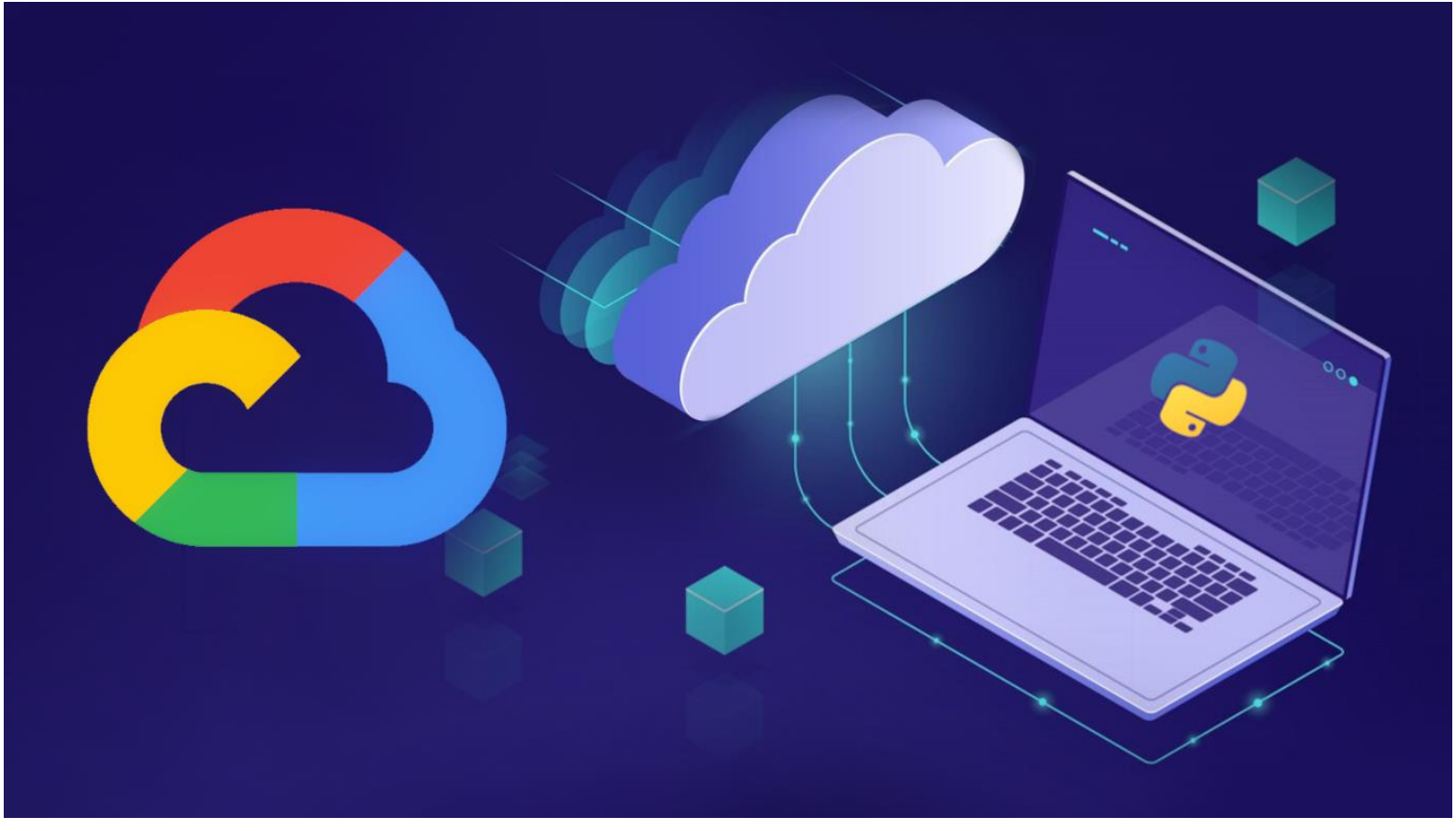
DEPLOYING PYTHON APPLICATIONS ON GOOGLE CLOUD PLATFORM (GCP)



SERVICES

Google Compute Engine
(GCE)

Google App Engine
(GAE)

Google Kubernetes Engine
(GKE)

Cloud Run

Cloud Functions

MORE CUSTOMIZABLE
MORE COMPLEX APPS
CONTINUOUS DEMAND

MORE AUTOMATIC
LESS COMPLEX APPS
SPORADIC DEMAND

GOOGLE COMPUTE ENGINE (GCE)

WHAT IS IT?

- ***Infrastructure as a Service*** (IaaS).
- Creation and management of virtual machines (VMs): instances.
- All management is done by the developer.

KEY FEATURES

- VMs with a high level of customization: CPU, memory, disk, operating system.
- Persistent disks attached to the VMs.
- Boot disks: choose or create your image.
- Network: easy connection to other GCP services and the internet.
- Allows ***load balancing*** and ***autoscaling***.
- Security: firewalls, identity/access management, and encryption.
- Best cost-benefit for long or continuous runs.

GOOGLE COMPUTE ENGINE (GCE)

USE CASES

- Web applications
- Data processing
- Machine learning
- Game Servers
- Development and testing

PROS

- Total control over the environment.
- Complex applications with specific needs.
- Cost-effective for long workloads.

CONS

- Need for management.
- More complex to set up and maintain.

GOOGLE APP ENGINE (GAE)

WHAT IS It?

- ***Serverless*** ℹ️ solution fully managed by GCP.
- Developers only care about the code.

KEY FEATURES

- Fully managed.
- Serverless.
- Scalability.
- Two deployment environments: standard and flexible.
- The cost is per resource used.

GOOGLE APP ENGINE (GAE)

USE CASES

- Web applications
- **APIs** **backend**
- **Microservices**

PROS

- Ease of use.
- Scalability.
- Serverless service flexibility with autoscaling.

CONS

- Vendor linking.
- Limited control.
- Pricing for large apps.

GOOGLE KUBERNETES ENGINE (GKE)

WHAT IS IT?

- Managed environment for deploying **containerized** applications.
- Built on top of the **Kubernetes** system (K8S, open source): nodes and clusters.
- Simplifies container orchestration: deployment, scaling, and load balancing.
- Developers only care about the code.

KEY FEATURES

- Fully managed, serverless.
- Almost unlimited scalability.
- High availability.
- Safety.
- Integration with GCP services.
- The cost is per resource used.

GOOGLE KUBERNETES ENGINE (GKE)

USE CASES

- Legacy application modernization.
- Cloud-native applications.
- ***DevOps*** and ***CI/CD*** pipelines.
- Machine Learning.

PROS

- Simplified Kubernetes management.
- Scalability and elasticity.
- Serverless service flexibility with autoscaling.

CONS

- Vendor linking.
- Limited control.
- Pricing for large apps.

CLOUD RUN

WHAT IS IT?

- Managed environment for deploying containerized applications.
- Designed to make it easy to build and deploy scalable applications.

KEY FEATURES

- Fully managed, serverless.
- Event-driven.
- The cost is per resource used.

CLOUD RUN

USE CASES

- Microservices.
- Web applications.
- API backend.
- Data processing pipelines.

PROS

- Easy to use.

CONS

- Cold start.
- Vendor linking.
- Limited control.

CLOUD FUNCTIONS

WHAT IS IT?

- Managed environment for deploying applications that perform functions.
- They are triggered by events such as HTTP requests, database changes, or messages.

KEY FEATURES

- Fully managed, serverless.
- Event-driven.
- The cost is per resource used.

CLOUD FUNCTIONS

USE CASES

- Simple web applications.
- API **endpoints** .
- Data processing.
- Image processing.
- Workflow automation.

PROS

- Easy to use.

CONS

- Limited runtime.
- Depuration.
- Cold start.
- Vendor linking.

IN THIS COURSE...

- Train a simple machine learning algorithm for image classification
- Develop a Flask application to use the trained algorithm to make predictions
- Deploy a simple app that serves the algorithm in a production environment, using the 5 GCP services presented.