

## Assignment 1, Cloud Application Development

Put all deliverables into github repository in your profile. Share link to google form 24 hours before defense. Defend by explaining deliverables and answering questions.

Deliverables: report in pdf

Google form:

[https://docs.google.com/forms/d/e/1FAIpQLSe0GyNdOYlvM1tX\\_I\\_CtlPod5jBf-ACLGdHYZq1gVZbUeBzlg/viewform?usp=sf\\_link](https://docs.google.com/forms/d/e/1FAIpQLSe0GyNdOYlvM1tX_I_CtlPod5jBf-ACLGdHYZq1gVZbUeBzlg/viewform?usp=sf_link)

### Exercise 1: Setting Up Google Cloud SDK

1. **Objective:** Install and configure the Google Cloud SDK on your local machine.
2. **Steps:**
  - Visit the Google Cloud SDK installation page.
  - Follow the instructions to download and install the SDK for your operating system.
  - After installation, run `gcloud init` to initialize the SDK and authenticate with your Google account.
  - Configure the default project and region.
  - Verify the installation by running `gcloud version` and `gcloud info`.
3. **Questions:**
  - What command did you use to authenticate with your Google account? `gcloud auth login`
  - How did you set the default project? `gcloud config set project`
  - What information does the `gcloud info` command provide?

SDK version, active configuration (default project, region, etc.).

python version and environment details, installation directory information on any available updates for the SDK, network diagnostic information if relevant

### Exercise 2: Exploring Cloud Shell

1. **Objective:** Familiarize yourself with the Google Cloud Shell environment.
2. **Steps:**
  - Open the Google Cloud Console and activate Cloud Shell.
  - Explore the environment by listing files and checking the available tools.
  - Run the command `gcloud config list` to see your current configuration.
  - Create a directory named `gcp-intro` and navigate into it.
  - Use the built-in code editor to create a simple `README.md` file describing your GCP project.
3. **Questions:**
  - What is the default home directory in Cloud Shell? `/home/<username>`
  - What tools are pre-installed in Cloud Shell? `gcloud, kubectl, git, python, pip, npm, node, vim, nano, etc`
  - How can you open the built-in code editor in Cloud Shell?  
`pencil icon or cloudshell edit .`

### Exercise 3: Managing Projects with Google Cloud SDK

1. **Objective:** Use Google Cloud SDK to manage projects.
2. **Steps:**
  - List all the projects associated with your Google account using `gcloud projects list`.
  - Create a new project with the command `gcloud projects create PROJECT_ID --name="My First GCP Project"`.
  - Set this new project as your default project.
  - Explore project metadata using `gcloud projects describe PROJECT_ID`.
  - Delete the project using `gcloud projects delete PROJECT_ID` after completing the exercise.
3. **Questions:**
  - How do you list all projects associated with your account? `gcloud projects list`
  - What command is used to set a default project? `gcloud config set project test-project-434508`
  - How do you describe project metadata? `gcloud projects describe test-project-434508`

## Exercise 4: Using Cloud Shell for Basic Operations

1. **Objective:** Perform basic file and directory operations in Cloud Shell.
2. **Steps:**
  - In Cloud Shell, create a directory structure that mimics a small project (e.g., `myproject/src`, `myproject/tests`, `myproject/docs`).
  - Create a few files in these directories and use commands like `touch`, `nano`, `cat`, and `rm` to manipulate them.
  - Use `gsutil` to create a new Cloud Storage bucket and upload a file from your Cloud Shell environment.
  - Verify the file upload by listing the contents of the bucket.
3. **Questions:**
  - What command did you use to create the directory structure? `mkdir -p myproject/src myproject/tests myproject/docs`
  - How did you upload a file to a Cloud Storage bucket? `gsutil mb gs://my-bucket-name`  
`gsutil cp path/to/your/file.txt gs://my-bucket-name`
  - How can you list the contents of a Cloud Storage bucket? `gsutil ls gs://my-bucket-name`

## Exercise 5: Automating Tasks with Shell Scripts in Cloud Shell

1. **Objective:** Write and execute a basic shell script in Cloud Shell.
2. **Steps:**
  - In Cloud Shell, create a new shell script named `setup.sh` in your `gcp-intro` directory.
  - The script should automate the creation of a new directory, a simple text file, and set up a basic Google Cloud configuration (e.g., set a default project).
  - Make the script executable using `chmod +x setup.sh`.
  - Run the script and verify that it performs the expected tasks.
3. **Questions:**

- What command did you use to make the script executable? `chmod +x setup.sh`
- How did you ensure the script was executed correctly? `./setup.sh`
- What steps did your script automate?

`create user and configure ssh connect`