

Assignment and Exam Content

App Engine

**Disable Your App Engine
when not using it.
Go To App Engine-> Settings**

Always Delete your Cloud Resources to Avoid \$\$ Charges.

App Engine lab

App Engine Lab Contains – Three major areas below to say complete Lab 😊

A

Create App Engine

App Engine is Locations Specific – Regional.

B

Deploy Application

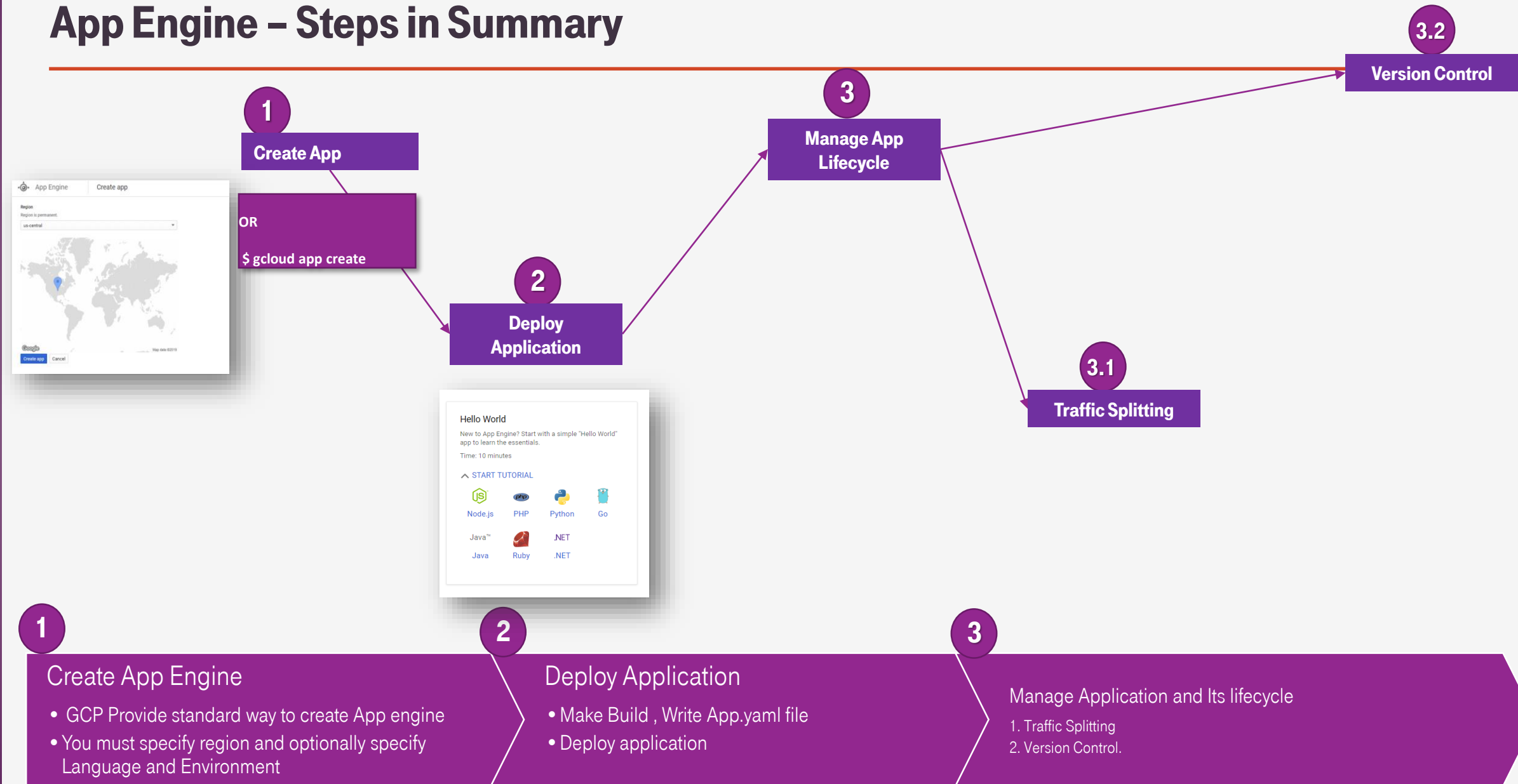
Once the cluster is ready , you can deploy Sample Application Bundle. Available to deploy.

3

Manage Application on in App Engine.

Monitor App, Traffic Splitting/ Canary Deployment etc etc

App Engine – Steps in Summary



1 App Engine Creation

Create App Engine

- GCP Provide standard way to create App engine
- You must specify region and optionally specify Language and Environment

Deploy Application

- Make Build , Write App.yaml file
- Deploy application

Manage Application and Its lifecycle

1. Traffic Splitting
2. Version Control.

1 Go To -> COMPUTE -> App Engine-> Dashboard

2 There are multiple way you can create App Engine – If you project have it.

Console (UI)

Dashboard -> Create App Engine -> Select Location (region) ->

Next page – Select Node.js or your own choice and Select Standard env and Click next. App Engine is created

Alternatively Go to Cloud Shell (CLI)

```
$ gcloud app create
```

For Current lab

Use Any method and create app engine with region of your choice.

You can disable App Engine – if you don't want Google to charge you continuously

Please disable App if not in use

The screenshots illustrate the process of creating and managing an App Engine application. The first screenshot shows the 'Create app' dialog with a map of the United States and a 'Create app' button. The second screenshot shows the 'Language' and 'Environment' selection screen, with 'Node.js' selected for language and 'Standard' for environment. The third screenshot shows the 'Next steps' page with links to documentation and SDKs. The fourth screenshot shows the 'Settings' page for the application, with a red box highlighting the 'Disable application' button. A purple box labeled 'Kubernetes Clusters' points to the 'App Engine' link in the left sidebar of the Google Cloud Platform dashboard.

2 Application Deployment

Create App Engine

- GCP Provide standard way to create App engine
- You must specify region and optionally specify Language and Environment

Deploy Application

- Make Build , Write App.yaml file
- Deploy application

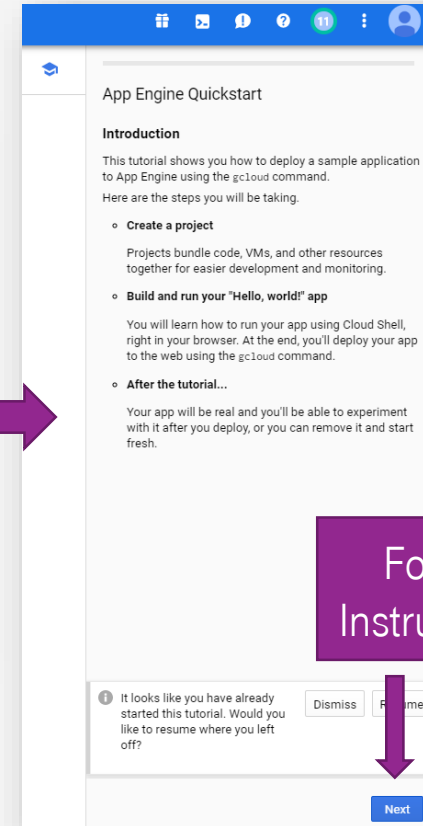
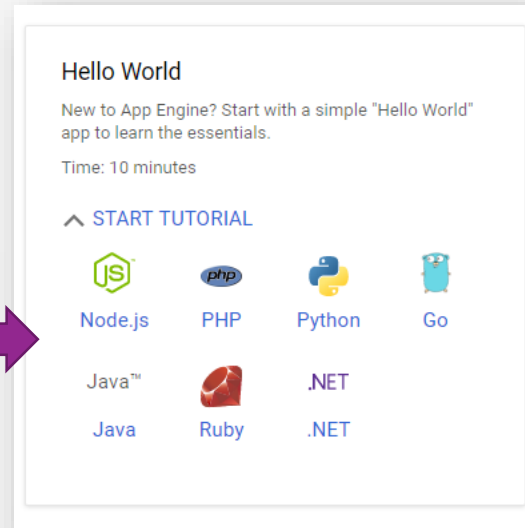
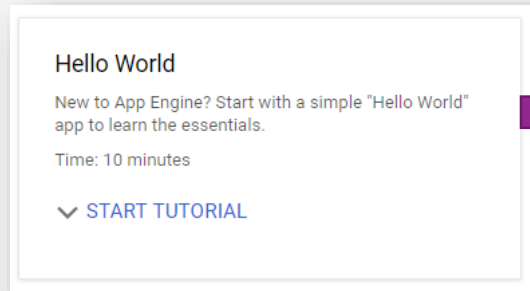
Manage Application and Its lifecycle

1. Traffic Splitting
2. Version Control.

2 You can deploy App using CLI and UI , The Best way to start and learn is UI and App Engine provides sample helloworld application for all supported language.

Go to Dashboard -> Hello World -> Start Tutorial

Choose language of your choice e.g. Node.js or Java



Follow Instructions

Congratulations
You have
deployed App.

If you are in project folder – You can deploy application using just one command

```
$ gcloud app deploy
```

or

```
$ gcloud app deploy ~/my_app/app.yaml
```

If you want to look at all options – Please check this link -> <https://cloud.google.com/sdk/gcloud/reference/app/deploy>

My Command Log

\$ git clone https://github.com/GoogleCloudPlatform/nodejs-getting-started

Cloning into 'nodejs-getting-started'...
remote: Enumerating objects: 3497, done.
remote: Total 3497 (delta 0), reused 0 (delta 0), pack-reused 3497
Receiving objects: 100% (3497/3497), 1.26 MiB | 0 bytes/s, done.
Resolving deltas: 100% (2642/2642), done.

\$ cd nodejs-getting-started/1-hello-world

\$ export PORT=8080 && npm install

> core-js@2.6.9 postinstall /home/ujjwalmusale/nodejs-getting-started/1-hello-world/node_modules/core-js

> node scripts/postinstall || echo "ignore"

Thank you for using core-js (https://github.com/zloirock/core-js) for polyfilling JavaScript standard library!

The project needs your help! Please consider supporting of core-js on Open Collective or Patreon:

> https://opencollective.com/core-js

> https://www.patreon.com/zloirock

Also, the author of core-js (https://github.com/zloirock) is looking for a good job -)

npm notice created a lockfile as package-lock.json. You should commit this file.

npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.9 (node_modules/fsevents):

npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.9: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})

added 856 packages from 672 contributors and audited 6675 packages in 39.961s

found 1 low severity vulnerability

run `npm audit fix` to fix them, or `npm audit` for details

\$ npm start

> nodejs-getting-started@1.0.0 start /home/ujjwalmusale/nodejs-getting-started/1-hello-world

> node app.js

App listening on port 8080

^C

\$ gcloud app deploy

Services to deploy:

descriptor: [/home/ujjwalmusale/nodejs-getting-started/1-hello-world/app.yaml]

source: [/home/ujjwalmusale/nodejs-getting-started/1-hello-world]

target project: [tokyo-data-11111]

target service: [default]

target version: [20190611t214249]

target url: [https://tokyo-data-243419.appspot.com]

Do you want to continue (Y/n)?

Beginning deployment of service [default]...

Uploading 6 files to Google Cloud Storage

File upload done.

Updating service [default]...done.

Setting traffic split for service [default]...done.

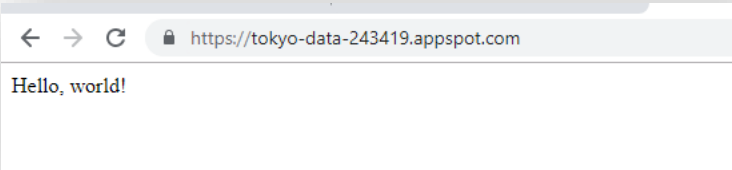
Deployed service [default] to [https://tokyo-data-243419.appspot.com]

You can stream logs from the command line by running:

\$ gcloud app logs tail -s default

To view your application in the web browser run:

\$ gcloud app browse



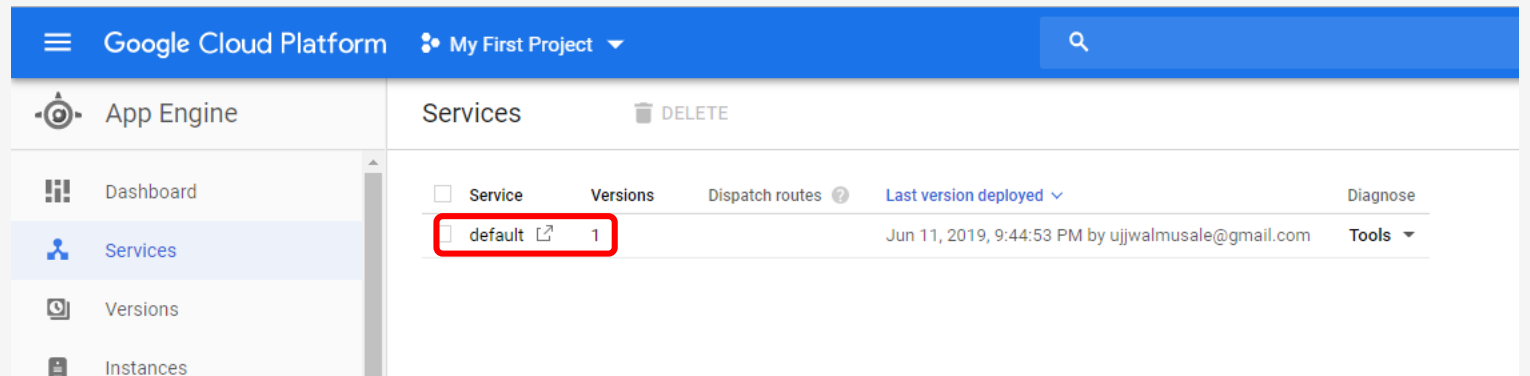
View Application

Go To Service

COMPUTE -> App Engine -> SERVICES

Click on Endpoint

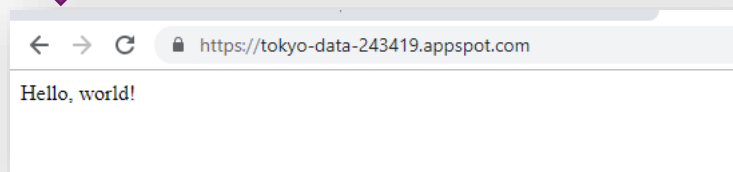
- You should see



Google Cloud Platform My First Project

App Engine Services DELETE

<input type="checkbox"/>	Service	Versions	Dispatch routes ?	Last version deployed ▾	Diagnose
<input type="checkbox"/>	default ↗	1		Jun 11, 2019, 9:44:53 PM by ujwalmusale@gmail.com	Tools ▾



3 Explore Application

Create App Engine

- GCP Provide standard way to create App engine
- You must specify region and optionally specify Language and Environment

Deploy Application

- Make Build , Write App.yaml file
- Deploy application

Manage Application and Its lifecycle

1. Traffic Splitting
2. Version Control.

Explore Application Make some request from browser

1 Go to Dashboard - See Summary, Billing Status , Current Load and Errors

2. Click On service and go to -> Diagnose. Under Tools You can find multiple options explore.

3. Go To Instance and See versions -. If you have one application – All Traffic must be served by same version.

Open app.js and add Version 2 along with Hello World

“Hello, world! Version 2”

Deploy application again.

Now You can see Multiple versions running

But only one version serving traffic.

The image displays three screenshots of the Google Cloud Platform (GCP) App Engine interface, illustrating the process of exploring and managing an application.

Top Screenshot: Dashboard

The Dashboard shows the application's summary, including a graph of Total requests (pink line) and Client (400) (orange line) over time. The graph shows a peak in requests around 10:00 PM on June 11, 2019. Below the graph, the Instance summary table shows the following data:

Resource	Usage	Billed	Price	Cost
Frontend Instance Hours	0.25 Instance Hours	0.00	\$0.05 / Hour	\$0.00
Outgoing Bandwidth	0.000001 GB	0.00	\$0.12 / GB	\$0.00
Estimated cost for the last 24 hours				\$0.00*

Middle Screenshot: Versions

The Versions page shows a table of application versions. The table has columns for Version, Status, Traffic Allocation, Instances, Runtime, Environment, Size, Deployed, Diagnose, and Config. The table shows one version, 20190611c214249, which is Serving and has 100% Traffic Allocation and 1 Instance.

Version	Status	Traffic Allocation	Instances	Runtime	Environment	Size	Deployed	Diagnose	Config
20190611c214249	Serving	100%	1	nodejs10	Standard	296.1 KB	Jun 11, 2019, 9:44:53 PM by ujijwalmusale@gmail.com	Tools	View

Bottom Screenshot: Versions

The Versions page shows a table of application versions. The table has columns for Version, Status, Traffic Allocation, Instances, Runtime, Environment, Size, Deployed, Diagnose, and Config. The table shows two versions, 20190611c220104 and 20190611c214249, both of which are Serving. The first version has 100% Traffic Allocation and 0 Instances, while the second version has 0% Traffic Allocation and 1 Instance.

Version	Status	Traffic Allocation	Instances	Runtime	Environment	Size	Deployed	Diagnose	Config
20190611c220104	Serving	100%	0	nodejs10	Standard	296.1 KB	Jun 11, 2019, 10:01:26 PM by ujijwalmusale@gmail.com	Tools	View
20190611c214249	Serving	0%	1	nodejs10	Standard	296.1 KB	Jun 11, 2019, 9:44:53 PM by ujijwalmusale@gmail.com	Tools	View

Traffic Splitting

We already saw we can do traffic Splitting using App Engine

- Select both versions and Click SPLIT TRAFFIC
- Select IP Address (Session Affinity)
- Define your own % for traffic.
- Now You should See Hello World message Depends on % Traffic



Version	Status	Traffic Allocation	Instances	Runtime	Environment	Size	Deployed	Diagnose	Config
20190611t220104	Serving	100%	0	nodejs10	Standard	296.1 KB	Jun 11, 2019, 10:01:26 PM by uj@walmusale@gmail.com	Tools	View
20190611t214249	Serving	0%	1	nodejs10	Standard	296.1 KB	Jun 11, 2019, 9:44:53 PM by uj@walmusale@gmail.com	Tools	View

You can split incoming traffic to different versions of your app. Traffic splitting is useful for slowly rolling out new versions or A/B testing different designs and features [Learn more](#)

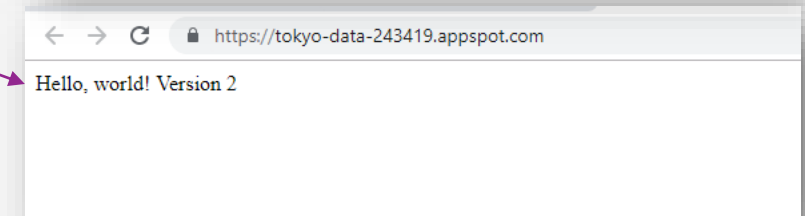
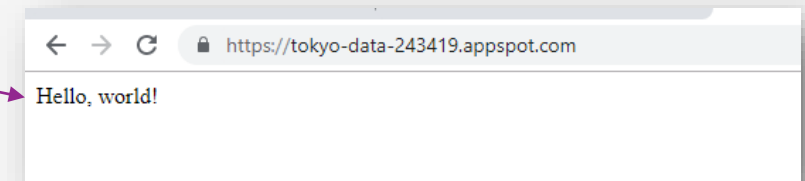
Split traffic by:
☒ IP address
☐ Cookie
☐ Random

Traffic allocation

Version	Allocation
20190611t220104	72 %
20190611t214249	28 %

[Add version](#)

[Save](#) [Cancel](#)



Additional workshops

- Try different programming as Sample Application
- Explorer remaining Menus in App Engine
- Explorer all available information in Instance
- Build your own deployment configuration for Flexible App Engine and provide Resources required.

**Disable Your App Engine
when not using it.
Go To App Engine-> Settings**

Important For Exam

- App Versions
- App Engine Location
- Multiple App Deployment - > Using Different Service.
- Applications with different Programming
- Providing Custom resources for Flexible App Engine.

**Disable Your App Engine
when not using it.
Go To App Engine-> Settings**

App Engine

**Disable Your App Engine
when not using it.
Go To App Engine-> Settings**

End of App Engine lab

**Always Delete your Cloud Resources to Avoid \$\$ Charges.
Disable App Engine**