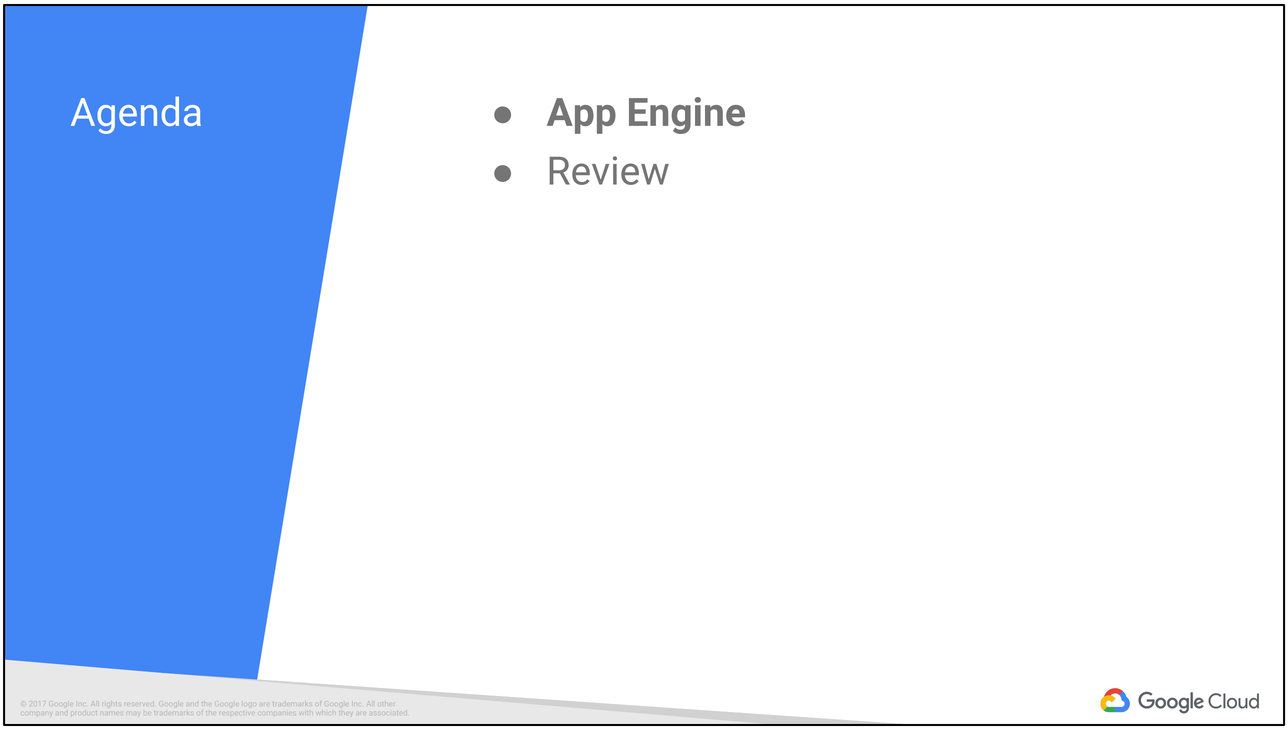
Application Development Services

Architecting with GCP Fundamentals: Infrastructure

APP ENGINE

Google Cloud

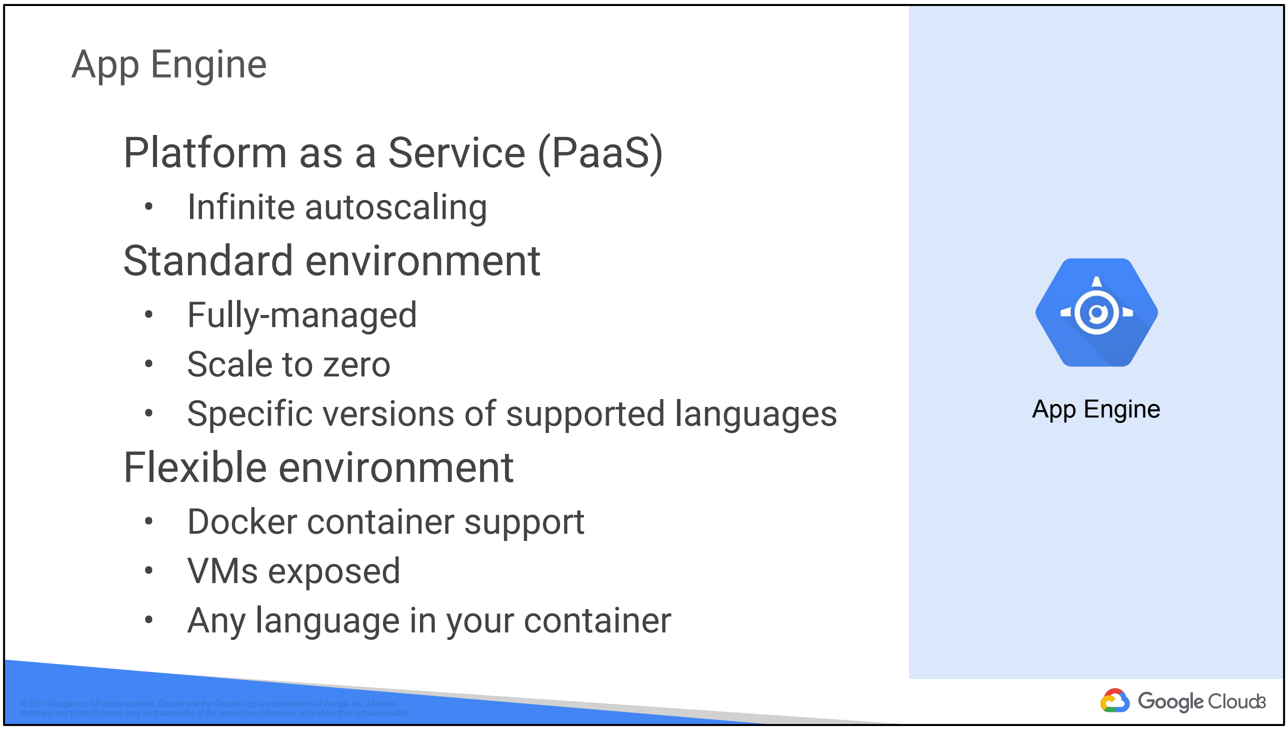
Last modified 2017-11-27



Agenda

- **App Engine**
- Review

Google Cloud



App Engine

Platform as a Service (PaaS)

- Infinite autoscaling

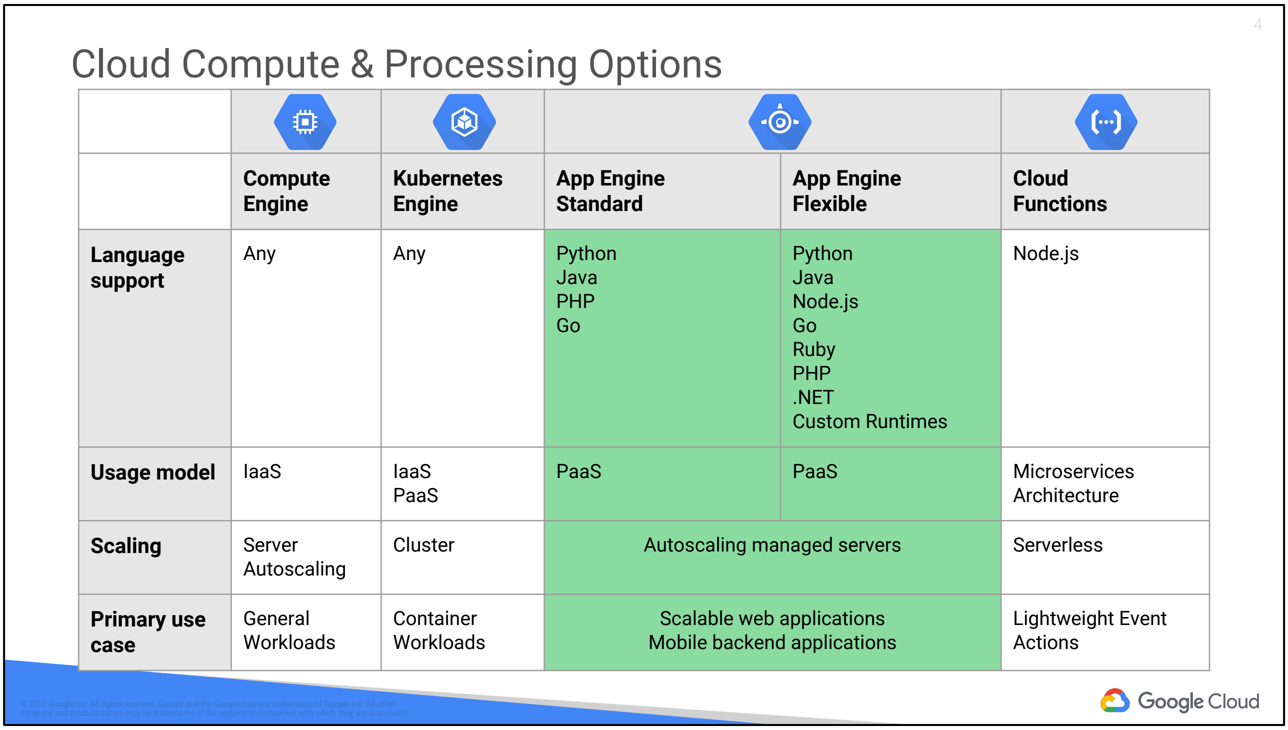
Standard environment

- Fully-managed
- Scale to zero
- Specific versions of supported languages

Flexible environment

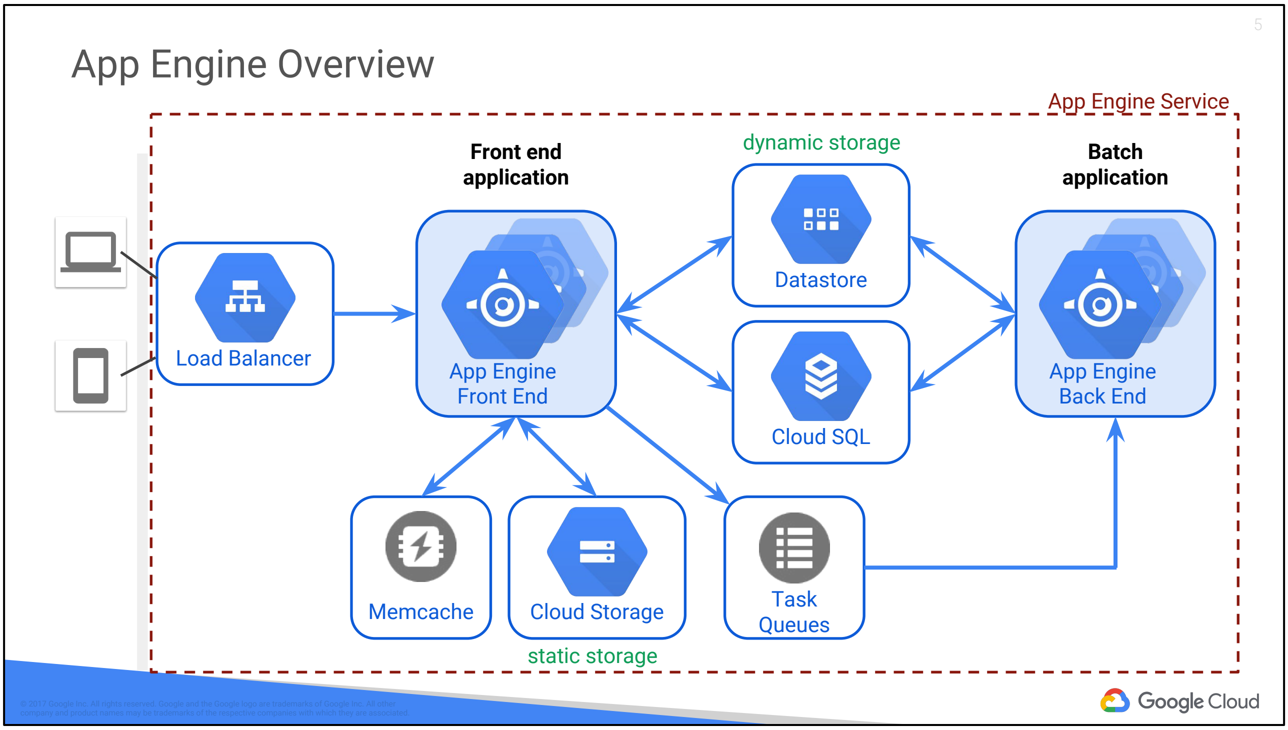
- Docker container support
- VMs exposed
- Any language in your container

Development tool compatibility: Eclipse, IntelliJ, Maven, Git, Jenkins, and PyCharm



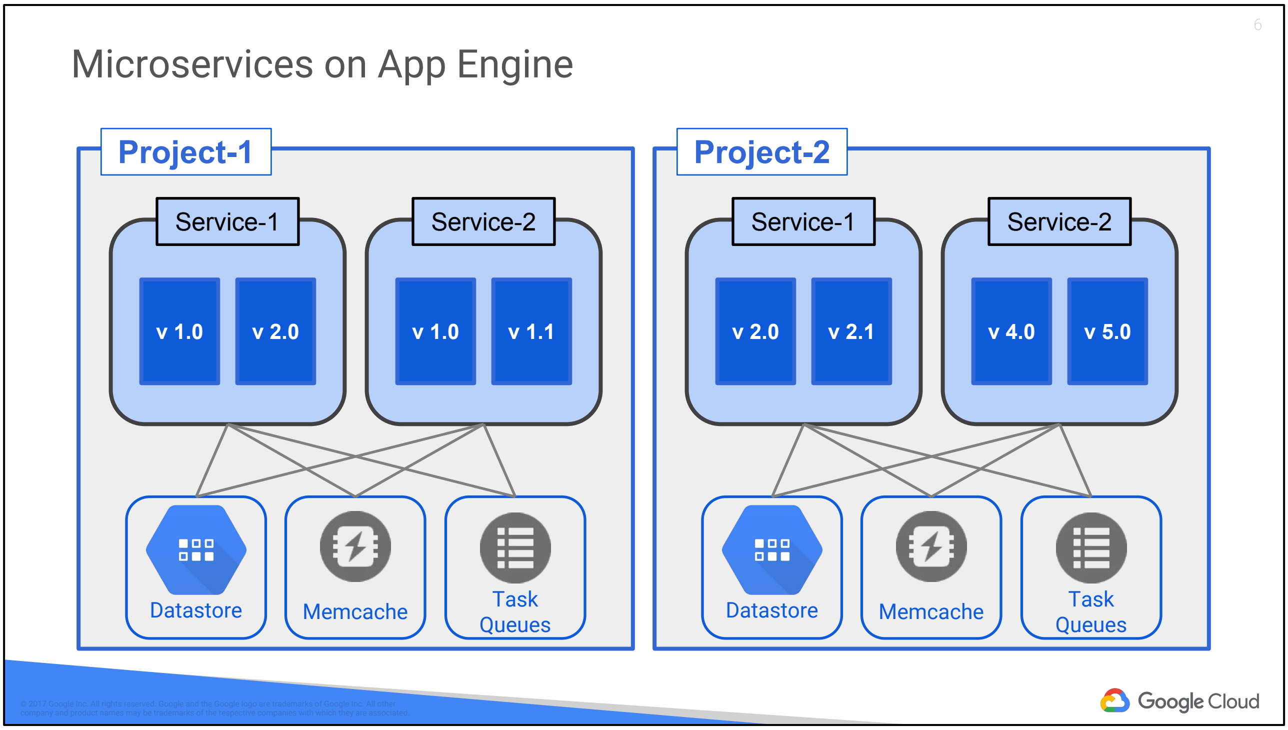
Cloud Compute & Processing Options

	Compute Engine	Kubernetes Engine	App Engine Standard	App Engine Flexible	Cloud Functions
Language support	Any	Any	Python Java PHP Go	Python Java Node.js Go Ruby PHP .NET Custom Runtimes	Node.js
Usage model	IaaS	IaaS PaaS	PaaS	PaaS	Microservices Architecture
Scaling	Server Autoscaling	Cluster	Autoscaling managed servers		Serverless
Primary use case	General Workloads	Container Workloads	Scalable web applications Mobile backend applications		Lightweight Event Actions



App Engine Overview

Front end is often critical to user experience (Load Balancer is built-in). Back end is for more intensive processing. This separation of function allows each part to scale as needed.
The App Engine service is a module. This example shows a single Service. More complex architectures are possible.



Microservices on App Engine

Services are a modular abstraction in App Engine; a way to break up the application into separate parts. Instead of architecting your entire application inside a single App Engine Service as shown on the previous slide, you can implement your application in multiple services. You can have more than one service in a project. You can also use multiple projects to further isolate services. Splitting traffic to different versions enables incremental rollouts and and A/B testing.

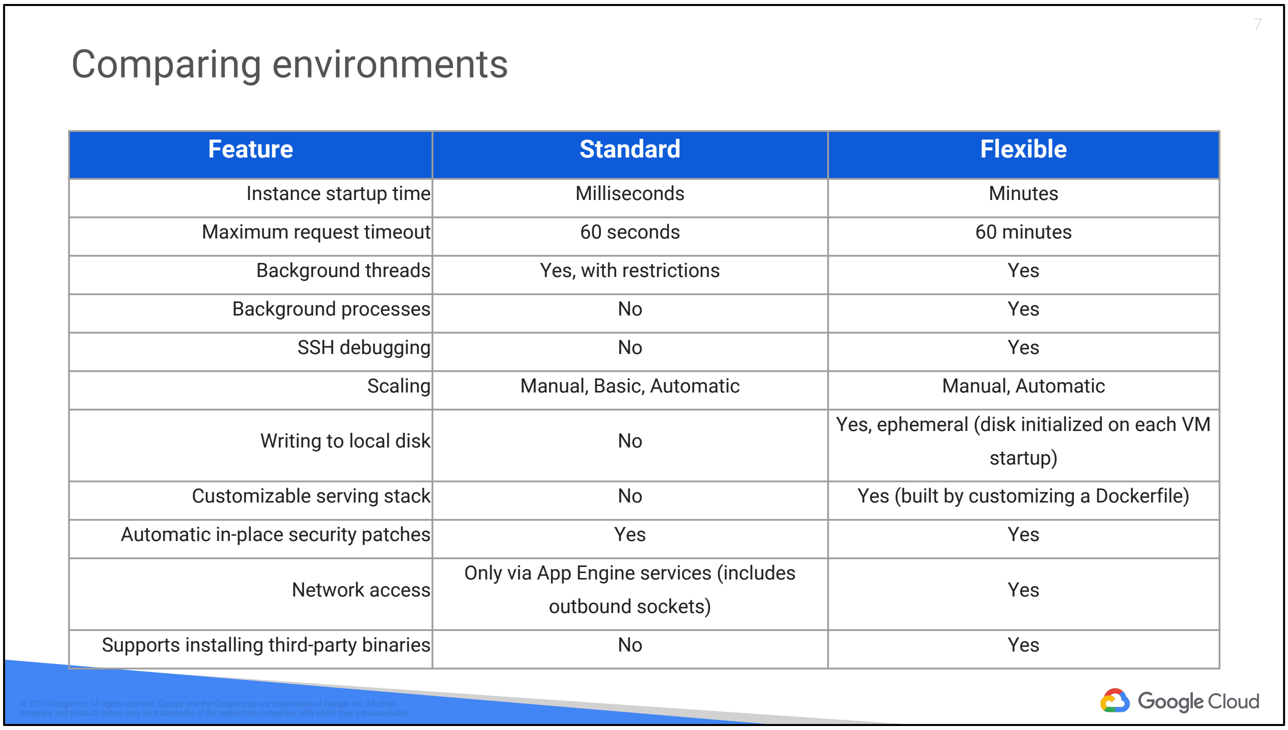
Each service can have multiple versions deployed simultaneously. Code is completely separate in versions, services, and projects.
Services in a single project share some resources, for example, Datastore, Memcache, and Task Queues.
Services in separate projects are completely isolated.
IAM -- you can assign different roles at the project, but not at the service-level.

When services are in the same project, they are isolated in some ways and share certain resources. Code in one service can't directly call code in another service. Code is deployed independently. Each service can be written in a different language. And the Autoscaling, load balancing, and machine instance type that we saw on the previous slide are independent for each service.

- Define strong contracts between the microservices
- Independent deployment cycles (and rollback if needed)
- Concurrent A/B release testing on subsystems

- Minimize test automation and quality-assurance overhead
- Logging and monitoring clarity
- More detailed cost accounting
- Increase application scalability and reliability

More information:
https://cloud.google.com/appengine/docs/python/microservices-on-app-engine



Comparing environments

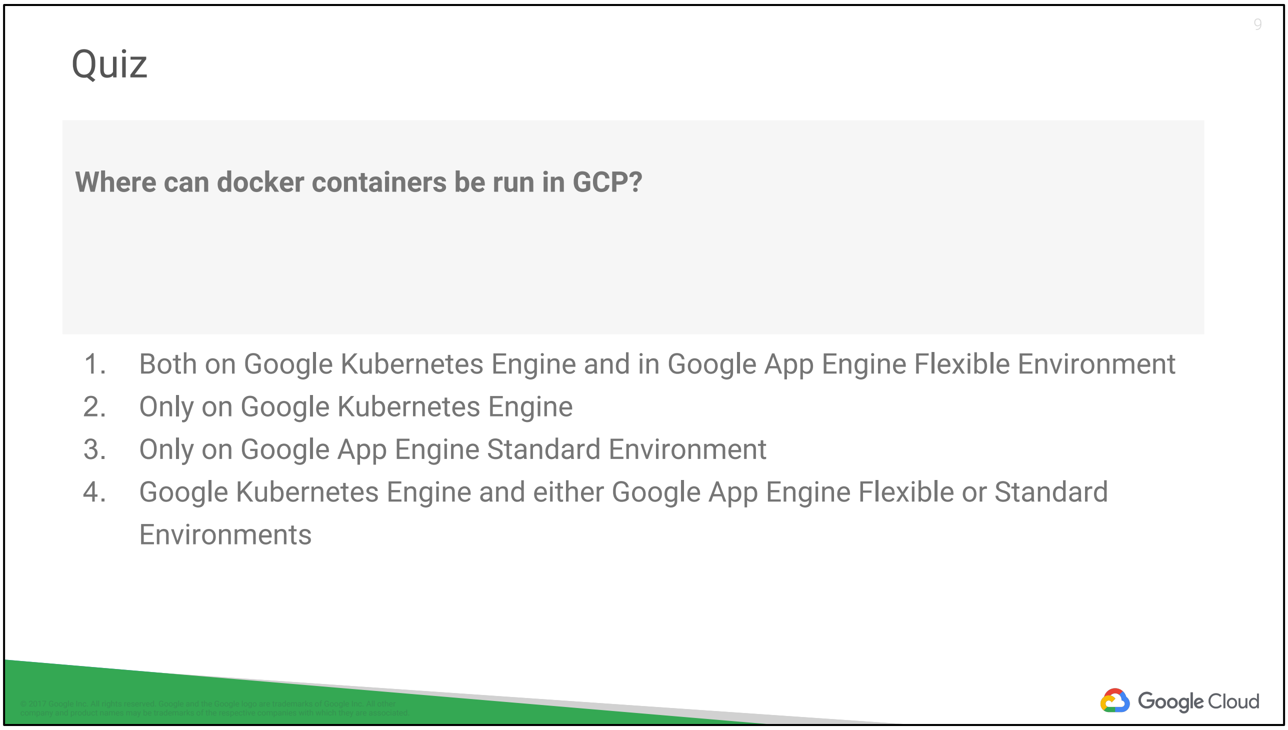
Feature	Standard	Flexible
Instance startup time	Milliseconds	Minutes
Maximum request timeout	60 seconds	60 minutes
Background threads	Yes, with restrictions	Yes
Background processes	No	Yes
SSH debugging	No	Yes
Scaling	Manual, Basic, Automatic	Manual, Automatic
Writing to local disk	No	Yes, ephemeral (disk initialized on each VM startup)
Customizable serving stack	No	Yes (built by customizing a Dockerfile)
Automatic in-place security patches	Yes	Yes
Network access	Only via App Engine services (includes outbound sockets)	Yes
Supports installing third-party binaries	No	Yes

Google Cloud

More information choosing an App Engine Environemnt:
https://cloud.google.com/appengine/docs/the-appengine-environments

Agenda

- App Engine
- **Quiz**

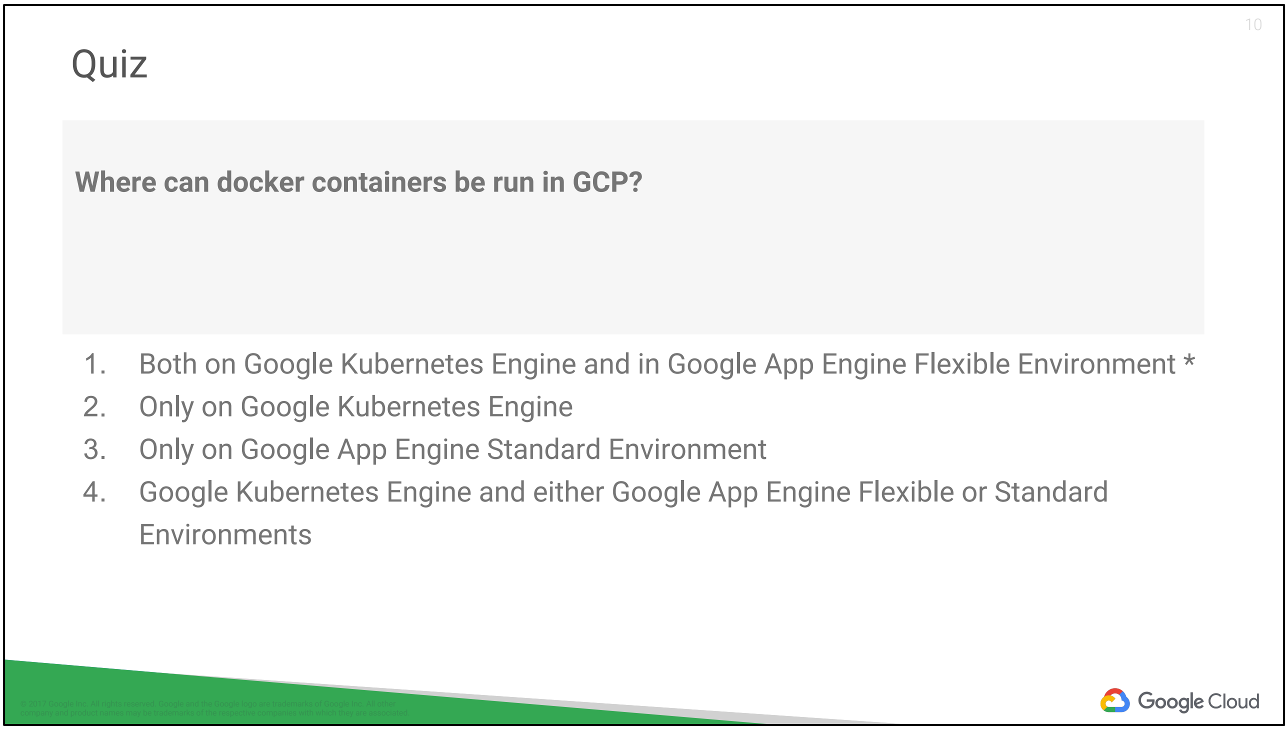


Quiz

Where can docker containers be run in GCP?

1. Both on Google Kubernetes Engine and in Google App Engine Flexible Environment
2. Only on Google Kubernetes Engine
3. Only on Google App Engine Standard Environment
4. Google Kubernetes Engine and either Google App Engine Flexible or Standard Environments

Google Cloud



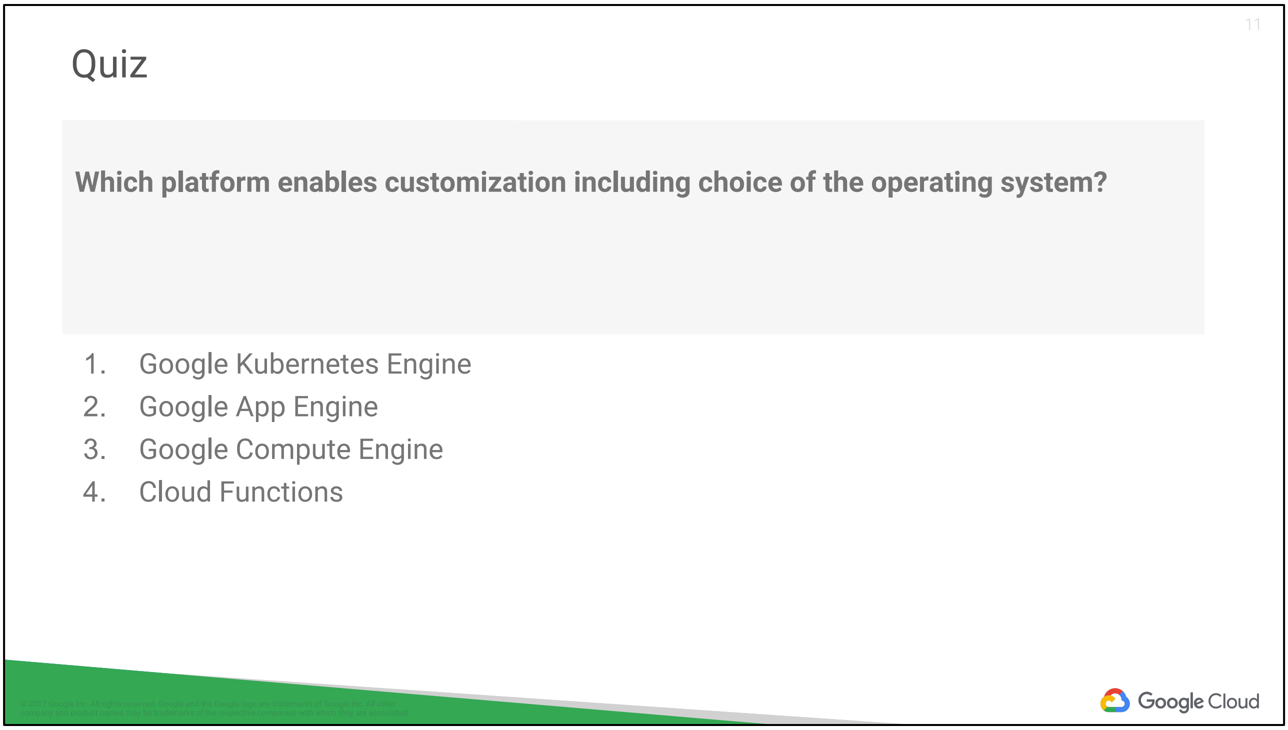
Quiz

Where can docker containers be run in GCP?

1. Both on Google Kubernetes Engine and in Google App Engine Flexible Environment *
2. Only on Google Kubernetes Engine
3. Only on Google App Engine Standard Environment
4. Google Kubernetes Engine and either Google App Engine Flexible or Standard Environments

Google Cloud

Explanation:
Google Kubernetes Engine and Google App Engine Flexible Environment only.

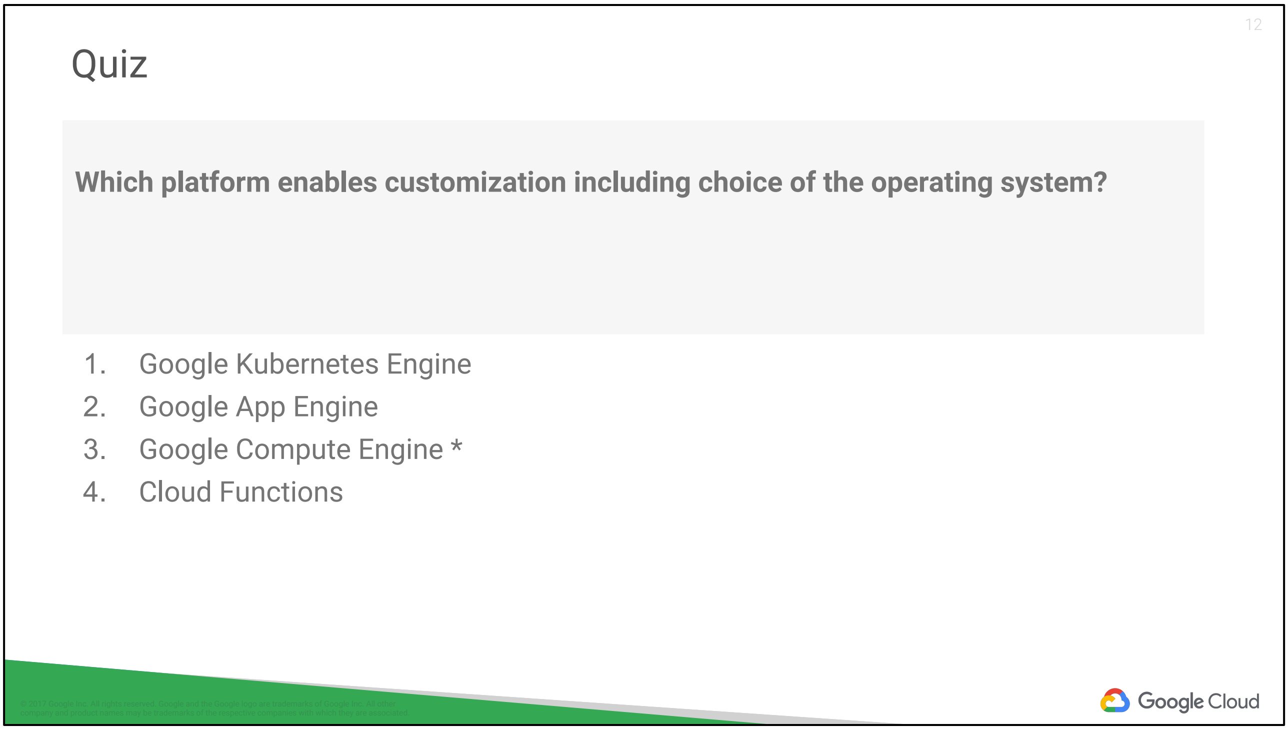


Quiz

Which platform enables customization including choice of the operating system?

1. Google Kubernetes Engine
2. Google App Engine
3. Google Compute Engine
4. Cloud Functions

Google Cloud

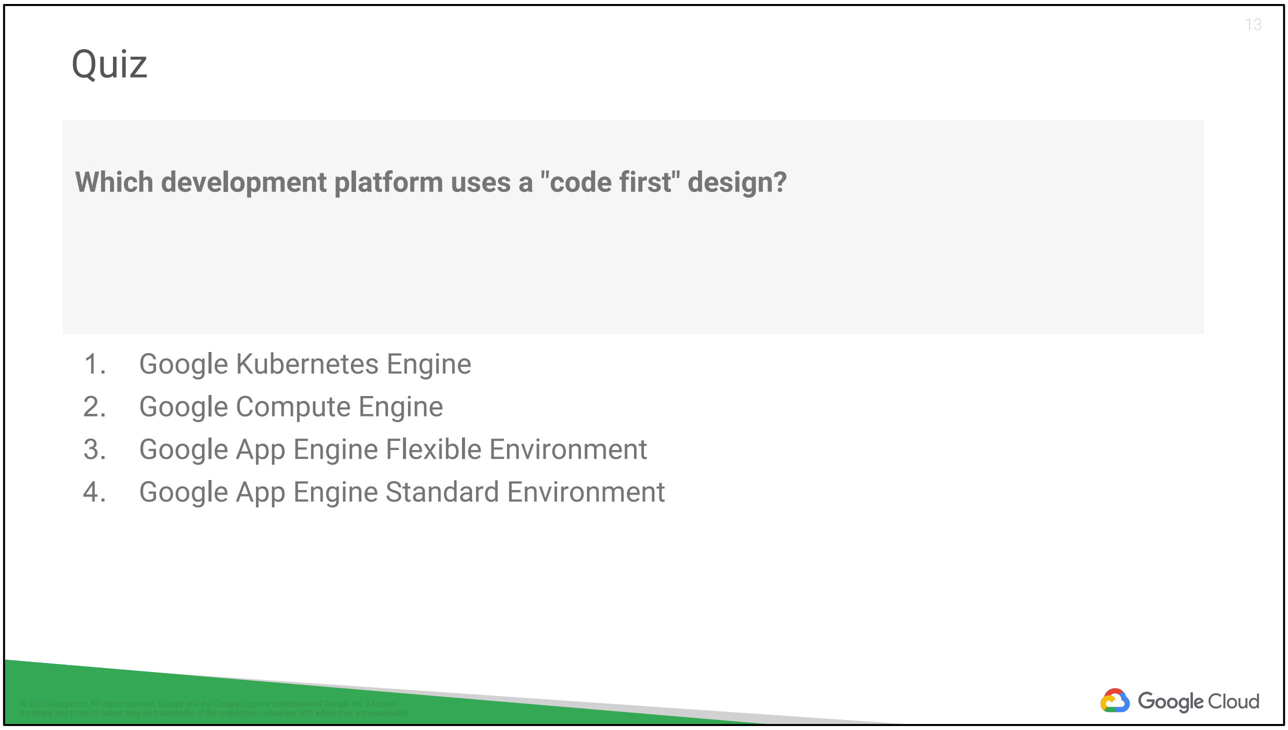


Quiz

Which platform enables customization including choice of the operating system?

1. Google Kubernetes Engine
2. Google App Engine
3. Google Compute Engine *
4. Cloud Functions

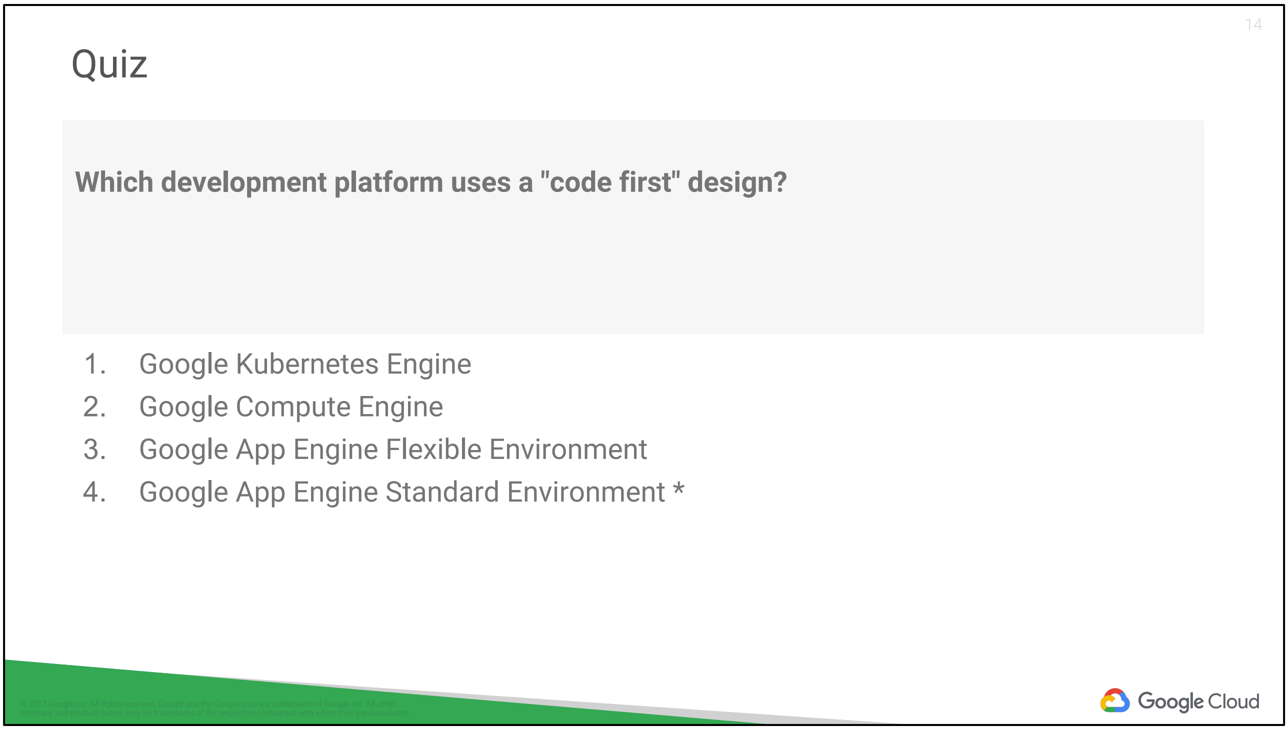
Explanation:
Compute Engine gives you complete control over the operating system.



Quiz

Which development platform uses a "code first" design?

1. Google Kubernetes Engine
2. Google Compute Engine
3. Google App Engine Flexible Environment
4. Google App Engine Standard Environment

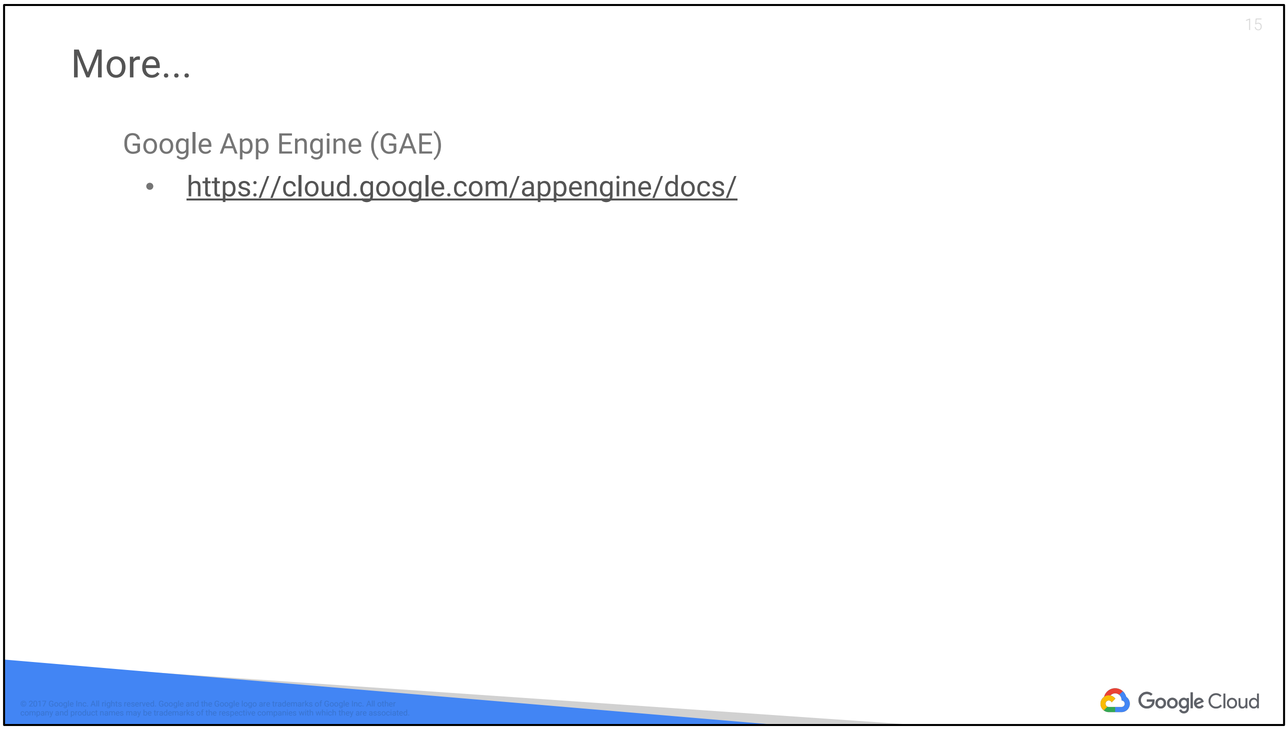


Quiz

Which development platform uses a "code first" design?

1. Google Kubernetes Engine
2. Google Compute Engine
3. Google App Engine Flexible Environment
4. Google App Engine Standard Environment *

Explanation:
Google App Engine Standard Environment gets apps running fast but offers less control and fewer options.



More...

Google App Engine (GAE)

- https://cloud.google.com/appengine/docs/

Google Cloud

More to learn on this subject. Here are some suggestions and links.

Google Cloud