

Session 03

Google App Engine

Objectives

- ❑ Describe the Google App Engine cloud environment
- ❑ Describe the various features of Google App Engine
- ❑ Explain the development and deployment in Google App Engine
- ❑ Explain the different types of storage services provided in Google App Engine
- ❑ List and describe the different types of API services offered by Google App Engine





Introduction

Google provides a distributed and scalable cloud computing environment called Google App Engine.

Google App Engine, also referred to as App Engine, is a platform that allows you to develop Web applications that are hosted on Google infrastructure.

The Google infrastructure provides necessary software to build and maintain the Web applications and scales millions of users' requests and data storage requirements.

The users can simply upload the ready-to-run Web applications to Google-managed servers that make applications accessible to outside world in a very short time.

Google App Engine platform combines building, testing, and maintaining of applications in a highly scalable cloud computing environment.

Features of Google App Engine 1-2

Google App Engine supports various languages, such as Java, Python, PHP, and Go to develop Web applications.

Each language has a runtime and a SDK containing tools that can be used to develop and deploy an application, and test it locally.

Google App Engine manages components of the Web application lifecycle such as request logs, application status check, and application version updating, and so on.

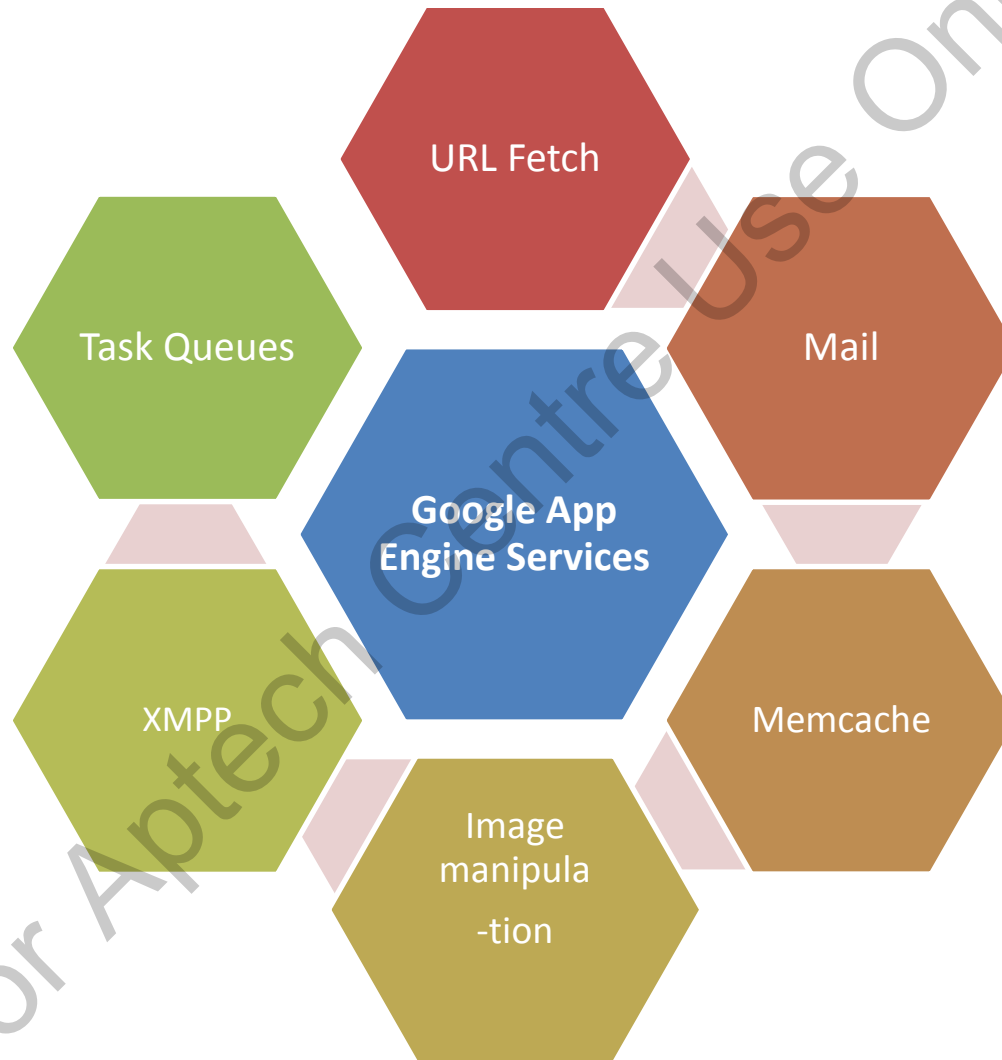
Google App Engine makes it easy to build Web applications that are robust and reliable, even under heavy load and with large amounts of data.

Google App Engine supports common Web technologies that can be used for building dynamic Web applications.

Google App Engine provides APIs for authenticating users and sending e-mail using Google accounts.

Google App Engine provides a development environment that facilitates the development and testing of Web applications on local machines. Google App Engine provides a secure environment referred to as Sandbox.

Features of Google App Engine 2-2



Google App Engine Quota

Google App Engine is a PaaS service that offers App Engines APIs as well as hosting services for the applications.

The users do not have to maintain any physical servers at their location, instead Google App Engine offers 1 GB of free data storage for the users.

Google App Engine offers three kinds of quotas that are as follows:

Free Quota

- It is set to provide a certain amount of free limit for deploying applications on the Google App Engine.

Billable Limits

- They are set by administrator and is applied for paid applications. Once set, the applications cannot exceed the limit.

Safety Limits

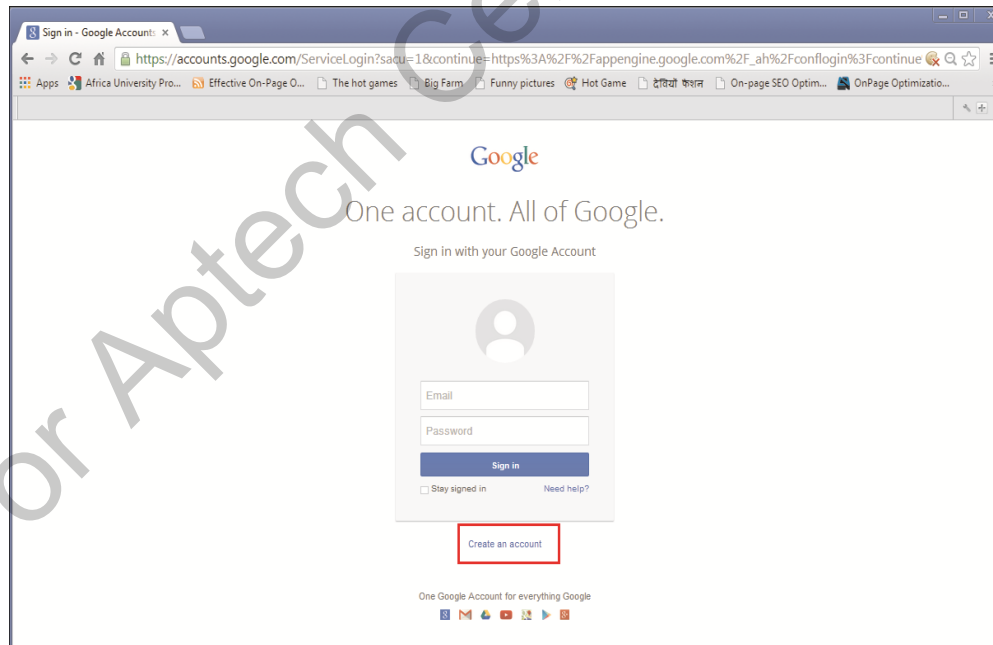
- They ensure that a single application cannot consume resources alone, leaving other applications starving for resources.

Signing Up for Google App Engine 1-6

To create an application and deploy it on Google App Engine, you need a Google App Engine account.

The steps to create a new Google account are as follows:

1. Type the address **`http://appengine.google.com`** to open the Google account page.
2. Click **Create an account** link to create a new Google account. Figure shows the Google account page.



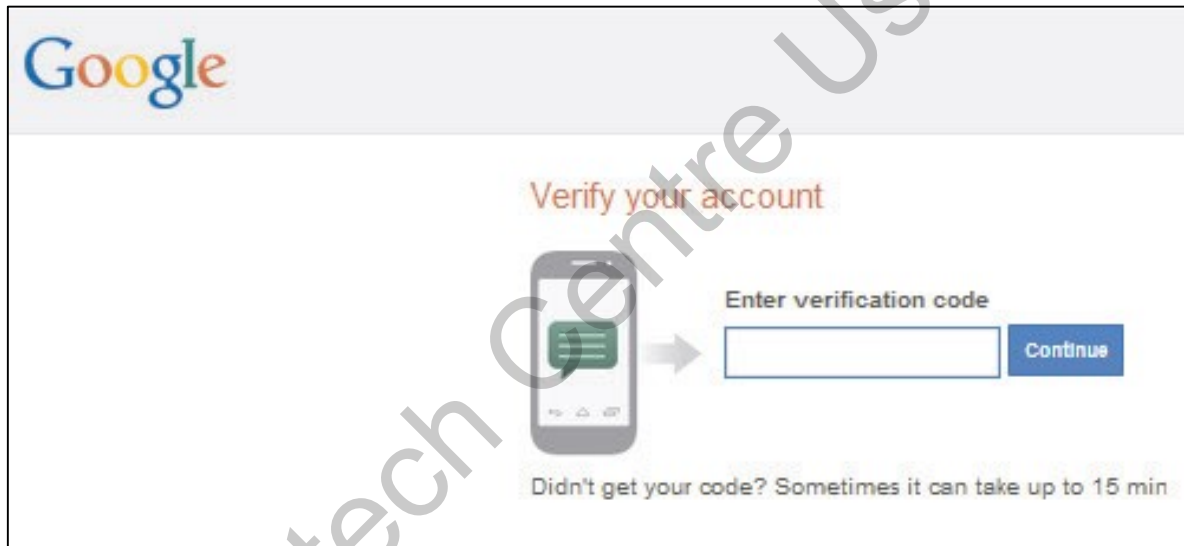
Signing Up for Google App Engine 2-6

After the link is clicked, figure shows the registration page to sign up a new account.

The screenshot shows the Google Accounts registration page in a web browser. The browser's address bar displays the URL: `https://accounts.google.com/SignUp?service=ah&continue=https%3A%2F%2Fappengine.google.com%2F_ah%2Fconflogin%3Fcontinue%`. The page features the Google logo at the top left and a 'Sign in' button at the top right. The main heading is 'Create your Google Account', followed by the text 'One account is all you need' and 'A single username and password gets you into everything Google.' Below this, there are icons for various Google services (Gmail, YouTube, etc.) and the text 'Make Google yours' and 'Set up your profile and preferences just the way you like.' A section titled 'Take it all with you' includes the text 'Switch between devices, and pick up wherever you left off.' and shows three profile pictures of people. On the right side, there is a registration form with the following fields: 'Name' (with sub-fields for 'John' and 'Bosco'), 'Choose your username' (with the value 'John.Bosco.Developer@gmail.com' and a note 'I prefer to use my current email address'), 'Create a password' and 'Confirm your password' (both masked with dots), 'Birthday' (with a dropdown for 'July', a date picker for '06', and a year field for '1980'), 'Gender' (with a dropdown set to 'Male'), 'Mobile phone' (with a country code dropdown set to '+1684'), 'Your current email address' (empty), and 'Prove you're not a robot'.

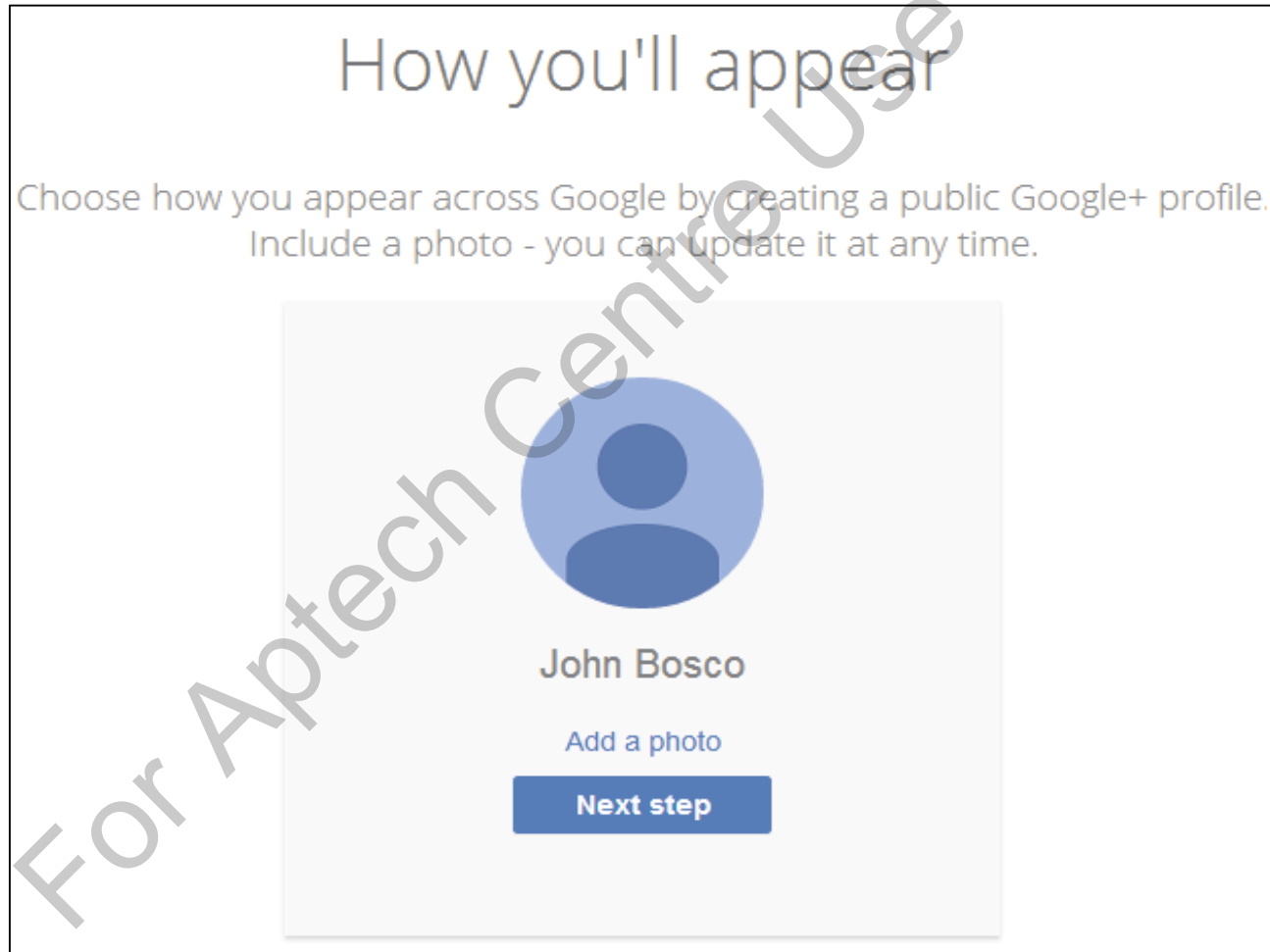
Signing Up for Google App Engine 3-6

3. Specify the details, accept the terms and conditions, and click **Next Step**.
4. Enter the verification code and click **Continue**.



Signing Up for Google App Engine 4-6

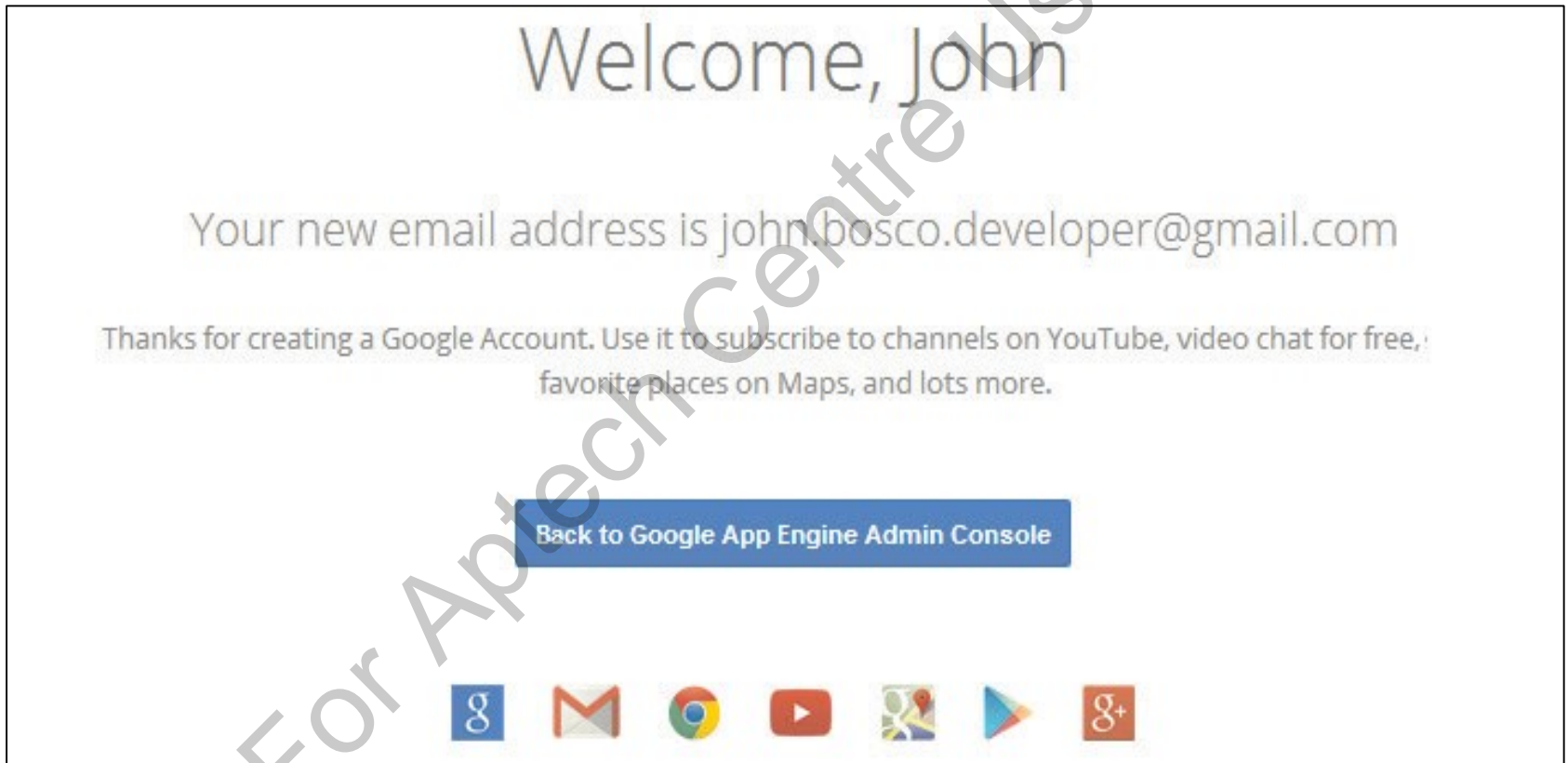
The next page informs users how the user name will appear on login page. Figure shows the user profile.



Signing Up for Google App Engine 5-6

5. Click **Next step** to finish creating the Google account.

Figure displays the Google account creation page with e-mail address.



Signing Up for Google App Engine 6-6

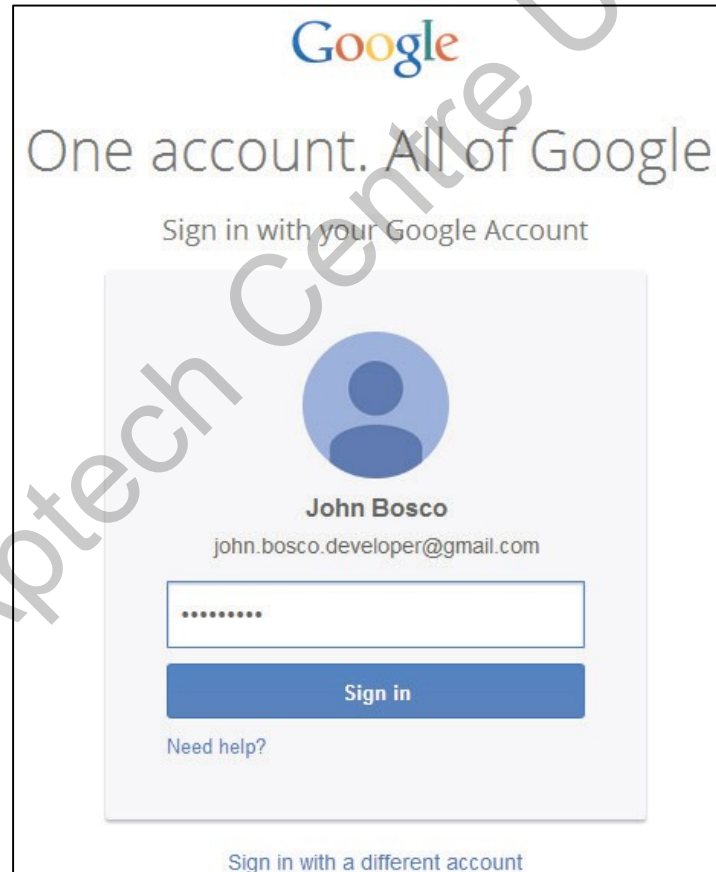
6. Click **Back to Google App Engine Admin Console** to display the Google App Engine start page as shown in figure.



Signing Up for Google App Engine-Creating and Registering an Application 1-6

The steps to create and register the first application are as follows:

1. Type **`http://appengine.google.com`** to login with your Google account credentials. Figure displays the Sign in page for Google account.



The image shows a Google sign-in page. At the top is the Google logo. Below it is the text "One account. All of Google." followed by "Sign in with your Google Account". In the center is a light gray box containing a blue circular profile picture placeholder, the name "John Bosco", and the email address "john.bosco.developer@gmail.com". Below the email is a password field with dots. A blue "Sign in" button is at the bottom of the box. Below the button is a link "Need help?". At the very bottom of the page is a link "Sign in with a different account".

Signing Up for Google App Engine-Creating and Registering an Application 2-6

2. Specify the login credentials and click **Sign in**.

This will display the Google App Engine start page as shown in figure.



Signing Up for Google App Engine-Creating and Registering an Application 3-6

3. Click **Create Application**. This will display **Create an Application** Web page as shown in figure.

You have 10 applications remaining.

Application Identifier:
 .appspot.com

All Google account names and certain offensive or trademarked names may not be used as Application Identifiers.
You can map this application to your own domain later. [Learn more](#)

Application Title:

Displayed when users access your application.

Authentication Options (Advanced): [Learn more](#)
Google App Engine provides an API for authenticating your users, including Google Accounts, Google Apps, and OpenID. If you choose to use this feature for some parts of your site, you'll need to specify now what type of users can sign in to your application:

☒ **Open to all Google Accounts users (default)**
If your application uses authentication, anyone with a valid Google Account may sign in.

☐ **Restricted to the following [Google Apps](#) domain:**

e.g. foo.com
If your application uses authentication, only members of this Google Apps domain may sign in. If your organization uses Google Apps, use this option to create an application (e.g. an HR tracking tool) that is only accessible to accounts on your Google Apps domain. This option cannot be changed once it has been set.

☐ **(Experimental) Open to all users with an OpenID Provider**
If your application uses authentication, anyone who has an account with an OpenID Provider may sign in.



Signing Up for Google App Engine-Creating and Registering an Application 4-6

4. Type your-app-id in the **Application Identifier** box. The application ID is a unique name given to the application on the domain **appspot.com**. Here, **appspot.com** is the free domain on which your application is identified with unique identifier. The full URL of the application will be

`http://your-app-id.appspot.com.`
5. Click **Check Availability** to check if the application ID is valid.
6. Type your-app-title in the **Application Title** box. This serves as a title for the application.

Signing Up for Google App Engine-Creating and Registering an Application 5-6

7. Click **I accept these terms** check box to accept the terms and conditions as shown in figure.

Google app engine John.Bosco.Developer@gmail.com | [My Account](#) | [Help](#) | [Sign out](#)

Create an Application

You have 10 applications remaining.

Application Identifier:
heloapp123 .appspot.com [Check Availability](#) Yes, "heloapp123" is available!
All Google account names and certain offensive or trademarked names may not be used as Application Identifiers.
You can map this application to your own domain later. [Learn more](#)

Application Title:
Python Application
Displayed when users access your application.

Authentication Options (Advanced): [Learn more](#)
Google App Engine provides an API for authenticating your users, including Google Accounts, Google Apps, and OpenID. If you choose to use this feature for some parts of your site, you'll need to specify now what type of users can sign in to your application:

- ☒ **Open to all Google Accounts users (default)**
If your application uses authentication, anyone with a valid Google Account may sign in.
- ☐ **Restricted to the following [Google Apps](#) domain:**

e.g. foo.com
If your application uses authentication, only members of this Google Apps domain may sign in. If your organization uses Google Apps, use this option to create an application (e.g. an HR tracking tool) that is only accessible to accounts on your Google Apps domain. This option cannot be changed once it has been set.
- ☐ **(Experimental) Open to all users with an OpenID Provider**
If your application uses authentication, anyone who has an account with an OpenID Provider may sign in.

Terms of Service:
Google processes and stores its own information of a similar type. Google has implemented at least industry standard systems and procedures to ensure the security and confidentiality of an Application and Customer Data, protect against anticipated threats or hazards to the security or integrity of an Application and Customer Data, and protect against unauthorized access to or use of an Application and Customer Data. Google may process and store an Application and Customer Data in the United States or any other country in which Google or its agents maintain facilities. By using the Services, Customer consents to this processing and storage of an Application and Customer Data. The parties agree that Google is merely a data processor.

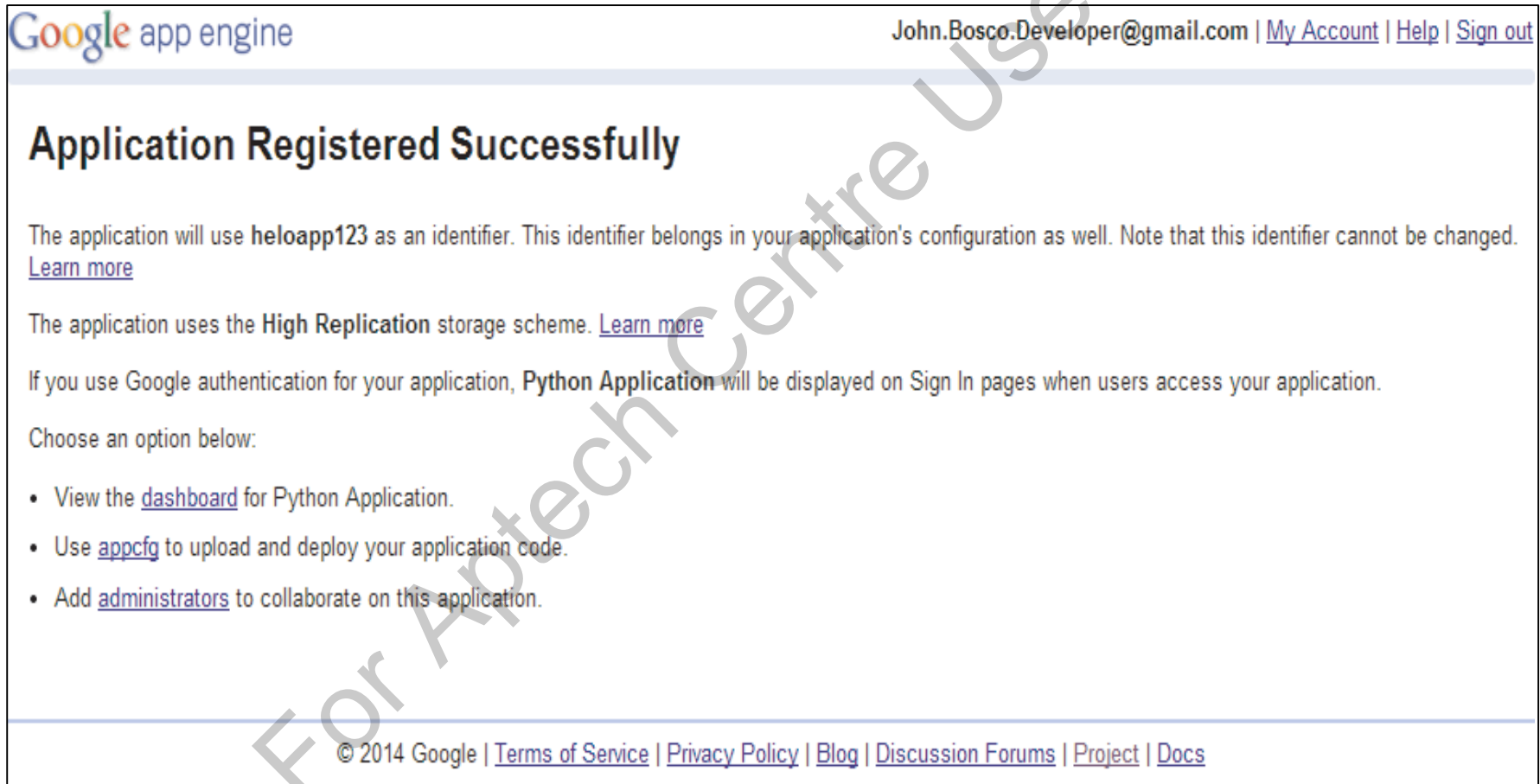
☒ I accept these terms.

[Create Application](#) [Cancel](#)

© 2014 Google | [Terms of Service](#) | [Privacy Policy](#) | [Blog](#) | [Discussion Forums](#) | [Project](#) | [Docs](#)

Signing Up for Google App Engine-Creating and Registering an Application 6-6

8. Click **Create Application** to create a new application. The **Application Registered Successfully** Web page is displayed as shown in figure.



The screenshot shows the Google App Engine 'Application Registered Successfully' page. At the top left is the 'Google app engine' logo. At the top right is the user's email 'John.Bosco.Developer@gmail.com' followed by links for 'My Account', 'Help', and 'Sign out'. The main heading is 'Application Registered Successfully'. Below this, a paragraph states: 'The application will use **heloapp123** as an identifier. This identifier belongs in your application's configuration as well. Note that this identifier cannot be changed.' followed by a 'Learn more' link. Another paragraph states: 'The application uses the **High Replication** storage scheme.' followed by a 'Learn more' link. A third paragraph states: 'If you use Google authentication for your application, **Python Application** will be displayed on Sign In pages when users access your application.' Below this, it says 'Choose an option below:' followed by a bulleted list: 'View the [dashboard](#) for Python Application.', 'Use [appcfg](#) to upload and deploy your application code.', and 'Add [administrators](#) to collaborate on this application.' At the bottom, there is a footer with '© 2014 Google' and links for 'Terms of Service', 'Privacy Policy', 'Blog', 'Discussion Forums', 'Project', and 'Docs'.

Google app engine John.Bosco.Developer@gmail.com | [My Account](#) | [Help](#) | [Sign out](#)

Application Registered Successfully

The application will use **heloapp123** as an identifier. This identifier belongs in your application's configuration as well. Note that this identifier cannot be changed. [Learn more](#)

The application uses the **High Replication** storage scheme. [Learn more](#)

If you use Google authentication for your application, **Python Application** will be displayed on Sign In pages when users access your application.

Choose an option below:

- View the [dashboard](#) for Python Application.
- Use [appcfg](#) to upload and deploy your application code.
- Add [administrators](#) to collaborate on this application.

© 2014 Google | [Terms of Service](#) | [Privacy Policy](#) | [Blog](#) | [Discussion Forums](#) | [Project](#) | [Docs](#)

Google App Engine Development Environments

Google App Engine platform supports different programming languages such as Java, Python, PHP, and Go for application development.

Each language provides a Software Development Kit (SDK) and a runtime environment that includes:

Application Programming Interfaces (APIs) classes to build an application.

Deployment tools to upload the developed application to the Google cloud environment.

Simulated Google App Engine sandbox environment that provides a full featured development environment on your local system.



Google App Engine Development Environments- Developing Web Application with Python 1-5

Python is a high-level object-oriented programming language. Its syntax helps the programmers to develop complex code easily as compared to other languages such as C or Java.

A pre-loaded Python interpreter on the Google App Engine is used to execute the application in a secured sandbox environment.

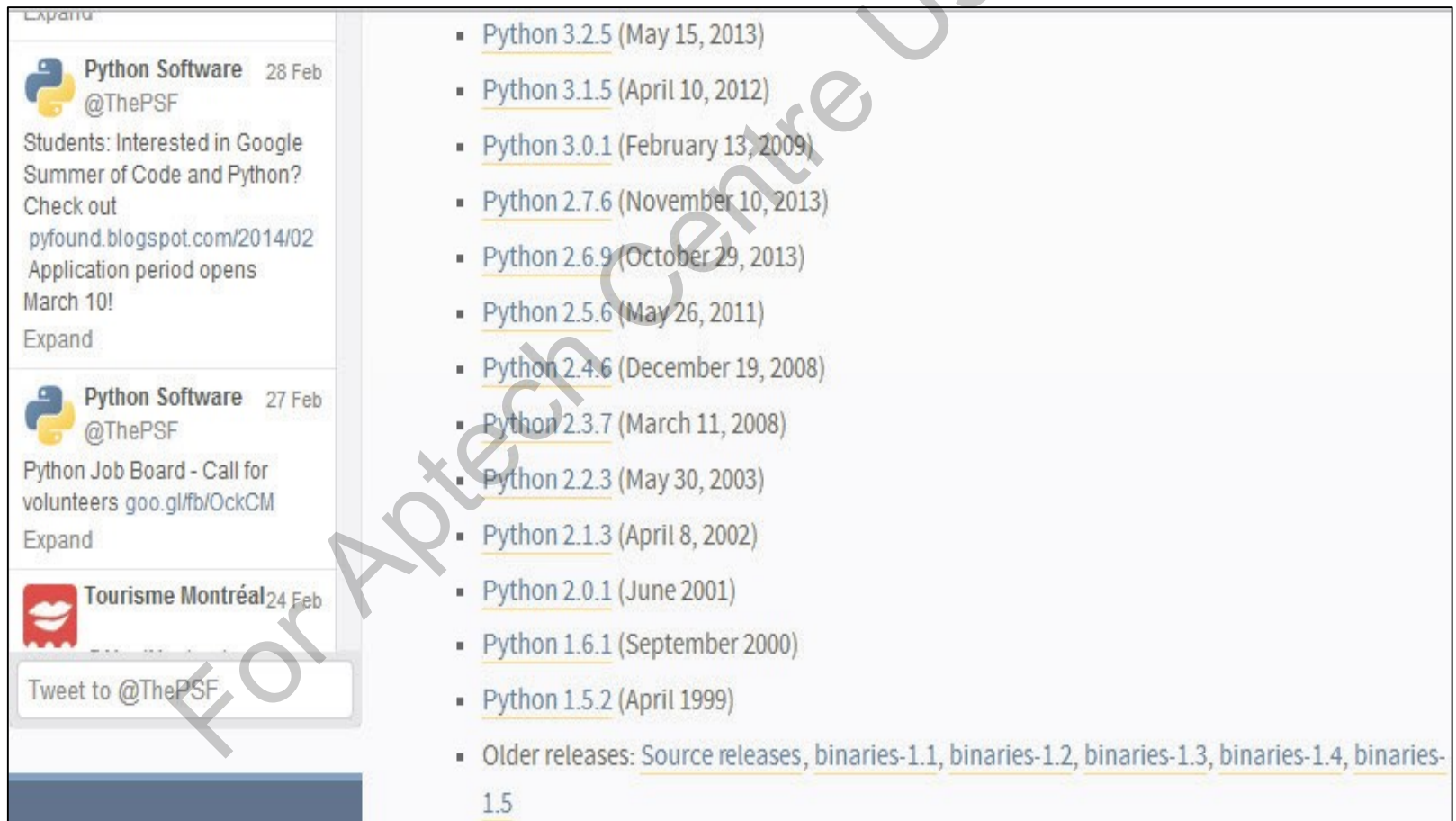
The application interacts with the environment and performs activities such as receiving Web requests, sending responses, and so on.

The Python SDK includes a Google App Engine Launcher that runs on the local machine and provides a graphical interface that simplifies the application development.

Google App Engine Development Environments- Developing Web Application with Python 2-5

The steps to create an application on Google App Engine using Python are as follows:

1. Download and install the latest version of Python from the Web site, <http://www.python.org/download/releases/> as shown in figure.



The screenshot displays the Python.org website's download releases page. On the left, there is a sidebar with social media links and a tweet from Python Software (@ThePSF) dated 28 Feb. The main content area lists various Python versions and their release dates:

- Python 3.2.5 (May 15, 2013)
- Python 3.1.5 (April 10, 2012)
- Python 3.0.1 (February 13, 2009)
- Python 2.7.6 (November 10, 2013)
- Python 2.6.9 (October 29, 2013)
- Python 2.5.6 (May 26, 2011)
- Python 2.4.6 (December 19, 2008)
- Python 2.3.7 (March 11, 2008)
- Python 2.2.3 (May 30, 2003)
- Python 2.1.3 (April 8, 2002)
- Python 2.0.1 (June 2001)
- Python 1.6.1 (September 2000)
- Python 1.5.2 (April 1999)
- Older releases: [Source releases](#), [binaries-1.1](#), [binaries-1.2](#), [binaries-1.3](#), [binaries-1.4](#), [binaries-1.5](#)

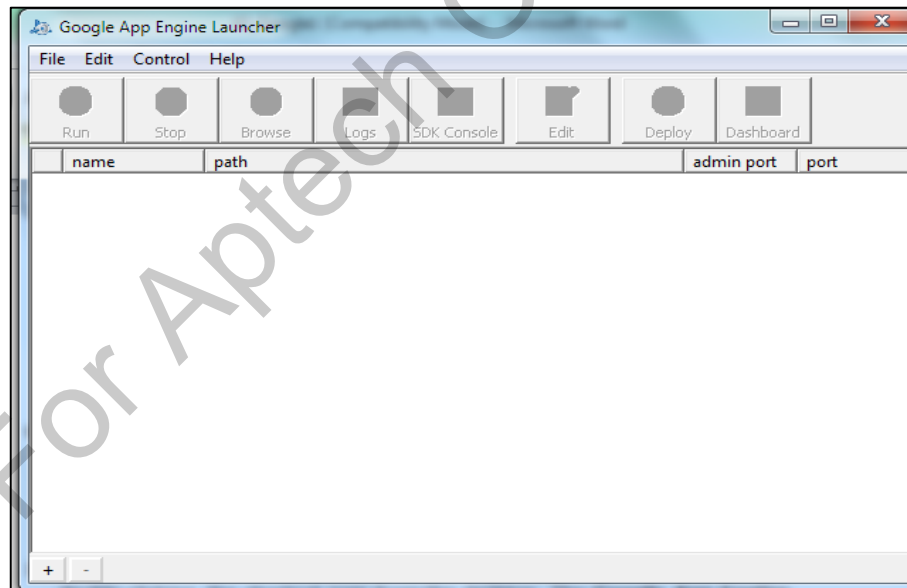
Google App Engine Development Environments- Developing Web Application with Python 3-5

2. Download Google App Engine launcher from

https://developers.google.com/appengine/downloads#Google_App_Engine_SDK_for_Python for the respective platform. For example, **GoogleAppEngine-1.9.0.msi**.

3. Install the **GoogleAppEngine-1.9.0.msi** on your local system.

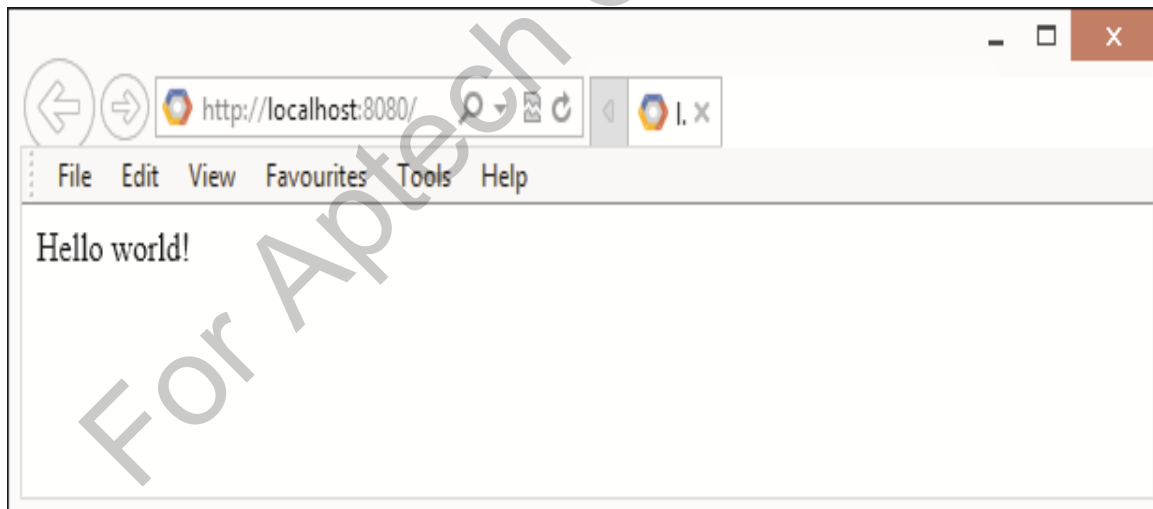
4. Launch the **Google App Engine Launcher** as shown in figure.



Google App Engine Development Environments- Developing Web Application with Python 4-5

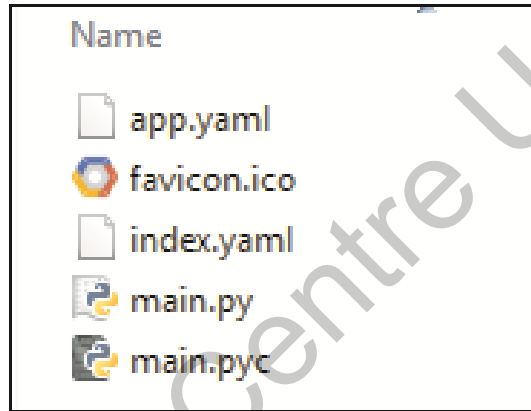
5. Click **File** → **Create New Application**.
6. Type **robbapp1234** in the **Application Name**.
7. Click **Create** to create the Web application.
8. Click **Run** on the Toolbar to start the application. Then, click **Browse** to open the Web browser and view the generated output of the application.

Following figure shows the output of the Python application.



Google App Engine Development Environments- Developing Web Application with Python 5-5

9. Click **Edit** → **Open in Explorer** to view the structure of the application created in the directory specified during application creation as shown in figure.



Following figure shows the contents of the `app.yaml` file.

```
application: robbapp1234
version: 1
runtime: python27
api_version: 1
threadsafe: yes


handlers:
- url: /favicon\.ico
  static_files: favicon.ico
  upload: favicon\.ico


- url: .*
  script: main.app


libraries:
- name: webapp2
  version: "2.5.2"
```






Google App Engine Development Environments- Developing Web Applications with Java

- 
- The Web applications can be built using standard Java technologies, such as Servlet and JSP for uploading on Google App Engine.

- 
- The Google App Engine provide Java SDK, which contains JVM and a Web server for testing Java applications.

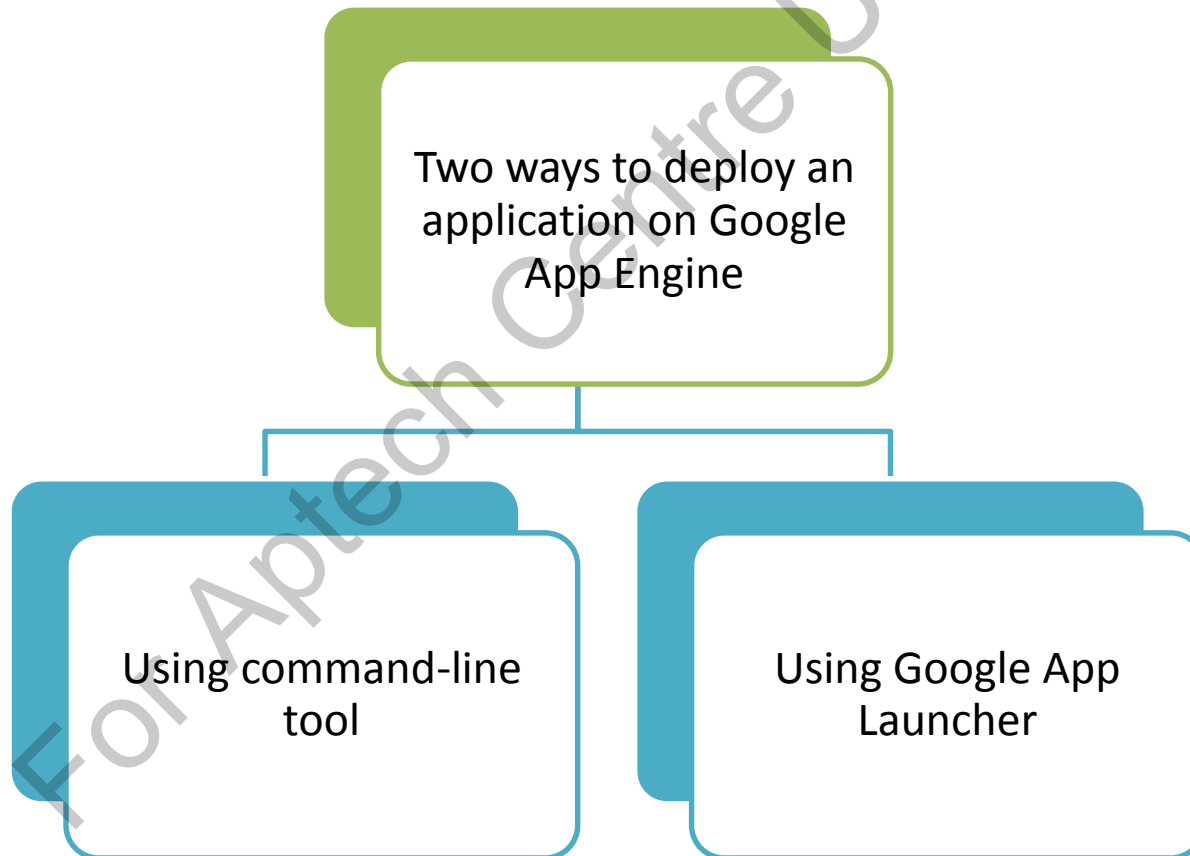
- 
- Java Virtual Machine (JVM) is used to run the Java applications in a secure sandbox environment similar to the Python.

- 
- The JVM is capable of executing any Java bytecode that operates within the restrictions of the sandbox.

- 
- The Eclipse Integrated Development Environment (IDE) can be used to develop, execute, and upload Java applications on Google App Engine.

Deploying an Application to Google App Engine 1-3

The Web application created on local system can be deployed on Google App Engine. The Google App Engine provides Administration Console, which manages the application running on its server.



Deploying an Application to Google App Engine 2-3

- ❑ Before an application is deployed on Google App Engine, you need to register the application with a unique application ID. Once the application is registered, open the `app.yaml` configuration file and change the application: setting from `your-app-id` to the registered application ID.
- ❑ For example, the Web application named **robbapp1234** created in Python can be changed to **heloapp123** application ID registered on the Google App Engine.

- **Using command-line**

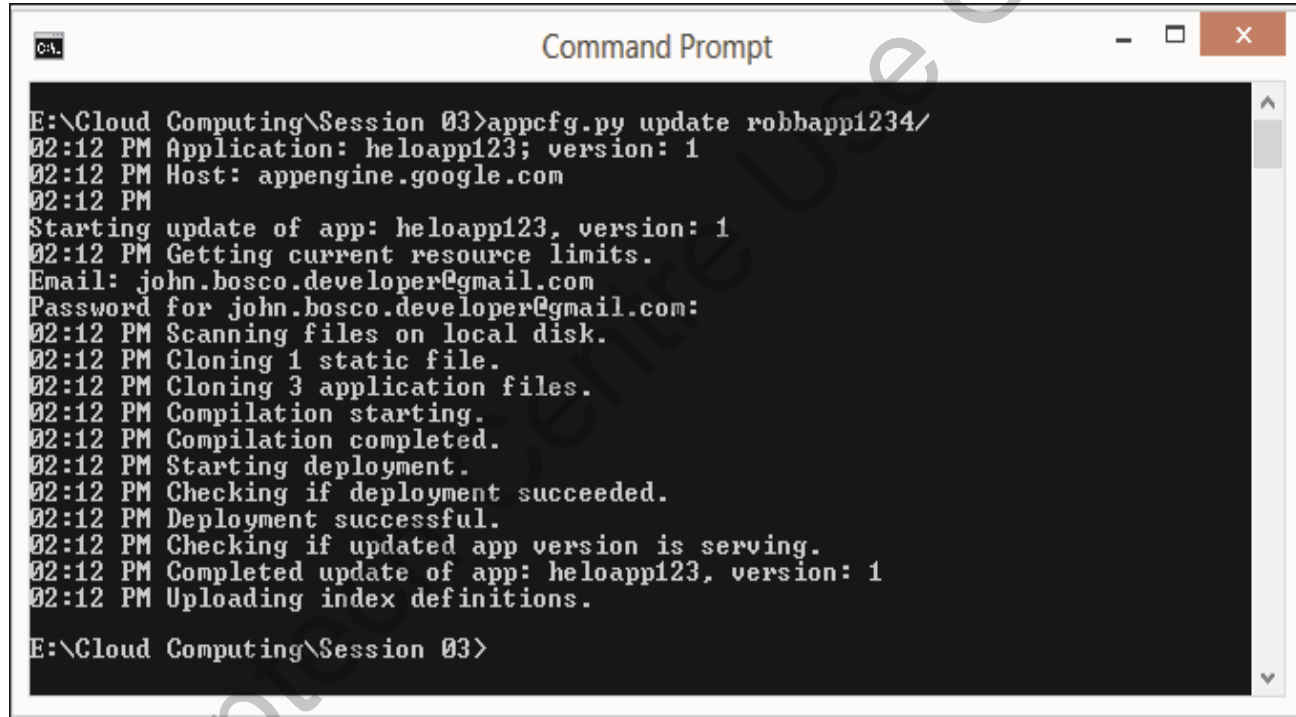
The `appcfg.py` command is used to upload the ready application on Google App Engine. In order to deploy an application, type the following command in the command prompt:

```
your-sdk-install-dir-path/appcfg.py update your-app-dir
```

For example, `appcfg.py update robbapp123/`. The `appcfg.py` refers to `app.yaml` file to get the application ID registered on the Google App Engine. It prompts the user for Google account credentials and after successful signing in, deploys the application on Google App Engine. The **robbapp123** is the root directory, which contains the `app.yaml` file.

Deploying an Application to Google App Engine 3-3

Figure shows the command prompt to deploy **heloapp123** application using **appcfg.py** command.



```
E:\Cloud Computing\Session 03>appcfg.py update robbapp1234/
02:12 PM Application: heloapp123; version: 1
02:12 PM Host: appengine.google.com
02:12 PM
Starting update of app: heloapp123, version: 1
02:12 PM Getting current resource limits.
Email: john.bosco.developer@gmail.com
Password for john.bosco.developer@gmail.com:
02:12 PM Scanning files on local disk.
02:12 PM Cloning 1 static file.
02:12 PM Cloning 3 application files.
02:12 PM Compilation starting.
02:12 PM Compilation completed.
02:12 PM Starting deployment.
02:12 PM Checking if deployment succeeded.
02:12 PM Deployment successful.
02:12 PM Checking if updated app version is serving.
02:12 PM Completed update of app: heloapp123, version: 1
02:12 PM Uploading index definitions.

E:\Cloud Computing\Session 03>
```

- **Using Google App Engine Launcher**

Click **Deploy** in the Google App Engine Launcher to deploy the application. It will prompt for Google account username and password to sign in, before application is deployed to Google App Engine.



Google App Engine Storage

Storing data for a Web application is very important because a user interacts with multiple Web servers, which in turn, interact with data storage accessible across multiple machines.

Google App Engine provides an effective solution to this.

The infrastructure of Google App Engine takes care of the distribution, replication, and balancing the data load.



Google App Engine Storage-App Engine Datastore 1-2

App Engine Datastore is a schemaless object datastore that provides scalable storage for the Web applications. Some of the features of App Engine Datastore are as follows:

It is highly reliable with less downtime.

It has transaction properties termed as Atomicity Consistency Integrity Durability (ACID).

It provides high availability for reads and writes.

It provides strong consistency for reads and ancestor queries are maintained.

It provides automatic caching of queries.

Google App Engine Storage-App Engine Datastore 2-2

Some of the terms related to objects stored by App Engine Datastore are as follows:

Entities

- Data objects in the Datastore are known as entities.

Properties

- Each entity store one or more values, which are referred to as properties. Each entity is defined by the properties.

Key

- Each entity in the datastore is identified by a unique identification number, which is referred as a key.

Google App Engine Storage-Google Cloud SQL 1-2

Google Cloud SQL is a service that facilitates creation, configuration, and use of relational databases present in Google's cloud.

It provides MySQL relational database, which is a fully-managed service available on Google's cloud for your Web applications written in Java, Python, PHP, and Go.

The Google Cloud SQL comes with MySQL client and administrative tools such as reporting tool to work with MySQL database.

It also facilitates easy movement of data, applications, and services in and out of the cloud.

With Google Cloud SQL, users can choose a billing option that fits their usage pattern. The **pay per use** billing option facilitates payment only for the time the user has accessed it.

Google App Engine Storage-Google Cloud SQL 2-2

The features of Google Cloud SQL are as follows:

Easy to Use

- It has a graphical user interface, which allows the user to create, configure, manage, and monitor the database instances.

Fully Managed

- It has in-built services for patch management, replication, or backups.

Flexible Configuration

- The instances can be changed with a few seconds of downtime.

Exceptional Security

- Security is a design component of each of Google's cloud computing elements.

Google App Engine Services

Google App Engine provides several services through APIs that are described as follows:

Memcache Service

- The memcache service enables your applications to easily access temporary and frequently used data by using an in-memory cache.

Image Service

- The image service enables applications to manipulate images easily. Developers can resize, crop, and modify images in several other ways using the image service APIs.

URL Fetch Service

- The URL fetch service enables applications to access resources on the Internet, such as Web services or other data.

Mail Service

- The mail service allows applications to send e-mail using the Google infrastructure.



Summary

- ❑ Google provides a distributed and scalable cloud computing environment called Google App Engine for developing applications and hosting them at Google data centers.
- ❑ The Google App Engine supports various programming languages to develop Web applications for Google App Engine. The primary languages supported are Python and Java.
- ❑ Google App Engine provides a secure environment referred to as Sandbox.
- ❑ Google App Engine offers three kinds of quotas that are namely, Free Quota, Billable Limits, and Safety Limits.
- ❑ A user can sign in through Google account to create and register your Web applications on Google App Engine.
- ❑ Datastore and Google Cloud SQL are few data storage services provided by Google App Engine.
- ❑ Memcache, Image, URL Fetch, and Mail are few API services supported by Google App Engine.