



eBook Gratuit

APPRENEZ

google-cloud-storage

eBook gratuit non affilié créé à partir des
contributeurs de Stack Overflow.

#google-

cloud-

storage

Table des matières

À propos	1
Chapitre 1: Démarrer avec google-cloud-storage	2
Remarques.....	2
Exemples.....	2
La configuration initiale.....	2
Téléchargement HTTP (objets publics).....	2
Télécharger des fichiers en utilisant Python.....	2
Crédits	5

À propos

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [google-cloud-storage](#)

It is an unofficial and free google-cloud-storage ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official google-cloud-storage.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapitre 1: Démarrer avec google-cloud-storage

Remarques

Google Cloud Storage est un service de stockage durable et hautement disponible pour les objets de toutes tailles. Vous pouvez utiliser Google Cloud Storage pour une gamme de scénarios, y compris la diffusion de contenu de site Web, le stockage de données pour l'archivage et la récupération d'urgence, ou la distribution d'objets de données volumineux aux utilisateurs via un téléchargement direct.

La documentation officielle de Google pour GCS (Google Cloud Storage) est disponible à l'adresse <https://cloud.google.com/storage/docs/> .

Exemples

La configuration initiale

Google maintient la documentation pour commencer ici: <https://cloud.google.com/storage/docs/quickstart-console>

Se préparer à utiliser GCS:

1. [Créez un projet Google Cloud](#) si vous n'en avez pas déjà.
2. [Activez la facturation pour votre projet](#) afin de permettre la création de compartiments.
3. (Facultatif) [Installez le SDK Google Cloud](#) , qui inclut gsutil, l'utilitaire de ligne de commande de GCS. Vous pouvez également utiliser gsutil directement à partir de l'interface utilisateur de [Google Cloud](#) à l'aide de [Google Cloud Shell](#) .

Téléchargement HTTP (objets publics)

Si vous souhaitez télécharger un objet à partir de GCS qui est accessible au public, la méthode la plus simple consiste à utiliser un navigateur Web ou un outil de ligne de commande pour extraire une URL avec ce modèle: <https://storage.googleapis.com/bucketName/objectName> .

Exemple: <https://storage.googleapis.com/pub/someOfTheTeam.jpg>

Télécharger des fichiers en utilisant Python

Importer les bibliothèques nécessaires:

```
from gcloud import storage
```

Définir les variables nécessaires:

Client : regroupe la configuration requise pour les demandes d'API

```
client = storage.Client()
```

Paramètres facultatifs pour le `Client()` :

- **projet** : le projet pour lequel le client agit. Sera passé lors de la création d'un sujet. Si ce n'est pas passé, revient à la valeur par défaut déduite de l'environnement.
- Informations d' **identification** : OAuth2 Informations d'identification utilisées pour la connexion. Si ce n'est pas passé, revient à la valeur par défaut déduite de l'environnement.
- **http** : objet HTTP pour faire des requêtes. S'il n'est pas transmis, un objet `http` est créé et lié aux informations d'identification de l'objet actuel.

Bucket : sélectionne le compartiment créé dans le projet via la console Google Cloud.

```
bucket = client.get_bucket('<your-bucket-name>')
```

Pour plus d'informations sur les fonctions du `Client` , voir [Storage Client](#).

Blob : Nom du fichier qui sera enregistré.

```
blob = bucket.blob('my-test-file.txt')
```

Vous pouvez également définir des répertoires comme celui-ci:

```
filename = "%s/%s" % (folder, filename)
blob = bucket.blob(filename)
```

Il existe plusieurs méthodes pour télécharger un fichier. Vous pouvez vous attendre à un fichier dans la charge utile d'une `POST` ou `PUT` , ou à l'avoir localement sur votre système de fichiers. Vous pouvez même envoyer du texte directement à un fichier texte.

```
# Uploading string of text
blob.upload_from_string('this is test content!')

# Uploading from a local file using open()
with open('photo.jpg', 'rb') as photo:
    blob.upload_from_file(photo)

# Uploading from local file without open()
blob.upload_from_filename('photo.jpg')
```

Pour plus d'informations sur les fonctions de téléchargement, reportez-vous à [Blob / Objects](#).

Si vous avez besoin de votre blob pour être public, vous pouvez définir la confidentialité du fichier public:

```
blob.make_public()
url = blob.public_url
```

Lire Démarrer avec google-cloud-storage en ligne: <https://riptutorial.com/fr/google-cloud-storage/topic/1356/demarrer-avec-google-cloud-storage>

Crédits

S. No	Chapitres	Contributeurs
1	Démarrer avec google-cloud-storage	Brandon Yarbrough , Community , Jorge Caballero