



National University of Sciences and Technology (NUST)
School of Electrical Engineering and Computer Science

Department of Computing

SE-315: Cloud Computing

Lab 06: Handling Storage: Cloud Storage and Cloud SQL

Date: 23.10.24



Lab 06: Handling Storage: Cloud Storage and Cloud SQL

Lab Objectives: In this lab, students will:

- Create a Cloud Storage bucket and place an image into it.
- Create a Cloud SQL instance and configure it.
- Connect to the Cloud SQL instance from a web server.
- Use the image in the Cloud Storage bucket on a web page.

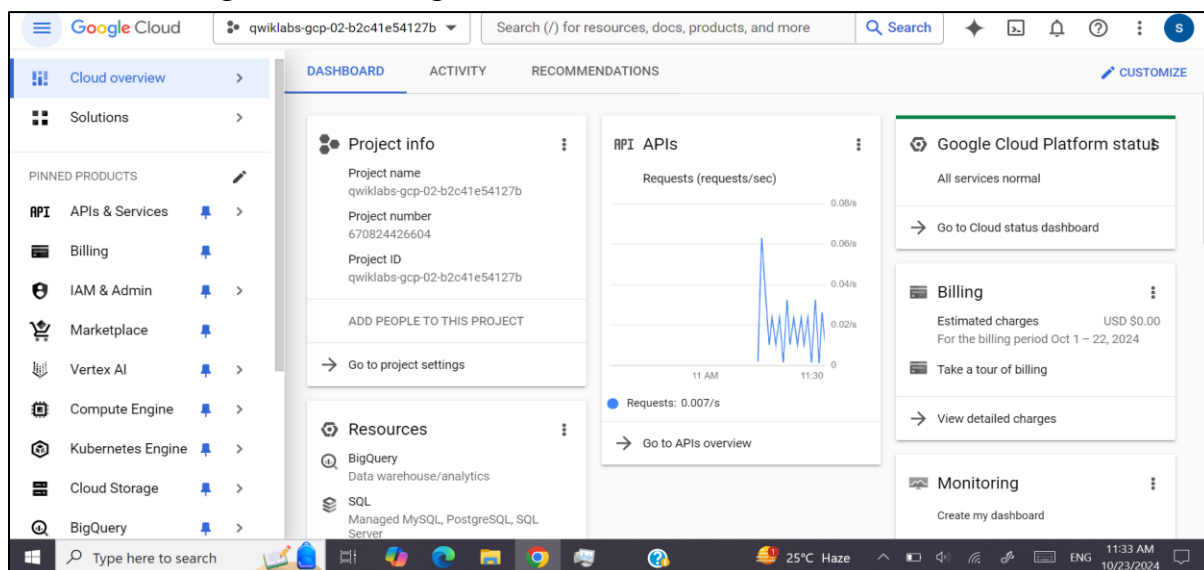
Lab Tasks

Go through the following link:

<https://www.cloudskillsboost.google/focuses/19064?parent=catalog>

which will take you to the 'Google Cloud Fundamentals: Getting Started with Cloud Storage and Cloud SQL' page. You have to start the lab and perform the tasks given below. Make sure to take screenshots of each task as you will need to add them in the solution section given below.

1. Sign in to the Google Cloud Console



2. Deploy a web server VM instance



National University of Sciences and Technology (NUST)

School of Electrical Engineering and Computer Science

Successfully created the mentioned instance

Notifications

✓

Create VM instance "bloghost" and its boot disk "bloghost"
qwiklabs-gcp-02-b2c41e54127b

Just now

These are the details for the vm

| Basic information | |
|-------------------------|-------------------------------------|
| Name | bloghost |
| Instance id | 6290715451969578473 |
| Description | None |
| Type | Instance |
| Status | Running |
| Creation time | Oct 23, 2024, 11:43:19 AM UTC+05:00 |
| Location | us-east4-a |
| Instance template | None |
| In use by | None |
| Reservations | Automatically choose |
| Labels | None |
| Tags | — |
| Deletion protection | Disabled |
| Confidential VM service | Disabled |
| Preserved state size | 0 GB |

3. Create a Cloud Storage bucket using the gcloud storage command line

```
Your Cloud Platform project in this session is set to quiklabs-gcp-02-b2c41e54127b.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
student_01_9be8b78423be@cloudshell:~ (quiklabs-gcp-02-b2c41e54127b)$
student_01_9be8b78423be@cloudshell:~ (quiklabs-gcp-02-b2c41e54127b)$ export LOCATION=US
student_01_9be8b78423be@cloudshell:~ (quiklabs-gcp-02-b2c41e54127b)$ gcloud storage buckets create -l US gs://$DEVSHHELL_PROJECT_ID
Creating gs://quiklabs-gcp-02-b2c41e54127b/...
student_01_9be8b78423be@cloudshell:~ (quiklabs-gcp-02-b2c41e54127b)$ gcloud storage cp gs://cloud-training/gcpfci/my-excellent-blog.png my-excellent-blog.png
Copying gs://cloud-training/gcpfci/my-excellent-blog.png to file:///my-excellent-blog.png
Completed files 1/1 | 8.2kiB/8.2kiB
student_01_9be8b78423be@cloudshell:~ (quiklabs-gcp-02-b2c41e54127b)$ gcloud storage cp my-excellent-blog.png gs://$DEVSHHELL_PROJECT_ID/my-excellent-blog.png
Copying file:///my-excellent-blog.png to gs://quiklabs-gcp-02-b2c41e54127b/my-excellent-blog.png
Completed files 1/1 | 8.2kiB/8.2kiB
student_01_9be8b78423be@cloudshell:~ (quiklabs-gcp-02-b2c41e54127b)$
```

Updating the access Control List of the image object I created :

```
student_01_9be8b78423be@cloudshell:~ (quiklabs-gcp-02-b2c41e54127b)$ gsutil acl ch -u allUsers:R gs://$DEVSHHELL_PROJECT_ID/my-excellent-blog.png
Updated ACL on gs://quiklabs-gcp-02-b2c41e54127b/my-excellent-blog.png
student_01_9be8b78423be@cloudshell:~ (quiklabs-gcp-02-b2c41e54127b)$
```

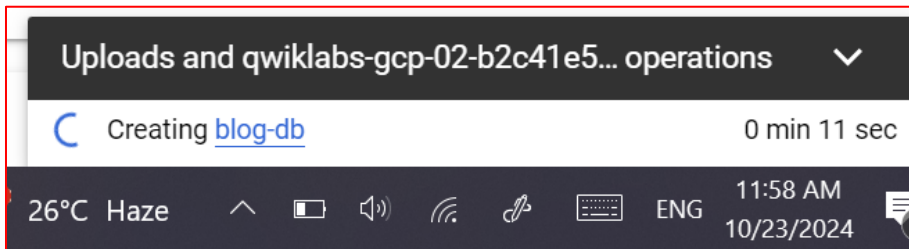
4. Create the Cloud SQL instance



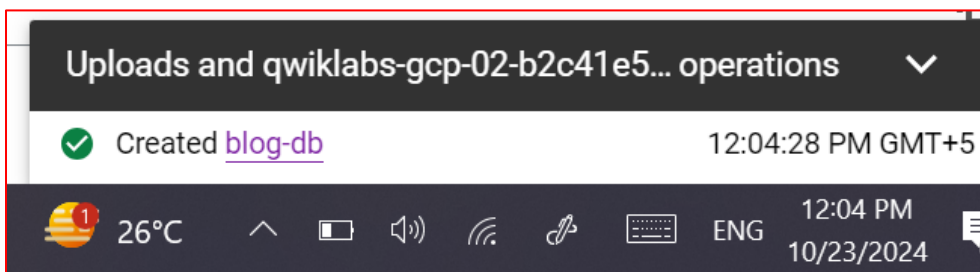
National University of Sciences and Technology (NUST)

School of Electrical Engineering and Computer Science

Creating the blog-db sql instance



Successfully created the sql instance



Creating a new db user “blogdbuser”

BLOG-DB

✓ **blog-db**

MySQL 8.0

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

+ ADD USER ACCOUNT

ADDED USERS

AUTHENTICATED IAM GROUP MEMBERS

These are accounts that you have granted instance access to, using either built-in or IAM authentication.

| | → User name | → Host name | → Authentication | → Password status | → |
|---|-------------|--------------|------------------|-------------------|---|
| ● | blogdbuser | % (any host) | Built-in | N/A | ⋮ |
| ● | root | % (any host) | Built-in | N/A | ⋮ |

Modified the sql db, these are the logs:



Operations and logs

| Creation Time | Completion Time | Type | Status |
|---------------------------|---------------------------|-------------|-----------------|
| Oct 23, 2024, 12:14:05 PM | Oct 23, 2024, 12:14:27 PM | Update | Update finished |
| Oct 23, 2024, 12:09:17 PM | Oct 23, 2024, 12:09:17 PM | Create user | User created |
| Oct 23, 2024, 12:02:52 PM | Oct 23, 2024, 12:04:23 PM | Backup | Backup finished |
| Oct 23, 2024, 11:58:39 AM | Oct 23, 2024, 12:04:26 PM | Create | Create finished |



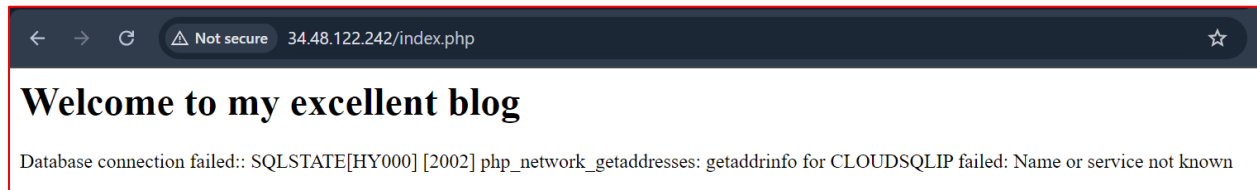
5. Configure an application in a Compute Engine instance to use Cloud SQL

This is the default html code provided which leads to a failed DB connection

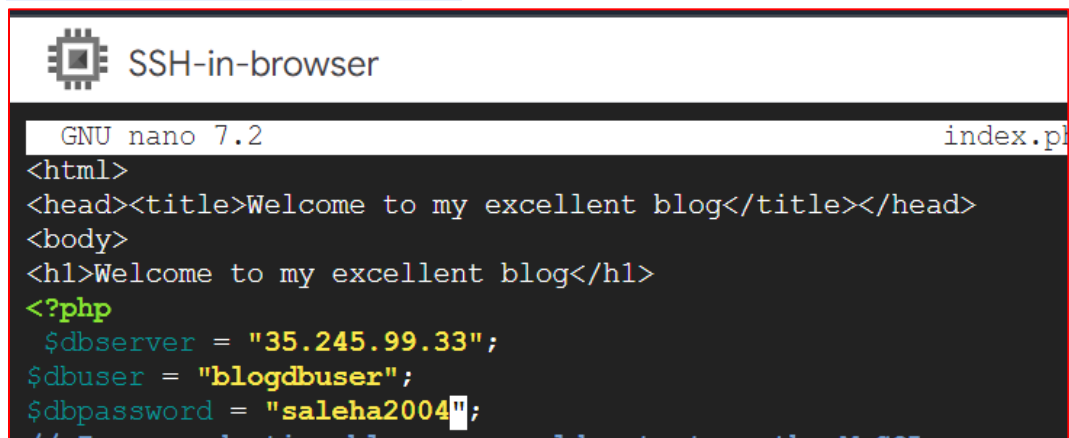
```
SSH-in-browser  UPLOAD FILE  DOW
GNU nano 7.2  index.php *
<html>
<head><title>Welcome to my excellent blog</title></head>
<body>
<h1>Welcome to my excellent blog</h1>
<?php
    $dbserver = "CLOUDSQLIP";
    $dbuser = "blogdbuser";
    $dbpassword = "DBPASSWORD";
    // In a production blog, we would not store the MySQL
    // password in the document root. Instead, we would store
    // it in a Secret Manger. For more information see
    // https://cloud.google.com/sql/docs/postgres/use-secret-manager

    try {
        $conn = new PDO("mysql:host=$dbserver;dbname=mysql", $dbuser, $dbpassword);
        // set the PDO error mode to exception
        $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
        echo "Connected successfully";
    } catch(PDOException $e) {
        echo "Database connection failed:: " . $e->getMessage();
    }
?>
</body></html>
```

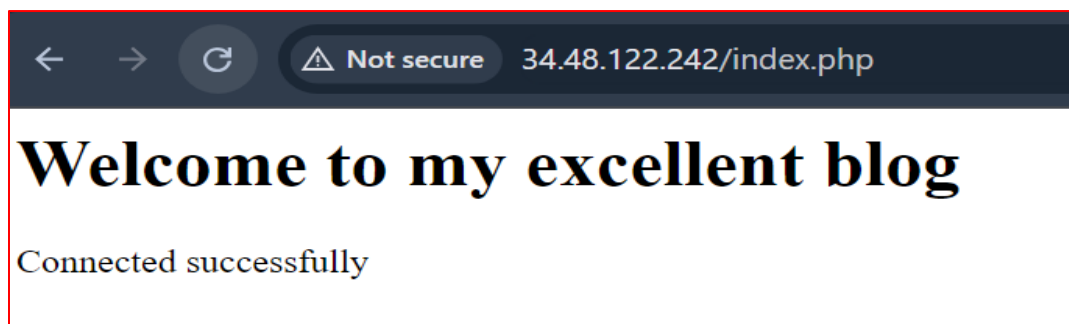
in a web browser tab, I pasted bloghost VM instance's external IP address followed by /index.php.



configured PHP's connection to myCloud SQL instance:
I've used my own credentials below



After providing the correct credentials, the DB has connected successfully:



6. Configure an application in a Compute Engine instance to use a Cloud Storage object.



National University of Sciences and Technology (NUST) School of Electrical Engineering and Computer Science

sh session on my **bloghost** VM instance:



```
GNU nano 7.2 index.php
<html>
<head><title>Welcome to my excellent blog</title></head>
<body>
<img src='https://storage.googleapis.com/qwiklabs-gcp-02-18aaba187d9/my-excellent-blog.png'>
<h1>Welcome to my excellent blog</h1>
<?php
$dbserver = "34.145.228.209";
$dbuser = "blogdbuser";
$dbpassword = "munazza";
// In a production blog, we would not store the MySQL
// password in the document root. Instead, we would store
// it in a Secret Manager. For more information see
// https://cloud.google.com/sql/docs/postgres/use-secret-manager

try {
    $conn = new PDO("mysql:host=$dbserver;dbname=mysql", $dbuser, $dbpassword);
    // set the PDO error mode to exception
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    echo "Connected successfully";
} catch(PDOException $e) {
    echo "Database connection failed: " . $e->getMessage();
}

?>
</body></html>
```

The content of the page now includes a banner image.

