-hello-cluster-3260-pvc-371c00f0-dc39-482c-8b89-55d70e238ce7	Disk - natural-bison-294112
gke-hello-cluster-3260-pvc-65c99632-a5b4-4efc-a79a-a950ac63db53	Disk - natural-bison-294112
gke-hello-cluster-3260-pvc-7a58c128-1dff-432b-8185-356d7ee7b47c	Disk - natural-bison-294112
gke-hello-cluster-3260-pvc-c586e87a-1fc0-4627-8bd9-5c4037599664	Disk - natural-bison-294112

MARKETPLACE

Marketplace Item
API
GKE Connect API