



www.linkedin.com/in/prafulpatel16



<https://github.com/prafulpatel16>



GCP BOOTCAMP PROJECT

GCP PROJECT 7: MIGRATION OF A STAND-ALONE

APPLICATION AND DATABASE FROM GCE (VM) TO A

'MODERN ARCHITECTURE' USING GCR & GKE (APP) +

CLOUD SQL (DB) WITH HIGH AVAILABILITY (REGION

FAILOVER)

IMPLEMENTED BY: PRAFUL PATEL

MIGRATION OF A STAND-ALONE APPLICATION AND DATABASE FROM GCE (VM)
TO A 'MODERN ARCHITECTURE' USING GKE (APP) + CLOUD SQL (DB) WITH HA



Page | 0

PRAFUL PATEL

www.linkedin.com/in/prafulpatel16

<https://github.com/prafulpatel16>



Date: MAR 21, 2022

➤ **Project Definition:**

MIGRATION OF A STAND-ALONE APPLICATION AND DATABASE FROM GCE (VM) TO A ‘MODERN ARCHITECTURE’ USING GKE (APP)+ CLOUD SQL (DB) WITH HA

An IT services provider company called **PRAfect Systems Inc.** is engaged into providing software development solutions. The web application name “**MIB Patient Management**” is running on compute engine vm’s and mysql database also running on another vm. At the present company needs to migrate an application and database which are running on GCP virtual machines on compute engine. Recently infrastructure team has observed that the maintenance is going to be critical and management team has noticed that due to running on VM’s the costing has increased so they have decided to migrate the entire application and database from vm to containerized environment using GKE service.

➤ **Application Environment:**

► Frontend tier:

1. Application name: “**MIB Patient Management**”
2. App Development: nodejs
3. Container Environment: Docker
4. Container Repository Tool: GCR (Google Container Registry)
5. Container Orchestration Tool: GKE (Google Kubernetes Engine)

► Database tier:

Technology: Google Cloud SQL (MySQL)

This project demonstrates an experience of containerizing an application using Google cloud container registry service and deploy in to Google kubernetes Engine along with Google Cloud SQL(My SQL) database service in a multi region with highly available environment.

➤ **Solution:**

This project is a migration project from virtual machines to containerize and run on highly available containerized environment. The web application will be running on container on google kubernetes engine which is an orchestration managed service from google cloud.

In order to achieve and deploy our frontend web layer in a containerized environment it needs to build a docker image from the web application source code and has to tag the image correctly. Once the image is created it needs to push to the google cloud container registry as a private repository where the artifactory will be stored. Then it needs to deploy the docker image to the orchestration tool using google kubernetes engine and needs to create a cluster with two nodes in order to achieve the high availability where the application as container will be run inside the pods and will be able to access using deployment and LoadBalancer service with external IP address. So with that the users from around the world can access the frontend web application.

The web application contains dynamic contact us web form where users can connect and enter the inquiry with the data fields name, email and subject and this should be dynamically send to the backend database storage in a highly available environment.

The database tier should be deployed with multi region in a highly available manner so that the application can be up and running all the time without any single point of failure and it will be achieved using Google Cloud SQL database service.

Once the application has connected correctly it will need to enter some data from the contact us page and meantime has to trigger the failover test from the Cloud SQL so that it can measure the performance and high availability of the web application.

This project demonstrates an experience of web application migration within intercontinental region using compute engine and storage and snapshot services. Also it is critical that within 30 minutes of downtime the web application should be up and running correctly.

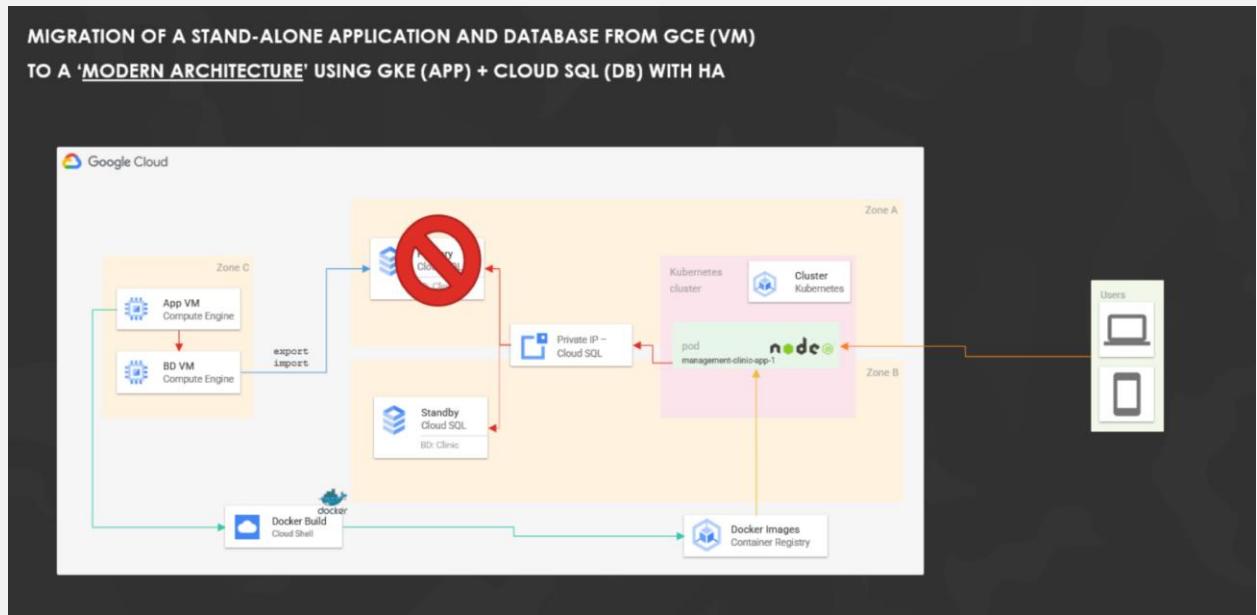
➤ **Project Cost Estimation:**

(Note: This cost is Not any actual cost, it's just an estimation based on high level requirement. Price may vary based on adding and removing services based on requirement.)

➤ **Tools & Technologies covered:**

- GCP Cloud
- Google Container Registry
- Google Kubernetes Engine
- Cloud SQL – MySQL
- Cloud Shell

➤ **Architectural Diagram:**



This migration project will be completed in 3 implementation phases.

➤ **Project implementation Phases:**

- Phase 1: Create an Application VM and database VM and run the application on VMs.
- Phase 2: Prepare for application and database VMs backup.
- Phase 3: Containerize the web app to docker image and push to google container registry.
- Phase 4: Deploy Cloud SQL with multi region and create a database and user & import a database back of database VM.
- Phase 5: Create a Kubernetes cluster and prepare deployment and service yaml.
- Phase 6: Deploy a application yaml to GKE and access the web application
- Phase 7: Test Application High Availability using Cloud SQL failover test.

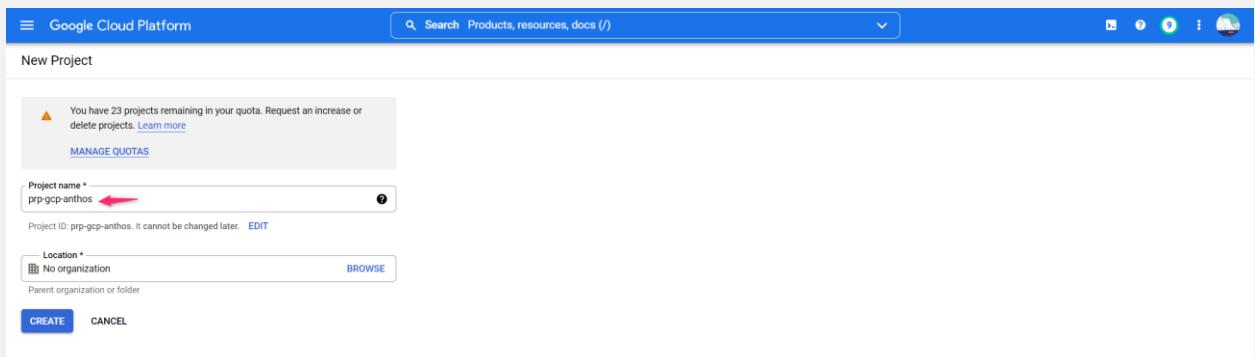
➤ **Implementation:**

- Phase 1: Create an Application VM and database VM and run the application on VMs.
 1. Create an application vm and install necessary packages.
 2. Create a database vm and install mysql packages
 3. Configure connection strings in application to connect database.
 4. Insert some data and verify the connectivity.
- Phase 2: Prepare for application and database VMs backup.
 1. Take an application VM backup
 2. Take a database VM backup.

- ❖ Phase 3: Containerize the web app to docker image and push to google container registry.
 1. Target to the web app source code and change the directory.
 2. Prepare a Dockerfile from the src folder to build the docker image.
 3. Give a proper tag and push the docker image to google container registry.
 4. Verify that web application is accessible using docker run.
- ❖ Phase 4: Deploy Cloud SQL with multi region and create a database and user & import a database back of database VM.
 1. Create a Cloud SQL- My SQL database instance.
 2. Create a required database with multi region.
 3. Create a user with password to access the database.
 4. Import database vm backup to Clous SQL
- ❖ Phase 5: Create a Kubernetes cluster and prepare deployment and service yaml.
 1. Create a GKE kubernetes cluster using autopilot.
 2. Create a web deployment and service yml
- ❖ Phase 6: Deploy a application yaml to GKE and access the web application.
 1. Deploy the yaml file
Kubectl apply -f < yaml file>.
 2. Verify that pods are created.
 3. Verify that web app deployment created.
 4. Verify that web app service is created with external LoadBalancer IP address.
 5. Access the frontend web application from browser and insert some data in contact us page.
- ❖ Phase 7: Test Application High Availability using Cloud SQL failover test.
 1. Go to Cloud SQL and observer the region in which the instance is running.
 2. Now click on Failover and trigger the test.
 3. Try to access and insert the data still within failover window.
 4. Verify that frontend web application is still up and running.
 5. Verify that sql instance has changed the region after failover and app is still running.

✓ Solution Implementation in Action:

1. Create a GCP Project



2. Check the billing is enable for the project

The top screenshot shows the 'MY BILLING ACCOUNTS' page. It has a 'CREATE ACCOUNT' button and a table with columns: 'Billing account name', 'Billing account ID', 'Status', 'Last 30 days' spend', 'Account type', 'Organization', and 'Health checks'. Two rows are shown: 'My Billing Account' (Active, \$0, Direct, Organization: -, Health checks: 1) and another row with a red arrow pointing to its 'Actions' column. The right sidebar is titled 'My Billing Account' with sections for 'Edit or delete permissions below or "Add Principal" to grant new' and 'Show inherited permissions'.

The bottom screenshot shows the 'Billing' page under 'Account management'. It has a sidebar with options like Overview, Reports, Cost table, Cost breakdown, Commitments, Commitment analysis, Budgets & alerts, Billing export, Pricing, Documents, Transactions, Payment settings, Payment method, and Account management (with a red arrow pointing to it). The main area shows 'Projects linked to this billing account' with a table:

Project name	Project ID	Actions
My First Project	ethereum-terra-343020	⋮
prafuls-projects	prafuls-projects	⋮
prp-gcp-anthos	prp-gcp-anthos	⋮ (with a red arrow)

The right sidebar is titled 'My Billing Account' with sections for 'Edit or delete permissions below or "Add Principal" to grant new' and 'Show inherited permissions'.

3. Enable 3 APIs

Enable the following APIs:

Cloud Resource Manager API

Cloud Resource Manager API

Creates, reads, and updates metadata for Google Cloud Platform containers.

OVERVIEW **DOCUMENTATION**

Overview
Creates, reads, and updates metadata for Google Cloud Platform containers.

Tutorials and documentation
[Learn more](#)

Select a project

RECENT	STARRED	ALL
Name	ID	
prp-gcp-anthos	prp-gcp-anthos	
prafuls-projects	prafuls-projects	
My First Project	ethereal-terra-343020	

CANCEL **OPEN**

What is your main goal for visiting Google Cloud Marketplace today?

- To find information about a specific product
- To browse products offered by the Marketplace
- To use or manage a product I have purchased
- To assess it as a potential procurement channel
- To view or accept a private offer
- To try out a specific product before purchasing
- To purchase a product
- To research and find solutions for my work
- Other

Compute Engine API

Compute Engine API

Compute Engine API

ENABLE **TRY THIS API**

OVERVIEW **DOCUMENTATION** **SUPPORT**

Overview

Create 01 db instance

GCE - Database (db)

Name: aus-db01

Region: australia

Zone: Australia-southeast1-c

Size: e2-micro

OS: Debian GNU/Linux 10 (buster)

```
gcloud compute instances create aus-db01 --machine-type f1-micro --zone
australia-southeast1-c
```

Installing, Setting up and Creating DB in the aus-db01 instance

Run these commands for OS update, repository and package installation:

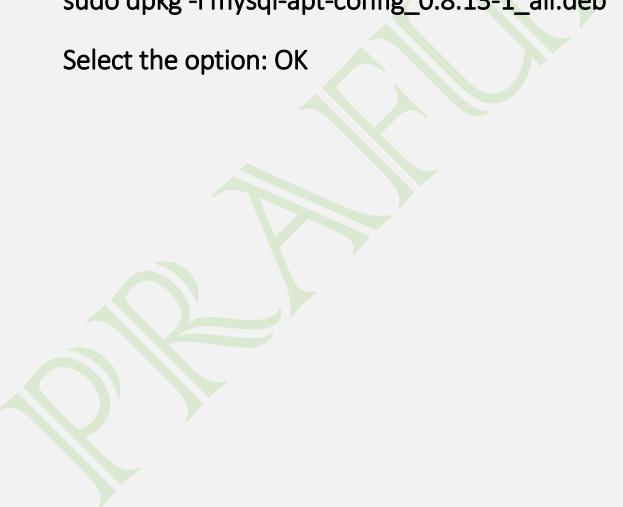
```
sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 467B942D3A79BD29
```

```
aystester1@aus-db02:~$ sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 467B942D3A79BD29
Executing: /tmp/apt-key-gpghome.TXeXJqB0jF/gpg.1.sh --keyserver keyserver.ubuntu.com --recv-keys 467B942D3A79BD29
gpg: key 467B942D3A79BD29: public key "MySQL Release Engineering <mysql-build@oss.oracle.com>" imported
gpg: Total number processed: 1
gpg: imported: 1
aystester1@aus-db02:~$
```

```
sudo apt update
```

```
sudo apt-get -y install wget
```

```
wget http://repo.mysql.com/mysql-apt-config_0.8.13-1_all.deb
```

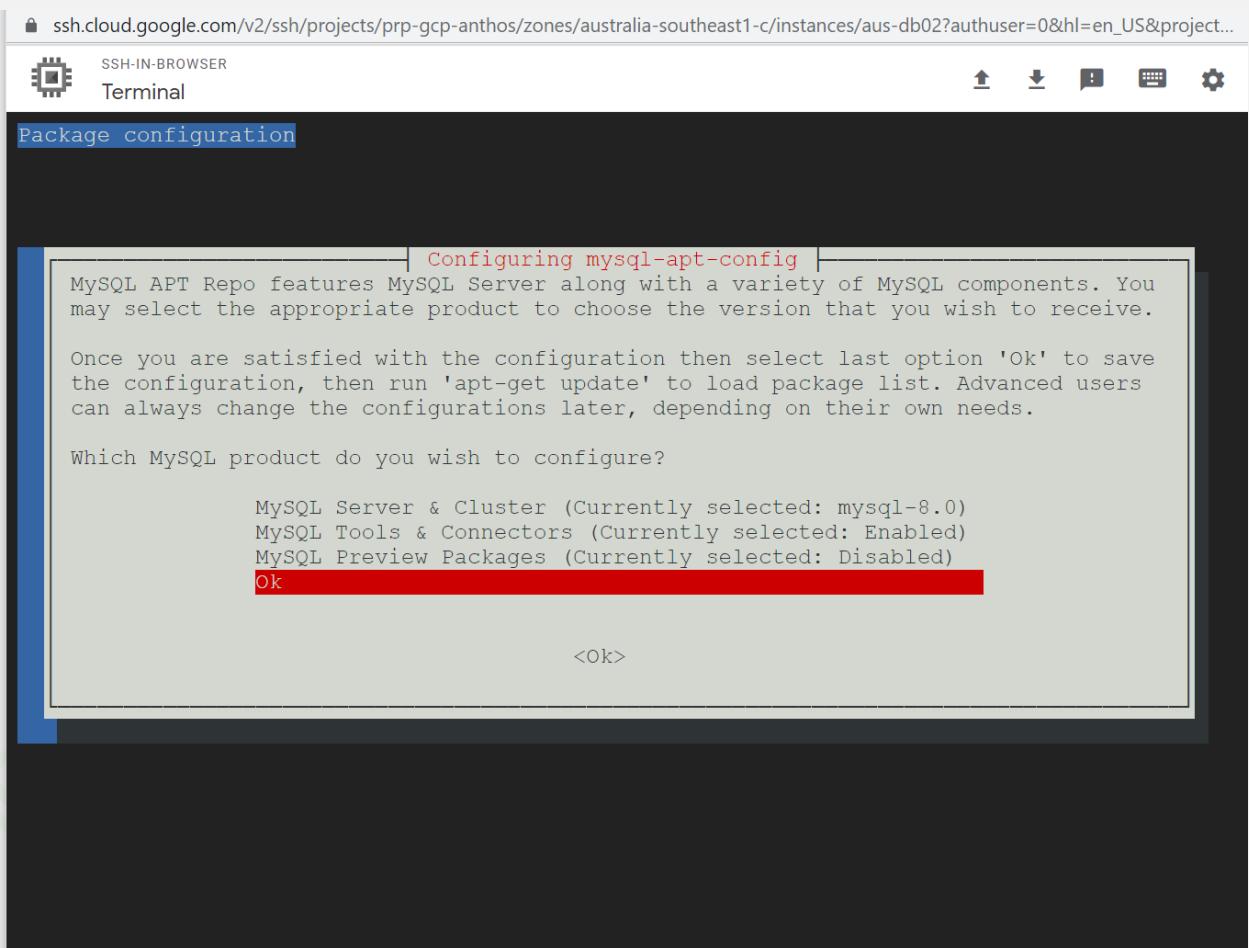


SSH-IN-BROWSER Terminal

```
Need to get 1115 kB of archives.  
After this operation, 3925 kB of additional disk space will be used.  
Get:1 http://deb.debian.org/debian buster/main amd64 libpcre2-8-0 amd64 10.32-5 [213 kB]  
Get:2 http://deb.debian.org/debian buster/main amd64 wget amd64 1.20.1-1.1 [902 kB]  
Fetched 1115 kB in 0s (41.2 MB/s)  
Selecting previously unselected package libpcre2-8-0:amd64.  
(Reading database ... 51342 files and directories currently installed.)  
Preparing to unpack .../libpcre2-8-0_10.32-5_amd64.deb ...  
Unpacking libpcre2-8-0:amd64 (10.32-5) ...  
Selecting previously unselected package wget.  
Preparing to unpack .../wget_1.20.1-1.1_amd64.deb ...  
Unpacking wget (1.20.1-1.1) ...  
Setting up libpcre2-8-0:amd64 (10.32-5) ...  
Setting up wget (1.20.1-1.1) ...  
Processing triggers for man-db (2.8.5-2) ...  
Processing triggers for libc-bin (2.28-10) ...  
aystester1@aus-db02:~$ wget http://repo.mysql.com/mysql-apt-config_0.8.13-1_all.deb  
--2022-03-21 22:33:02-- http://repo.mysql.com/mysql-apt-config_0.8.13-1_all.deb  
Resolving repo.mysql.com (repo.mysql.com)... 23.1.23.97  
Connecting to repo.mysql.com (repo.mysql.com)|23.1.23.97|:80... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 35560 (35K) [application/x-debian-package]  
Saving to: 'mysql-apt-config_0.8.13-1_all.deb'  
  
mysql-apt-config_0.8.1 100%[=====] 34.73K --.-KB/s in 0s  
  
2022-03-21 22:33:02 (73.7 MB/s) - 'mysql-apt-config_0.8.13-1_all.deb' saved [35560/35560]  
  
aystester1@aus-db02:~$
```

`sudo dpkg -i mysql-apt-config_0.8.13-1_all.deb`

Select the option: OK

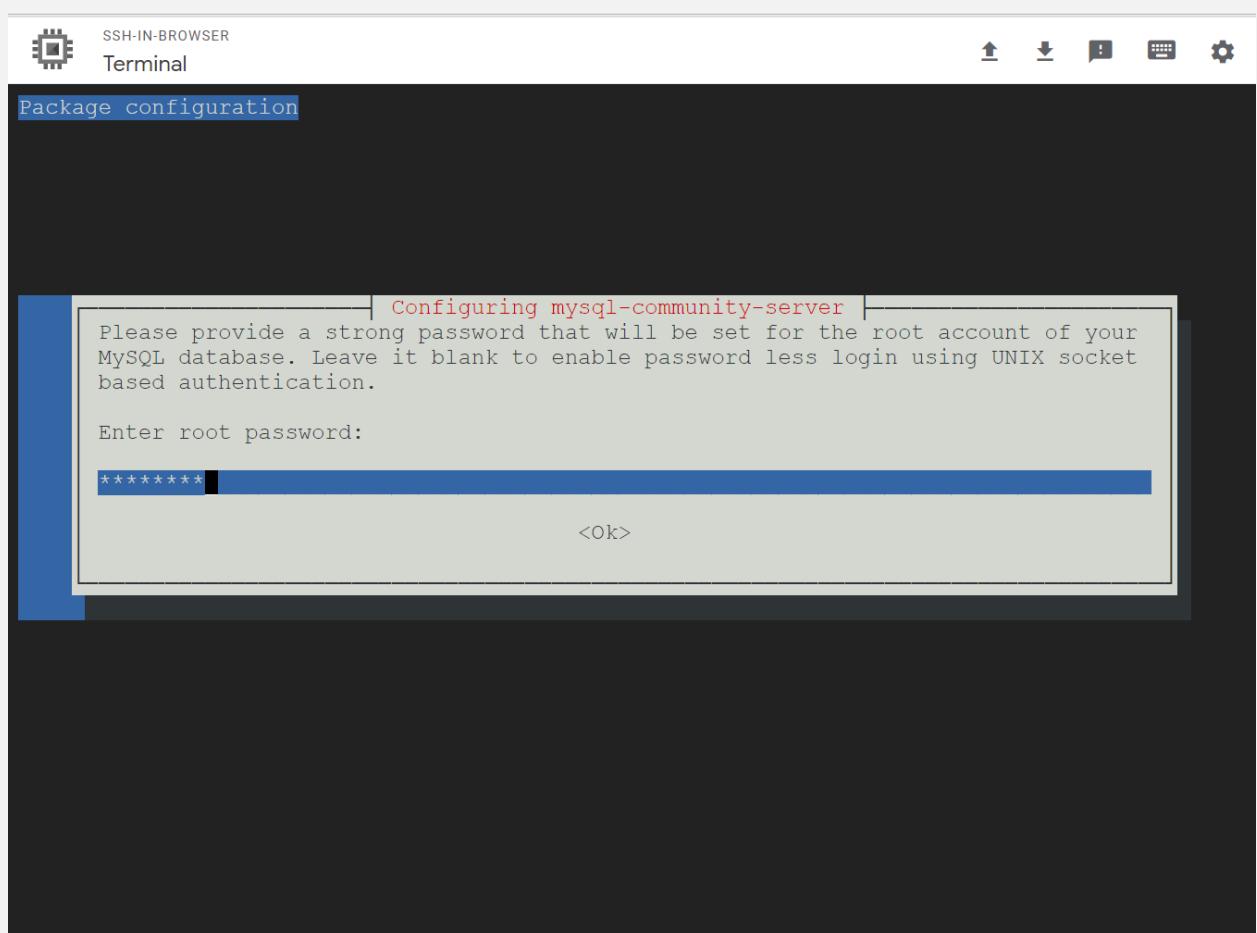


Installing the MySQL Server

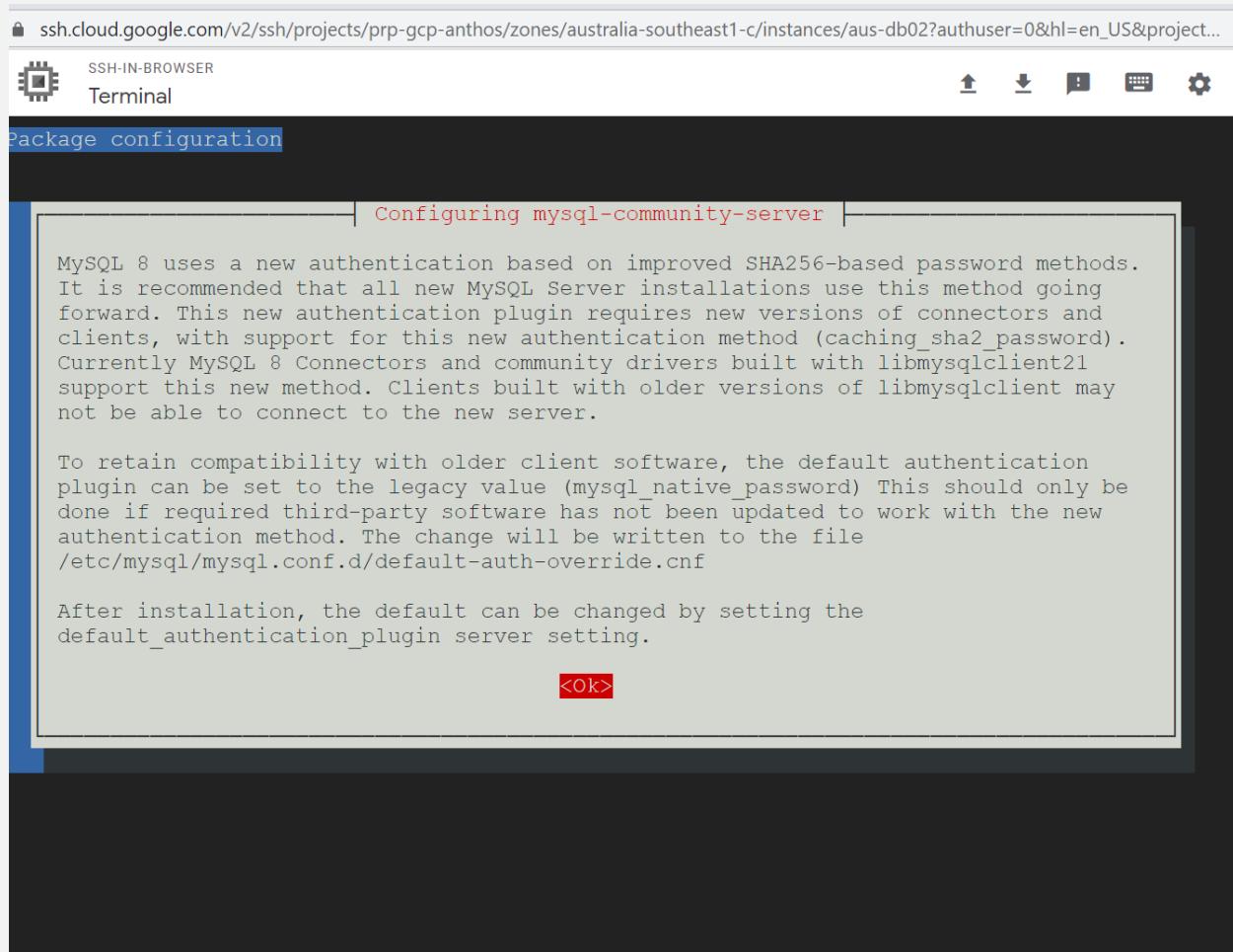
`sudo apt update`

`sudo apt install mysql-server -y`

Insert and repeat the password: welcome1



Click on Ok.



The screenshot shows a terminal window titled "Package configuration" with the URL "ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&project..." at the top. The terminal interface includes icons for SSH-IN-BROWSER, Terminal, and various system controls. The main content is a configuration message for MySQL:

```
Configuring mysql-community-server

MySQL 8 uses a new authentication based on improved SHA256-based password methods. It is recommended that all new MySQL Server installations use this method going forward. This new authentication plugin requires new versions of connectors and clients, with support for this new authentication method (caching_sha2_password). Currently MySQL 8 Connectors and community drivers built with libmysqlclient21 support this new method. Clients built with older versions of libmysqlclient may not be able to connect to the new server.

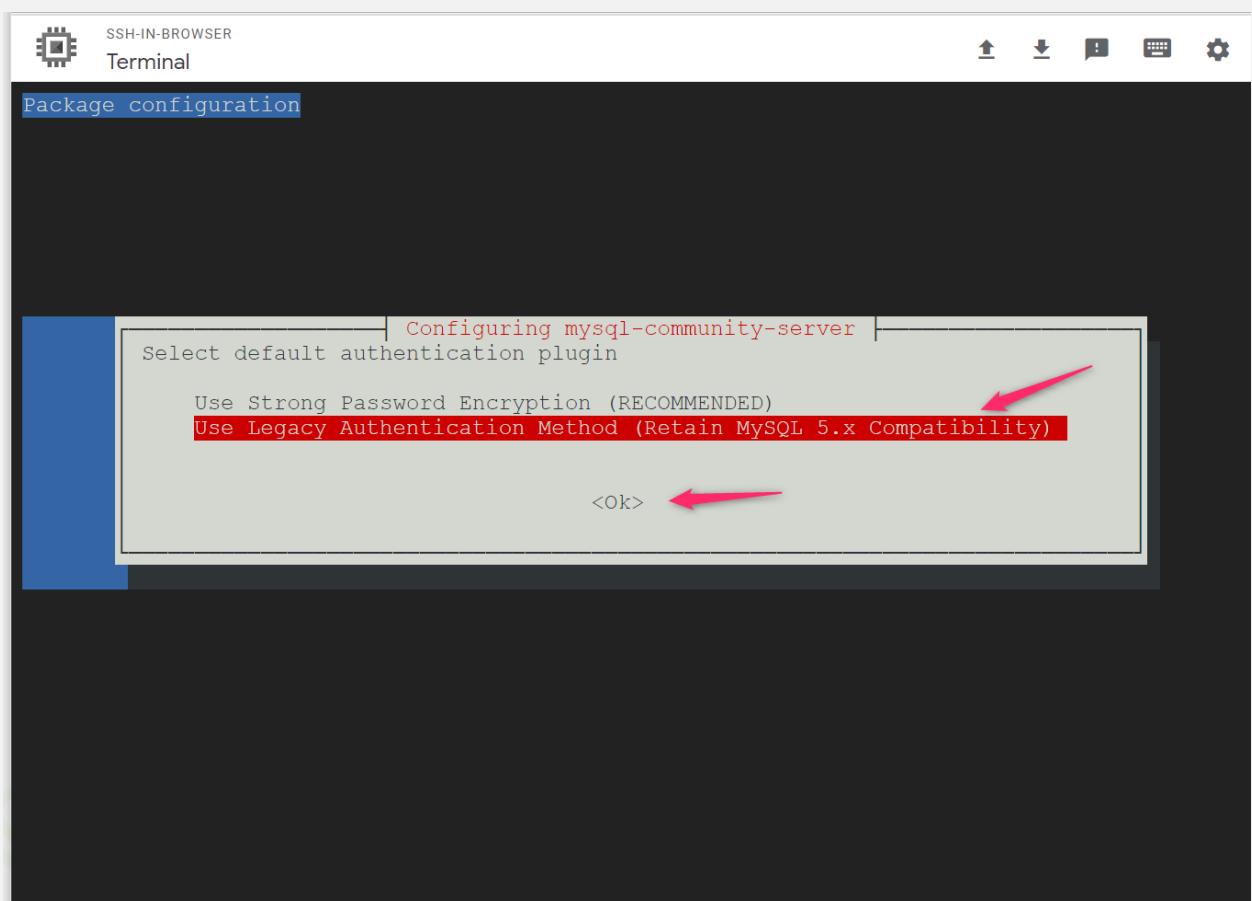
To retain compatibility with older client software, the default authentication plugin can be set to the legacy value (mysql_native_password). This should only be done if required third-party software has not been updated to work with the new authentication method. The change will be written to the file /etc/mysql/mysql.conf.d/default-auth-override.cnf

After installation, the default can be changed by setting the default_authentication_plugin server setting.
```

A red button labeled "<ok>" is visible at the bottom of the configuration window.

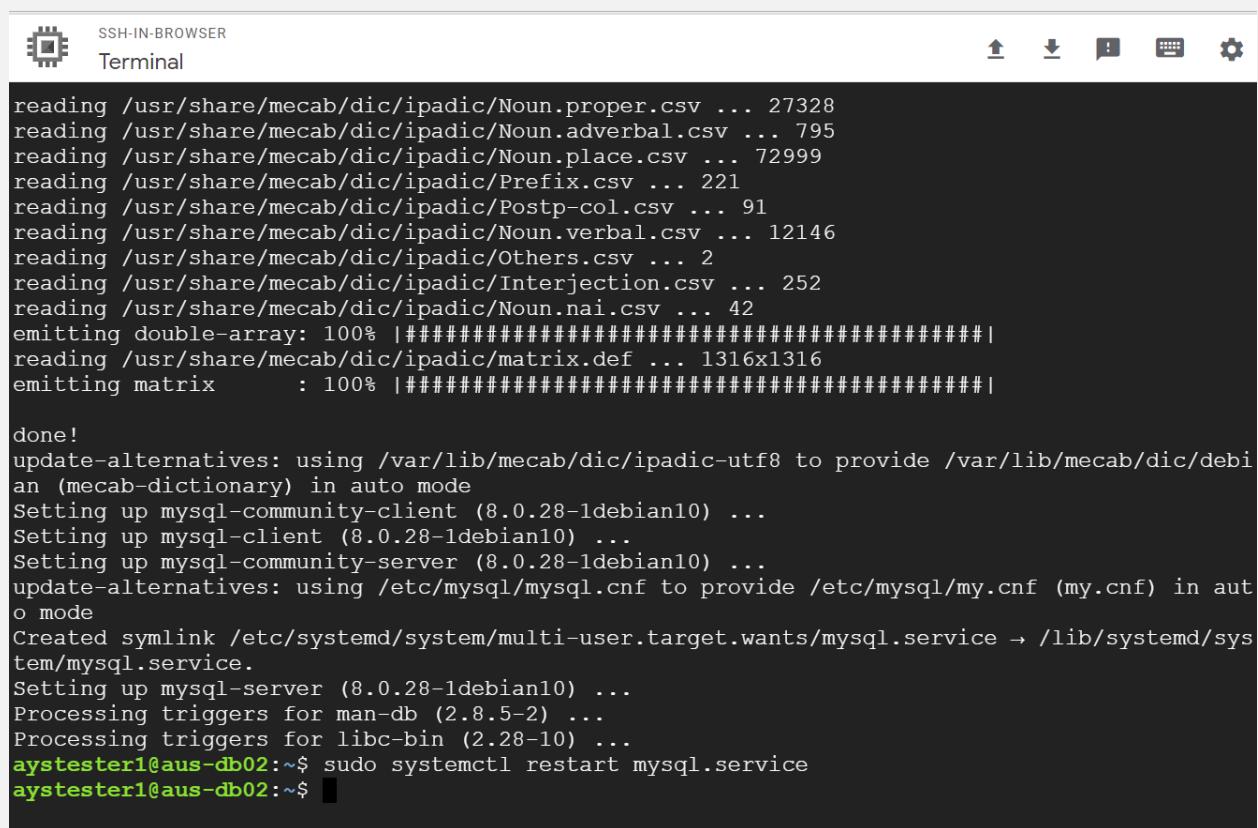
Select default authentication plugin

Select Use Legacy Authentication Method (Retain MySQL 5.x Compatibility)



Restarting the MySQL service:

```
sudo systemctl restart mysql.service
```



The screenshot shows a terminal window titled "Terminal" with the subtitle "SSH-IN-BROWSER". The window has a toolbar with icons for upload, download, copy, paste, and settings. The terminal output is as follows:

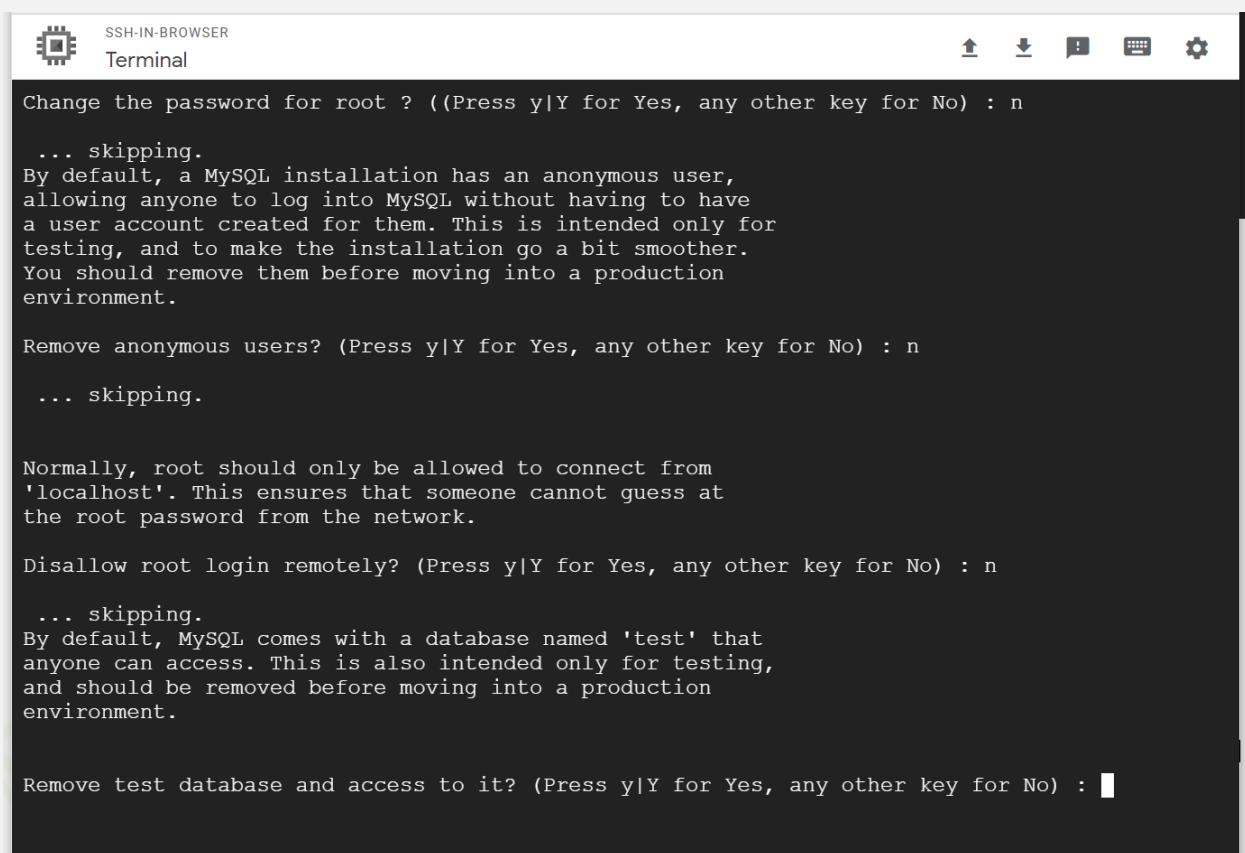
```
reading /usr/share/mecab/dic/ipadic/Noun.proper.csv ... 27328
reading /usr/share/mecab/dic/ipadic/Noun.adverbial.csv ... 795
reading /usr/share/mecab/dic/ipadic/Noun.place.csv ... 72999
reading /usr/share/mecab/dic/ipadic/Prefix.csv ... 221
reading /usr/share/mecab/dic/ipadic/Postp-col.csv ... 91
reading /usr/share/mecab/dic/ipadic/Noun.verbal.csv ... 12146
reading /usr/share/mecab/dic/ipadic/Others.csv ... 2
reading /usr/share/mecab/dic/ipadic/Interjection.csv ... 252
reading /usr/share/mecab/dic/ipadic/Noun.nai.csv ... 42
emitting double-array: 100% #####|#####
reading /usr/share/mecab/dic/ipadic/matrix.def ... 1316x1316
emitting matrix      : 100% #####|#####
done!
update-alternatives: using /var/lib/mecab/dic/ipadic-utf8 to provide /var/lib/mecab/dic/debian (mecab-dictionary) in auto mode
Setting up mysql-community-client (8.0.28-1debian10) ...
Setting up mysql-client (8.0.28-1debian10) ...
Setting up mysql-community-server (8.0.28-1debian10) ...
update-alternatives: using /etc/mysql/mysql.cnf to provide /etc/mysql/my.cnf (my.cnf) in auto mode
Created symlink /etc/systemd/system/multi-user.target.wants/mysql.service → /lib/systemd/system/mysql.service.
Setting up mysql-server (8.0.28-1debian10) ...
Processing triggers for man-db (2.8.5-2) ...
Processing triggers for libc-bin (2.28-10) ...
aystester1@aus-db02:~$ sudo systemctl restart mysql.service
aystester1@aus-db02:~$
```

Setting up the MySQL

sudo mysql_secure_installation

Inform the password: welcome1

Answer N for all questions



The screenshot shows a terminal window titled "SSH-IN-BROWSER Terminal". It displays the following MySQL setup process:

```
Change the password for root ? ((Press y|Y for Yes, any other key for No) : n
... skipping.
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : n
... skipping.

Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : n
... skipping.
By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No) : [REDACTED]
```

Downloading the .sql file:

```
wget https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-storage-db-en.sql
```

The screenshot shows a terminal window titled "SSH-IN-BROWSER Terminal". The terminal output is as follows:

```
Remove test database and access to it? (Press y|Y for Yes, any other key for No) : n
... skipping.
Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : n
... skipping.
All done!
aystester1@aus-db02:~$ wget https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-storage-db-en.sql
--2022-03-21 22:38:27--  https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-storage-db-en.sql
Resolving storage.googleapis.com (storage.googleapis.com)... 142.250.67.16, 172.217.24.48, 172.217.167.112, ...
Connecting to storage.googleapis.com (storage.googleapis.com)|142.250.67.16|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 316 [application/octet-stream]
Saving to: 'bootcamp-gcp-storage-db-en.sql'

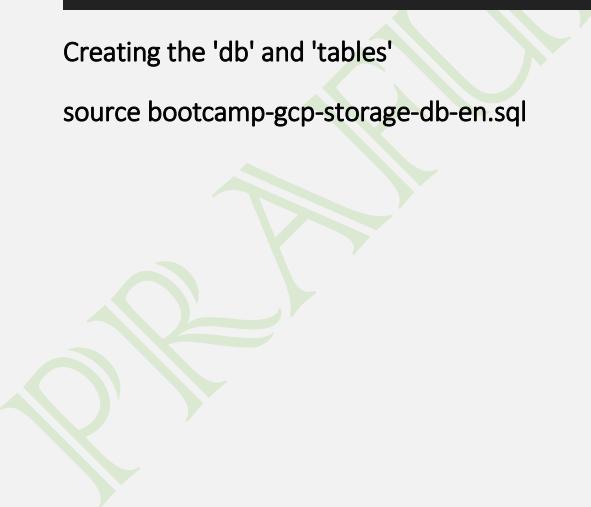
bootcamp-gcp-storage-d 100%[=====] 316 --.-KB/s in 0s
2022-03-21 22:38:28 (293 MB/s) - 'bootcamp-gcp-storage-db-en.sql' saved [316/316]

aystester1@aus-db02:~$
```

Connecting to DB

mysql -u root -p

Type the password: welcome1



SSH-IN-BROWSER Terminal

```
--2022-03-21 22:38:27-- https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-storage-db-en.sql
Resolving storage.googleapis.com (storage.googleapis.com)... 142.250.67.16, 172.217.24.48, 172.217.167.112, ...
Connecting to storage.googleapis.com (storage.googleapis.com)|142.250.67.16|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 316 [application/octet-stream]
Saving to: 'bootcamp-gcp-storage-db-en.sql'

bootcamp-gcp-storage-d 100%[=====] 316 --.-KB/s in 0s

2022-03-21 22:38:28 (293 MB/s) - 'bootcamp-gcp-storage-db-en.sql' saved [316/316]

aystester1@aus-db02:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.28 MySQL Community Server - GPL

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 
```

Creating the 'db' and 'tables'

source bootcamp-gcp-storage-db-en.sql

SSH-IN-BROWSER Terminal

```
affiliates. Other names may be trademarks of their respective owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> source bootcamp-gcp-storage-db-en.sql
Query OK, 1 row affected (0.00 sec)

Database changed
Query OK, 0 rows affected, 1 warning (0.02 sec)

+-----+
| Tables_in_clinic |
+-----+
| patient          |
+-----+
1 row in set (0.00 sec)           ↗

+-----+-----+-----+-----+-----+
| Field    | Type      | Null | Key | Default | Extra       |
+-----+-----+-----+-----+-----+
| id       | int unsigned | NO   | PRI | NULL    | auto_increment |
| name     | varchar(50)  | NO   |     | NULL    |             |
| address  | varchar(100) | NO   |     | NULL    |             |
| phone    | varchar(15)  | YES  |     | NULL    |             |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> 
```

Creating an user and changing the privileges

```
CREATE USER app@'%' IDENTIFIED BY 'welcome1';
```

```
GRANT ALL PRIVILEGES ON clinic.* TO app@'%';
```

```
FLUSH PRIVILEGES;
```

```
Exit
```

SSH-IN-BROWSER
Terminal

```

Query OK, 0 rows affected, 1 warning (0.02 sec)

+-----+
| Tables_in_clinic |
+-----+
| patient          |
+-----+
1 row in set (0.00 sec)

+-----+-----+-----+-----+-----+
| Field      | Type       | Null | Key | Default | Extra        |
+-----+-----+-----+-----+-----+
| id         | int unsigned | NO   | PRI | NULL    | auto_increment |
| name       | varchar(50)  | NO   |     | NULL    |               |
| address    | varchar(100) | NO   |     | NULL    |               |
| phone      | varchar(15)   | YES  |     | NULL    |               |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> CREATE USER app@'%' IDENTIFIED BY 'welcome1';
Query OK, 0 rows affected (0.02 sec)

mysql> GRANT ALL PRIVILEGES ON clinic.* TO app@'%';
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)

mysql> 

```

Create 02 instances with following settings:

GCE - Application (app)

Name: aus-app01

Region: australia

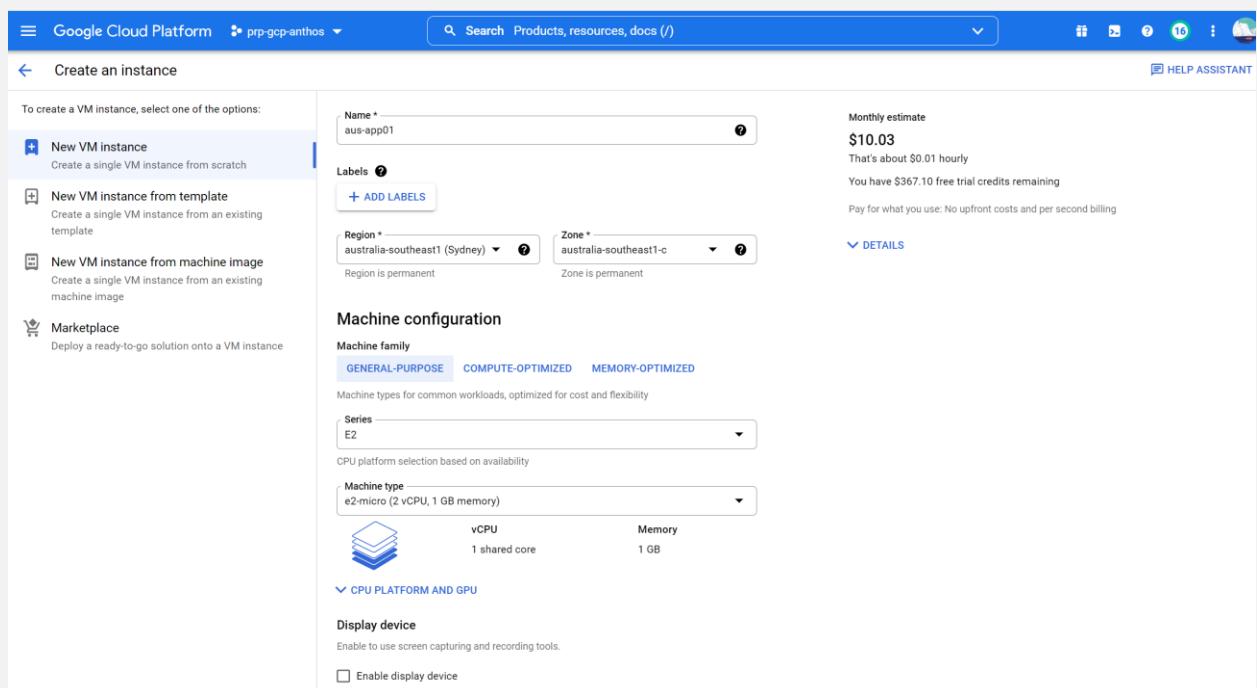
Zone: australia-southeast1-c

Size: e2-micro

OS: Debian GNU/Linux 10 (buster)

```
gcloud compute instances create aus-app01 --machine-type f1-micro --zone
australia-southeast1-c
gcloud compute instances create aus-db01 --machine-type f1-micro --zone
australia-southeast1-c
```

Setting up the VM aus-app01 for the Application



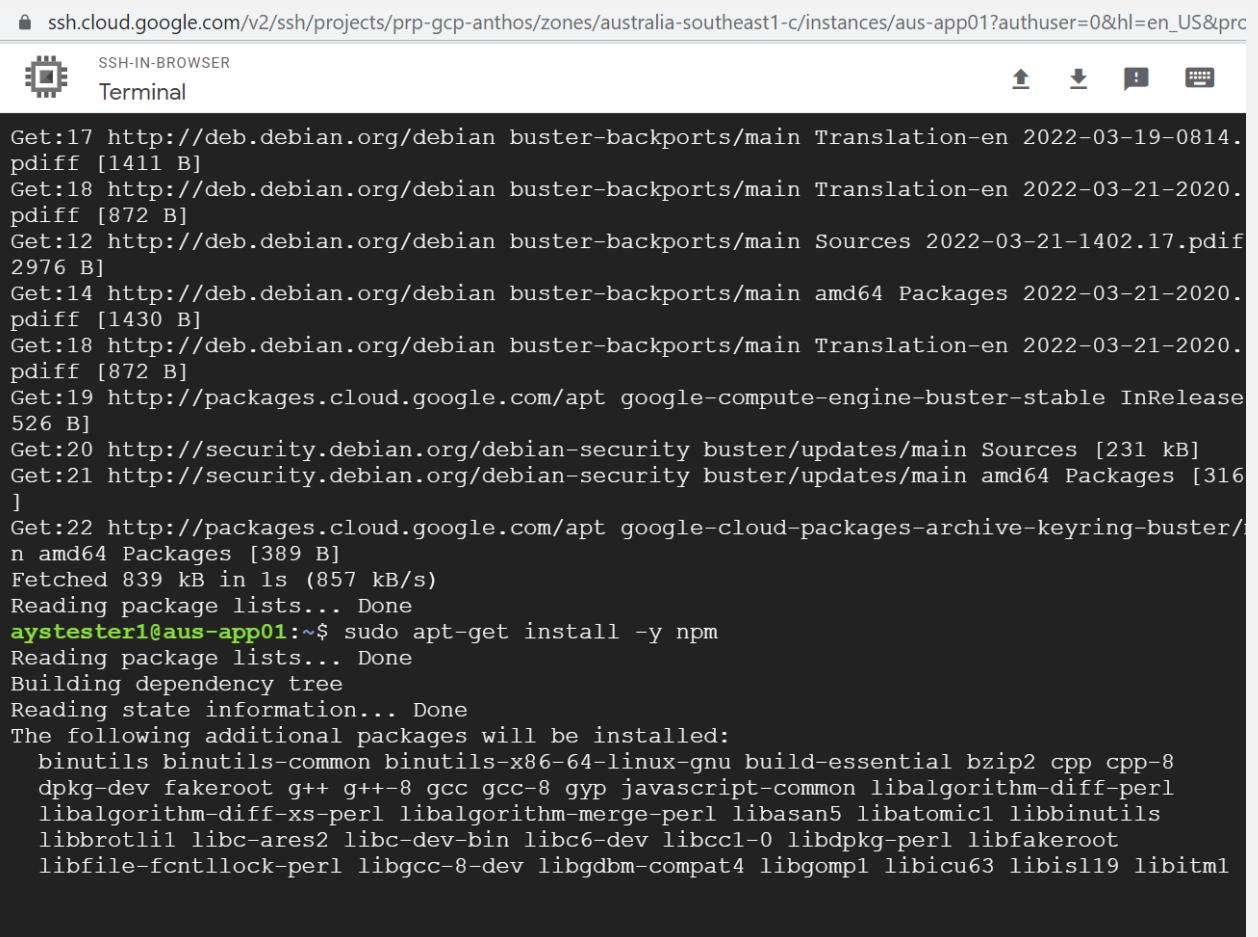
Updating the OS and the packages

```
sudo apt-get update
```

```
sudo apt-get install -y npm
```

```
sudo apt-get install -y zip
```

```
sudo apt-get install -y wget
```



```

SSH-IN-BROWSER
Terminal

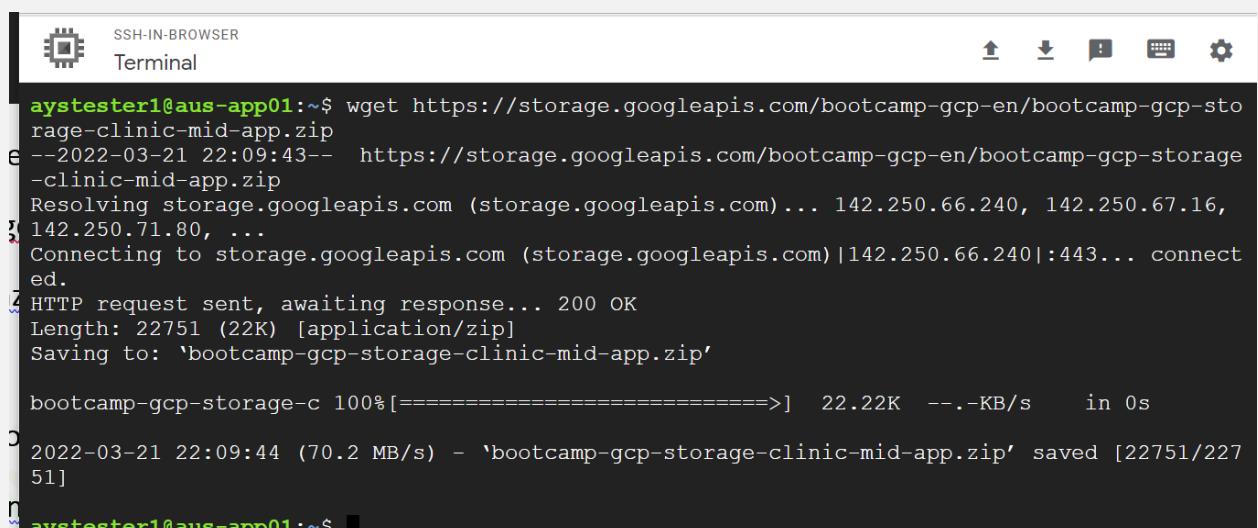
Get:17 http://deb.debian.org/debian buster-backports/main Translation-en 2022-03-19-0814.
pdiff [1411 B]
Get:18 http://deb.debian.org/debian buster-backports/main Translation-en 2022-03-21-2020.
pdiff [872 B]
Get:12 http://deb.debian.org/debian buster-backports/main Sources 2022-03-21-1402.17.pdif
2976 B]
Get:14 http://deb.debian.org/debian buster-backports/main amd64 Packages 2022-03-21-2020.
pdiff [1430 B]
Get:18 http://deb.debian.org/debian buster-backports/main Translation-en 2022-03-21-2020.
pdiff [872 B]
Get:19 http://packages.cloud.google.com/apt google-compute-engine-buster-stable InRelease
526 B]
Get:20 http://security.debian.org/debian-security buster/updates/main Sources [231 kB]
Get:21 http://security.debian.org/debian-security buster/updates/main amd64 Packages [316
]
Get:22 http://packages.cloud.google.com/apt google-cloud-packages-archive-keyring-buster/
n amd64 Packages [389 B]
Fetched 839 kB in 1s (857 kB/s)
Reading package lists... Done
aystester1@aus-app01:~$ sudo apt-get install -y npm
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
binutils binutils-common binutils-x86-64-linux-gnu build-essential bzip2 cpp cpp-8
dpkg-dev fakeroot g++ g++-8 gcc gcc-8 gyp javascript-common libalgorithm-diff-perl
libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan5 libatomic1 libbinutils
libbrotli1 libc-ares2 libc-dev-bin libc6-dev libcc1-0 libdpkg-perl libfakeroot
libfile-fcntllock-perl libgcc-8-dev libgdbm-compat4 libgomp1 libicu63 libis119 libitm1

```

Creating and accessing the folder to download application files (Node.js)

`wget https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-storage-clinic-mid-app.zip`

`unzip bootcamp-gcp-storage-clinic-mid-app.zip`



```

SSH-IN-BROWSER
Terminal

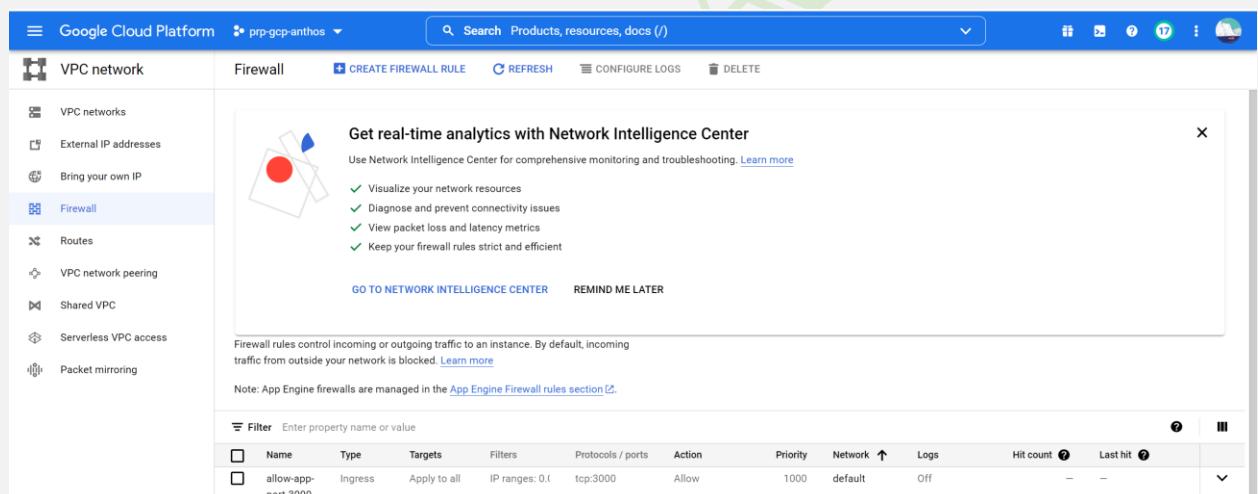
aystester1@aus-app01:~$ wget https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-storage-clinic-mid-app.zip
--2022-03-21 22:09:43-- https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-storage-clinic-mid-app.zip
Resolving storage.googleapis.com (storage.googleapis.com)... 142.250.66.240, 142.250.67.16, 142.250.71.80, ...
Connecting to storage.googleapis.com (storage.googleapis.com)|142.250.66.240|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 22751 (22K) [application/zip]
Saving to: 'bootcamp-gcp-storage-clinic-mid-app.zip'

bootcamp-gcp-storage-c 100%[=====] 22.22K --.-KB/s in 0s

2022-03-21 22:09:44 (70.2 MB/s) - 'bootcamp-gcp-storage-clinic-mid-app.zip' saved [22751/22751]

aystester1@aus-app01:~$ 

```



Google Cloud Platform prp-gcp-anthos

VPC network Firewall CREATE FIREWALL RULE REFRESH CONFIGURE LOGS DELETE

Get real-time analytics with Network Intelligence Center

Use Network Intelligence Center for comprehensive monitoring and troubleshooting. [Learn more](#)

- ✓ Visualize your network resources
- ✓ Diagnose and prevent connectivity issues
- ✓ View packet loss and latency metrics
- ✓ Keep your firewall rules strict and efficient

GO TO NETWORK INTELLIGENCE CENTER REMIND ME LATER

Firewall rules control incoming or outgoing traffic to an instance. By default, incoming traffic from outside your network is blocked. [Learn more](#)

Note: App Engine firewalls are managed in the [App Engine Firewall rules section](#).

Name	Type	Targets	Filters	Protocols / ports	Action	Priority	Network	Logs	Hit count	Last hit
allow-app-port-3000	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:3000	Allow	1000	default	Off	-	-

Npm install

```

aystester1@aus-app01:~/bootcamp-gcp-storage-clinic-mid-app/src$ cd ..
aystester1@aus-app01:~/bootcamp-gcp-storage-clinic-mid-app$ node src/index.js
internal/modules/cjs/loader.js:638
    throw err;
^

Error: Cannot find module 'express'
    at Function.Module._resolveFilename (internal/modules/cjs/loader.js:636:15)
    at Function.Module._load (internal/modules/cjs/loader.js:562:25)
    at Module.require (internal/modules/cjs/loader.js:692:17)
    at require (internal/modules/cjs/helpers.js:25:18)
    at Object.<anonymous> (/home/aystester1/bootcamp-gcp-storage-clinic-mid-app/src/index.js:1:17)
    at Module._compile (internal/modules/cjs/loader.js:778:30)
    at Object.Module._extensions..js (internal/modules/cjs/loader.js:789:10)
    at Module.load (internal/modules/cjs/loader.js:653:32)
    at tryModuleLoad (internal/modules/cjs/loader.js:593:12)
    at Function.Module._load (internal/modules/cjs/loader.js:585:3)
aystester1@aus-app01:~/bootcamp-gcp-storage-clinic-mid-app$ npm install
npm WARN npm npm does not support Node.js v10.24.0
npm WARN npm You should probably upgrade to a newer version of node as we
npm WARN npm can't make any promises that npm will work with this version.
npm WARN npm Supported releases of Node.js are the latest release of 4, 6, 7, 8, 9.
npm WARN npm You can find the latest version at https://nodejs.org/
npm WARN clinic-mib-crud@1.0.0 No repository field.
npm WARN clinic-mib-crud@1.0.0 No license field.

added 62 packages from 50 contributors in 1.849s
aystester1@aus-app01:~/bootcamp-gcp-storage-clinic-mid-app$ █

```

Grab the private ip of db02

10.152.0.5

	Status	Name	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input type="checkbox"/>	✓	aus-app01	australia-southeast1-c			10.152.0.4 (nic0)	34.151.85.55	SSH
<input type="checkbox"/>	✓	aus-db02	australia-southeast1-c			10.152.0.5 (nic0)	34.116.107.220	SSH
<input type="checkbox"/>	○	migration-vm	us-east1-b			10.142.0.12 (nic0)	None	SSH

From the folder bootcamp-gcp-storage-clinic-mid-app,

run the following command to edit the file index.js

nano src/index.js

Once inside of the file index.js, in the Middlewares section, replace:

host: to the Private IP address of the vm aus-db01.

user: to app,

password: to welcome1

database: to clinic

```

SSH-IN-BROWSER
Terminal
GNU nano 3.2          index.js          Modified
` app.set('port', process.env.PORT || 3000);
` app.set('views', path.join(__dirname, 'views'));
` app.set('view engine', 'ejs');

// middlewares
app.use(morgan('dev'));
app.use(myConnection(mysql, {
  host: '10.152.0.5', 1
  user: 'app', 2
  password: 'welcome1', 3
  port: 3306, 4
  database: 'clinic' 5
}), 'single');
app.use(express.urlencoded({extended: false}));

// routes
app.use('/', patientRoutes);

// static files
app.use(express.static(path.join(__dirname, 'public')));

// starting the server
app.listen(app.get('port'), () => {
  console.log(`server on port ${app.get('port')}`);
}

^G Get Help      ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify    ^C Cur Pos
^X Exit         ^R Read File     ^\ Replace     ^U Uncut Text   ^T To Spell   ^_ Go To Line

```

And save the file!

From bootcamp-gcp-storage-clinic-mid-app folder run the command below to install the NPM package:

`npm install`

Start the Node.js application:

`node src/index.js`

```

https://ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-app01?authuser=0&hl=en_US&...
ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-app01?authuser=0&hl=en_US&proj...
SSH-IN-BROWSER Terminal
aystester1@aus-app01:~/bootcamp-gcp-storage-clinic-mid-app$ cd src/
aystester1@aus-app01:~/bootcamp-gcp-storage-clinic-mid-app/src$ ls -ltr
total 16
drwxrwxr-x 2 aystester1 aystester1 4096 Aug 20 2021 controllers
drwxrwxr-x 2 aystester1 aystester1 4096 Aug 20 2021 routes
-rw-rw-r-- 1 aystester1 aystester1 878 Aug 20 2021 index.js
drwxrwxr-x 3 aystester1 aystester1 4096 Aug 21 2021 views
aystester1@aus-app01:~/bootcamp-gcp-storage-clinic-mid-app/src$ sudo nano index.js
aystester1@aus-app01:~/bootcamp-gcp-storage-clinic-mid-app/src$ cd ..
aystester1@aus-app01:~/bootcamp-gcp-storage-clinic-mid-app$ node src/index.js ←
server on port 3000 ←

```

Creating a firewall rule to allow the TCP access on 3000 port

From Firewall services, click on Create Firewall Rule:

Name: allow-app-port-3000

Targets: All instances in the network

Source IP ranges: 0.0.0.0/0

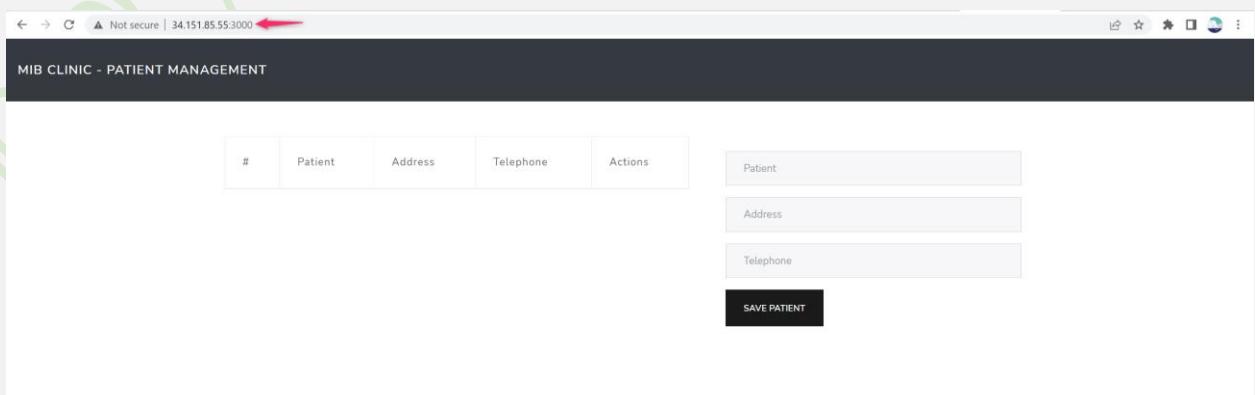
Select TCP and the port 3000

Click on Create

Copy the public IP of the instance aus-app01 and paste in the browser

Go to Browser

<http://34.151.85.55:3000/>



There isn't any data loaded into app yet.

Now, let's download and load data into the database: aus-db01.

```
wget https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-module-db.zip
```

The screenshot shows an SSH session in a browser window titled "SSH-IN-BROWSER Terminal". The terminal output is as follows:

```
aystester1@aus-db02:~$ wget https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-module-db.zip
--2022-03-21 22:53:06-- https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-module-db.zip
Resolving storage.googleapis.com (storage.googleapis.com)... 172.217.167.80, 142.250.66.240,
142.250.67.16, ...
Connecting to storage.googleapis.com (storage.googleapis.com)|172.217.167.80|:443... connected.
HTTP request sent, awaiting response... 200 OK ←
Length: 23244 (23K) [application/zip]
Saving to: 'bootcamp-gcp-module-db.zip'

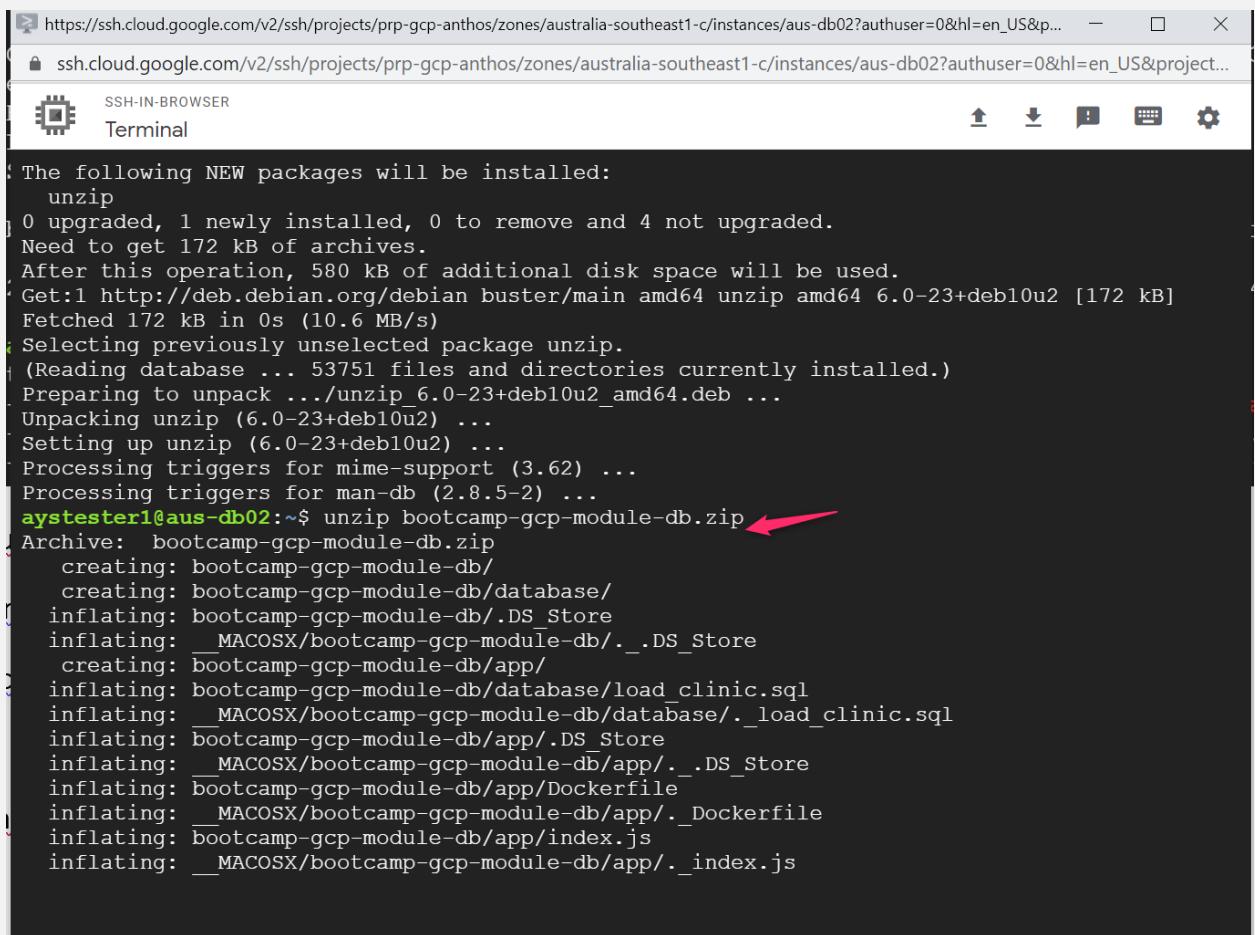
bootcamp-gcp-module-db 100%[=====] 22.70K --.-KB/s   in 0s ←
2022-03-21 22:53:07 (70.0 MB/s) - 'bootcamp-gcp-module-db.zip' saved [23244/23244]

aystester1@aus-db02:~$ ls -ltr
total 64
-rw-r--r-- 1 aystester1 aystester1 35560 Apr 29 2019 mysql-apt-config_0.8.13-1_all.deb
-rw-r--r-- 1 aystester1 aystester1 316 Aug 20 2021 bootcamp-gcp-storage-db-en.sql ←
-rw-r--r-- 1 aystester1 aystester1 23244 Aug 30 2021 bootcamp-gcp-module-db.zip ←
```

Red arrows highlight the "200 OK" status code, the progress bar during download, and the file name in the ls command output.

```
sudo apt-get install unzip
```

```
unzip bootcamp-gcp-module-db.zip
```



The following NEW packages will be installed:
unzip
0 upgraded, 1 newly installed, 0 to remove and 4 not upgraded.
Need to get 172 kB of archives.
After this operation, 580 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian buster/main amd64 unzip amd64 6.0-23+deb10u2 [172 kB]
Fetched 172 kB in 0s (10.6 MB/s)
Selecting previously unselected package unzip.
(Reading database ... 53751 files and directories currently installed.)
Preparing to unpack .../unzip_6.0-23+deb10u2_amd64.deb ...
Unpacking unzip (6.0-23+deb10u2) ...
Setting up unzip (6.0-23+deb10u2) ...
Processing triggers for mime-support (3.62) ...
Processing triggers for man-db (2.8.5-2) ...
aystester1@aus-db02:~\$ unzip bootcamp-gcp-module-db.zip ↗
Archive: bootcamp-gcp-module-db.zip
 creating: bootcamp-gcp-module-db/
 creating: bootcamp-gcp-module-db/database/
 inflating: bootcamp-gcp-module-db/.DS_Store
 inflating: __MACOSX/bootcamp-gcp-module-db/.DS_Store
 creating: bootcamp-gcp-module-db/app/
 inflating: bootcamp-gcp-module-db/database/load_clinic.sql
 inflating: __MACOSX/bootcamp-gcp-module-db/database/_load_clinic.sql
 inflating: bootcamp-gcp-module-db/app/.DS_Store
 inflating: __MACOSX/bootcamp-gcp-module-db/app/.DS_Store
 inflating: bootcamp-gcp-module-db/app/Dockerfile
 inflating: __MACOSX/bootcamp-gcp-module-db/app/_Dockerfile
 inflating: bootcamp-gcp-module-db/app/index.js
 inflating: __MACOSX/bootcamp-gcp-module-db/app/_index.js

cd ~/bootcamp-gcp-module-db/database

verify that all the data is going to be installed into database



```

https://ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&projectNumber=...
ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&projectNumber=25...
SSH-IN-BROWSER
Terminal

total 72
-rw-r--r-- 1 aystester1 aystester1 35560 Apr 29 2019 mysql-apt-config_0.8.13-1_all.deb
-rw-r--r-- 1 aystester1 aystester1 316 Aug 20 2021 bootcamp-gcp-storage-db-en.sql
drwx----- 4 aystester1 aystester1 4096 Aug 30 2021 bootcamp-gcp-module-db
-rw-r--r-- 1 aystester1 aystester1 23244 Aug 30 2021 bootcamp-gcp-module-db.zip
drwxr-xr-x 3 aystester1 aystester1 4096 Mar 21 22:54 __MACOSX
aystester1@aus-db02:~$ cd bootcamp-gcp-module-db1
aystester1@aus-db02:~/bootcamp-gcp-module-db$ ls -ltr2
total 8
drwx----- 5 aystester1 aystester1 4096 Aug 30 2021 app
drwx----- 2 aystester1 aystester1 4096 Aug 30 2021 database3
aystester1@aus-db02:~/bootcamp-gcp-module-db$ cd database4
aystester1@aus-db02:~/bootcamp-gcp-module-db/database$ ls -ltr5
total 4
-rw----- 1 aystester1 aystester1 1043 Aug 30 2021 load_clinic.sql
aystester1@aus-db02:~/bootcamp-gcp-module-db/database$ cat load_clinic.sql6
use clinic;

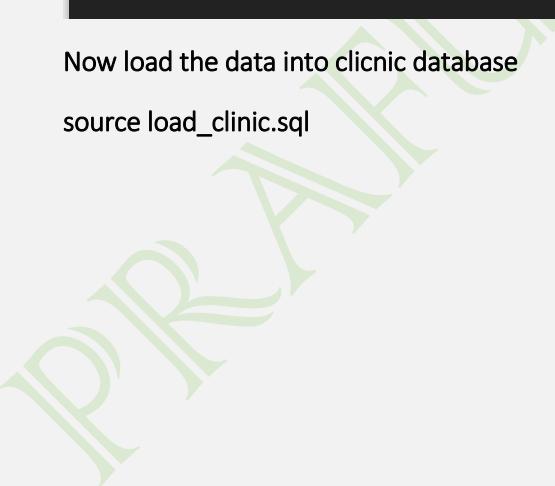
INSERT INTO patient(name,address,phone) values ('Kevin Park','76 Princess Dr. Lakeland, FL 33801','294720472');
INSERT INTO patient(name,address,phone) values ('Joe Felix','33 Holly Ave. Williamstown, NJ 08094','494720472');
INSERT INTO patient(name,address,phone) values ('Mary Phonenix','4 Arch Street Millville, NJ 08332','894720472');
INSERT INTO patient(name,address,phone) values ('Josh Koe','693 Victoria St. Macungie, PA 18062','6794720472');
INSERT INTO patient(name,address,phone) values ('Mariah Steven','74 Somerset St. Sioux Falls, SD 57103','294720472');
INSERT INTO patient(name,address,phone) values ('Mark Plum','8652 Delaware Road Inman, SC 29349','194720472');
INSERT INTO patient(name,address,phone) values ('Larry Worth','8233 Roosevelt St. Burnsville, MN 55337','538329298');
INSERT INTO patient(name,address,phone) values ('Sarah Lee','540 Sussex Rd. Ozone Park, NY 11417','9058747322');
INSERT INTO patient(name,address,phone) values ('Martin McLee','48 Blue Spring Road Milwaukee');

aystester1@aus-db02:~/bootcamp-gcp-module-db/database$ 

```

mysql -u root -p

Password: welcome1



```
SSH-IN-BROWSER
Terminal
-rw----- 1 aystester1 aystester1 1043 Aug 30 2021 load_clinic.sql
aystester1@aus-db02:~/bootcamp-gcp-module-db/database$ cat load_clinic.sql
use clinic;

INSERT INTO patient(name,address,phone) values ('Kevin Park','76 Princess Dr. Lakeland, FL 33801','294720472');
INSERT INTO patient(name,address,phone) values ('Joe Felix','33 Holly Ave. Williamstown, NJ 08094','494720472');
INSERT INTO patient(name,address,phone) values ('Mary Phonenix','4 Arch Street Millville, NJ 08332','894720472');
INSERT INTO patient(name,address,phone) values ('Josh Koe','693 Victoria St. Macungie, PA 18062','6794720472');
INSERT INTO patient(name,address,phone) values ('Mariah Steven','74 Somerset St. Sioux Falls, SD 57103','294720472');
INSERT INTO patient(name,address,phone) values ('Mark Plum','8652 Delaware Road Inman, SC 29349','194720472');
INSERT INTO patient(name,address,phone) values ('Larry Worth','8233 Roosevelt St. Burnsville, MN 55337','538329298');
INSERT INTO patient(name,address,phone) values ('Sarah Lee','540 Sussex Rd. Ozone Park, NY 11417','9058747322');
INSERT INTO patient(name,address,phone) values ('Martin McLee','48 Blue Spring Road Milwaukee');
aystester1@aus-db02:~/bootcamp-gcp-module-db/database$ mysql -u root -p
Enter password: ←
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.28 MySQL Community Server - GPL

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

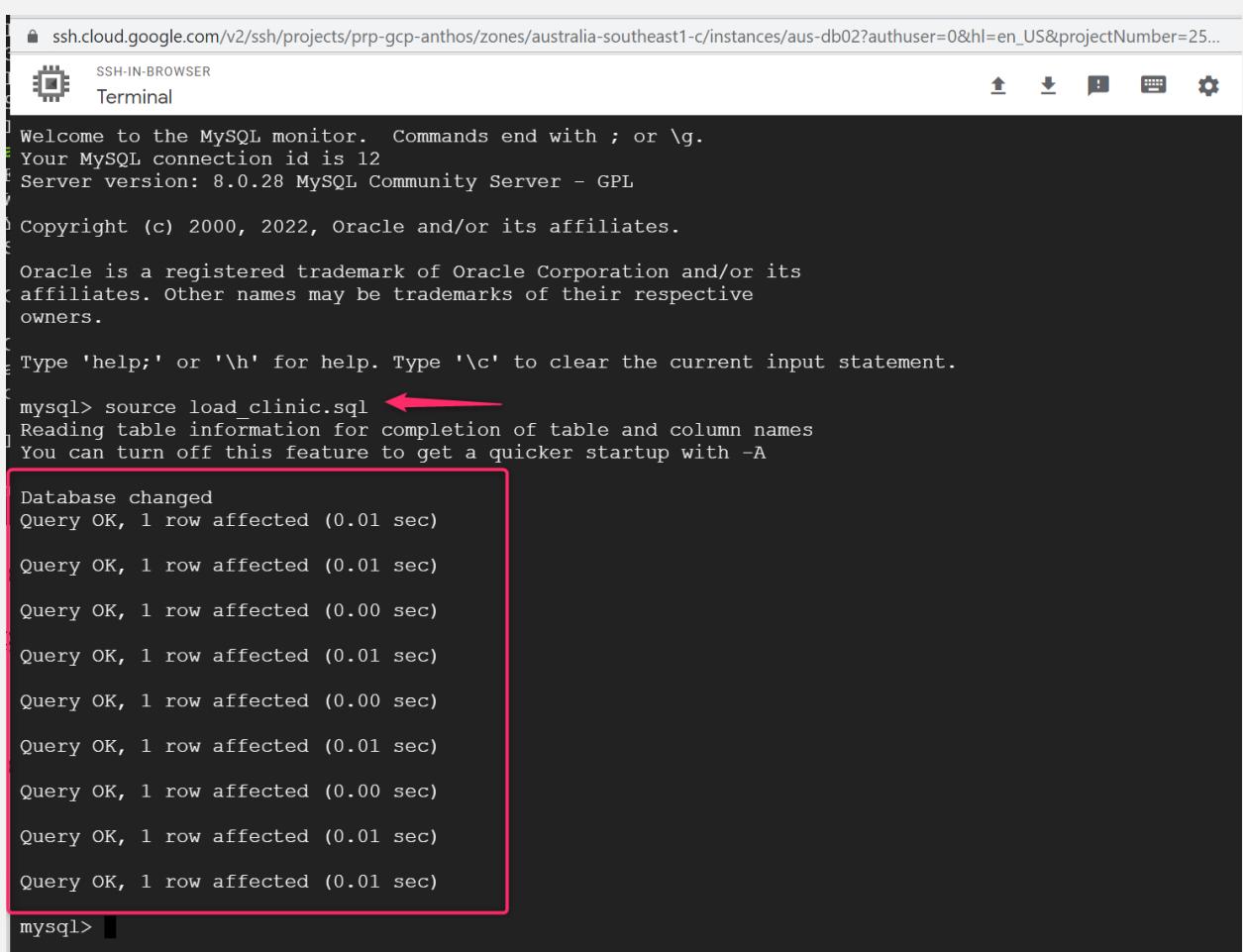
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> [REDACTED] ←
```

Now load the data into clinic database

source load_clinic.sql

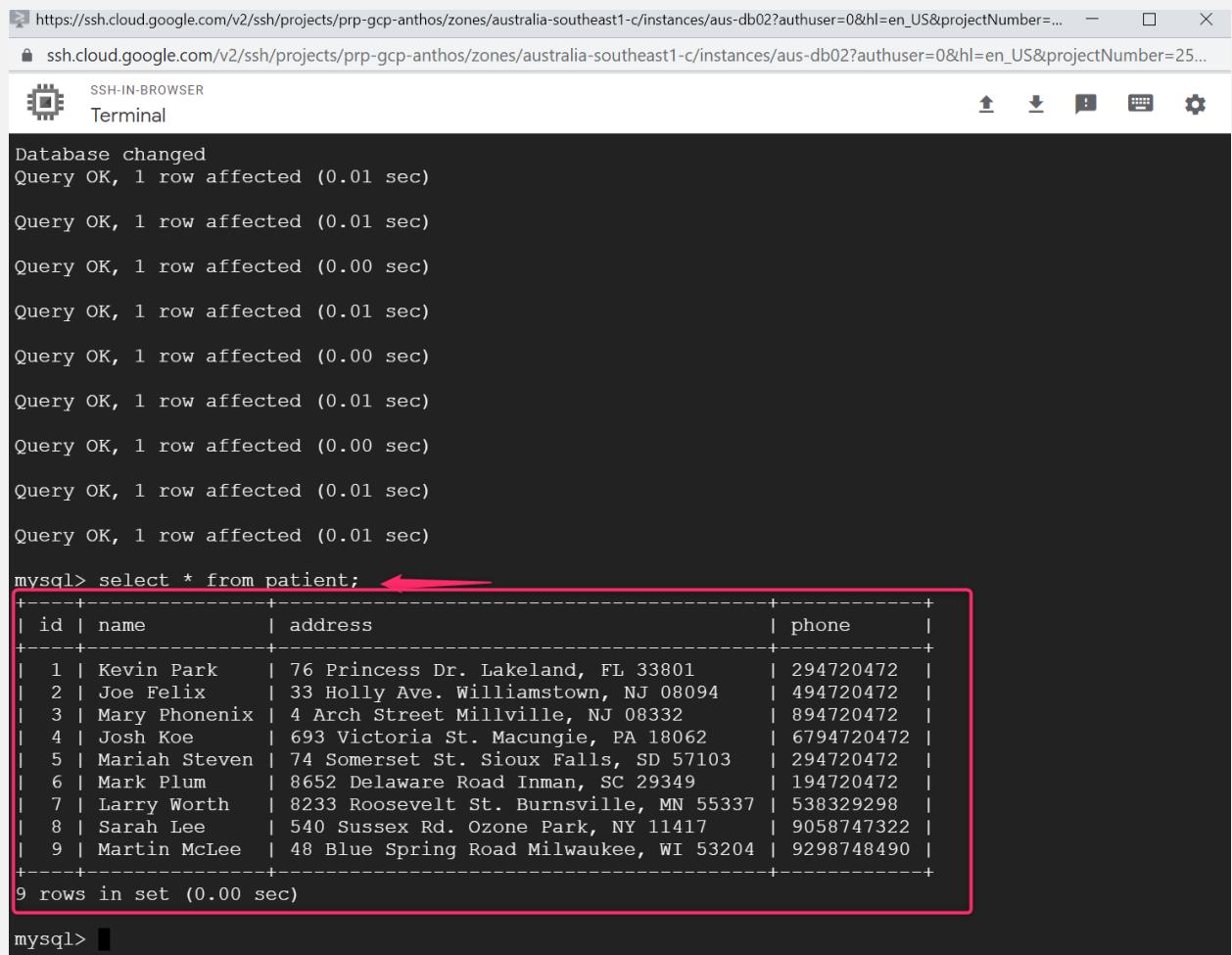


```
ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&projectNumber=25...
```

SSH-IN-BROWSER Terminal

```
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 12  
Server version: 8.0.28 MySQL Community Server - GPL  
  
Copyright (c) 2000, 2022, Oracle and/or its affiliates.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
mysql> source load_clinic.sql ←  
Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A  
  
Database changed  
Query OK, 1 row affected (0.01 sec)  
  
Query OK, 1 row affected (0.01 sec)  
  
Query OK, 1 row affected (0.00 sec)  
  
Query OK, 1 row affected (0.01 sec)  
  
Query OK, 1 row affected (0.00 sec)  
  
Query OK, 1 row affected (0.01 sec)  
  
Query OK, 1 row affected (0.00 sec)  
  
Query OK, 1 row affected (0.01 sec)  
  
Query OK, 1 row affected (0.01 sec)  
  
mysql>
```

Verify that tables contains some data



The screenshot shows an SSH session in a browser terminal window. The user has run several MySQL queries against a database named 'patient'. The results of the last query, which selects all columns from the 'patient' table, are highlighted with a red box. An arrow points from the text 'Now, access the application via browser, and check if the records were loaded successfully.' to the highlighted query result.

```
Database changed
Query OK, 1 row affected (0.01 sec)

Query OK, 1 row affected (0.01 sec)

Query OK, 1 row affected (0.00 sec)

Query OK, 1 row affected (0.01 sec)

Query OK, 1 row affected (0.00 sec)

Query OK, 1 row affected (0.01 sec)

Query OK, 1 row affected (0.00 sec)

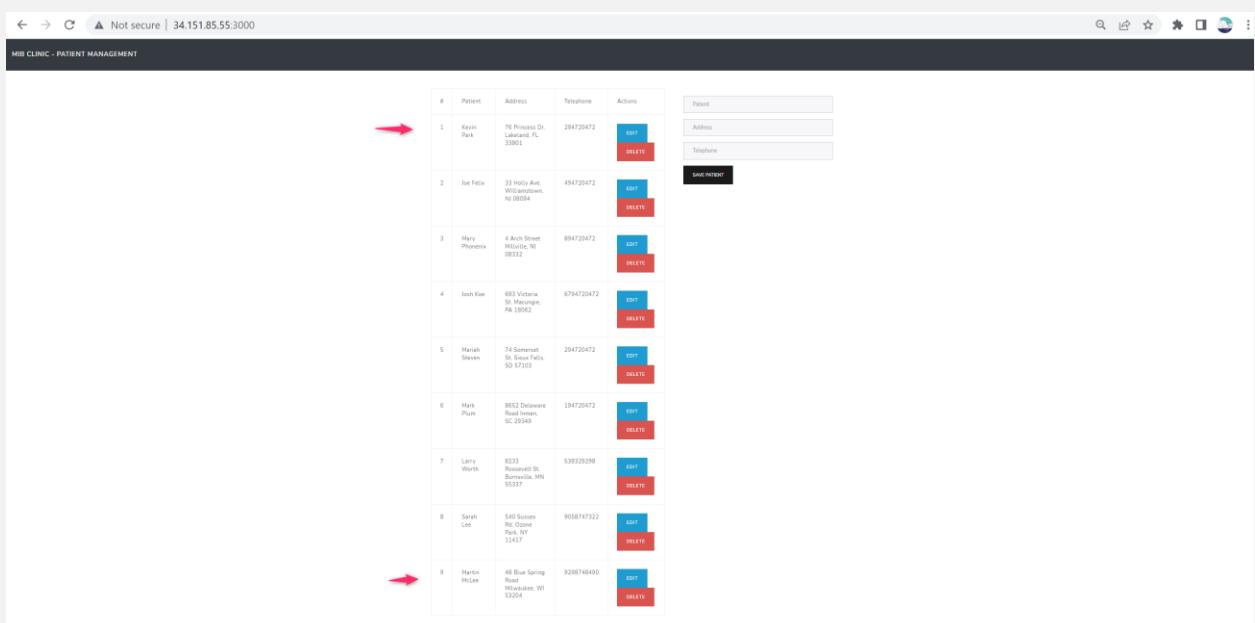
Query OK, 1 row affected (0.01 sec)

Query OK, 1 row affected (0.01 sec)

mysql> select * from patient;
+----+-----+-----+-----+
| id | name      | address          | phone   |
+----+-----+-----+-----+
| 1  | Kevin Park | 76 Princess Dr. Lakeland, FL 33801 | 294720472 |
| 2  | Joe Felix   | 33 Holly Ave. Williamstown, NJ 08094 | 494720472 |
| 3  | Mary Phonex | 4 Arch Street Millville, NJ 08332 | 894720472 |
| 4  | Josh Koe    | 693 Victoria St. Macungie, PA 18062 | 6794720472 |
| 5  | Mariah Steven | 74 Somerset St. Sioux Falls, SD 57103 | 294720472 |
| 6  | Mark Plum   | 8652 Delaware Road Inman, SC 29349 | 194720472 |
| 7  | Larry Worth  | 8233 Roosevelt St. Burnsville, MN 55337 | 538329298 |
| 8  | Sarah Lee    | 540 Sussex Rd. Ozone Park, NY 11417 | 9058747322 |
| 9  | Martin McLee | 48 Blue Spring Road Milwaukee, WI 53204 | 9298748490 |
+----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql>
```

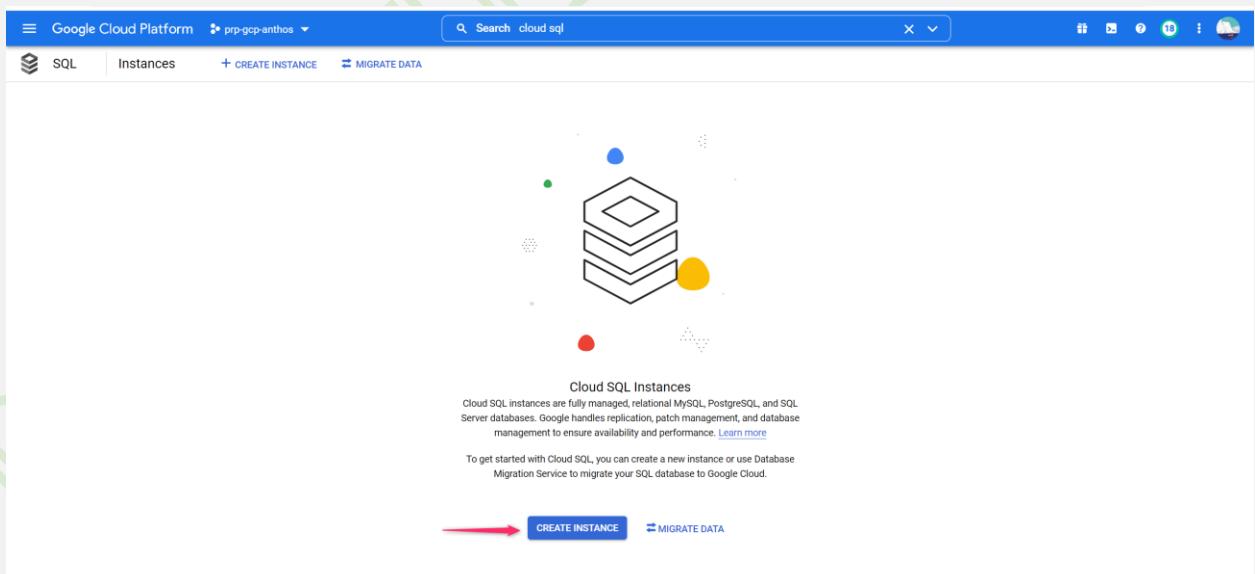
Now, access the application via browser, and check if the records were loaded successfully.

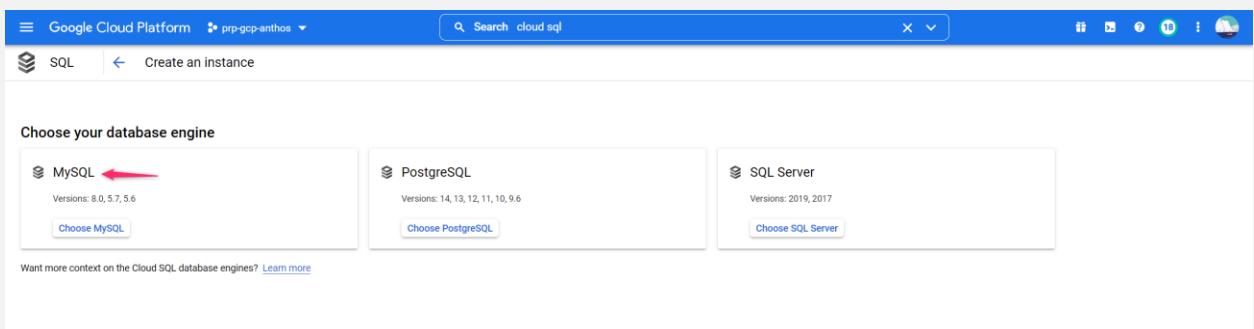


#	Patient	Address	Telephone	Actions
1	Karen Park	78 Princess Dr, Lakeland, FL 33801	234720472	EDIT DELETE
2	Joe Felix	33 Holly Ave, Indianapolis, IN 46204	494720472	EDIT DELETE
3	Mary Phoenix	4 Arch Street, Philadelphia, PA 19102	694720472	EDIT DELETE
4	Josh Koe	693 Victoria Avenue, PA 19002	6794720472	EDIT DELETE
5	Mariah Stevens	24 Somerset Drive, Falls, ID 83110	234720472	EDIT DELETE
6	Mark Blue	895 Delaware Road, Greenville, SC 29610	194720472	EDIT DELETE
7	Larry Worth	1011 Roosevelt St, Burnsville, MN 55337	530329298	EDIT DELETE
8	Sarah Lee	507 Orange Rd, Orange Park, NY 13420	9058747322	EDIT DELETE
9	Marta McLee	All Blue Spring Road, Milwaukee, WI 53204	9298748490	EDIT DELETE

Modernization process to Cloud SQL + Kubernetes

Creating Cloud SQL instance (Australia region)





Instance ID: tcb-gcp-aus-db01;

Password: welcome1;

Version: MySQL 5.7;

Region: (australia-southeast1);

Enable HA: Multiple Zones;

Primary zone: australia-southeast1-a

Secondary zone: australia-southeast1-b

The screenshot shows the 'Create a MySQL instance' form. The 'Instance info' section includes fields for Instance ID (tcb-gcp-aus-db01), Password (redacted), and Database version (MySQL 5.7). The 'Choose region and zonal availability' section includes a Region dropdown (australia-southeast1 (Sydney)) and a Zonal availability section with 'Multiple zones (Highly available)' selected. The 'Summary' section provides details about the instance configuration, such as Region (australia-southeast1 (Sydney)), DB Version (MySQL 5.7), vCPUs (4 vCPU), Memory (26 GB), Storage (100 GB), Network throughput (1,000 of 2,000 MB/s), Disk throughput (Read: 48.0 of 240.0 MB/s, Write: 48.0 of 240.0 MB/s), IOPS (Read: 3,000 of 15,000, Write: 3,000 of 15,000), Connections (Public IP), Backup (Automated), Availability (Multiple zones (Highly available)), and Point-in-time recovery (Enabled).

Customize your instance;

Machine type -> Select Lightweight;

Storage -> Storage capacity -> Select 10GB;

Machine type

- Lightweight (1)
- 1 vCPU, 3.75 GB
- 2 vCPU, 3.75 GB
- 4 vCPU, 3.75 GB
- Custom

Storage

Storage type: Choice is permanent. Storage type affects performance.

- SSD (Recommended) (2)
- HDD

Storage capacity: 10 - 65,536 GB. Higher capacity improves performance, up to the limits set by the machine type. Capacity can't be decreased later.

- 10 GB (2)
- 20 GB
- 100 GB
- 200 GB
- Custom

Summary

Region	australia-southeast1 (Sydney)
DB Version	MySQL 5.7
vCPUs	1 vCPU
Memory	3.75 GB
Storage	10 GB
Network throughput (MB/s)	250 of 2,000
Disk throughput (MB/s)	Read: 4.8 of 240.0 Write: 4.8 of 72.0
IOPS	Read: 300 of 15,000 Write: 300 of 4,500
Connections	Public IP
Backup	Automated
Availability	Multiple zones (Highly available)
Point-in-time recovery	Enabled

Connectivity

Unckeck Public IP;

Select Private IP;

Network, select default;

Click on Set Up Connection;

Click on Enable API to enable API Service Networking;

Allocate an IP range -> Select Use an automatically allocated IP range.

Click on Continue;

Click on Create Connection;

Private services access connection:

- Are per VPC network and can be used across all managed services such as Memystore, Tensorflow and SQL.
- Are between your VPC network and network owned by Google using a VPC peering, enabling your instances and services to communicate exclusively by using internal IP addresses.
- Create an isolated project for you on the service-producer side, meaning no other customers share it. You will be billed for only the resources you provision.

SHOW DIAGRAM

Enable Service Networking API

Allocate an IP range

Create a connection

Network default
Allocated IP range default-ip-range (automatically allocated)

CREATE CONNECTION CANCEL

Private services access connection required

Your network "default" requires a private services access connection. This connection enables your services to communicate exclusively by using internal IP addresses. [Learn more](#)

SET UP CONNECTION

SHOW ALLOCATED IP RANGE OPTION

Public IP

Assigns an external, internet-accessible IP address. Requires using an authorized network or the Cloud SQL Proxy to connect to this instance. [Learn more](#)

Authorized networks

You can specify CIDR ranges to allow IP addresses in those ranges to access your instance. [Learn more](#)

Summary

Region	australia-southeast1 (Sydney)
DB Version	MySQL 5.7
vCPUs	1 vCPU
Memory	3.75 GB
Storage	10 GB
Network throughput (MB/s)	250 of 2,000
Disk throughput (MB/s)	Read: 4.8 of 240.0 Write: 4.8 of 72.0
IOPS	Read: 300 of 15,000 Write: 300 of 4,500
Connections	Private IP
Backup	Automated
Availability	Multiple zones (Highly available)
Point-in-time recovery	Enabled

Associated networking

Select a network to create a private connection

Network * default

Private services access connection for network default has been successfully created. You will now be able to use the same network across all your project's managed services. If you would like to change this connection, please visit the [Networking page](#).

SHOW ALLOCATED IP RANGE OPTION

Public IP

Assigns an external, internet-accessible IP address. Requires using an authorized network or the Cloud SQL Proxy to connect to this instance. [Learn more](#)

Authorized networks

You can specify CIDR ranges to allow IP addresses in those ranges to access your instance. [Learn more](#)

Backups

Automatic backups enabled. Point-in-time recovery (via binary logs) enabled.

Maintenance

Updates may occur any day of the week. Cloud SQL chooses the maintenance timing.

Flags

No flags set

Click on Create Instance.

Summary

Region	australia-southeast1 (Sydney)
DB Version	MySQL 5.7
vCPUs	1 vCPU
Memory	3.75 GB
Storage	10 GB
Network throughput (MB/s)	250 of 2,000
Disk throughput (MB/s)	Read: 4.8 of 240.0 Write: 4.8 of 72.0
IOPS	Read: 300 of 15,000 Write: 300 of 4,500
Connections	Private IP
Backup	Automated
Availability	Multiple zones (Highly available)
Point-in-time recovery	Enabled

After created the Cloud SQL instance. Click on Databases. Then, click on Create Database.

Databases

All instances > tcb-gcp-aus-db01 **tcb-gcp-aus-db01** (red arrow)

+ CREATE DATABASE

Name	Collation	Character set	Type
information_schema	utf8_general_ci	utf8	System
mysql	utf8_general_ci	utf8	System
performance_schema	utf8_general_ci	utf8	System
sys	utf8_general_ci	utf8	System

Uploads and prp-gcp-anthos operations
Created tcb-gcp-aus-db01 5:39:38 PM GMT-6

Database name: clinic;

Click on Create.

Create a database

Database Name * clinic

Character set * utf8

Collation Default collation

CREATE CANCEL

Creating an user in the Cloud SQL instance

Click on Users.

Add user account:

Username: app

Password: welcome1

Host name | Allow any host (%)

Add a user account to instance tcb-gcp-aus-db01

Choose how to authenticate

Built-in authentication

User name * app

Password (Optional) *****

Host name * Allow any host (%)

Cloud IAM

ADD CANCEL

Click on Add

Google Cloud Platform - prp-gcp-anthos - Search cloud sql

SQL

Users

PRIMARY INSTANCE

All instances > tcb-gcp-aus-db01

tcb-gcp-aus-db01

MySQL 5.7

Overview Connections Users Databases Backups Replicas Operations

User accounts enable users and applications to connect to your instance. [Learn more](#)

ADD USER ACCOUNT

User name	Host name	Authentication
app	% (any host)	Built-in
root	% (any host)	Built-in

Containerizing (docker) the application

Open the Cloud Shell

Download the application files

```
wget https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-module-db.zip
```

```
unzip bootcamp-gcp-module-db.zip
```

Google Cloud Platform - prp-gcp-anthos - Search cloud sql

SQL

Users

PRIMARY INSTANCE

All instances > tcb-gcp-aus-db01

tcb-gcp-aus-db01

MySQL 5.7

Overview Connections Users Databases Backups Release Notes

User accounts enable users and applications to connect to your instance. [Learn more](#)

ADD USER ACCOUNT

User name	Host name	Authentication
app	% (any host)	Built-in
root	% (any host)	Built-in

CLOUD SHELL Terminal (prp-gcp-anthos) + Open Editor

```
astester1@cloudshell:~ (prp-gcp-anthos)$ wget https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-module-db.zip
--2022-03-21 23:48:47-- https://storage.googleapis.com/bootcamp-gcp-en/bootcamp-gcp-module-db.zip
Resolving storage.googleapis.com (storage.googleapis.com)... 74.125.132.128, 74.125.201.128, 74.125.69.128, ...
Connecting to storage.googleapis.com (storage.googleapis.com)|74.125.132.128|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 23244 (23K) [application/zip]
Saving to: 'bootcamp-gcp-module-db.zip'

bootcamp-gcp-module-db.zip          100%[=====] 22.70K --.-KB/s   in 0s

2022-03-21 23:48:47 (76.6 MB/s) - 'bootcamp-gcp-module-db.zip' saved [23244/23244]

Archive: bootcamp-gcp-module-db.zip
  creating: bootcamp-gcp-module-db/
  creating: bootcamp-gcp-module-db/database/
  inflating: bootcamp-gcp-module-db/.DS_Store
  inflating: MACOSX/bootcamp-gcp-module-db/_DS_Store
  creating: bootcamp-gcp-module-db/app/
  inflating: bootcamp-gcp-module-db/database/load_clinic.sql
  inflating: MACOSX/bootcamp-gcp-module-db/database/load_clinic.sql
```

Creating a new image:

```
cd ~/bootcamp-gcp-module-db/app
```

```

ls -ltr
total 8
drwxr-xr-x  5 aystesterl aystesterl 4096 Aug 30 2021 app
drwxr-xr-x  2 aystesterl aystesterl 4096 Aug 30 2021 database

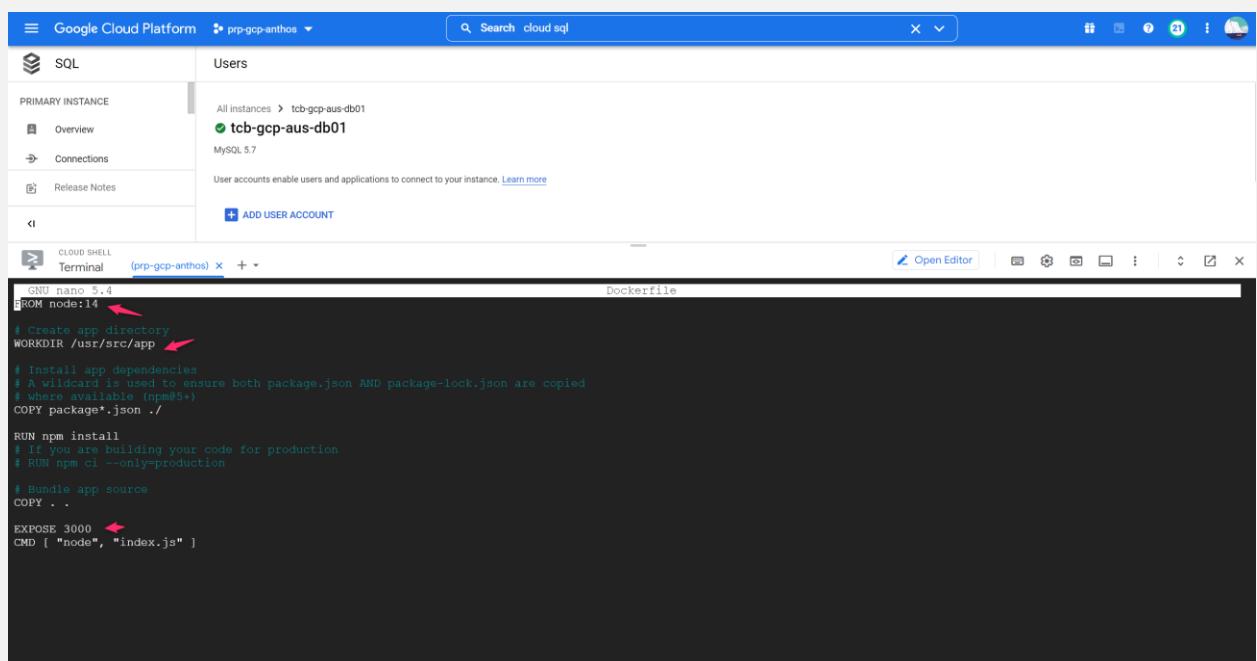
```

```

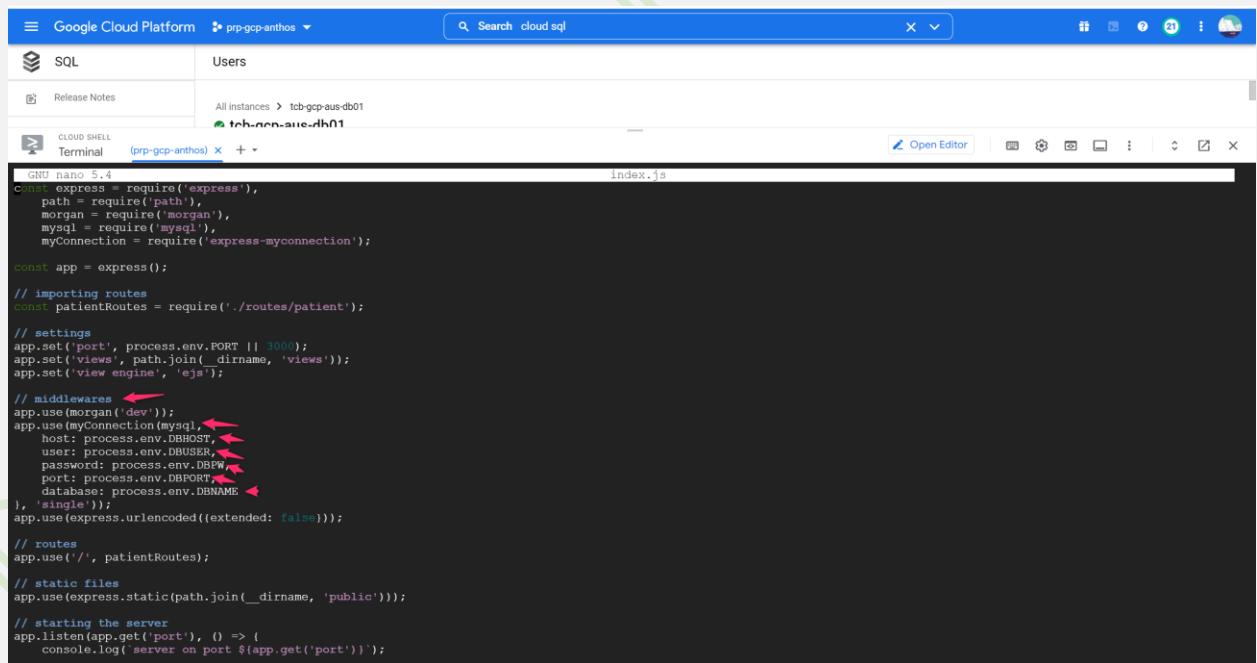
ls -ltr
total 44
-rw-rw-r-- 1 aystesterl aystesterl 17031 Feb 22 2021 package-lock.json
-rw-rw-r-- 1 aystesterl aystesterl 381 Mar  2 2021 Dockerfile
drwxrwxr-x  2 aystesterl aystesterl 4096 Aug 20 2021 controllers
drwxrwxr-x  2 aystesterl aystesterl 4096 Aug 20 2021 routes
-rw-rw-r-- 1 aystesterl aystesterl 421 Aug 20 2021 package.json
drwxrwxr-x  3 aystesterl aystesterl 4096 Aug 21 2021 views
-rw-rw-r-- 1 aystesterl aystesterl 930 Aug 30 2021 index.js

```

Let's observe Dockerfile



Lets Observe index.js file



Now let's build a docker image from app file

docker build -t tcb-clinic-app .

Don't worry about these errors:

```
ayesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ ls -ltr
total 44
-rw-r--r-- 1 ayesterl ayesterl 17031 Feb 22 2021 package-lock.json
-rw-r--r-- 1 ayesterl ayesterl 381 Mar 2 2021 Dockerfile
drwxrwxr-x 2 ayesterl ayesterl 4096 Aug 20 2021 controllers
drwxrwxr-x 2 ayesterl ayesterl 4096 Aug 20 2021 routes
-rw-rw-r-- 1 ayesterl ayesterl 421 Aug 20 2021 package.json
drwxrwxr-x 3 ayesterl ayesterl 4096 Aug 21 2021 views
-rwx----- 1 ayesterl ayesterl 930 Aug 30 2021 index.js
ayesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ docker build -t tcb-clinic-app .
Step 7/7 : FROM node:14
14: Pulling from library/node
d172cccd78ea: Pull complete
4f105b790398: Download complete
01ae525c10a7: Download complete
72215048f783: Download complete
8cfaf9c61a0ab: Download complete
7f6b791c006f: Download complete
6138e26477a6: Download complete
62d956a575b5: Download complete
8ac0f5d179f7: Download complete
[]
```

npm WARN clinica-mib-crud@1.0.0 No repository field.

npm WARN clinica-mib-crud@1.0.0 No license field.

```
4f105b790398: Pull complete
01ae525c10a7: Pull complete
72215048f783: Pull complete
8cfaf9c61a0ab: Pull complete
7f6b791c006f: Pull complete
6138e26477a6: Pull complete
62d956a575b5: Pull complete
8ac0f5d179f7: Pull complete
Digest: sha256:7d38b5e42b2ac006c3a79ef8ad9fle912bde6cb4cb4243c188689d5aalaa437
Status: downloaded newer image for node:14
--> 903c2c873e4
Step 2/7 : WORKDIR /usr/src/app
--> Running in elab4c513aff
Removing intermediate container elab4c513aff
--> e08929...
Step 3/7 : COPY package*.json .
--> 02b9e607b468
Step 4/7 : RUN npm install
--> Running in c84ce15d34ae
npm WARN clinica-mib-crud@1.0.0 No repository field.
npm WARN clinica-mib-crud@1.0.0 No license field.

added 62 packages from 50 contributors and audited 62 packages in 1.962s
found 0 vulnerabilities

Removing intermediate container c84ce15d34ae
--> e010a30cac5a
Step 5/7 : COPY .
--> 6c8843f8a454
Step 6/7 : EXPOSE 3000
--> 85351ab864cd
Removing intermediate container 53f683398d8c
--> c244cca45590
Step 7/7 : CMD [ "node", "index.js" ]
--> Running in 85351ab864cd
Removing intermediate container 85351ab864cd
--> fedd21aefb09
Successfully built fedd21aefb09
Successfully tagged tcb-clinic-app:latest
ayesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ []
```

Adding tag and uploading it to the Container Registry

docker tag tcb-clinic-app asia.gcr.io/\$DEVSHELL_PROJECT_ID/tcb-clinic-app:latest

```

Google Cloud Platform prp-gcp-anthos
Cloud Shell Terminal (prp-gcp-anthos) + ×
Search cloud sql × v
SQL Users
Release Notes All instances > tcb-gcp-aus-db01
tcb-gcp-aus-db01

01ae525c10a7: Pull complete
72215048f783: Pull complete
8cfaf9c61a0ab: Pull complete
7f6b791c006f: Pull complete
6138e264a555: Pull complete
6293a5175b5: Pull complete
8ac0f5d179f7: Pull complete
Digest: sha256:7d38b5ed42b2a006c3a79ef8ad9fle912bde6cb4cb4243c188689d5aalaa437
Status: Downloaded newer image for node:14
--> 903c2c873ea4
Step 2/7 : WORKDIR /usr/src/app
--> Running in elab4c513aff
Removing intermediate container elab4c513aff
--> a10891a25748
Step 3/7 : COPY package*.json .
--> 02b9e607b468
Step 4/7 : RUN npm install
--> Running in c84ce15d34ae
npm WARN clinic-mib-crud@1.0.0 No repository field.
npm WARN clinic-mib-crud@1.0.0 No license field.

added 62 packages from 50 contributors and audited 62 packages in 1.962s
found 0 vulnerabilities

Removing intermediate container c84ce15d34ae
--> ea10a30cac5a
Step 5/7 : COPY .
--> 6c8843f8a45f
Step 6/7 : EXPOSE 3000
--> Running in 53f683398d8c
Removing intermediate container 53f683398d8c
--> c244ca4559e
Step 7/7 : CMD [ "node", "index.js" ]
--> Running in 85351ab864cd
Removing intermediate container 85351ab864cd
--> fedd21aefb09
Successfully built fedd21aefb09
Successfully tagged tcb-clinic-app:latest
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ docker tag tcb-clinic-app asia.gcr.io/$DEVSHELL_PROJECT_ID/tcb-clinic-app:latest
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$

```

`docker push asia.gcr.io/$DEVSHELL_PROJECT_ID/tcb-clinic-app:latest`

```

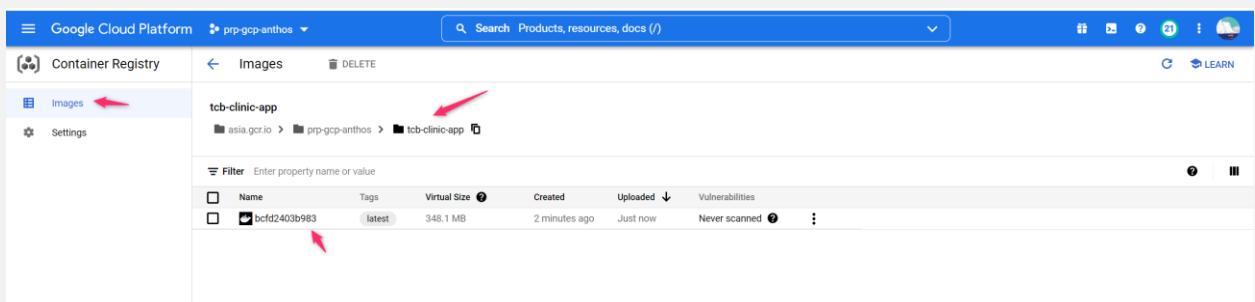
Google Cloud Platform prp-gcp-anthos
Cloud Shell Terminal (prp-gcp-anthos) + ×
Search cloud sql × v
SQL Users
Release Notes All instances > tcb-gcp-aus-db01
tcb-gcp-aus-db01

Step 4/7 : RUN npm install
--> Running in c84ce15d34ae
npm WARN clinic-mib-crud@1.0.0 No repository field.
npm WARN clinic-mib-crud@1.0.0 No license field.

added 62 packages from 50 contributors and audited 62 packages in 1.962s
found 0 vulnerabilities

Removing intermediate container c84ce15d34ae
--> ea10a30cac5a
Step 5/7 : COPY .
--> 6c8843f8a45f
Step 6/7 : EXPOSE 3000
--> Running in 53f683398d8c
Removing intermediate container 53f683398d8c
--> c244ca4559e
Step 7/7 : CMD [ "node", "index.js" ]
--> Running in 85351ab864cd
Removing intermediate container 85351ab864cd
--> fedd21aefb09
Successfully built fedd21aefb09
Successfully tagged tcb-clinic-app:latest
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ docker tag tcb-clinic-app asia.gcr.io/$DEVSHELL_PROJECT_ID/tcb-clinic-app:latest
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ docker push asia.gcr.io/$DEVSHELL_PROJECT_ID/tcb-clinic-app:latest
The push refers to repository [asia.gcr.io/prp-gcp-anthos/tcb-clinic-app]
4e8e9e096d3: Pushed
95407758d5dd: Pushed ←
e4b382af46a: Pushed ←
e3cb3a4b6d3b: Pushed ←
4e1fb3a4: Layer already exists
e7ed1a7261b: Layer already exists
b959df1ffd88: Layer already exists
27ec3cf84b82: Layer already exists
276f4f550b43: Layer already exists
f8dd5cf64c8e: Layer already exists
118470c69019: Layer already exists
7096dc109574: Layer already exists
af9da282c8c4: Layer already exists
latest: digest: sha256:bcfd2403b9830704b55b38304b3alfbeaf17962967112d022c7bba3f180fedcd size: 3050
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$

```

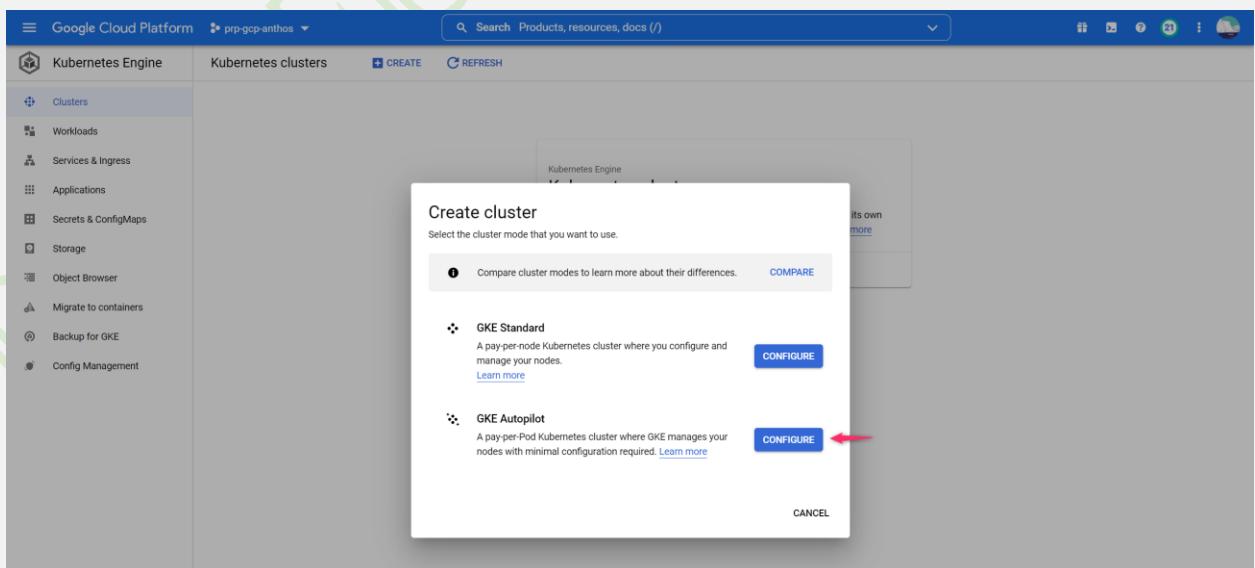
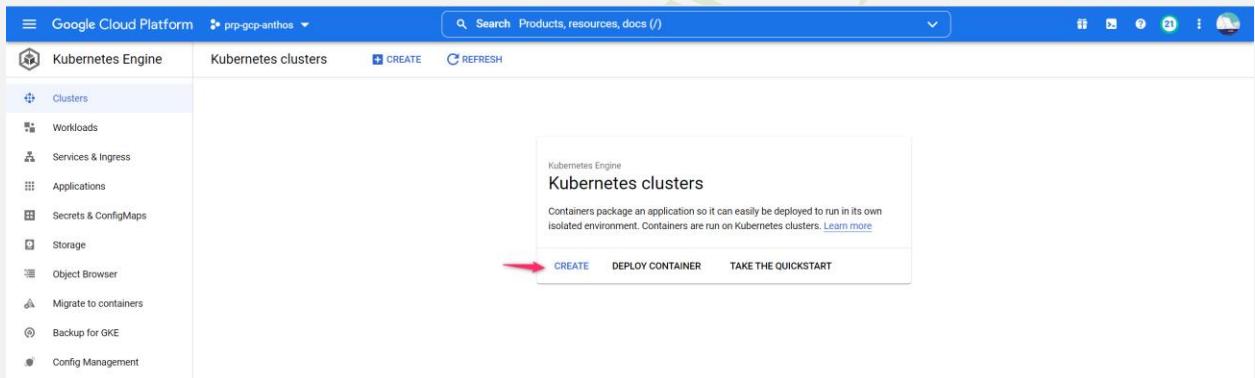


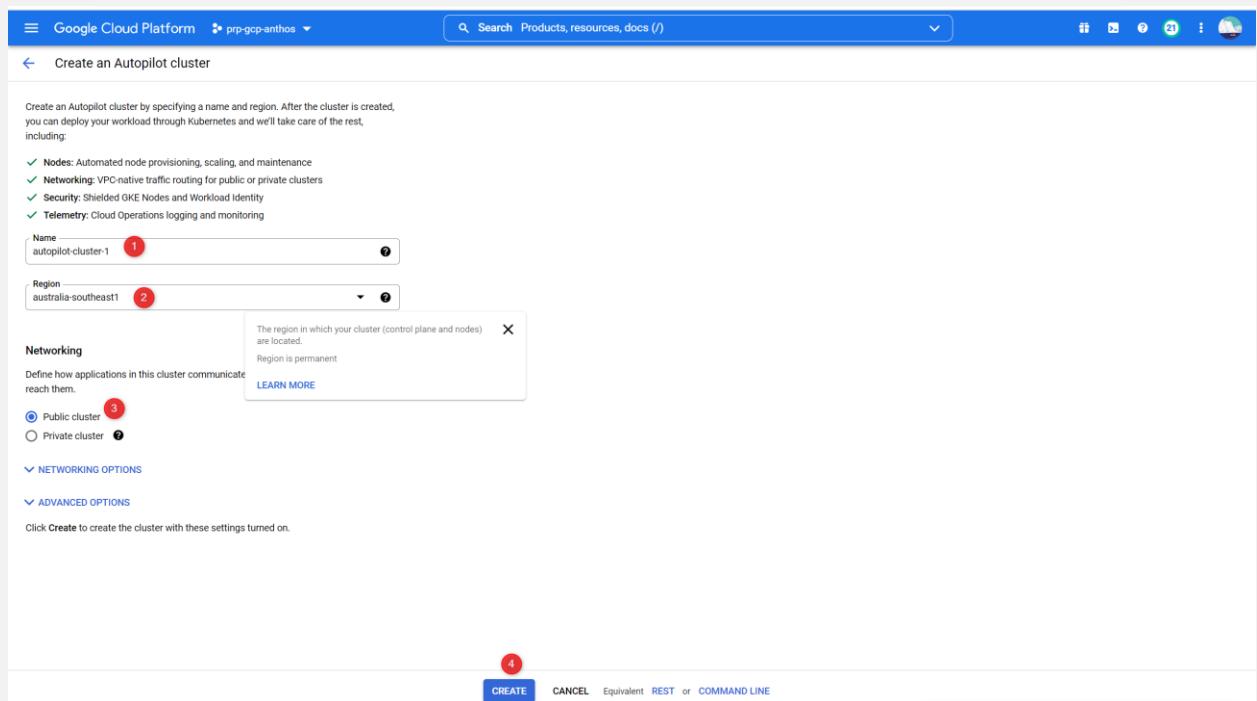
From the Kubernetes Engine services, click on [+] Create | Autopilot cluster | Configure

Name: autopilot-cluster-1

Region: australia-southeast1

Click on Create.





In the Cloud Shell, downloading the file tcb-clinic.yaml

```
 wget https://storage.googleapis.com/bootcamp-gcp-en/tcb-clinic.yaml
```

The screenshot shows a Cloud Shell terminal window. The command 'wget https://storage.googleapis.com/bootcamp-gcp-en/tcb-clinic.yaml' is being run. The output shows the file is being downloaded from storage.googleapis.com to the current directory. A red arrow points to the progress bar at the end of the terminal window.

```
ayester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ wget https://storage.googleapis.com/bootcamp-gcp-en/tcb-clinic.yaml
--2022-03-22 00:22:23-- https://storage.googleapis.com/bootcamp-gcp-en/tcb-clinic.yaml
Resolving storage.googleapis.com (storage.googleapis.com)... 142.251.6.128, 142.250.159.128, 142.251.120.128, ...
Connecting to storage.googleapis.com (storage.googleapis.com)|142.251.6.128|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1117 (1.1K) [application/x-yaml]
Saving to: 'tcb-clinic.yaml'

tcb-clinic.yaml          100%[=====]  1.09K --.-KB/s   in 0s

2022-03-22 00:22:23 (7.32 MB/s) - 'tcb-clinic.yaml' saved [1117/1117]

ayester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$
```

```

Google Cloud Platform prp-gcp-anthos Terminal (prp-gcp-anthos) x + v
Search cloud sql
SQL Users
CLOUD SHELL Terminal (prp-gcp-anthos) x + v
Open Editor
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ wget https://storage.googleapis.com/bootcamp-gcp-en/tcb-clinic.yaml
--2022-03-22 00:22:23-- https://storage.googleapis.com/bootcamp-gcp-en/tcb-clinic.yaml
Resolving storage.googleapis.com (storage.googleapis.com)... 142.251.6.128, 142.250.159.128, 142.251.120.128, ...
Connecting to storage.googleapis.com (storage.googleapis.com)|142.251.6.128|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1117 (1.1K) [application/x-yaml]
Saving to: 'tcb-clinic.yaml'

tcb-clinic.yaml          100%[=====]   1.09K --.-KB/s   in 0s

2022-03-22 00:22:23 (7.32 MB/s) - 'tcb-clinic.yaml' saved [1117/1117]

aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ ls -ltr
total 48
drwxr-x--- 1 aystesterl 17031 Feb 22 2021 package-lock.json
-rw-r--r-- 1 aystesterl 381 Mar 2 2021 Dockerfile
drwxrwxr-x 2 aystesterl aystesterl 4096 Aug 20 2021 controllers
drwxrwxr-x 2 aystesterl aystesterl 4096 Aug 20 2021 routes
-rw-rw-r-- 1 aystesterl aystesterl 421 Aug 20 2021 package.json
drwxrwxr-x 3 aystesterl aystesterl 4096 Aug 21 2021 views
-rw-r--r-- 1 aystesterl aystesterl 930 Aug 30 2021 index.js
-rw-r--r-- 1 aystesterl aystesterl 1117 Aug 30 2021 tcb-clinic.yaml
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ sudo vi tcb-clinic.yaml
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$

```

Edit the file `tcb-clinic.yaml` using the Cloud Editor, change the Private IP (DBHOST) of the Cloud SQL:

image path (image: `asia.gcr.io/<project-name>/tcb-clinic-app:latest`)

```

Google Cloud Platform prp-gcp-anthos Terminal (prp-gcp-anthos) x + v
Search cloud sql
SQL Users
CLOUD SHELL Terminal (prp-gcp-anthos) x + v
Open Editor
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ vi tcb-clinic.yaml
apiVersion: v1
kind: Service
metadata:
  name: tcb-clinic
spec:
  ports:
  - port: 3000
  selector:
    app: tcb-clinic
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: tcb-clinic
spec:
  replicas: 1
  selector:
    matchLabels:
      app: tcb-clinic
  strategy:
    rollingUpdate:
      maxSurge: 1
      maxUnavailable: 1
    minReadySeconds: 5
  template:
    metadata:
      labels:
        app: tcb-clinic
    spec:
      containers:
      - name: tcb-clinic
        image: asia.gcr.io/prp-gcp-anthos/tcb-clinic-app:latest
        ports:
        - containerPort: 3000
        resources:
          requests:
            cpu: 100m
            memory: 100Mi
        env:
          - name: DBHOST
            value: "21.0.2"
          - name: DBUSER
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$
aystesterl@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$

```

Change the DBHost name to Cloud SQL db host

```

apiVersion: v1
kind: Service
metadata:
  name: tcb-clinic
spec:
  type: LoadBalancer
  ports:
    - protocol: TCP
      port: 80
      targetPort: 3000
  selector:
    app: tcb-clinic

```

YAML configuration for tcb-clinic service:

```

rollingUpdate:
  maxSurge: 1
  maxUnavailable: 1
minReadySeconds: 5
template:
  metadata:
    labels:
      app: tcb-clinic
  spec:
    containers:
      - name: tcb-clinic
        image: asia.gcr.io/prp-gcp-anthos/tcb-clinic-app:latest
        ports:
          - containerPort: 3000
        resources:
          requests:
            cpu: 100m
            memory: 100Mi
        env:
          - name: DBHOST
            value: "10.103.240.2" ←
          - name: DBUSER
            value: "app"
          - name: DBPW
            value: "welcomeme1"
          - name: DBPORT
            value: "3306"
          - name: DBNAME
            value: "clinic"
    ...

```

Instance ID	Type	Public IP address	Private IP address	Instance connection name	High availability	Location	Storage used	Actions
tcb-gcp-aus-db01	MySQL 5.7		10.103.240.2	prp-gcp-anthos:austr...	ENABLED	australia-southeast1-a		

```

apiVersion: v1
kind: Service
metadata:
  name: tcb-clinic
spec:
  type: LoadBalancer
  ports:
    - protocol: TCP
      port: 80
      targetPort: 3000
  selector:
    app: tcb-clinic

```

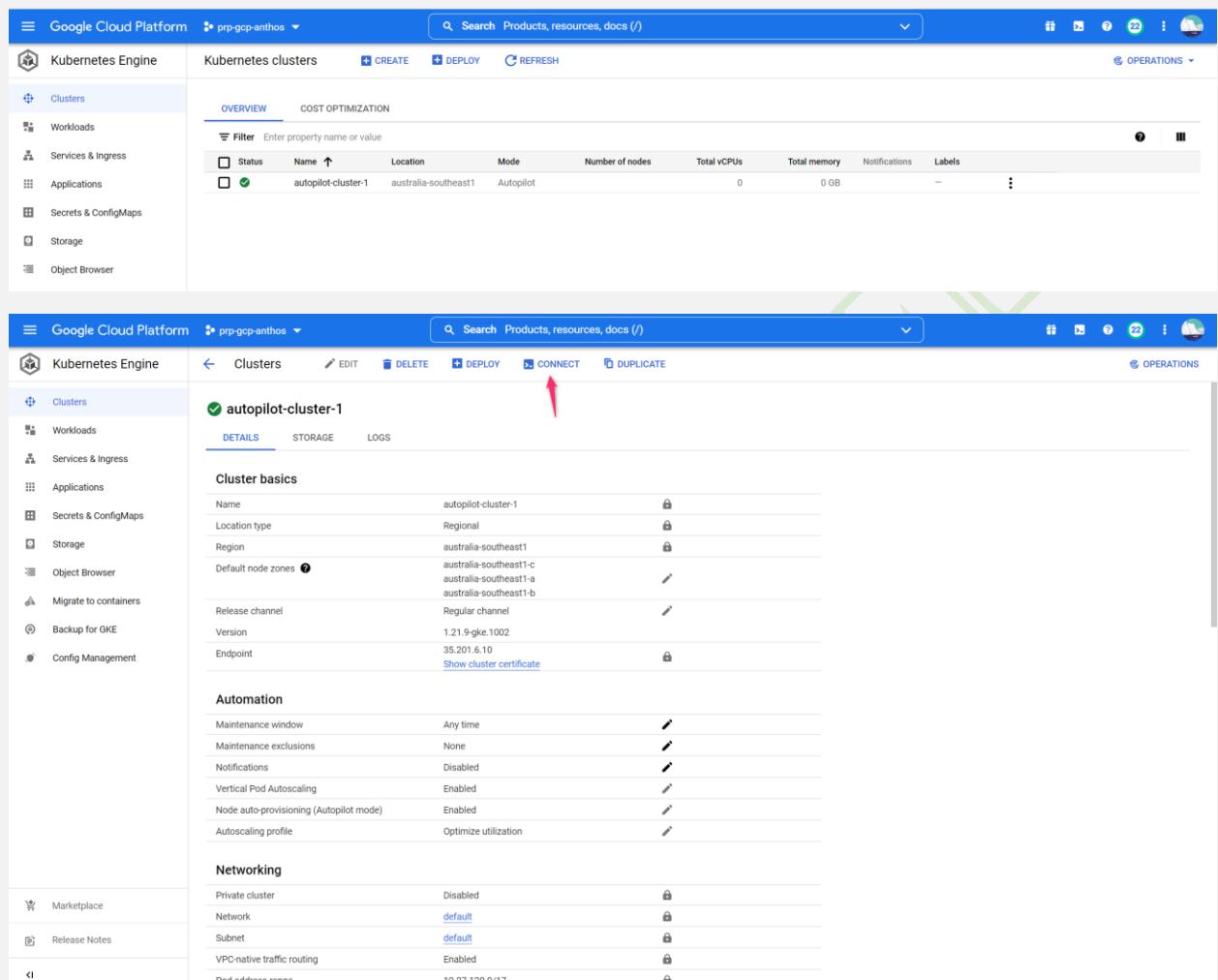
Terminal output showing tcb-clinic deployment configuration:

```

rollingUpdate:
  maxSurge: 1
  maxUnavailable: 1
minReadySeconds: 5
template:
  metadata:
    labels:
      app: tcb-clinic
  spec:
    containers:
      - name: tcb-clinic
        image: asia.gcr.io/prp-gcp-anthos/tcb-clinic-app:latest
        ports:
          - containerPort: 3000
        resources:
          requests:
            cpu: 100m
            memory: 100Mi
        env:
          - name: DBHOST
            value: "10.103.240.2" ←
          - name: DBUSER
            value: "app"
          - name: DBPW
            value: "welcomeme1"
          - name: DBPORT
            value: "3306"
          - name: DBNAME
            value: "clinic"
    ...

```

Connecting to the Cluster (Console -> GKE -> the Cluster created -> Connect)

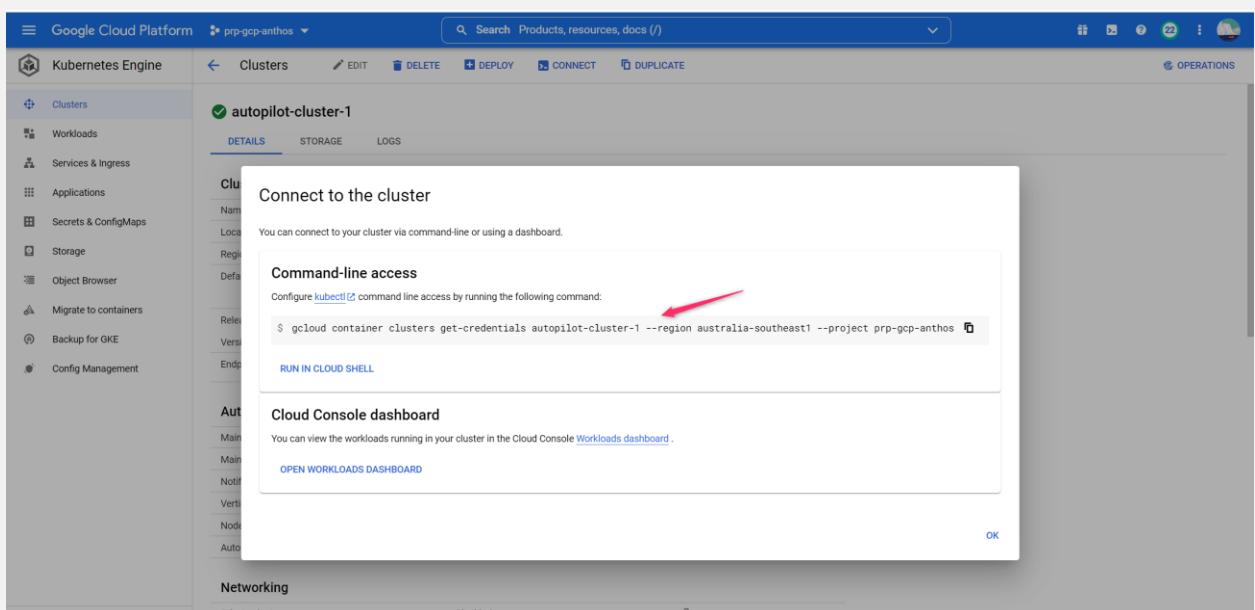


The screenshot shows two views of the Google Cloud Platform Kubernetes Engine Clusters interface.

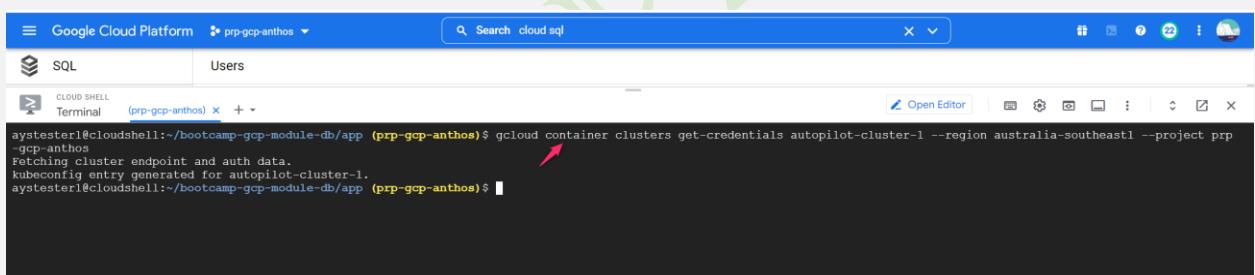
Top View: The 'Clusters' section of the 'Kubernetes Engine' dashboard. It lists a single cluster named 'autopilot-cluster-1' located in 'australia-southeast1' mode 'Autopilot'. The cluster has 0 nodes and 0 GB total memory. There are tabs for 'OVERVIEW' and 'COST OPTIMIZATION'.

Bottom View: A detailed view of the 'autopilot-cluster-1' cluster. The 'CONNECT' button is highlighted with a red arrow. The cluster details include:

- Cluster basics:**
 - Name: autopilot-cluster-1
 - Location type: Regional
 - Region: australia-southeast1
 - Default node zones: australiasoutheast1-c, australiasoutheast1-a, australiasoutheast1-b
 - Release channel: Regular channel
 - Version: 1.21.9-gke.1002
 - Endpoint: 35.201.6.10 (Show cluster certificate)
- Automation:**
 - Maintenance window: Any time
 - Maintenance exclusions: None
 - Notifications: Disabled
 - Vertical Pod Autoscaling: Enabled
 - Node auto-provisioning (Autopilot mode): Enabled
 - Autoscaling profile: Optimize utilization
- Networking:**
 - Private cluster: Disabled
 - Network: default
 - Subnet: default
 - VPC-native traffic routing: Enabled
 - Port address range: 10 27 128 0/17

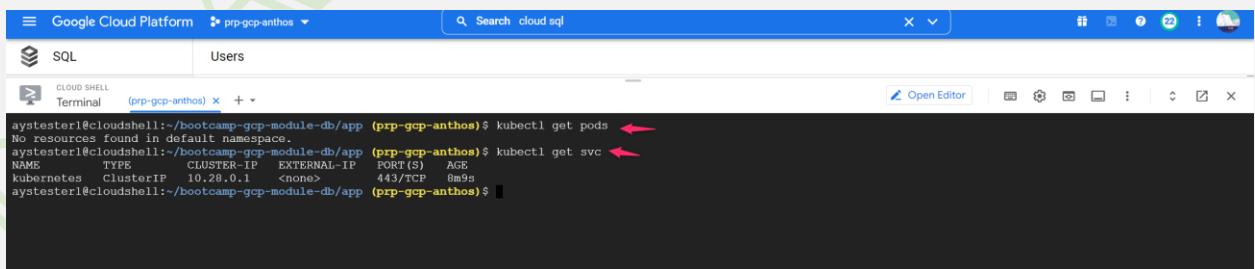


In the Command-line access, click on Run in Cloud Shell [Enter]



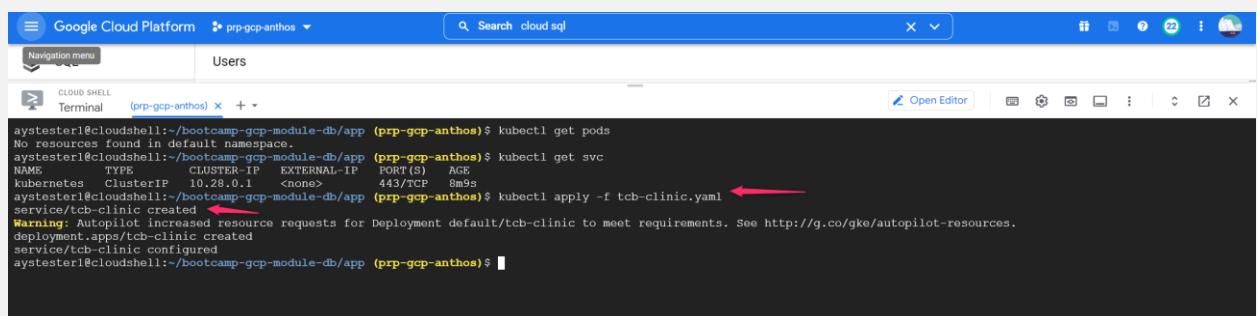
Deploying the application from the yaml file:

Let's check before deploying to kubernetes



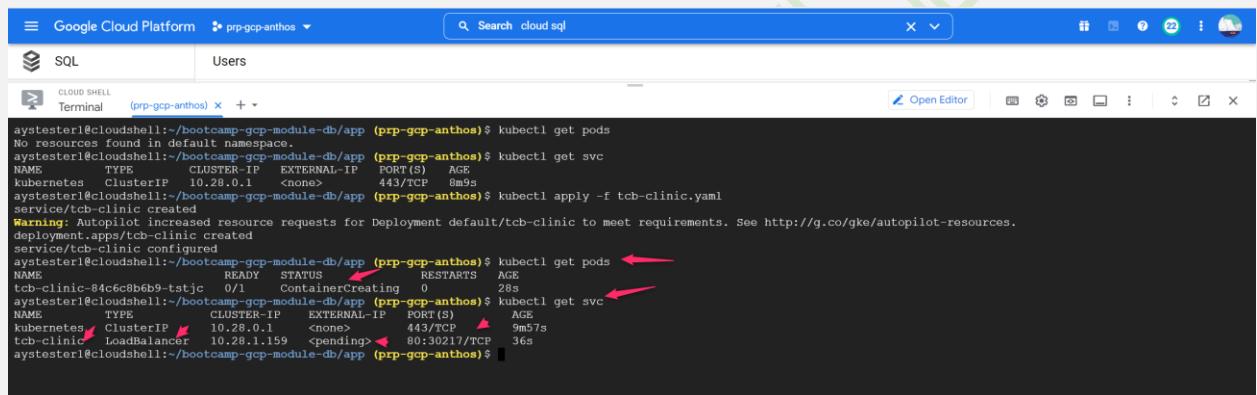
Let's deploy the app to kubernetes

`kubectl apply -f tcb-clinic.yaml`



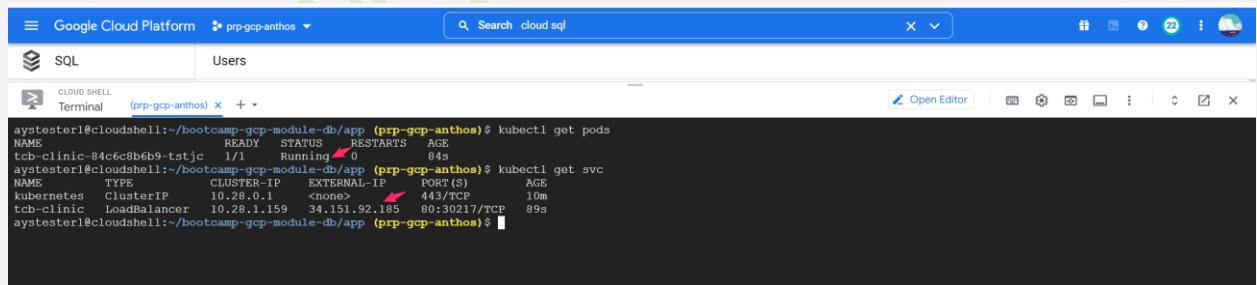
```
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ kubectl get pods
No resources found in default namespace.
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ kubectl get svc
NAME      TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP  10.28.0.1   <none>        443/TCP   8m9s
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ kubectl apply -f tcb-clinic.yaml
service/tcb-clinic created
Warning: Autopilot increased resource requests for Deployment default/tcb-clinic to meet requirements. See http://g.co/gke/autopilot-resources.
deployment.apps/tcb-clinic created
service/tcb-clinic configured
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$
```

Lets verify if the pod and svc is created for clinic app



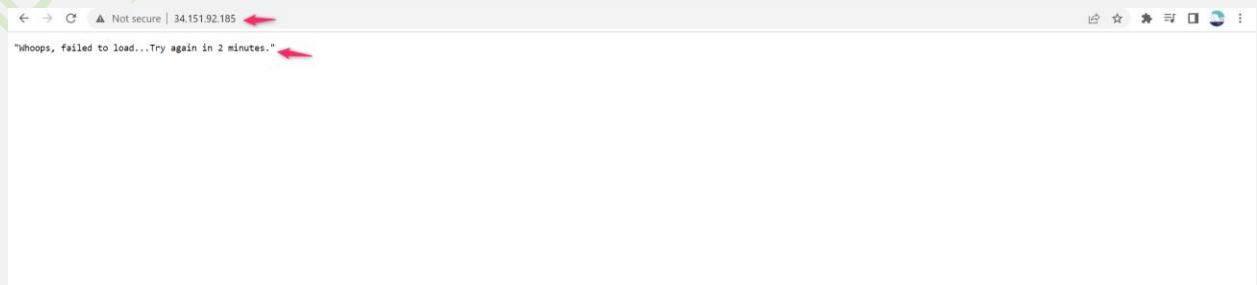
```
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ kubectl get pods
No resources found in default namespace.
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ kubectl get svc
NAME      TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP  10.28.0.1   <none>        443/TCP   8m9s
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ kubectl apply -f tcb-clinic.yaml
service/tcb-clinic created
Warning: Autopilot increased resource requests for Deployment default/tcb-clinic to meet requirements. See http://g.co/gke/autopilot-resources.
deployment.apps/tcb-clinic created
service/tcb-clinic configured
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
tcb-clinic-84c6c8b6b9-tstjc  0/1   ContainerCreating   0          28s
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ kubectl get svc
NAME      TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP  10.28.0.1   <none>        443/TCP   9m57s
tcb-clinic   LoadBalancer  10.28.1.159  <pending>   80:30217/TCP  36s
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$
```

Application is running now



```
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
tcb-clinic-84c6c8b6b9-tstjc  1/1   Running   0          84s
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$ kubectl get svc
NAME      TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP  10.28.0.1   <none>        443/TCP   10m
tcb-clinic   LoadBalancer  10.28.1.159  34.151.92.185  80:30217/TCP  89s
aystester1@cloudshell:~/bootcamp-gcp-module-db/app (prp-gcp-anthos)$
```

Access application using External ip



As there is no database table is created in Cloud SQL so it show's an error like this

Database Migration Process

Insert a new data to front end app before data migration

#	Patient	Address	Telephone	Actions
1	Kevin Park	76 Princess Dr. Lakeland, FL 33801	294720472	<button>EDIT</button> <button>DELETE</button>
2	Joe Felix	33 Holly Ave. Williamstown, NJ 08094	494720472	<button>EDIT</button> <button>DELETE</button>
3	Mary Phonex	4 Arch Street Millville, NJ 08332	894720472	<button>EDIT</button> <button>DELETE</button>
4	Josh Koe	693 Victoria St. Macungie, PA 18062	6794720472	<button>EDIT</button>

Praful Patel 1
 2121 Saskatchewan Dr. Regina, Canada 2
 966 565 5645 3
SAVE PATIENT

4	Josh Koe	693 Victoria St. Macungie, PA 18062	6794720472	<button>EDIT</button> <button>DELETE</button>
5	Mariah Steven	74 Somerset St. Sioux Falls, SD 57103	294720472	<button>EDIT</button> <button>DELETE</button>
6	Mark Plum	8652 Delaware Road Inman, SC 29349	194720472	<button>EDIT</button> <button>DELETE</button>
7	Larry Worth	8233 Roosevelt St. Burnsville, MN 55337	538329298	<button>EDIT</button> <button>DELETE</button>
8	Sarah Lee	540 Sussex Rd. Ozone Park, NY 11417	9058747322	<button>EDIT</button> <button>DELETE</button>
9	Martin McLee	48 Blue Spring Road Milwaukee, WI 53204	9298748490	<button>EDIT</button> <button>DELETE</button>
10	Praful Patel	2121 Saskatchewan Dr. Regina, Canada	966 565 5645	<button>EDIT</button> <button>DELETE</button>

ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&projectNumber=25

```

SSH-IN-BROWSER Terminal

Query OK, 1 row affected (0.01 sec)

mysql> select * from patient;
+---+-----+-----+-----+
| id | name      | address          | phone   |
+---+-----+-----+-----+
| 1  | Kevin Park | 76 Princess Dr. Lakeland, FL 33801 | 294720472 |
| 2  | Joe Felix   | 33 Holly Ave. Williamstown, NJ 08094 | 494720472 |
| 3  | Mary Phonenix | 4 Arch Street Millville, NJ 08332 | 894720472 |
| 4  | Josh Koe    | 693 Victoria St. Macungie, PA 18062 | 6794720472 |
| 5  | Mariah Steven | 74 Somerset St. Sioux Falls, SD 57103 | 294720472 |
| 6  | Mark Plum    | 8652 Delaware Road Inman, SC 29349 | 194720472 |
| 7  | Larry Worth  | 8233 Roosevelt St. Burnsville, MN 55337 | 538329298 |
| 8  | Sarah Lee    | 540 Sussex Rd. Ozone Park, NY 11417 | 9058747322 |
| 9  | Martin McLee | 48 Blue Spring Road Milwaukee, WI 53204 | 9298748490 |
+---+-----+-----+-----+
9 rows in set (0.00 sec)

mysql> select * from patient;
+---+-----+-----+-----+
| id | name      | address          | phone   |
+---+-----+-----+-----+
| 1  | Kevin Park | 76 Princess Dr. Lakeland, FL 33801 | 294720472 |
| 2  | Joe Felix   | 33 Holly Ave. Williamstown, NJ 08094 | 494720472 |
| 3  | Mary Phonenix | 4 Arch Street Millville, NJ 08332 | 894720472 |
| 4  | Josh Koe    | 693 Victoria St. Macungie, PA 18062 | 6794720472 |
| 5  | Mariah Steven | 74 Somerset St. Sioux Falls, SD 57103 | 294720472 |
| 6  | Mark Plum    | 8652 Delaware Road Inman, SC 29349 | 194720472 |
| 7  | Larry Worth  | 8233 Roosevelt St. Burnsville, MN 55337 | 538329298 |
| 8  | Sarah Lee    | 540 Sussex Rd. Ozone Park, NY 11417 | 9058747322 |
| 9  | Martin McLee | 48 Blue Spring Road Milwaukee, WI 53204 | 9298748490 |
| 10 | Praful Patel | 2121 Saskatchewan Dr. Regina, Canada | 966 565 5645 |
+---+-----+-----+-----+
10 rows in set (0.00 sec)

mysql>
```

Accessing the Compute Engine service and turn off the GCE aus-app01

Exporting the DB of the MySQL GCE

Access the aus-db01 instance via ssh

Run this command to export the data:

`mysqldump --add-drop-table -u root -p clinic > clinic.sql`

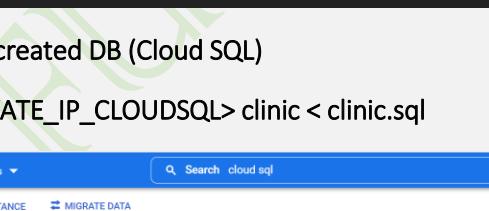


```
https://ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&projectNumber=25...
ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&projectNumber=25...
SSH-IN-BROWSER Terminal
aystester1@aus-db02:~$ mysqldump --add-drop-table -u root -p clinic > clinic.sql
Enter password:
aystester1@aus-db02:~$ ls -ltr
total 76
-rw-r--r-- 1 aystester1 aystester1 35560 Apr 29 2019 mysql-apt-config_0.8.13-1_all.deb
-rw-r--r-- 1 aystester1 aystester1 316 Aug 20 2021 bootcamp-gcp-storage-db-en.sql
drwx----- 4 aystester1 aystester1 4096 Aug 30 2021 bootcamp-gcp-module-db
-rw-r--r-- 1 aystester1 aystester1 23244 Aug 30 2021 bootcamp-gcp-module-db.zip
drwxr-xr-x 3 aystester1 aystester1 4096 Mar 21 22:54 _MACOSX
-rw-r--r-- 1 aystester1 aystester1 2680 Mar 22 00:47 clinic.sql
aystester1@aus-db02:~$ cat clinic.sql
```

The terminal window shows the command `mysqldump` being run to export the `clinic` database to a file named `clinic.sql`. The output directory is shown as `~` (home directory). The file `clinic.sql` is created in the current directory. Two red circles are overlaid on the terminal window: circle 1 is at the top right of the terminal title bar, and circle 2 is at the bottom right of the terminal window.

#Workaround to the error ERROR 1273 (HY000) at line 25: Unknown collation:'utf8mb4_0900_ai_ci'

```
sed -e 's/utf8mb4_0900_ai_ci/utf8mb4_unicode_ci/g' -i clinic.sql
```

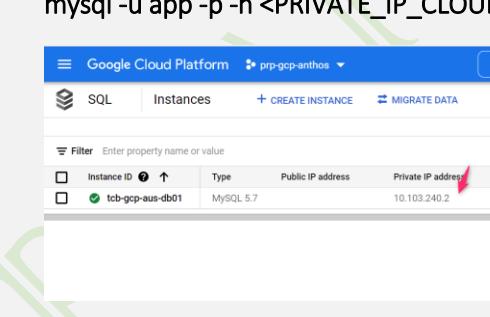


```
https://ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&projectNumber=25...
ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&projectNumber=25...
SSH-IN-BROWSER Terminal
aystester1@aus-db02:~$ sed -e 's/utf8mb4_0900_ai_ci/utf8mb4_unicode_ci/g' -i clinic.sql
aystester1@aus-db02:~$
```

The terminal window shows the command `sed` being used to replace the collation `utf8mb4_0900_ai_ci` with `utf8mb4_unicode_ci` in the `clinic.sql` file. The file is modified in place with the `-i` option. The terminal prompt ends with `~\$`.

Importing data into the created DB (Cloud SQL)

```
mysql -u app -p -h <PRIVATE_IP_CLOUDSQL> clinic < clinic.sql
```



Google Cloud Platform prp-gcp-anthos

SQL Instances + CREATE INSTANCE MIGRATE DATA

No instances selected

MONITORING LABELS

10.103.240.2

MySQL 5.7

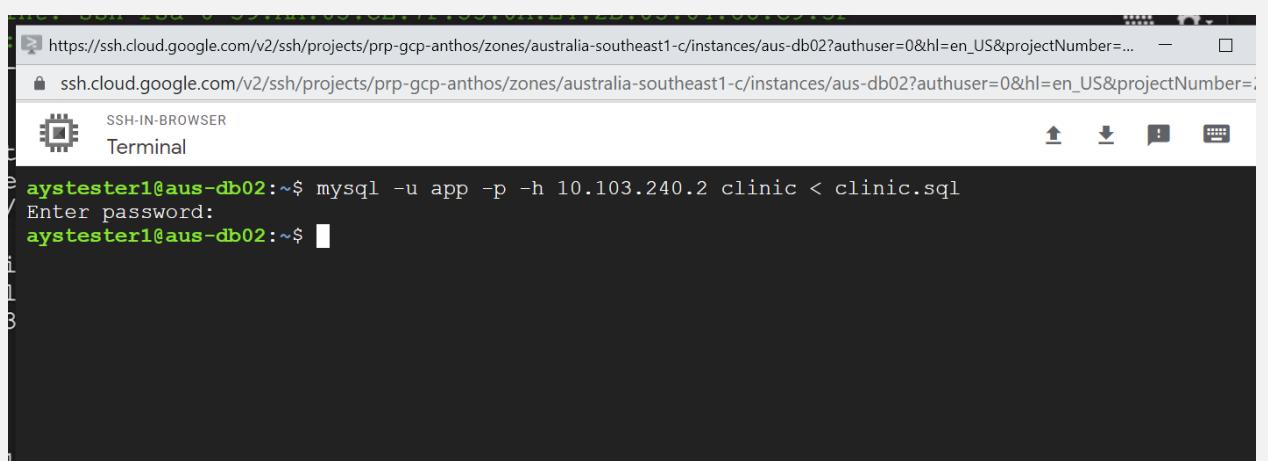
10.103.240.2 prp-gcp-anthos:aus... ENABLED australiasoutheast1-a

HIDE INFO PANEL

This screenshot shows the Google Cloud Platform Cloud SQL interface. It displays a table with one instance listed: `tcb-gcp-aus-db01` (MySQL 5.7, Public IP address 10.103.240.2, Private IP address 10.103.240.2, Instance connection name `prp-gcp-anthos:aus...`, High availability `ENABLED`, Location `australiasoutheast1-a`). The table has columns for Instance ID, Type, Public IP address, Private IP address, Instance connection name, High availability, Location, Storage used, and Actions. A red arrow points to the Public IP address column. The right side of the interface shows monitoring and labeling options, and a note that no instances are selected.

10.103.240.2

```
mysql -u app -p -h 10.103.240.2 clinic < clinic.sql
```

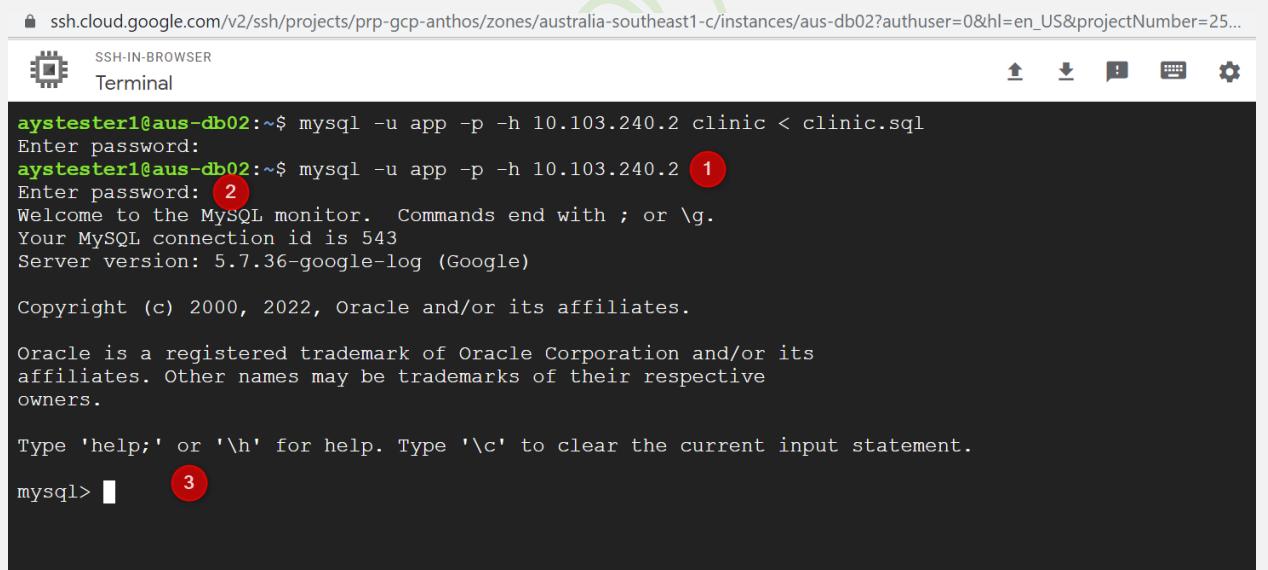


```
https://ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&projectNumber=... - Terminal
aystester1@aus-db02:~$ mysql -u app -p -h 10.103.240.2 clinic < clinic.sql
Enter password:
aystester1@aus-db02:~$
```

Connecting to Cloud SQL DB and checking the imported data:

```
mysql -u app -p -h <PRIVATE_IP_CLOUDSQL>
```

```
mysql -u app -p -h 10.103.240.2
```



```
ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&projectNumber=25...
aystester1@aus-db02:~$ mysql -u app -p -h 10.103.240.2 clinic < clinic.sql
Enter password:
aystester1@aus-db02:~$ mysql -u app -p -h 10.103.240.2 1
Enter password: 2
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 543
Server version: 5.7.36-google-log (Google)

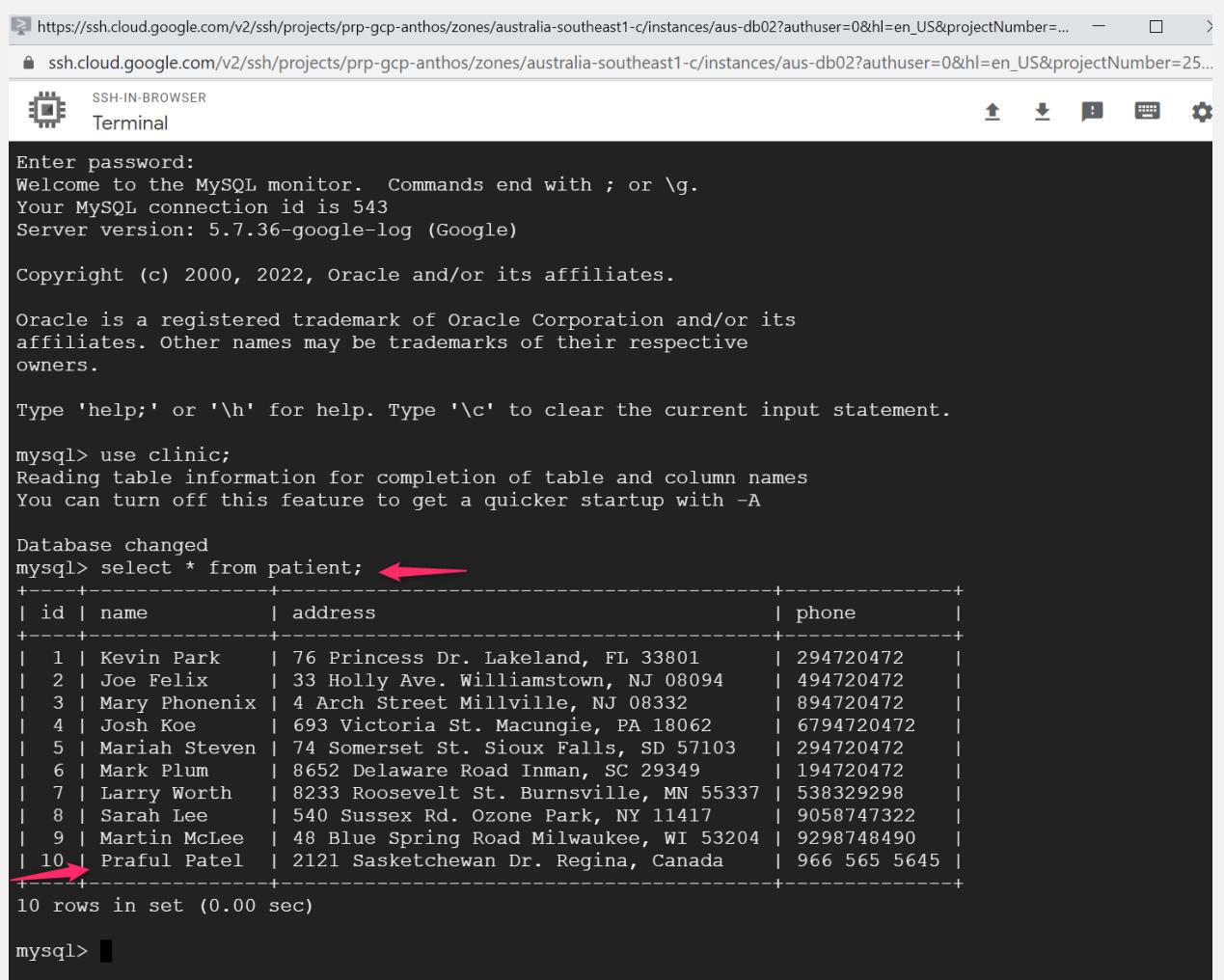
Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> 3
```

```
use clinic;
```

```
select * from patient;
```



```

https://ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&projectNumber=...
ssh.cloud.google.com/v2/ssh/projects/prp-gcp-anthos/zones/australia-southeast1-c/instances/aus-db02?authuser=0&hl=en_US&projectNumber=25...
SSH-IN-BROWSER Terminal
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 543
Server version: 5.7.36-google-log (Google)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

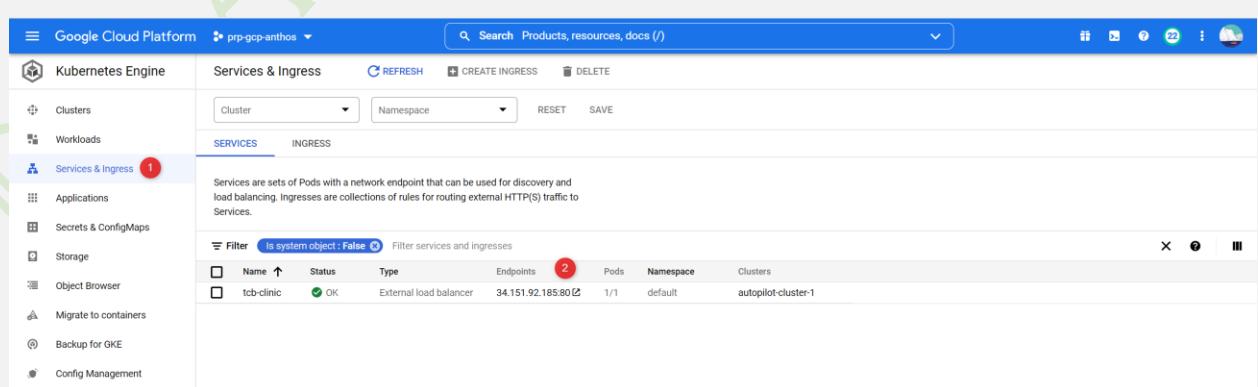
mysql> use clinic;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from patient;
+----+-----+-----+-----+
| id | name      | address          | phone   |
+----+-----+-----+-----+
| 1  | Kevin Park | 76 Princess Dr. Lakeland, FL 33801 | 294720472 |
| 2  | Joe Felix   | 33 Holly Ave. Williamstown, NJ 08094 | 494720472 |
| 3  | Mary Phonenix | 4 Arch Street Millville, NJ 08332 | 894720472 |
| 4  | Josh Koe    | 693 Victoria St. Macungie, PA 18062 | 6794720472 |
| 5  | Mariah Steven | 74 Somerset St. Sioux Falls, SD 57103 | 294720472 |
| 6  | Mark Plum    | 8652 Delaware Road Inman, SC 29349 | 194720472 |
| 7  | Larry Worth   | 8233 Roosevelt St. Burnsville, MN 55337 | 538329298 |
| 8  | Sarah Lee    | 540 Sussex Rd. Ozone Park, NY 11417 | 9058747322 |
| 9  | Martin McLee | 48 Blue Spring Road Milwaukee, WI 53204 | 9298748490 |
| 10 | Praful Patel | 2121 Saskatchewan Dr. Regina, Canada | 966 565 5645 |
+----+-----+-----+-----+
10 rows in set (0.00 sec)

mysql>

```

Testing the access to the application with the External IP of the Load Balancing.



Name	Status	Type	Endpoints	Pods	Namespace	Clusters
tcb-clinic	OK	External load balancer	34.151.92.185:80	1/1	default	autopilot-cluster-1

MIB CLINIC - PATIENT MANAGEMENT

#	Patient	Address	Telephone	Actions
1	Kevin Park	76 Princess Dr, Lakeland, FL 33801	294720472	<button>EDIT</button> <button>DELETE</button>
2	Joe Felix	33 Holly Ave, Williamstown, NJ 08094	494720472	<button>EDIT</button> <button>DELETE</button>
3	Mary Phonex	4 Arch Street Millville, NJ 08332	894720472	<button>EDIT</button> <button>DELETE</button>
4	Josh Koe	693 Victoria St, Macungie, PA 18062	6794720472	<button>EDIT</button> <button>DELETE</button>
5	Mariah Steven	74 Somerset St, Sioux Falls, SD 57103	294720472	<button>EDIT</button> <button>DELETE</button>

#	Patient	Address	Telephone	Actions
5	Mariah Steven	74 Somerset St, Sioux Falls, SD 57103	294720472	<button>EDIT</button> <button>DELETE</button>
6	Mark Plum	8652 Delaware Road Inman, SC 29349	194720472	<button>EDIT</button> <button>DELETE</button>
7	Larry Worth	8233 Roosevelt St, Burnsville, MN 55337	538329298	<button>EDIT</button> <button>DELETE</button>
8	Sarah Lee	540 Sussex Rd, Ozone Park, NY 11417	9058747322	<button>EDIT</button> <button>DELETE</button>
9	Martin McLee	48 Blue Spring Road Milwaukee, WI 53204	9298748490	<button>EDIT</button> <button>DELETE</button>
10	Praful Patel	2121 Saskatchewan Dr, Regina, Canada	966 565 5645	<button>EDIT</button> <button>DELETE</button>

Testing the HA of the DB

Access the Cloud SQL

Access the Cloud SQL instance

Click on Failover

Google Cloud Platform prp-gcp-anthos

SQL Overview EDIT IMPORT EXPORT RESTART STOP DELETE CLONE FAILOVER

All instances > tcb-gcp-aus-db01
tcb-gcp-aus-db01

MySQL 5.7

Chart CPU utilization

1 hour 6 hours 1 day 7 days 30 days Custom

UTC-8 8:00 PM 10:00 PM Mar 21 2:00 AM 4:00 AM 6:00 AM 8:00 AM 10:00 AM 12:00 PM 2:00 PM 4:00 PM 6:00 PM

15%
10%
5%
0%

Connect to this instance

Private IP address: 10.103.248.2

Associated networking: projects/prp-gcp-anthos/global/networks/default

Connection name: prp-gcp-anthos:australia-southeast1:tcb-gcp-aus-db01

Configuration

vCPUs: 1	Memory: 3.75 GB	SSD storage: 10 GB
----------	-----------------	--------------------

Database version is MySQL 5.7.36
Auto storage increase is enabled
Automated backups are enabled
Point-in-time recovery is enabled
Located in australia-southeast1-a
Highly available (regional)

Zone a

SQL Overview EDIT IMPORT EXPORT RESTART STOP DELETE CLONE FAILOVER

PRIMARY INSTANCE

- Overview
- Connections
- Users
- Databases
- Backups
- Replicas
- Operations

Connect to this instance

Private IP address: 10.103.248.2

Associated networking: projects/prp-gcp-anthos/global/networks/default

Connection name: prp-gcp-anthos:australia-southeast1:tcb-gcp-aus-db01

Need help connecting?

Review the documentation to learn about the many ways to connect to your instance. [Learn more](#)

To connect using gcloud, [OPEN CLOUD SHELL](#)

To learn about connecting with a Compute Engine VM, [START TUTORIAL](#)

Configuration

vCPUs	Memory	SSD storage
1	3.75 GB	10 GB

Database version is MySQL 5.7.36

Auto storage increase is enabled

Automated backups are enabled

Point-in-time recovery is enabled

Located in australia-southeast1-a

Highly available (regional)

No database flags set

No labels set

[Edit configuration](#)

Service account

p254756840759-uzh2cu@gcp-sa-cloud-sql.iam.gserviceaccount.com

Maintenance

Maintenance window: Updates may occur any day of the week.

Order of update: Cloud SQL chooses the maintenance timing.

Notifications

Check the name of the instance;

SQL Overview EDIT IMPORT EXPORT RESTART STOP DELETE CLONE FAILOVER

PRIMARY INSTANCE

- Overview
- Connections
- Users
- Databases
- Backups
- Replicas
- Operations

Connect to this instance

Private IP address: 10.103.248.2

Associated networking: projects/prp-gcp-anthos/global/networks/default

Connection name: prp-gcp-anthos:australia-southeast1:tcb-gcp-aus-db01

Manually trigger a failover

This operation cannot be undone.

Your instance will not be available for several minutes while the failover is taking place.

To confirm the failover, enter the instance ID tcb-gcp-aus-db01 below:

TRIGGER FAILOVER

Configuration

vCPUs	Memory	SSD storage
1	3.75 GB	10 GB

Database version is MySQL 5.7.36

Auto storage increase is enabled

Automated backups are enabled

Point-in-time recovery is enabled

Located in australia-southeast1-a

Highly available (regional)

No database flags set

No labels set

[Edit configuration](#)

Maintenance

Then, click on Trigger Failover.

The screenshot shows the Google Cloud Platform Cloud SQL Overview page for a primary instance named UTC-6. The configuration section indicates that a failover is in progress. Other details shown include vCPUs (1), Memory (3.75 GB), and SSD storage (10 GB). The instance was created at 10:00 PM on Mar 21.

Application is still up & running

The screenshot shows a web browser displaying the MIB CLINIC - PATIENT MANAGEMENT application. The page lists four patients with columns for #, Patient, Address, Telephone, and Actions (Edit and Delete buttons). To the right, there is a form for adding a new patient with fields for Patient, Address, and Telephone, and a 'SAVE PATIENT' button.

#	Patient	Address	Telephone	Actions
1	Kevin Park	76 Princess Dr, Lakeland, FL 33801	294720472	EDIT DELETE
2	Joe Felix	33 Holly Ave, Williamstown, NJ 08094	494720472	EDIT DELETE
3	Mary Phonex	4 Arch Street Millville, NJ 08332	894720472	EDIT DELETE
4	Josh Koe	693 Victoria St, Macungie, PA 18062	6794720472	EDIT DELETE

In the Configuration section, pay attention that the zone will be changed after the failover process finish!

Google Cloud Platform - prp-gcp-anthos - Search cloud sql

SQL Overview EDIT IMPORT EXPORT RESTART STOP DELETE CLONE FAILOVER

PRIMARY INSTANCE

UTC-6 10:00 PM Mar 21 2:00 AM 4:00 AM 6:00 AM 8:00 AM 10:00 AM 12:00 PM 2:00 PM 4:00 PM 6:00 PM

Connect to this instance

Private IP address: 10.103.240.2

Associated networking: projects/prp-gcp-anthos/global/networks/default

Connection name: prp-gcp-anthos:australia-southeast1:tcb-gcp-aus-db01

Need help connecting?

Review the documentation to learn about the many ways to connect to your instance. [Learn more](#)

To connect using gcloud, [OPEN CLOUD SHELL](#)

To learn about connecting with a Compute Engine VM, [START TUTORIAL](#)

Service account

p254756840759-uzh2cu@gcp-sa-cloud-sql.iam.gserviceaccount.com

Configuration

vCPUs: 1 Memory: 3.75 GB SSD storage: 10 GB

- Database version is MySQL 5.7.36
- Auto storage increase is enabled
- Automated backups are enabled
- Point-in-time recovery is enabled
- Located in australia-southeast1-b**
- Highly available (regional)
- No database flags set
- No labels set

Maintenance

Maintenance window: 7:01:42 PM GMT-6

Updates may occur any time.

Order of update: Triggered failover for tcb-gcp-aus-db01

Cloud SQL operations

During the failover process, try to access the application, refreshing the browser:

← → ⌂ Not secure | 34.151.92.185

MIB CLINIC - PATIENT MANAGEMENT

#	Patient	Address	Telephone	Actions
1	Kevin Park	76 Princess Dr, Lakeland, FL 33801	294720472	EDIT DELETE
2	Joe Felix	33 Holly Ave, Williamstown, NJ 08094	494720472	EDIT DELETE
3	Mary Phonex	4 Arch Street Millville, NJ 08332	894720472	EDIT DELETE
4	Josh Koe	693 Victoria St, Macungie, PA 18062	6794720472	EDIT DELETE

Patient

Address

Telephone

SAVE PATIENT

Whoops, failed to load...Try again in 2 minutes.

It means that the application is not able to connect to the DB. The failover process takes less than 2 minutes. Refresh until the connection is re-established after failover.

Congratulations!!!! 🎉

PRAFUL PATEL