**LAB1  
App Dev: Setting up a Development Environment v1.1**

**Task 1: Creating a Compute Engine Virtual Machine Instance**

In this lab, you will provision a Google Compute Engine virtual machine and install software libraries for Node.js software development on Google Cloud Platform.

1. In the Cloud Platform Console, on the Navigation menu, click Compute Engine 🡪 VM Instances page, click Create.
2. On the Create an instance page, for Name type Name for instance 🡪 select (us-central1) for region and (us-central1-a) for the zone
3. In the **Identity and API access > Access Scopes** section, select **Allow full access to all Cloud APIs**.
4. In the **Firewall** section,Enable **Allow HTTP traffic** (this will create firewall rule).
5. Leave the remaining settings as their defaults and click **Create**.
6. On the VM instances page, click SSH of your vm instance to open instance window.

**Task 2: Install software on the VM instance**

1. To update the Debian package list, execute the following command:

sudo apt-get update

1. To install Git, execute the following command:

sudo apt-get install git

If prompted, press Enter.

1. To download the Node.js setup script, execute the following command:

curl -sL https://deb.nodesource.com/setup\_6.x | sudo -E bash -

1. To install npm and Node.js, execute the following command:

sudo apt install nodejs

**Task 3: Configure the VM to Run Application Software**

1. To check the version of Node.js,use:

node -v

1. To clone the class repository, use:

git clone https://github.com/GoogleCloudPlatform/training-data-analyst

To change the working directory, use:

cd ~/training-data-analyst/courses/developingapps/nodejs/devenv/

1. To run a simple web server, use:

sudo node server/app.js

1. Return to the Cloud Console VM instances list, and click on the External IP address for the dev-instance.

You should see a Hello GCP dev! message from Node.js.

1. Return to the SSH window, and stop the application by pressing Ctrl+C.
2. To install the Node.js library for Compute Engine use:

npm install

1. To run a simple Node.js application that lists Compute Engine instances, use:

node list-gce-instances.js

END.

**LAB 2**

**Explore a BigQuery Public Dataset.**

In this lab you:

* Query a public dataset
* Create a custom table
* Load data into a table
* Query a table

Navigate to BigQuery in the GCP console.

* 1. Navigate to ADD DATA > Explore public datasets 🡪 we will be exploring the USA NAMES dataset.

**Task 1: Query the USA Name dataset**

1. In the Query editor text area enter the following and click RUN:

SELECT

name, gender,

SUM(number) AS total

FROM

`bigquery-public-data.usa\_names.usa\_1910\_2013`

GROUP BY

name, gender

ORDER BY

total DESC

LIMIT

10

**Task 2: Create a custom table**

1. Download the [baby names zip file](http://www.ssa.gov/OACT/babynames/names.zip) to your local computer and unzip. We will be using the file yob2014.txt. The file is a comma-separated value (CSV) file with the following three columns: name, sex (M or F), and number of children with that name. The file has no header row.
2. Note the location of the yob2014.txt file so that you can find it later.

**Task 3: Create a dataset**

1. Back in the console, in the left pane, in the Resources section, click your GCP Project ID and then click CREATE DATASET.
2. On the Create dataset page: For Dataset ID, enter babynames. For Data location, choose United States (US). For Default table expiration, leave the default value 🡪 create dataset.

**Task 4: Load the data into a new table**

In this section, you load data into the table you made.

1. Click **babynames** found in the left pane in the **Resources** section, and then click **Create table**. Use the default values for all settings unless otherwise indicated.
2. On the **Create table** page:

* For **Source**, choose **Upload** from the Create table from: dropdown menu.
* For **Select file**,the yob2014.txt file and click **Open**.
* For **File format**, choose **CSV** from the dropdown menu.
* For **Table name**, enter table name.
* In the **Schema** section, click the **Edit as text** toggle and paste the following:

name:string,gender:string,count:integer

* Click **Create table**

In the **Query editor - to** retrieve the top 5 baby names for US males in 2014.

SELECT

name, count

FROM

`babynames.names\_2014`

WHERE

gender = 'M'

ORDER BY count DESC LIMIT 5.

END.

**LAB3**

**Google Cloud Fundamentals: Getting Started with Cloud Marketplace.**

In this lab, you use Cloud Marketplace to quickly and easily deploy a LAMP stack on a Compute Engine instance.

**Task 1: Use Cloud Marketplace to deploy a LAMP stack**

1. In the GCP Console, on the Navigation menu go to Marketplace. In the search bar, type LAMP. Click on LAMP Certified by Bitnami. and Launch.
2. For **Zone**, select the deployment zone that Qwiklabs assigned. If you are prompted to accept the GCP Marketplace Terms of Service, do so.
3. Click **Deploy**.

When the deployment of the infrastructure is complete, the status changes to **lampstack-1 has been deployed**.

**Task 3: Verify your deployment.**

When the deployment is complete, you can visit the Site address link in the right pane.

1. On the GCP Console, under Get started with LAMP Certified by Bitnami, click SSH.
2. In the SSH window, to change the current working directory to /opt/bitnami, use:

cd /opt/bitnami

1. To copy the phpinfo.php script from the installation directory to a publicly accessible location under the web server document root, use :

sudo sh -c 'echo "<?php phpinfo(); ?>" > apache2/htdocs/phpinfo.php'

The phpinfo.php script displays your PHP configuration. It is often used to verify a new PHP installation.

1. To close the SSH window,type:

exit

1. In a new browser tab, Type the following URL, and replace SITE\_ADDRESS with the URL in the Site address field in the right pane of the **lampstack** page.

http://SITE\_ADDRESS/phpinfo.php

A summary of the PHP configuration of your server is displayed.

1. Close the **phpinfo** tab.

END.