

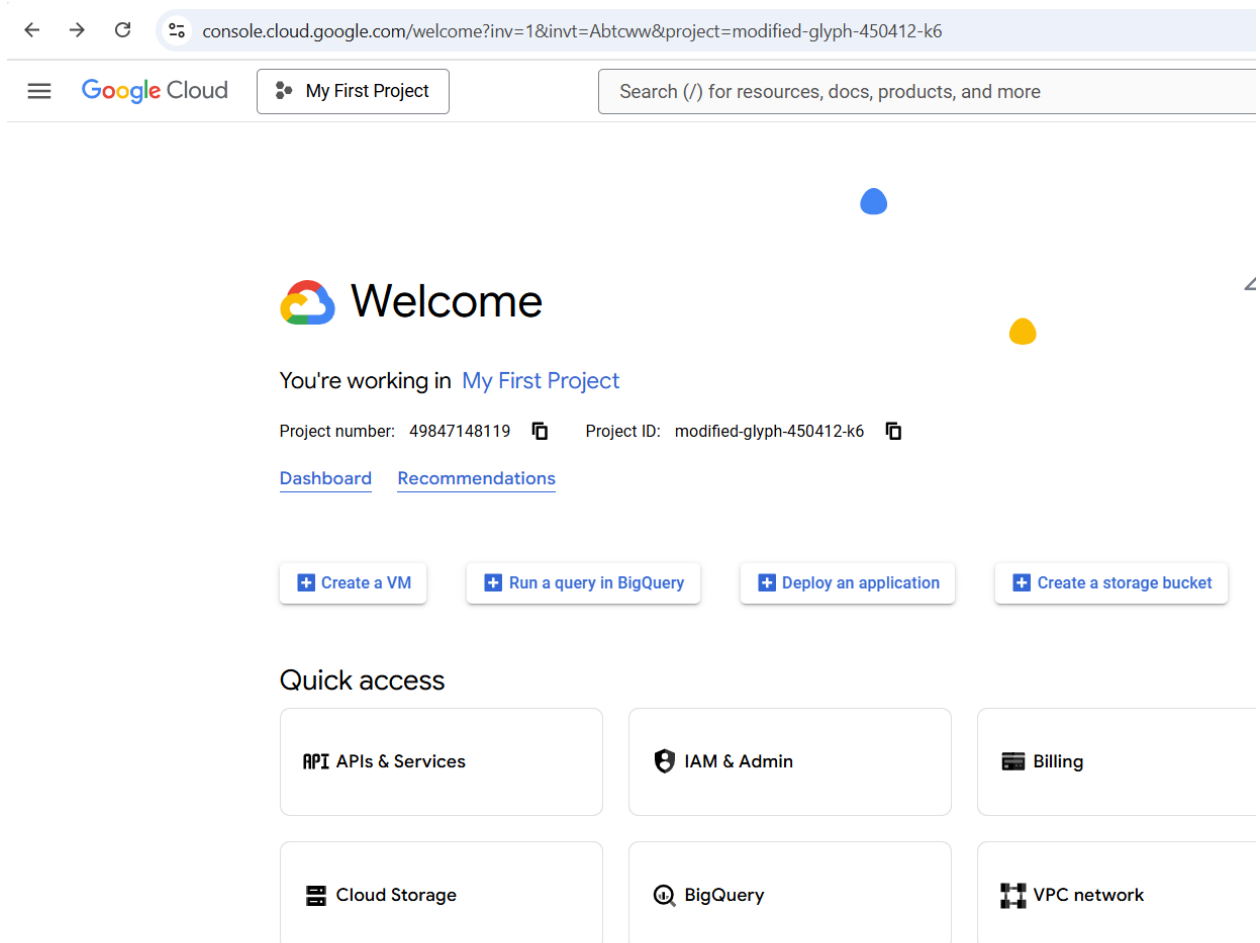
## Homework\_Lesson31\_Cloud1

1. Создайте новый бакет Amazon S3/GCP Cloud Storage тремя способами: через GUI (консоль в браузере), с помощью CLI, конфиг в terraform.
2. Создайте Amazon EC2/GCP Compute Engine и настройте масштабирование, чтобы автоматически добавлять новые экземпляры в случае увеличения нагрузки. Сделайте это тремя способами: через GUI (консоль в браузере) с помощью CLI, конфиг в terraform.
- 3.\* Создайте базу через GUI (консоль в браузере), с помощью CLI, конфиг в terraform данных Amazon RDS/GCP CloudSQL и подключитесь к ней из виртуальной машины, которую вы создали ранее. Сделайте это тремя способами: через GUI (консоль в браузере) с помощью CLI, конфиг в terraform.

### Выполнение 1-ого задания:

#### 1.1 Создание нового бакета GCP Cloud Storage через GUI:

- 1) Переходим в Google Cloud Console (<https://console.cloud.google.com/>).
- 2) Выбираем проект, в котором хотим создать бакет.
- 3) Перейходим в раздел "Cloud Storage".
- 4) Нажимаем "Create a storage bucket".
- 5) Указываем имя bucket.
- 6) Выбираем регион или Multi-Region.
- 7) Нажимаем "Create".



← → ↻ console.cloud.google.com/storage/create-bucket?inv=1&inv=Abtcww&project=modified-glyph-450412-k6

Google Cloud My First Project Search (/) for resources, docs, products, and more

Cloud Storage Create a bucket

- Overview
- Buckets
- Monitoring
- Storage Insights
- Settings

Marketplace Release Notes

<|

### Get Started

Pick a globally unique, permanent name. [Naming guidelines](#)

my-first-bucket-gcp-gui

Tip: Don't include any sensitive information

Optimize storage for data-intensive workloads

Labels (optional)

Continue

### Choose where to store your data

Location: us (multiple regions in United States)

Location type: Multi-region

### Choose a storage class for your data

Default storage class: Standard

### Choose how to control access to objects

Public access prevention: On

Access control: Uniform

Следующая страница, после нажатия (Create a storage bucket).

Здесь мы указываем имя bucket (которое должно быть уникальным). В нашем случае название “my-first-bucket-gcp-gui”. Также можно выбрать регион или Multi-Region и настроить другие параметры такие как например, класс хранения и параметры доступа и нажимаем “Create”.

console.cloud.google.com/storage/browser/my-first-bucket-gcp-gui;tab=objects?inv=1&inv=Abtcww&project=modified-glyph-450412-k6&prefix=&forceOnObjectsSorting

Google Cloud Мой первый проект Поиск (/) ресурсов, документов, продуктов и многого другого Поиск

Облачное хранилище Подробности о ковше

Обзор Buckets Мониторинг Информация о хранении Настройки

### мой-первый-ведро-gcp-gui

Расположение: us (несколько регионов в Соединенных Штатах) Класс хранения: Стандарт Публичный доступ: Не публичный Защита: Мягкое удаление

Объекты Конфигурация Разрешения Защита Жизненный цикл Наблюдаемость Новый Отчет

Браузер папок

my-first-bucket-gcp-gui

Ведра > мой-первый-ведро-gcp-gui

Создать папку Загрузить Передача данных Другие услуги

Фильтровать только по префиксу имени Фильтр Показывать Только

Имя	Размер	Тип	Созданный	Класс хранения	Последний
Нет строк для отображения					

Наш первый созданный bucket my-first-bucket-gcp-gui с помощью GCP Cloud Storage через GUI

## 1.2 Создание bucket через CLI:

1) Проверяем в Debian, в нашем случае WSL, под каким аккаунтом находимся с помощью команды:

```
$ gcloud config get-value account
```

```
$ gcloud config list
```

```
root@DESKTOP-UG6J7T7:~# gcloud config list
[core]
account = seriousmkr@gmail.com
disable_usage_reporting = True
project = modified-glyph-450412-k6

Your active configuration is: [default]
root@DESKTOP-UG6J7T7:~# |
```

2) Вводим команду на создание нашего bucket:

```
$ gsutil mb -l US gs://my-first-bucket-gcp-cli/
```

```
root@DESKTOP-UG6J7T7:~# gcloud config get-value account
seriousmkr@gmail.com
root@DESKTOP-UG6J7T7:~# gsutil mb -l US gs://my-first-bucket-gcp-cli/
Creating gs://my-first-bucket-gcp-cli/...
root@DESKTOP-UG6J7T7:~#
```

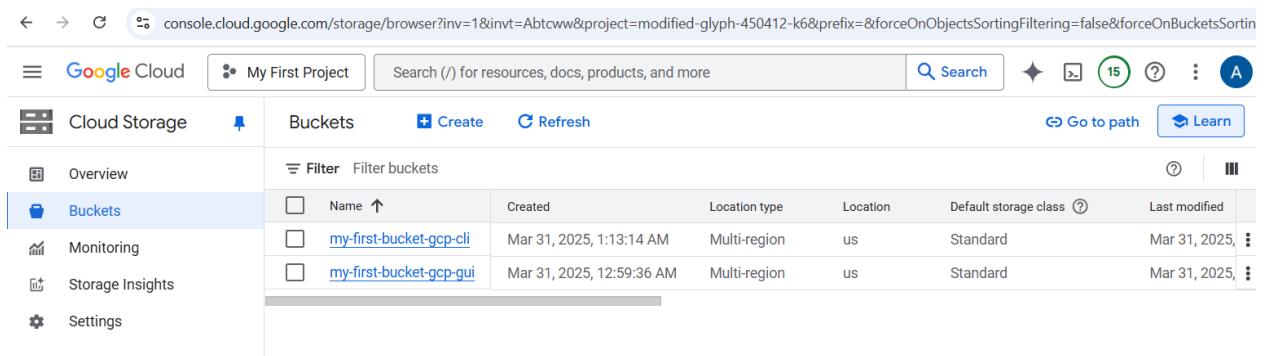
Для проверки на нашем аккаунте bucket вводим команду:

```
$ gsutil ls
```

```
root@DESKTOP-UG6J7T7:~# gsutil ls
gs://my-first-bucket-gcp-cli/
gs://my-first-bucket-gcp-gui/
root@DESKTOP-UG6J7T7:~# |
```

Вывод наших созданных bucket GCP

Также заходим на сайт в наш аккаунт GCP Buckets:



Name	Created	Location type	Location	Default storage class	Last modified
my-first-bucket-gcp-cli	Mar 31, 2025, 1:13:14 AM	Multi-region	us	Standard	Mar 31, 2025
my-first-bucket-gcp-gui	Mar 31, 2025, 12:59:36 AM	Multi-region	us	Standard	Mar 31, 2025

Созданный нами первый Bucket с помощью CLI на WSL (Debian)

### 1.3 Создание bucket с помощью terraform на WSL:

1) В корне директории ~/Homework\_Lesson31\_Cloud\_1 создаем файл main.tf и пишем скрипт на создание bucket:

```
provider "google" {
  credentials = file("/home/makarov/modified-glyph-450412-k6-a498aa55c17d.json")
  project = "modified-glyph-450412-k6"
  region = "US"
}

resource "google_storage_bucket" "default" {
  name = "my-first-bucket-gcp-terraform"
  location = "US"
  storage_class = "STANDARD"
}
```

Также для запуска выполнения terraform нужно указать credentials gcloud, т.е. создать ключи пользователя для нашего аккаунта. Для этого нужно с сайта GCP, создать пользователя или, у имеющего пользователя с правами создать файл ключа \*\*\*.json и проложить в какое-нибудь безопасное место. В нашем случае прокладываем в home-директорию пользователя WSL и в файле main.tf добавляем строку credentials и указываем путь, где лежит наш ключ. Если этого не сделать, при выполнении команды terraform init будет выдавать ошибку такого плана:

```
makarov@DESKTOP-UG6J7T7:~/Aliaksandr_Makarau_DOS24/Homework_Lesson31_Cloud_1$ terraform apply
Error: Attempted to load application default credentials since neither `credentials` nor `access_token` was set in the provider block. No credentials loaded. To
e your gcloud credentials, run `gcloud auth application-default login`. Original error: google: could not find default credentials. See https://cloud.google.com/d
/authentication/external/set-up-adc for more information

with provider["registry.terraform.io/hashicorp/google"],
on main.tf line 1, in provider "google":
1: provider "google" {
```

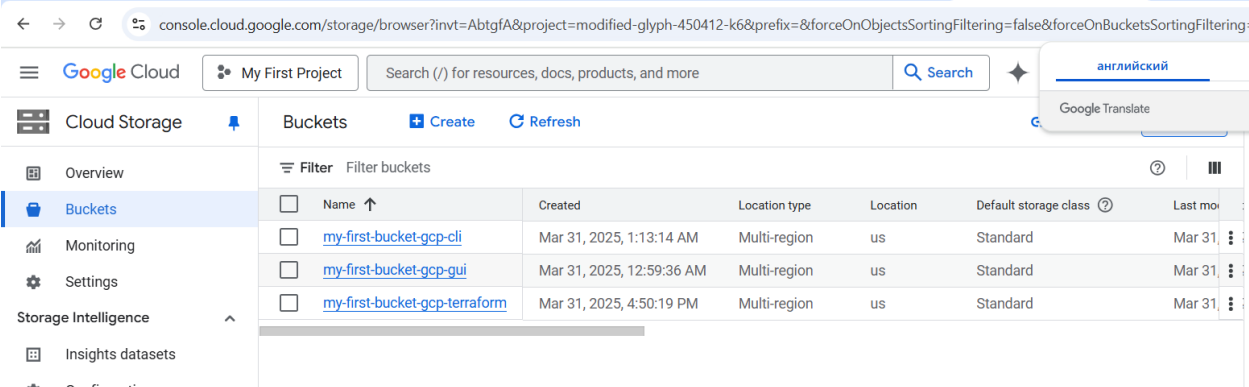
Ошибка связана с отсутствием необходимых учетных данных для доступа в Gcloud

После всех настроек, запускаем terraform:

\$ terraform init // инициализация каталога terraform

\$ terraform apply // команда запуска на создание. После выполнения, получаем запрос на подтверждение действий (Enter a value: yes)

Наш bucket после подтверждения создан в Gcloud, можем проверить с помощью команды “gsutil ls” или перейти в браузер GCP GUI:



The screenshot shows the Google Cloud Storage console interface. The left sidebar contains navigation links for Overview, Buckets, Monitoring, Settings, Storage Intelligence, Insights datasets, and Configuration. The main area displays a table of buckets under the 'Buckets' tab. The table has columns for Name, Created, Location type, Location, Default storage class, and Last modified. Three buckets are listed: 'my-first-bucket-gcp-cli', 'my-first-bucket-gcp-gui', and 'my-first-bucket-gcp-terraform'. The bucket 'my-first-bucket-gcp-terraform' is highlighted in blue.

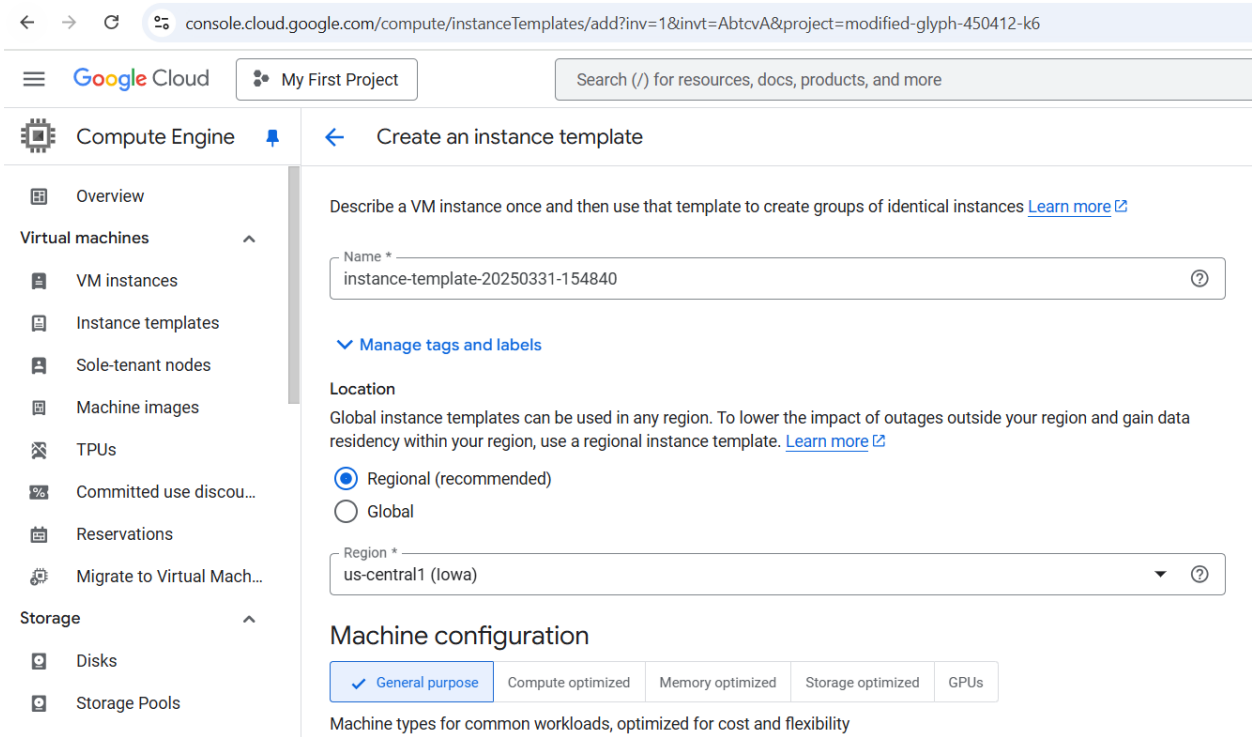
Name	Created	Location type	Location	Default storage class	Last modified
my-first-bucket-gcp-cli	Mar 31, 2025, 1:13:14 AM	Multi-region	us	Standard	Mar 31, 2025, 1:13:14 AM
my-first-bucket-gcp-gui	Mar 31, 2025, 12:59:36 AM	Multi-region	us	Standard	Mar 31, 2025, 12:59:36 AM
my-first-bucket-gcp-terraform	Mar 31, 2025, 4:50:19 PM	Multi-region	us	Standard	Mar 31, 2025, 4:50:19 PM

Созданный нами bucket с помощью terraform.

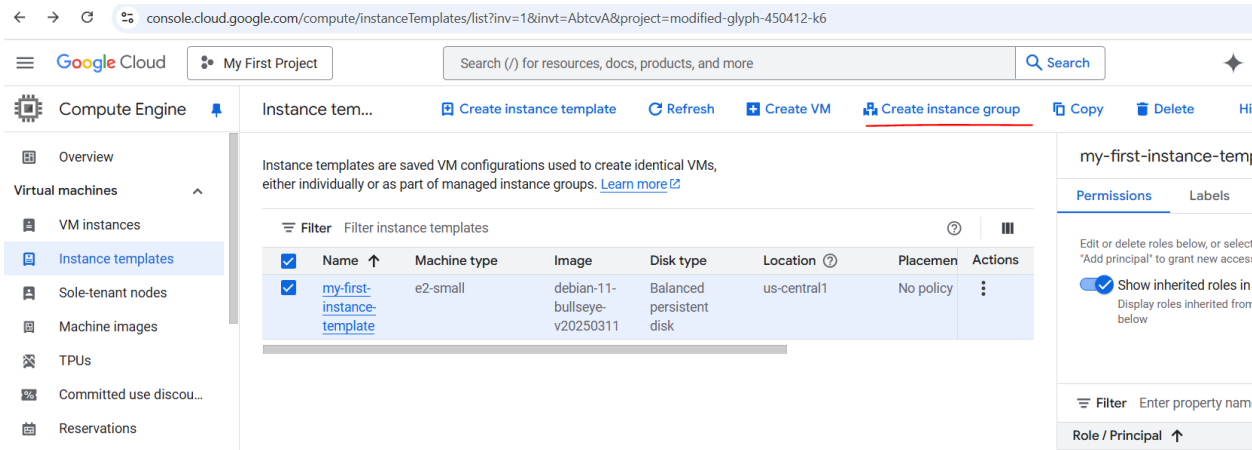
**Выполнение 2-ого задания:**

**2.1 Создание через GUI (Google Cloud Console):**

- 1) Входим в Google Cloud Console и переходим в раздел Compute Engine.
- 2) Нажимаем "Instance templates", создаем новый шаблон, и задаем настройки шаблона.
- 3) Переходим в раздел "Instance groups" и нажимаем "Create instance group".
- 4) Выбираем "Managed instance group" и указываем ранее созданный шаблон.
- 5) Настраиваем минимальное и максимальное количество экземпляров, а также правила для масштабирования (на основе нагрузки CPU).
- 6) Завершаем создание.



Выбираем “Create an instance template” и настраиваем все необходимые параметры



Созданный наш первый шаблон “my-first-instance-template”

Google Cloud

My First Project

Search (/) for resources, docs, products, and more

Sea

Compute Engine / Instance groups

Overview

Virtual machines

VM instances

Instance templates

Sole-tenant nodes

Machine images

TPUs

Committed use discou...

Reservations

Migrate to Virtual Mach...

Storage

Disks

Storage Pools

Snapshots

### Instance Groups

Instance groups let you organize VM instances or use them in a load-balancing backend service. You can group existing instances or create a group based on an instance template. [Learn more](#)

Create Instance GroupTake the quick start

Переходим в раздел "Instance groups" и нажимаем "Create instance group"

## Автомасштабирование

Используйте автомасштабирование для автоматического добавления и удаления экземпляров в группу в периоды высокой и низкой нагрузки. [Узнать больше](#)

Режим автомасштабирования

Вкл.: добавлять и удалять экземпляры в группе

Минимальное количество экземпляров

1

Максимальное количество экземпляров

10

### Автомасштабирование сигналов

Используйте сигналы, чтобы определить, когда следует масштабировать группу. [Узнать больше](#)

Загрузка ЦП: 60% (по умолчанию)

Предиктивное автомасштабирование отключено

(Не сохранено)

Настройки автомасштабирования в разделе "Create instance group"

Compute Engine / Instance groups

Overview

Virtual machines

VM instances

Instance templates

Sole-tenant nodes

Machine images

TPUs

Committed use discou...

Reservations

Instance groups

Create instance group

Refresh

Delete

Instance groups are collections of VM instances that use load balancing and automated services, like autoscaling and autohealing. [Learn more](#)

Filter Enter property name or value

Status	Name ↑	Instances	Template	Group type	Creation time	Recommendation
<input type="checkbox"/>	<a href="#">my-first-instance-group-1</a>	0 → 1	<a href="#">my-first-instance-template (Regional)</a>	Managed	Mar 31, 2025, 6:47:47 PM UTC+03:00	

## 2.2 Создание и настройка масштабирования через CLI:

1) Создаем шаблон экземпляра:

```
$ gcloud compute images list | grep debian* // список дистрибутивов debian
$ gcloud compute machine-types list // список имеющихся типов машин и CPU
$ gcloud compute zones list // список имеющихся зон
```

Создаем с имеющейся информации шаблон:

```
$ gcloud compute instance-templates create my-first-template-cli --machine-type=e2-micro -
-image-family=debian-11 --image-project=debian-cloud
```

```
makarov@DESKTOP-UG6J7T7:~/Aliaksandr_Makarau_DOS24/Homework_Lesson31_Cloud_1$ gcloud compute instance-templates create my-first-template-cli --machine-type=e2-micro -
-image-family=debian-11 --image-project=debian-cloud
Created [https://www.googleapis.com/compute/v1/projects/modified-glyph-450412-k6/global/instanceTemplates/my-first-template-cli].
NAME MACHINE_TYPE PREEMPTIBLE CREATION_TIMESTAMP
my-first-template-cli e2-micro 2025-03-31T14:09:14.206-07:00
makarov@DESKTOP-UG6J7T7:~/Aliaksandr_Makarau_DOS24/Homework_Lesson31_Cloud_1$
```

2) Создаем управляемую группу экземпляров:

```
$ gcloud compute instance-groups managed create my-first-instance-group-cli --base-instanc
e-name my-instance --size=1 --template=my-first-template-cli --zone=us-central1-a
```

```
makarov@DESKTOP-UG6J7T7:~/Aliaksandr_Makarau_DOS24/Homework_Lesson31_Cloud_1$ gcloud compute instance-groups managed create my-first-instance-group-cli --base-instanc
e-name my-instance --size=1 --template=my-first-template-cli --zone=us-central1-a
Created [https://www.googleapis.com/compute/v1/projects/modified-glyph-450412-k6/zones/us-central1-a/instanceGroupManagers/my-first-instance-group-cli].
NAME LOCATION SCOPE BASE_INSTANCE_NAME SIZE TARGET_SIZE INSTANCE_TEMPLATE AUTOSCALED
my-first-instance-group-cli us-central1-a zone my-instance 0 1 my-first-template-cli no
makarov@DESKTOP-UG6J7T7:~/Aliaksandr_Makarau_DOS24/Homework_Lesson31_Cloud_1$ gcloud compute instance-groups list
NAME LOCATION SCOPE NETWORK MANAGED INSTANCES
my-first-instance-group-cli us-central1-a zone default Yes 1
makarov@DESKTOP-UG6J7T7:~/Aliaksandr_Makarau_DOS24/Homework_Lesson31_Cloud_1$
```

3) Настраиваем масштабирование:

```
$ gcloud compute instance-groups managed set-autoscaling my-first-instance-group-cli --max-
num-replicas=5 --min-num-replicas=1 --target-cpu-utilization=0.6 --zone=us-central1-a
```

```
makarov@DESKTOP-UG6J7T7:~/Aliaksandr_Makarau_DOS24/Homework_Lesson31_Cloud_1$ gcloud compute instance-groups managed set-autoscaling my-first-instance-group-cli --max-num-replic
s=5 --min-num-replicas=1 --target-cpu-utilization=0.6 --zone=us-central1-a
Created [https://www.googleapis.com/compute/v1/projects/modified-glyph-450412-k6/zones/us-central1-a/autoscalers/my-first-instance-group-cli-3va3].
---
autoscalingPolicy:
  coolDownPeriodSec: 60
  cpuUtilization:
    utilizationTarget: 0.6
  maxNumReplicas: 5
  minNumReplicas: 1
  mode: ON
creationTimestamp: '2025-03-31T14:27:07.767-07:00'
id: '5920433835904251460'
kind: compute#autoscaler
name: my-first-instance-group-cli-3va3
selflink: https://www.googleapis.com/compute/v1/projects/modified-glyph-450412-k6/zones/us-central1-a/autoscalers/my-first-instance-group-cli-3va3
status: ACTIVE
target: https://www.googleapis.com/compute/v1/projects/modified-glyph-450412-k6/zones/us-central1-a/instanceGroupManagers/my-first-instance-group-cli
zone: https://www.googleapis.com/compute/v1/projects/modified-glyph-450412-k6/zones/us-central1-a
```

## 2.3 Создание и настройка масштабирования через Terraform.

1) В корне директории ~/Homework\_Lesson31\_Cloud\_1 создаем файл папку "terraform\_inst\_templates" и в нем файл main.tf, где будем создавать и настраивать масштабирование:

```
provider "google" {
  credentials = file("/home/makarov/modified-glyph-450412-k6-a498aa55c17d.json")
  project = "modified-glyph-450412-k6"
  region = "us-central1"
}

resource "google_compute_instance_template" "default" {
  name = "my-first-template-terraform"
  machine_type = "e2-micro"

  disk {
    auto_delete = true
    boot = true
    source_image = "debian-cloud/debian-11"
  }

  network_interface {
    network = "default"
    access_config {

    }
  }
}

resource "google_compute_instance_group_manager" "default" {
  name = "my-first-instance-group-terraform"
  version {
    instance_template = google_compute_instance_template.default.id
  }
  base_instance_name = "my-first-instance"
  target_size = 1
  zone = "us-central1-a"
}

resource "google_compute_autoscaler" "default" {
  name = "default-autoscaler"
  zone = "us-central1-a"
  target = google_compute_instance_group_manager.default.id
  autoscaling_policy {
    max_replicas = 3
    min_replicas = 1
    cpu_utilization {
      target = 0.6
    }
  }
}
```

Выполняем команды для запуска:

```
$ terraform init
$ terraform plan
$ terraform apply
Enter: yes
```



```
}
}
}

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

google_compute_autoscaler.default: Creating...
google_compute_autoscaler.default: Creation complete after 7s [id=projects/modified-glyph-450412-k6/zones/us-central1-a/autoscalers/default-autoscaler]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
makarov@DESKTOP-UG6J7T7:~/Aliaksandr_Makarau_DOS24/Homework_Lesson31_Cloud_1/terraform_inst_templates$
```

Успешное создание Compute Engine и настроен пробное масштабирование через Terraform

Проверяем через браузер об успешном создании с помощью terraform:

<input type="checkbox"/>	<a href="#">my-first-template-terraform</a>	e2-micro	debian-11	Standard persistent disk	global	No policy	⋮
--------------------------	---	----------	-----------	--------------------------	--------	-----------	---

Наш созданный template с помощью terraform

<input checked="" type="checkbox"/>	<a href="#">my-first-instance-group-terraform</a>	1	<a href="#">my-first-template-terraform</a>	Managed	Apr 2, 2025, 12:21:05 PM UTC+03:00	On: Target CPU utilization 60%	us-central1-a
-------------------------------------	---	---	---	---------	------------------------------------	--------------------------------	---------------

Instance-group

Autoscaling	
Autoscaling mode	On
Minimum # of instances	1
Maximum # of instances	3
Initialization period	60 seconds
Autoscaling signal	
CPU utilization	60%
Predictive autoscaling	Off
Scale in controls	Off
Scaling schedules	<a href="#">Manage schedules</a>

Autoscaling (настроено масштабирование)