Exercise 1: Understanding Cloud Computing Models

What are the main differences between IaaS, PaaS, and SaaS?

The main differences between these models is that IaaS manages the OS, applications, and middleware, PaaS focuses on building applications, Saas delivers fully managed applications accessible over the internet.

Which GCP services fall under each of these models?

IaaS: Compute Engine (Virtual Machines), Cloud Storage
PaaS: App Engine, Cloud Functions, Google Kubernetes Engine (GKE)
SaaS: Google Workspace, BigQuery, Google Cloud Pub/Sub

Provide a real-world example where each cloud service model might be the most appropriate choice.

IaaS: Project that needs full control over its infrastructure to run a complex web application.
PaaS: It will be helpful for a developer who wants to focus on coding a web app and doesn't want to manage the infrastructure.
SaaS: For companies that use Google Workspace like Gmail, Google Docs and etc. for daily business operations without managing servers.

Exercise 2: Exploring Google Cloud Platform's Core Services

What is the primary use case of Compute Engine?

It allows users to create and manage applications requiring custom VMs with full control over configurations.

How does Google Kubernetes Engine (GKE) simplify the management of containerized applications?

It allows developers to focus on their apps without checking underlying infrastructure like deployment, scaling, and management of containerized applications.

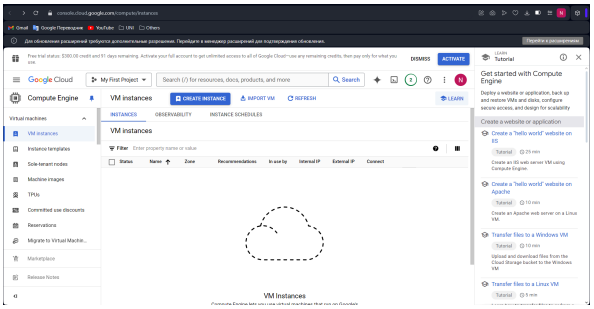
What advantages does Cloud Storage offer for data management?

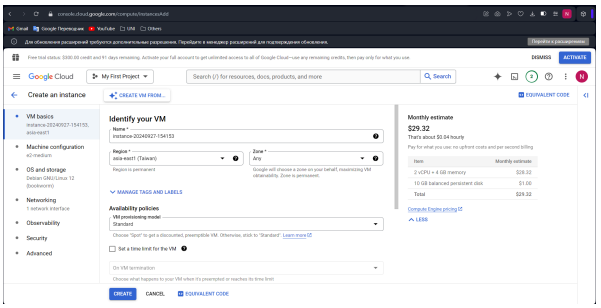
It is ideal for storing unstructured data, backups, and media files, with features like automatic redundancy and easy global access.

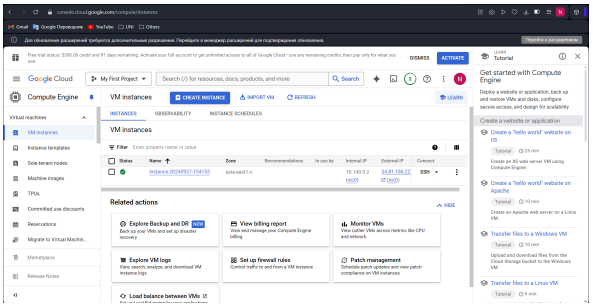
Exercise 3: Creating and Managing Virtual Machines with Compute Engine

What steps did you follow to create the VM?

1. Navigate to the Google Cloud Console.
2. Then select "Compute Engine" and click "Create Instance."
3. Configure the instance with CPU, memory, OS, and region preferences.
4. Launch the VM.





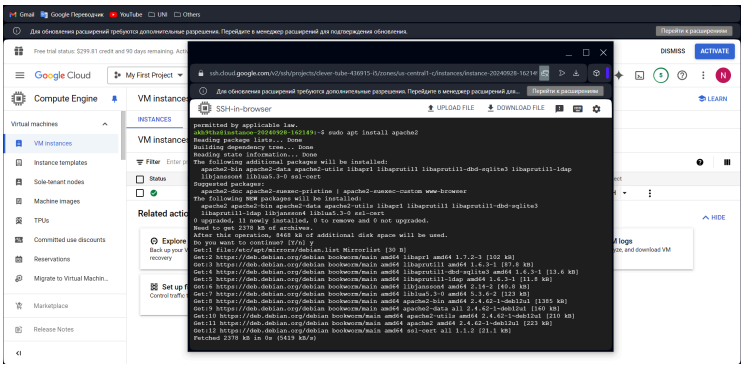


How did you connect to the VM, and what commands did you use to install the web server?

Used SSH to connect to the VM via terminal. Then Installed a web server using commands like:

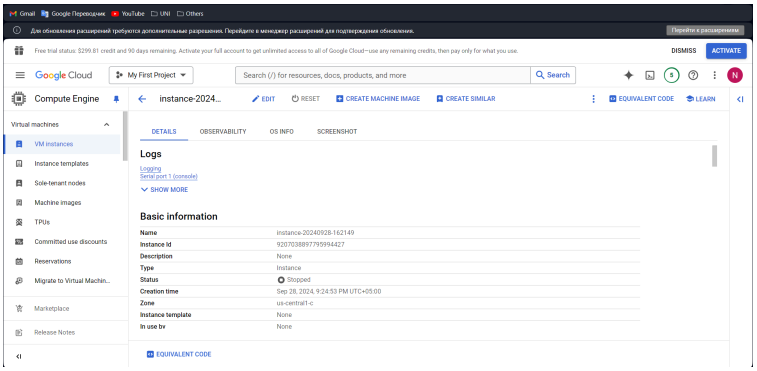
sudo apt update

sudo apt install apache2

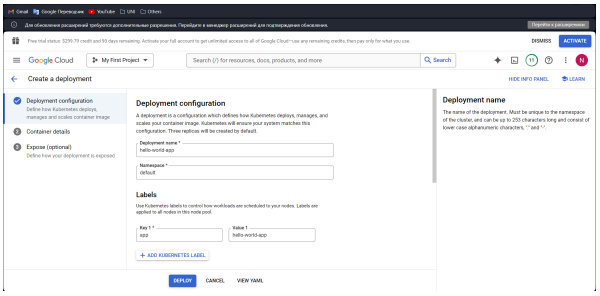
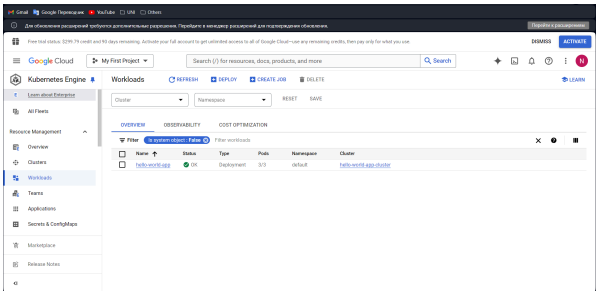
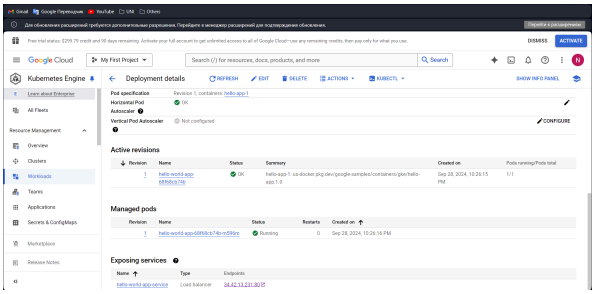
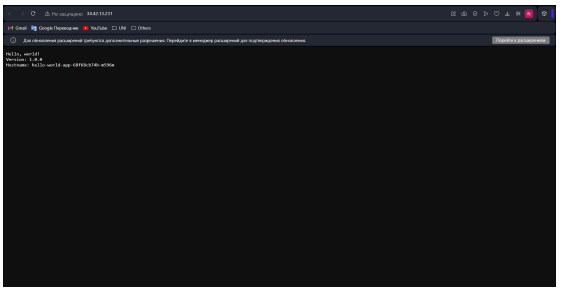


What happens to the VM and its data when it is stopped versus when it is deleted?

When stopped the VM is turned off, but the disk and data remain intact. And after deleting the VM its associated data are removed permanently.



Exercise 4: Deploying a Containerized Application on Google Kubernetes

How did you create and push the Docker container to GCR?

Firstly I build an image by command (using ProjectId and image name): docker build -t gcr.io/clever-tube-436915-i5/my-web-app

Then pushed the image to GCR: docker push gcr.io/clever-tube-436915-i5/my-web-app

What steps were involved in setting up the GKE cluster?

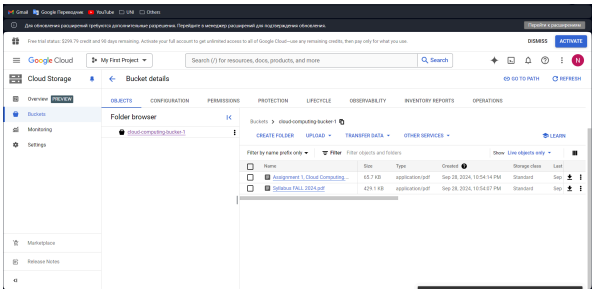
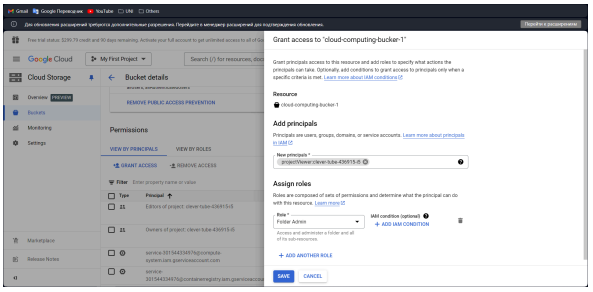
1. Pressed the button "Create Cluster" in "Kubernetes Engine".
2. Configured settings like region and machine types.
3. Deploy the cluster.

How did you verify that your application was successfully deployed and accessible?

By checking the status of the deployment with kubectl get pods or kubectl get services.

Access the service with the external IP assigned by GKE.

Exercise 5: Storing and Accessing Data in Google Cloud Storage

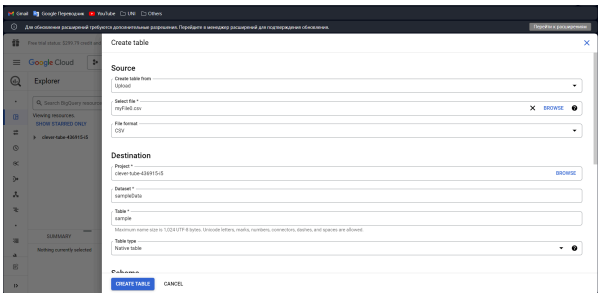
How do you create a Cloud Storage bucket, and what options are available during setup?

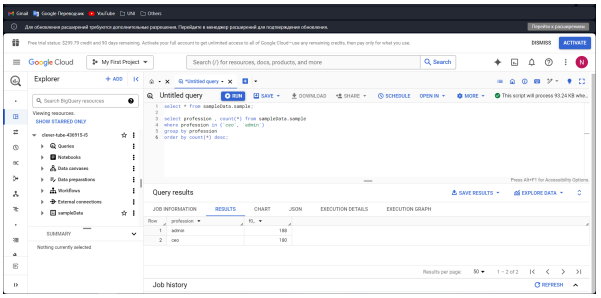
1. Click "Create Bucket" in the Cloud Storage section and set a name.
2. Select storage class (Standard, Nearline, Coldline) and region.
3. Configure access control (public or private).

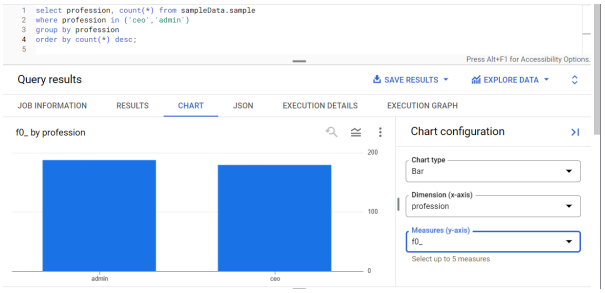
What are the differences between setting a bucket to public versus private?
Public: Files in the bucket can be accessed by anyone with the link.
Private: Only authorized users can access the files.

How can you manage access permissions for individual files in a bucket?
By using Google Cloud IAM roles or assigning specific permissions to users at the file level, granting roles like Viewer or Editor.

Exercise 6: Analyzing Data with BigQuery







What steps did you take to create a dataset and table in BigQuery?

1. Click "Create Dataset." in the BigQuery
2. Name the dataset and configure location and expiration settings.
3. Create a table by uploading data or defining a schema.

How did you write and execute SQL queries in BigQuery?
It is very similar to other SQL applications. I accessed the BigQuery console, selected the dataset and table, and then wrote queries for filtering, aggregating, and sorting data, using standard SQL commands like `SELECT`, `WHERE`, `GROUP BY`, and `ORDER BY`. The process is intuitive and very much like working with other SQL databases.

What insights were you able to derive from the data analysis?
Trends in user behavior, sales performance, or any relevant dataset pattern based on the queries executed.