

Assignment and Exam Content

Cloud pubsub

Cloud Gurus Seattle, USA
Training material

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

Cloud Pub/Sub Lab

Cloud Pub/Sub Lab Contains following topics

A

Create Topic , Create Subscriptions , Publish topic and Pull Message

Console and CLI

B

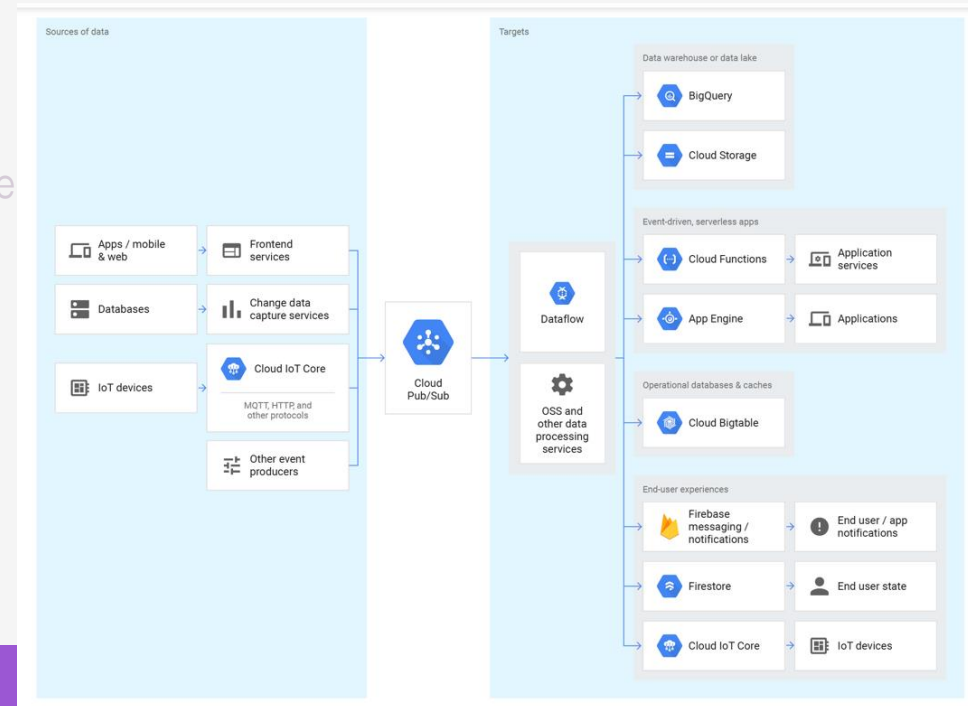
Create Cloud Storage Notification using Cloud PubSub (Console and CLI)

C

Exam TIPS

Cloud Pub/Sub is a fully-managed real-time messaging service that allows you to send and receive messages between independent applications.

<https://cloud.google.com/pubsub/>



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

A

Create PubSub Topic

Cloud PubSub

1

Go To -> BIGDATA -> Cloud PubSub -> Topics -> CREATE TOPIC

2

Create topic

Create Topic takes only two parameter

Name : Name of topic – Your choice

Encryption Method : Google managed – by default

Customer managed key -> Your own key.

Topic is created now

Create a topic

A topic forwards messages from publishers to subscribers.

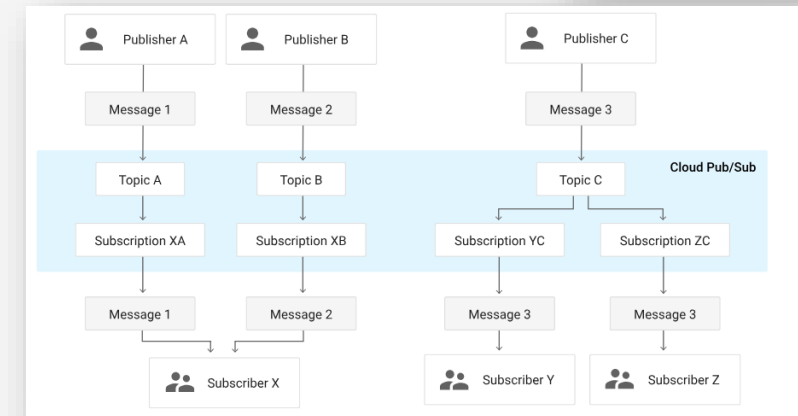
Name *
mytopic

Topic ID: projects/tokyo-data-243419/topics/mytopic

Encryption

- ☒ Google-managed key
No configuration required
- ☐ Customer-managed key
Manage via Google Cloud Key Management Service

CANCEL CREATE TOPIC



Google Cloud Platform My First Project

Pub/Sub

Topics

Subscriptions

Snapshots

Topic details

mytopic

Topic ID: projects/tokyo-data-243419/topics/mytopic

Encryption key: Google-managed key

Labels: -

SUBSCRIPTIONS

SNAPSHOTS

CREATE SUBSCRIPTION

Filter table

Subscription name	Subscription id	Project
No rows to display		

- **Topic**: A named resource to which messages are sent by publishers.
- **Subscription**: A named resource representing the stream of messages from a single, specific topic, to be delivered to the subscribing application. For more details about subscriptions and message delivery semantics, see the [Subscriber Guide](#).
- **Message**: The combination of data and (optional) attributes that a publisher sends to a topic and is eventually delivered to subscribers.
- **Message attribute**: A key-value pair that a publisher can define for a message. For example, key `iana.org/language_tag` and value `en` could be added to messages to mark them as readable by an English-speaking subscriber.

A

Create PubSub Subscription

3

Create Subscription

Provide name of subscription

Topic ID – from topic created.

Delivery Type – Pull and Push

Subscription expiry in number of days or you can choose non Expiry.

Acknowledgement deadline for delivery of message.

- 10 Sec by default

Message retention duration

- Amount of time the message saved

Retain ack. Messages : Saved for 7 days if enabled.

Hit **Create**

Google Cloud Platform My First Project

Pub/Sub Create subscription

A subscription directs messages on a topic to subscribers. Messages can be pushed to subscribers immediately, or subscribers can pull messages as needed.

Subscription name *
mysubs

Subscription ID: projects/tokyo-data-243419/subscriptions/mysubs

Topic ID *
projects/tokyo-data-243419/topics/mytopic

Existing Cloud Pub/Sub topic to subscribe from. The topic ID should be in the format of projects/<project-id>/topics/<topic-name>

Delivery type ?

☒ Pull
☒ Push

Subscription expiration ?

☒ Expire after this many days of inactivity (up to 365)
31 Days
☐ Never expire
The subscription will never expire no matter the activity.

Acknowledgement deadline ?
Deadline time is from 10 seconds to 600 seconds
10 Seconds

Message retention duration
Deadline time is from 10 minutes to 7 days
Days: 7 Hours: 0 Minutes: 0

Retain acknowledged messages ?
☐ Enable

CREATE

Push Delivery type needs endpoint URL

Delivery type ?

☐ Pull
☒ Push

Endpoint URL *

Endpoint URL is required for push subscriptions.

☐ Enable authentication [Learn More](#)

Your subscriptions is created.

A

Cloud Pub/Sub –

Subscriptions Details -

Observe subscription ID , Topic ID, Monitoring Graphs

Pull Messages

You can also create Snapshots -

The screenshot displays the Google Cloud Platform console interface for a Pub/Sub subscription. The top navigation bar shows 'Google Cloud Platform' and 'My First Project'. The left sidebar contains a menu with 'Pub/Sub', 'Topics', 'Subscriptions', and 'Snapshots'. The main content area is titled 'Subscription details' and shows the subscription 'mysubs'. Two monitoring graphs are visible: 'Unacked message count' and 'Oldest unacked message age'. Both graphs show a flat line at zero with a warning icon and the message 'No data is available for the selected time frame.' The right sidebar lists subscription details: Subscription ID (projects/tokyo-data-243419/subscriptions/mysubs), Topic ID (projects/tokyo-data-243419/topics/mytopic), Delivery type (Pull), Subscription expiration (31 days), Acknowledgement deadline (10 Seconds), Message retention duration (7 Days), Retained acknowledged messages (Yes), and Labels (-).

Google Cloud Platform My First Project

Pub/Sub

Subscription details EDIT PULL MESSAGES CREATE SNAPSHOT SEEK DELETE

Topics

Subscriptions

Snapshots

mysubs

1 hour 6 hours 1 day 7 days 30 days RESET ZOOM

Unacked message count

Oldest unacked message age

Subscription ID
projects/tokyo-data-243419/subscriptions/mysubs

Topic ID
projects/tokyo-data-243419/topics/mytopic

Delivery type
Pull

Subscription expiration
Subscription expires in 31 days if there is no activity.

Acknowledgement deadline
10 Seconds

Message retention duration
7 Days

Retained acknowledged messages
Yes

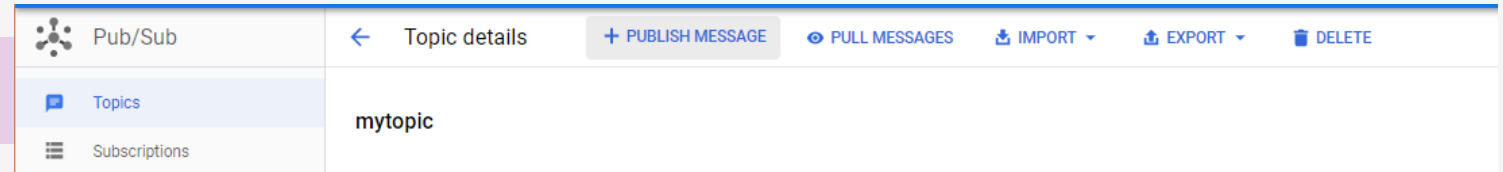
Labels
-

A

Cloud Pub/sub – Publish and Pull message

Go to Topic , Click on **PUBLISH MESSAGE**

1



Publish Message

Enter Message Body - you like
Attributes – you like

- E.g. given here...

Once you fill details – Click **PUBLISH**

← Publish message

Topic ID
projects/tokyo-data-243419/topics/mytopic

Message body ⓘ
Message
This is my 1st message on Cloud pubsub

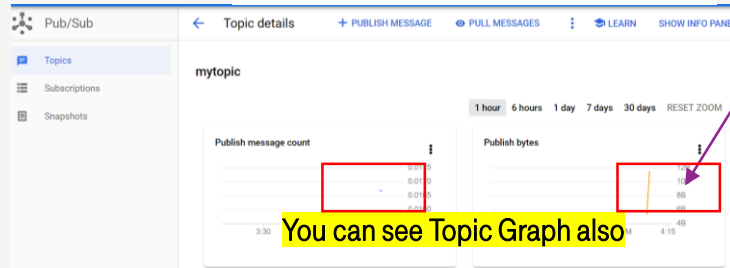
The message you want to publish to this topic. Either message or attribute will be required to publish.

Message attributes ⓘ

Key	Value
Name	Cloud Gurus
location	Seattle

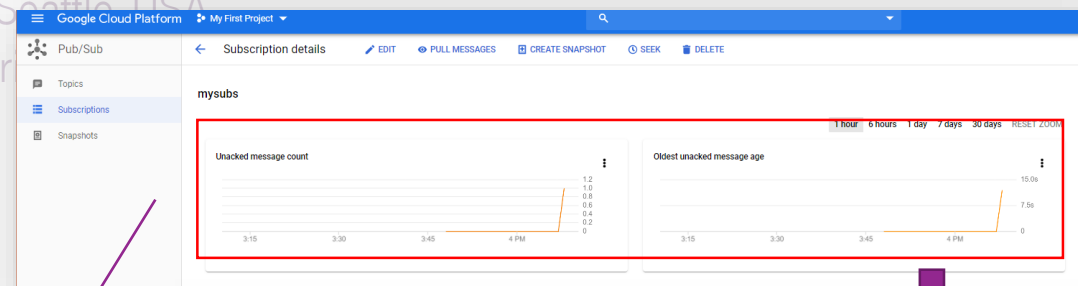
[+ ADD AN ATTRIBUTE](#)

PUBLISH CANCEL



2

Go to Subscriptions : You can see message shown in graph.
, Click on **PULL MESSAGES**



Messages

Click Pull to view messages and temporarily delay message delivery to other subscribers. Select **Enable ACK** messages and then click **ACK** next to the message to permanently prevent message delivery to other subscribers. Use this option cautiously in production environments. If you miss the acknowledgement deadline (10 seconds), the message will be sent again if no other subscribers of this subscription acknowledged the message. [Learn more](#)

PULL ☐ Enable ack messages

Filter table

Publish time	Message body	attribute.location	attribute.Name	Ack ↑
Jul 11, 2019, 4:07:42 PM	This is my 1st message on Cloud pubsub	Seattle	Cloud Gurus	ACK

A

Try Sample Commands using CLI.

4

Create Topic

```
gcloud pubsub topics create my-topic
```

Create Subscription

```
gcloud pubsub subscriptions create --topic my-topic my-sub
```

Publish Message on Topic

```
gcloud pubsub topics publish my-topic --message "hello"
```

Pull Message using Subscriptions

```
gcloud pubsub subscriptions pull --auto-ack my-sub
```

```
gcloud pubsub subscriptions pull --auto-ack cloudstorage
```

```
gcloud pubsub subscriptions pull --auto-ack cloudstorage
```

Gsutils used for Cloud Storage

Seattle, USA

Update Dataset

```
bq update --description "Description of mydataset" mydataset
```

```
bq update --default_table_expiration 7200 mydataset
```

Remove

```
bq rm -r -f -d [PROJECT_ID]:[DATASET]
```

Access Control Access.

```
bq show --format=prettyjson [PROJECT_ID]:[DATASET] > [PATH_TO_FILE]
```

```
bq update --source [PATH_TO_FILE] [PROJECT_ID]:[DATASET]
```

Listing Dataset

Listing

```
bq ls --format=prettyjson --project_id
```

```
bq ls --format=prettyjson
```

```
bq show --format=prettyjson [PROJECT_ID]:[DATASET]
```

```
bq query --nouse_legacy_sql \
```

```
'SELECT * EXCEPT(schema_owner) FROM INFORMATION_SCHEMA.SCHEMATA'
```

Try Yourself – Follow demo and create Cloud Storage Notifications on Cloud PubSub

4

Cloud Storage Notification

Create Cloud Storage Bucket

```
gsutil notification create -f json gs://mycsdemobucket  
gsutil notification create -f json -t files-to-process gs://example-bucket  
gsutil notification create -p photos/ gs://example-bucket  
gsutil notification list gs://example-bucket  
gsutil notification delete gs://example-bucket  
gsutil notification delete projects/_/buckets/example-  
bucket/notificationConfigs/1
```

Pull the message from Cloud PubSub.

```
gcloud pubsub subscriptions pull --auto-ack mycsdemobucket  
gcloud pubsub subscriptions create --topic mycsdemobucket mycs  
gcloud pubsub subscriptions pull --auto-ack mycs
```

Gsutils used for Cloud Storage

Any Problems executing –
Please check Demo for Cloud Pub Sub.

Cloud Pub/Sub : Try Yourself

1

Try Different Notifications - Cloud Function / Cloud Storage

Delete all Topics and Subscriptions

2

Exam Tips

Important concepts are

1. Integration framework – High performance
2. Scalability – Fully Scalable integration framework
3. Automatic Scalable – No infrastructure to manage.

Cloud Gurus Seattle, USA
Training material

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

Cloud Gurus Seattle, USA
Training material

End of Cloud Pub/Sub Assignment