

**BESPIN GLOBAL**

HELPING YOU ADOPT CLOUD.

# 베스핀글로벌의 GCP(구글 클라우드 플랫폼) 가이드



## 구글 클라우드 프리미어 파트너 베스핀글로벌



- 구글 클라우드 프리미어 파트너
- 국내 유일 구글 클라우드 공인 트레이닝 파트너
- 국내 최대 규모 구글 클라우드 플랫폼(GCP) 전문가 보유
- 구글 클라우드 전문가를 통한 최적의 아키텍처 컨설팅 및 구축 서비스 제공

### Helping You Adopt Cloud.

베스핀글로벌은 클라우드의 여정에 어려움을 풀어주는 해결사입니다.

구글 프리미어 파트너사로서 클라우드 플랫폼(GCP) 및 G Suite GCP 전담팀이 안정적인 클라우드 도입 및 이전을 책임집니다. 더불어 구글 클라우드 공인 트레이닝 파트너인 베스핀글로벌은 클라우드에 많은 경험을 가진 엔지니어들이 기업의 전문 교육 및 컨설팅도 제공하고 있습니다.

베스핀글로벌은 다년간의 경험 노하우를 통해 수립한 방법론과 베스트 프랙티스를 기반으로 하는 전문가로 고객사는 다양한 서비스의 혜택을 누리실 수 있습니다.

베스핀글로벌은 고객사 사업 특성에 적합한 제품과 서비스 가이드로 해당 기업에 가장 알맞은 구글 클라우드 도입, 마이그레이션 및 운영까지 전 단계 성공을 보장합니다.

비즈니스 성장에 따른 서비스 설계는 비용 절감을 가능케 하며, 클라우드에 특화된 베스핀글로벌의 운영 인력 확보는 IT 팀의 재교육, 인원 충원 등에 드는 간접 비용 또한 줄이는 방법이 될 것입니다.

여기에 클라우드 공급자가 제공하는 IaaS와 PaaS 영역 외에 애플리케이션과 네트워크 모니터링, 백업과 DR까지 통합적인 지원 서비스를 받으실 수 있습니다.

베스핀글로벌은 국내뿐 아니라 중국, 미국, 유럽에 걸친 지원 체계를 기반으로 언제 어디서나 이슈와 장애에 신속하게 대응합니다. SLA를 훨씬 뛰어넘는 99.99% 이상의 높은 가용성, 베스핀글로벌의 매니지드 서비스가 특별한 이유입니다.

## Introduce

빠르게 변하는 IT 기술, 특히 클라우드 관련 기술은 날이 다르게 변화하고 있습니다. 한정적인 IT 인원으로서는 다양한 기술들을 모두 따라잡을 수 없습니다. 베스핀글로벌은 각 분야에 특화된 100여명 이상의 클라우드 운영팀이 빠르게 변하는 클라우드 기술을 고객 환경에 최적화된 형태로 공급합니다.

이러한 전문성은 베스핀글로벌이 APAC(Asia Pacific Accreditation Cooperation) 내 구글 클라우드 관련 최다 자격증을 보유한 점으로 증명되어 집니다.

베스핀글로벌의 아키텍처 전략은 다음과 같습니다.

### 베스핀글로벌의 아키텍처 전략

- ▶ 전략 컨설팅
- ▶ 환경/요구사항 분석, 거버넌스 및 보안체계 구축
- ▶ 클라우드 선택과 설계
- ▶ 솔루션 선택/도입, 애플리케이션 재설계
- ▶ 클라우드로의 이전
- ▶ IT 프로세스 재정의, 데브옵스(DevOps) 구현
- ▶ 구축 & 구현
- ▶ 프로세스 자동화
- ▶ 관리 & 최적화

구글 클라우드 도입 부터 마이그레이션 등 까지 처음부터 끝까지 믿을 만한 파트너를 찾는다면, 베스핀글로벌이 정답입니다.

가트너 매직 쿼드런트  
퍼블릭 클라우드 MSP부분  
한중일 최초 3연속 등재



국내 유일 ISO, ISMS, GS인증을 확보한 MSP

## 목차

1. 구글 클라우드 플랫폼의 Cloud Healthcare API 소개
2. 구글 클라우드 플랫폼의 Datalab 및 BigQuery를 이용한 빅 데이터 분석
3. 구글 클라우드 플랫폼에 IoT Analytics Pipeline 구축하기

## 1장

# 구글 클라우드 플랫폼의 Cloud Healthcare API 소개



### 구글 클라우드 플랫폼의 Cloud Healthcare API 소개

구글 클라우드 플랫폼의 Cloud Healthcare API는 기존 케어 시스템과 Google Cloud에 호스팅 된 애플리케이션을 연결시키는 기능을 하고, Google Cloud Platform(GCP)에서 의료 데이터를 저장하고 액세스 할 수 있는 관리 솔루션을 제공합니다.

Cloud Healthcare API는 기존 헬스케어 기술의 표준 데이터 형식 및 프로토콜을 지원하며, 데이터 처리에 유용한 Cloud Dataproc, 빅데이터 분석을 위한 BigQuery 및 머신러닝을 위한 Cloud ML 엔진과 같은 Google Cloud 기능을 연결해서 사용자의 어플리케이션 개발 및 디바이스 통합을 단순화시켜줍니다.

API를 사용하면 데이터 분석, 머신러닝 및 응용프로그램 개발을 위한 기능을 활용할 수 있으며, 이러한 기능을 활용해 차세대 의료 솔루션을 구축 할 수 있습니다.

API는 헬스케어 데이터에 대한 업계의 주요 표준을 이행하는 세 가지 특정 인터페이스로 구성됩니다.

※ 구글 클라우드 플랫폼의 Cloud Healthcare API는 현재 알파버전 상태이며, Google의 테스터 프로그램을 통해 제공되고 있습니다.

### 헬스케어 데이터에 대한 업계의 주요 표준을 이행하는 세 가지 특정 API 인터페이스

#### 1. FHIR (Fast Healthcare Interoperability Resources): 건강 데이터 교환을 위한 신규 표준

FHIR (Fast Healthcare Interoperability Resources)는 차세대 의료정보표준 플랫폼으로 전자 의료 정보를 교환하기 위한 목적을 지니며 의료 환경에서 다루어지는 다양한 정보들을 'Resource' 형태로 구현하고 이를 이용하여 다양한 의료정보 간 상호 운용성을 보장하는 표준안입니다. FHIR은 REST API를 이용하여 'Resource'를 효율적으로 관리하도록 합니다. Google Cloud의 FHIR API는 STU3 리소스를 완벽하게 지원합니다. 또한 Google Cloud는 분석 및 머신러닝 톨에 데이터 처리를 단순화하기 위해 다른 형식의 데이터를 FHIR 리소스로 변환해주는 톨과 유틸리티들을 개발 중 입니다.

#### 2. DICOM (Digital Imaging and Communications in Medicine): 방사선과 영상관련 분야에서 대표로 사용되는 표준

의료 이미징 정보 및 관련 데이터의 통신 및 관리를 의미합니다. 이 규정은 주로 초음파진단기, X-ray, MRI, CT, RI 등 의료영상이 나오는 장비에 적용됩니다. DICOMweb API를 사용하면 기존 이미징 장치, PACS 솔루션 및 뷰어가 Cloud Healthcare API와 직접 또는 기존 DICOM DIMSE 프로토콜을 지원하도록 설계된 오픈 소스 어댑터를 통해 상호 작용할 수 있습니다. DICOMweb API는 재해복구를 위해 연구, 머신러닝 모델 트레이닝 및 저장용 이미징 데이터 세트의 큐레이션 및 export를 지원합니다. 또한 머신러닝이나 다른 프로세싱 모듈을 기존 클리닉 워크플로우에 통합하는데 사용할 수 있습니다.

### 3. HL7v2 (Health Level7): 보건시스템 통합에 가장 많이 사용되는 방법

HL7v2는 레거시 임상 시스템과 통합하려는 모든 application의 필수적인 통신 방식입니다. HL7v2 API는 HL7v2 메시지를 수집, 전송, 검색을 위해 REST 인터페이스를 사용합니다. 또한 Google Cloud는 MLLP(Minimal Lower Layer Protocol)를 통해 메시지를 주고받을 수 있는 오픈소스 어댑터를 제공합니다. 이 어댑터는 Google Kubernetes Engine에서 실행되어 신속한 프로비저닝을 제공하고, Cloud Pub/Sub 상에서 통신을 수행해 확장성을 제공하며, Cloud VPN을 통해 전송 보안을 지원합니다.

## Cloud Healthcare API 주요 특징

### 1. 표준 준수

Google은 여러 의료 표준 단체 활동을 통해 표준 기반의 기능을 제공하고 있습니다. Cloud Healthcare API의 각 양식별 data store 및 관련 API는 해당 표준을 따르고 있습니다. 예를 들면 FHIR store는 FHIR의 현재 버전에서 사용하는 STU3를 사용하며, DICOM store는 의료 이미지 교환을 위한 웹 기반 표준인 DICOMweb을 사용하고 있습니다. Cloud Healthcare API 향후 업데이트에서 추가적인 표준 버전과 표준 이외의 요청들을 처리할 수 있도록 지원할 예정입니다.

### 2. 개인정보보호 규정 준수

GCP는 미국의 HIPAA, 캐나다의 PIPEDA 및 다른 글로벌 개인정보보호 표준을 ([cloud.google.com/security/compliance](https://cloud.google.com/security/compliance)) 준수하는 방법에 대한 자세한 가이드를 규정하고 있습니다.

### 3. 데이터 위치 제어

Cloud Healthcare API는 데이터 위치를 API의 핵심 구성 요소로 취급합니다. 현재 사용 가능한 위치 중에서 데이터 세트의 저장 위치를 선택할 수 있습니다. 사용 가능한 위치는 지리적으로 구분됩니다. 미래에는 저장소를 더 넓은 지역으로 분산시킬 수 있도록 할 예정입니다.

### 4. 보안

Cloud Healthcare API 보안 모델은 Google에서 기 입증된 IAM (Identity and Access Management) 시스템을 기반으로 하고 있습니다. IAM의 세분화된 권한을 통해 의료 데이터에 대한 액세스를 완벽하게 제어할 수 있습니다. 또한 광범위의 위협 탐지 및 트래픽 관리 기능을 제공하는 Apigee API 관리 시스템 기반 오픈 소스 프록시를 제작하여 중요한 전자 의료정보를 안전하게 사용될 수 있도록 했습니다.

### 5. Bulk import and export

Cloud Healthcare API의 DICOM 및 FHIR 양식은 대량의 데이터 입출력을 지원하며, 이는 Cloud Storage 시스템을 통해 보다 쉽게 데이터 전송을 수행할 수 있습니다.

### 6. 비식별화

DICOM에 대한 비식별화 지원으로 실험 및 기타 목적의 연구에서 환자 정보를 보다 쉽게 가공할 수 있습니다. 비식별화는 데이터 저장소 기준으로 동작합니다.

### 7. Auditability

Cloud Healthcare API에 대한 관리 및 데이터 액세스 요청을 모두 audit 할 수 있습니다. Google Cloud의 Stackdriver 하이브리드 모니터링 시스템을 통해 로그를 확인할 수 있습니다.

### 8. 고가용성

Google Cloud의 안전한 인프라를 통해 mission-critical한 환경에서도 적용 가능합니다.



## 2장

# 구글 클라우드 플랫폼의 Datalab 및 BigQuery를 이용한 빅 데이터 분석

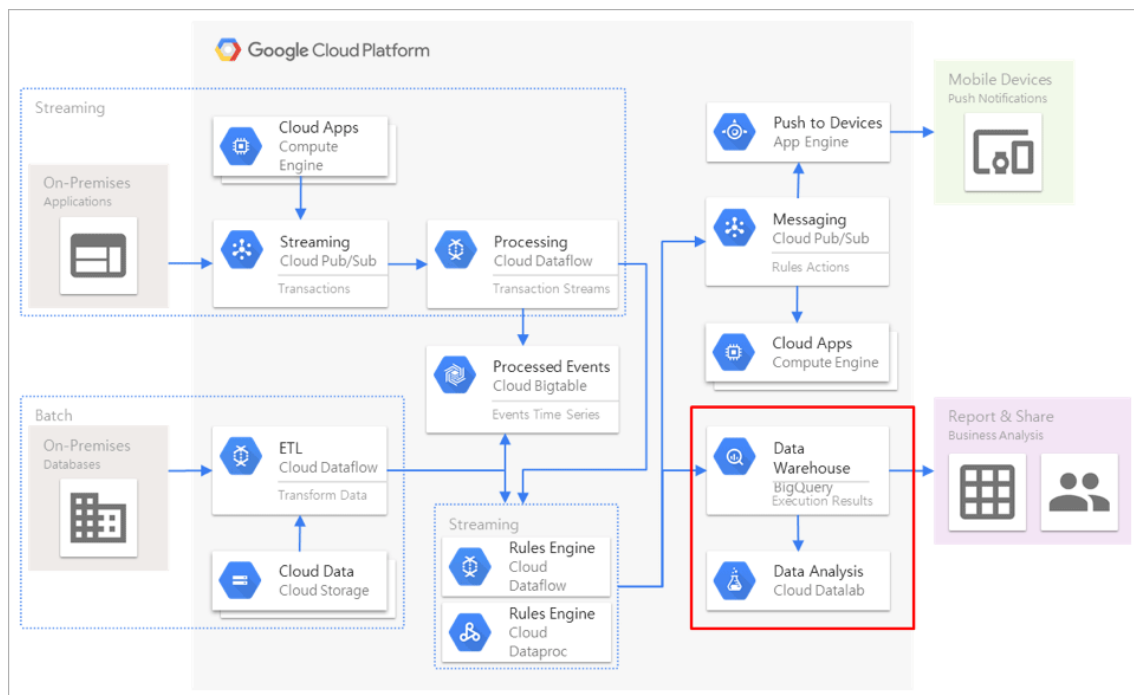


### 구글 클라우드 플랫폼의 Datalab 및 BigQuery를 이용한 빅 데이터 분석

이번 테크블로그에서는 Datalab 및 BigQuery를 이용한 빅데이터 분석에 대해 살펴보겠습니다.

- Google BigQuery는 대규모(페타바이트 급) 데이터 분석을 위한 빠르고 경제적인 완전 관리형 기업용 데이터 웨어하우스입니다.
- 관리할 인프라가 없기 때문에 익숙한 SQL을 이용하여 데이터를 분석해 의미 있는 유용한 정보를 찾는 데 집중할 수 있으며, 데이터 베이스 관리자가 필요하지 않습니다.
- Cloud Datalab은 데이터 탐색, 분석, 시각화를 위한 도구로 Python, SQL 과 같은 익숙한 언어를 이용하여, 데이터 분석 및 변환을 Interactive하게 수행합니다.

[그림 1] 빅데이터 분석 표준 아키텍처



이 글에서는 Google BigQuery 및 Cloud Datalab을 사용하여 대규모 (1억 3천 7백만 행) 출생 데이터 세트를 분석하는 실습을 수행합니다. 실습에서는 다음과 같은 과정을 수행합니다.

- Cloud Datalab 실행
- BigQuery 쿼리 호출
- Datalab에서 그래프 만들기
- 머신러닝을 위한 Data Export

## <실습 절차>

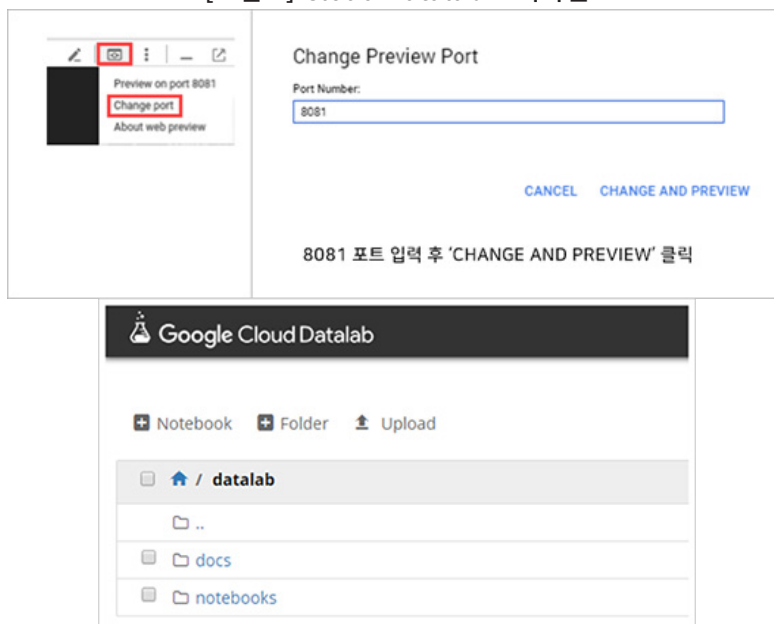
### 1. Cloud Datalab 실행

1) Datalab을 시작하려면 먼저 VM 인스턴스를 만들어야 합니다. Cloud Shell에서 다음을 입력합니다.

```
$ datalab create babyweight -zone us-central1-c
```

2) Datalab은 생성하는데 약 5분이 소요되며, Cloud Shell의 상단에 웹 미리보기 아이콘으로 접속합니다. 생성 되는 시간에 Big Query를 살펴 봅니다.

[그림 2] Cloud Datalab 초기화면

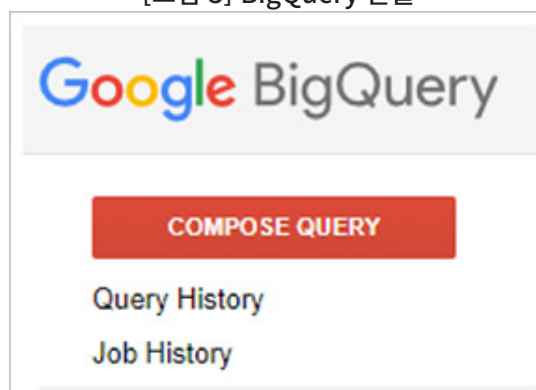


### 2. 개인정보보호 규정 준수

GCP에서는 BigQuery 분석 예시용 Public Dataset을 제공합니다. 본 실습에서는 출산율 관련 Public Dataset을 사용합니다. (bigquery-public-data.samples.natality)

1) BigQuery를 수행하기 위하여, BigQuery 콘솔 창으로 이동합니다. 콘솔에서 Compose Query를 클릭합니다.

[그림 3] BigQuery 콘솔



2) 쿼리 텍스트 상자에서 다음을 입력하고 Run Query를 클릭합니다.

**SELECT**

plurality,  
COUNT(1) AS num\_babies,  
AVG(weight\_pounds) AS ave\_weight

**FROM**

`bigquery-public-data.samples.natality`

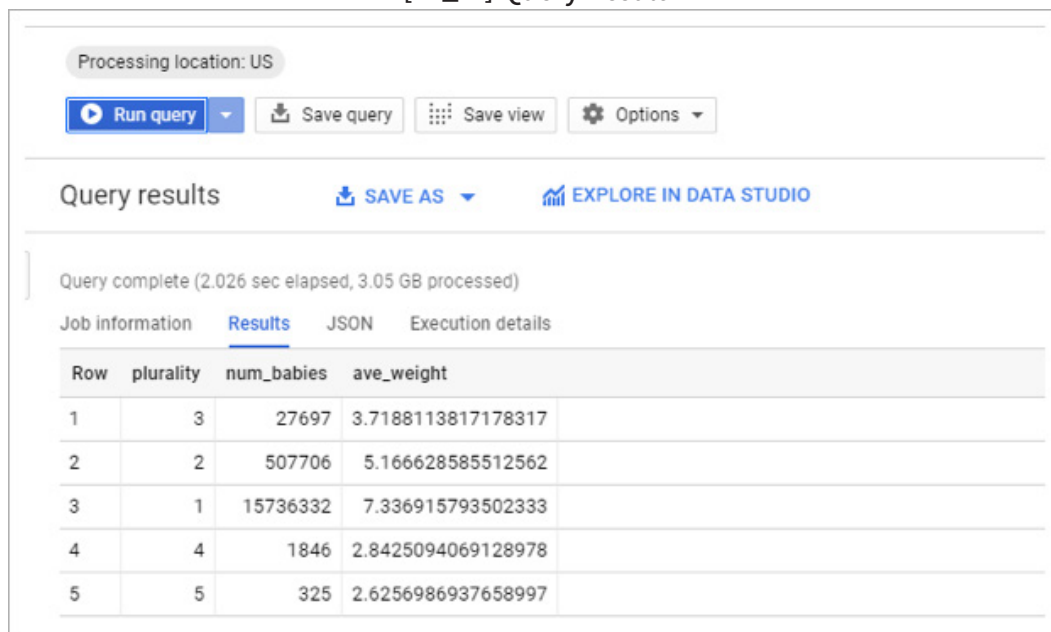
**WHERE**

year > 2000 AND year < 2005

**GROUP BY**

plurality

[그림 4] Query Result



Processing location: US

Run query Save query Save view Options

Query results SAVE AS EXPLORE IN DATA STUDIO

Query complete (2.026 sec elapsed, 3.05 GB processed)

Job information Results JSON Execution details

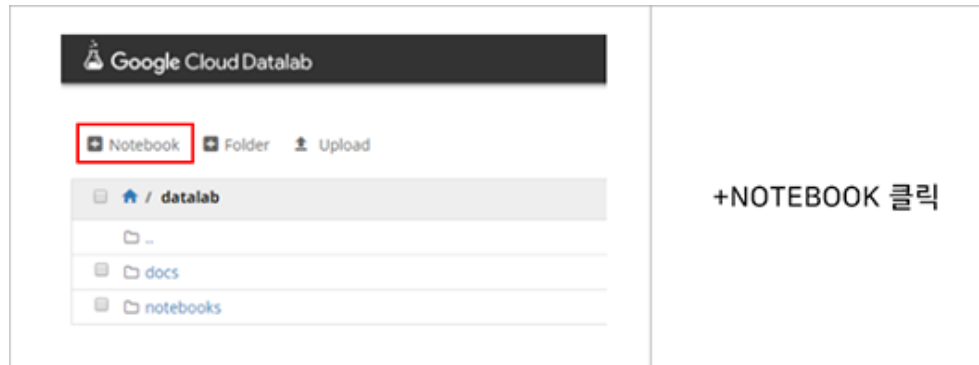
Row	plurality	num_babies	ave_weight
1	3	27697	3.7188113817178317
2	2	507706	5.166628585512562
3	1	15736332	7.336915793502333
4	4	1846	2.8425094069128978
5	5	325	2.6256986937658997

쿼리 결과는 2000 ~ 2005년 사이의 다둥이 출생아 수치를 보여줍니다.

Ex) 세 쌍둥이는 총 27,697명이 태어 났으며, 평균 몸무게는 3.71 파운드입니다.

### 3. Visualize data in Cloud Datalab

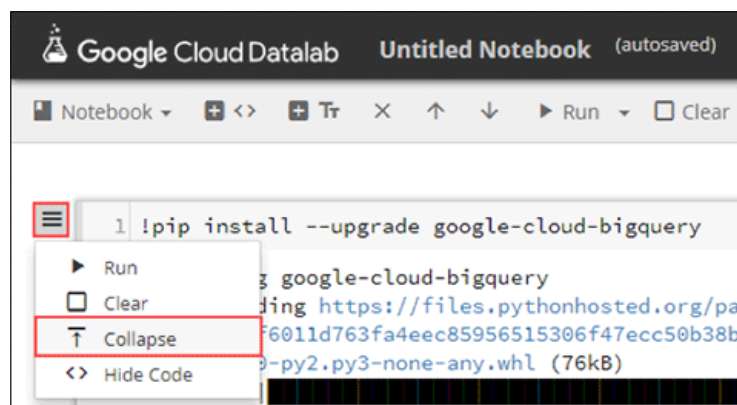
1) Datalab 화면에서 Notebook을 추가합니다.



2) 첫 번째 셀에서 BigQuery Python 라이브러리를 최신으로 업데이트합니다. 셀에서 다음을 입력하고 상단에 RUN 버튼을 눌러 실행합니다.

```
!pip install --upgrade google-cloud-bigquery
```

결과 값이 많이 출력됨으로 셀을 다음과 같이 접어서 숨길 수 있습니다.



3) 다음 코드를 삽입하여 BigQuery Python 클라이언트 라이브러리를 가져오고 클라이언트를 초기화합니다. BigQuery 클라이언트는 BigQuery API에서 메시지를 주고받는 데 사용됩니다. Shift+Enter를 사용하여 셀을 실행합니다. (RUN 버튼 클릭과 동일)

```
from google.cloud import bigquery  
client = bigquery.Client()
```

4) 출생정보 관련 Public DataSet을 사용하여 1969년 ~ 2008년 까지 등록된 다둥이의 출산율을 조사합니다. 다음의 쿼리를 입력하고 Shift+Enter를 이용하여 실행합니다.

```
sql = """ SELECT plurality, COUNT(1) AS count, year
FROM `bigquery-public-data.samples.natality`
WHERE NOT IS_NAN(plurality) AND plurality > 1
GROUP BY plurality, year ORDER BY count DESC
```

"""

```
df = client.query(sql).to_dataframe()
df.head()
```

쿼리 결과 데이터 프레임의 5행이 다음과 같이 출력됩니다.

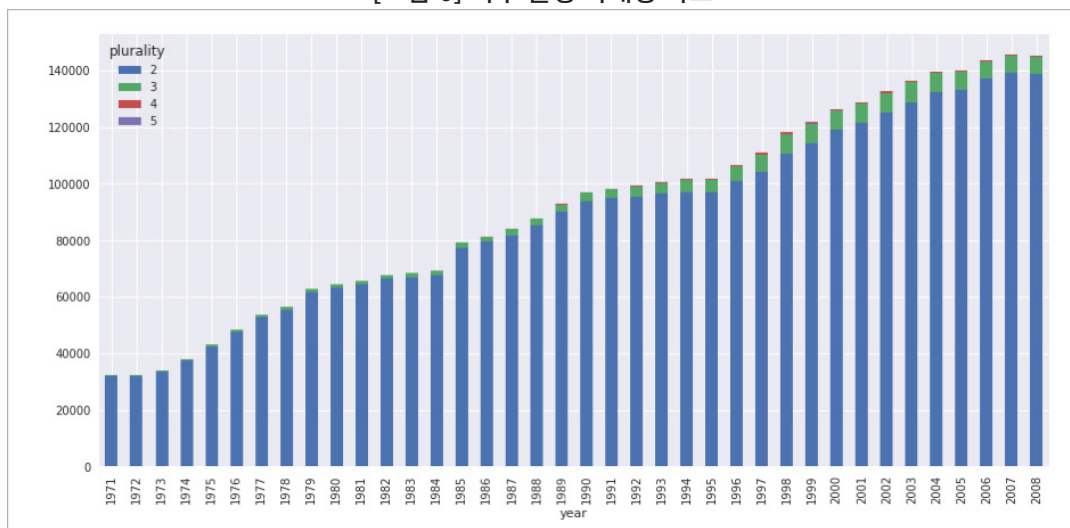
[그림 5] Query Result

	plurality	count	year
0	2	139209	2007
1	2	138866	2008
2	2	137239	2006
3	2	133285	2005
4	2	132344	2004

5) 다음코드에서 4)번에서 실행한 데이터 프레임을 이용하여 데이터를 피벗하고 시간경과에 따른 복수 출생 횟수의 누적 막대형 차트를 만듭니다.

```
pivot_table = df.pivot(index='year', columns='plurality', values='count')
pivot_table.plot(kind='bar', stacked=True, figsize=(15,7));
```

[그림 6] 복수 출생 막대형 차트



## 구글 클라우드 플랫폼의 Datalab 및 BigQuery를 이용한 빅 데이터 분석

6) 다음으로 성별과 아기 체중의 상관성을 살펴봅니다. 다음 셀에 아래의 쿼리 문을 입력하고, 막대 그래프를 생성합니다.

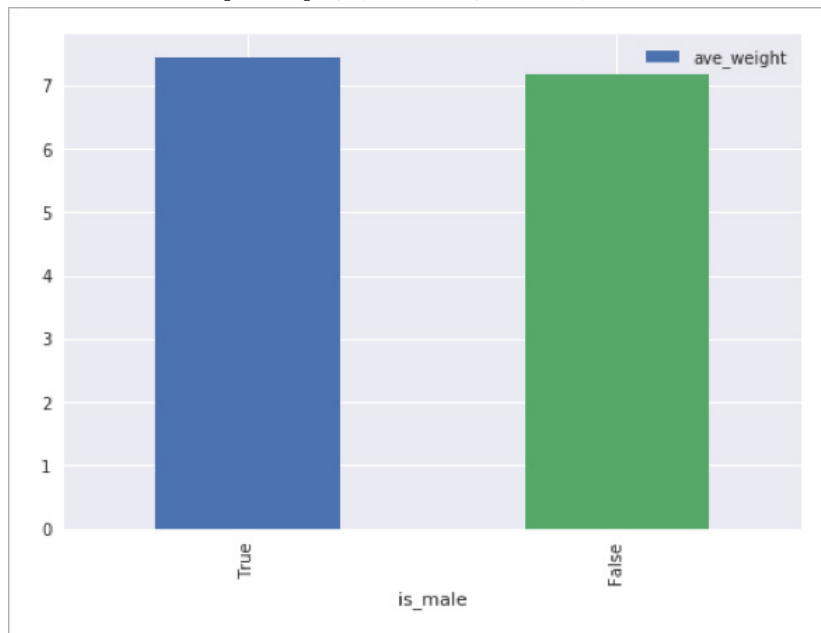
```
sql = """ SELECT is_male, AVG(weight_pounds) AS ave_weight

FROM `bigquery-public-data.samples.natality`
GROUP BY is_male
"""

df = client.query(sql).to_dataframe()
df.plot(x='is_male', y='ave_weight', kind='bar');
```

미세하게 남자 아이가 더 무겁다는 사실을 알 수 있지만, 성별과 큰 차이는 없다는 사실을 알 수 있습니다.

[그림 7] 아기 성별에 따른 평균 무게



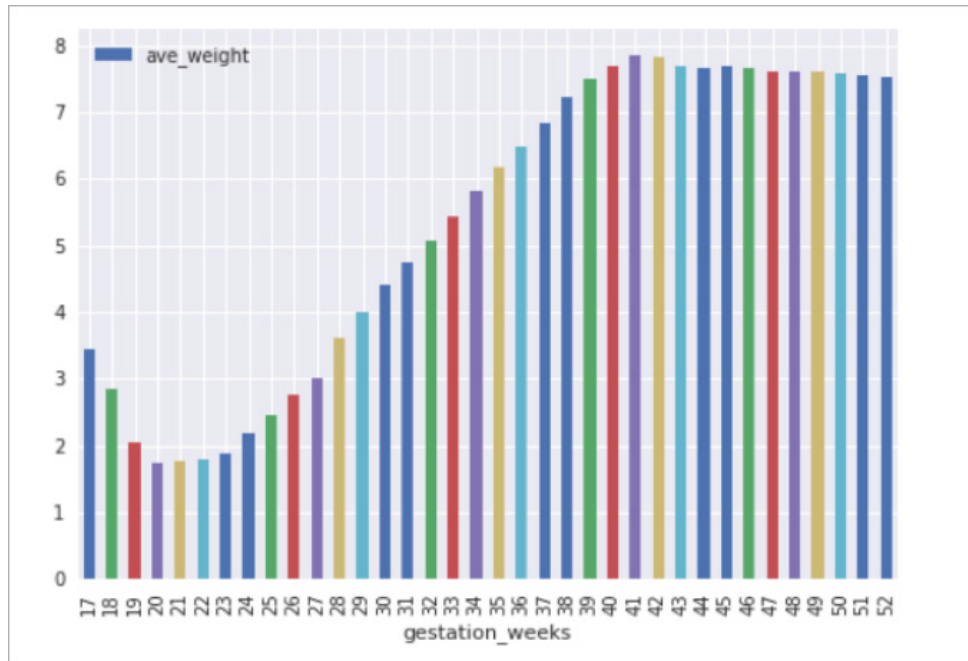
7) 마지막으로 임신 주차별 태아의 몸무게 평균 값을 구합니다. 다음의 쿼리를 입력하여 실행합니다.

```
sql = """ SELECT gestation_weeks, AVG(weight_pounds) AS ave_weight

FROM `bigquery-public-data.samples.natality`
WHERE NOT IS_NAN(gestation_weeks) AND gestation_weeks <> 99
GROUP BY gestation_weeks ORDER BY gestation_weeks
"""

df = client.query(sql).to_dataframe()
df.plot(x='gestation_weeks', y='ave_weight', kind='bar');
```

[그림 8] 임신 주차별 태아 몸무게





## 3장

# 구글 클라우드 플랫폼에 IoT Analytics Pipeline 구축하기



- Google Cloud IoT는 완전 관리형 통합 서비스로, 간편하고 안전하게 전 세계에 분산된 기기에서 IoT 데이터에 대한 연결, 관리, 내부 데이터 처리를 대규모로 진행할 수 있습니다.
- 또한, 이 데이터를 실시간으로 처리하고 분석 및 시각화하여 작업 변경사항을 적용하고 필요에 따라 조치할 수 있습니다.

The diagram illustrates the Google Cloud IoT Core architecture, divided into two main sections: **Data Analytics in Cloud** (light blue background) and **Data Usage** (light green background).

**Data Analytics in Cloud:**

- Devices** (represented by a smartphone icon) send **Data** to the **Cloud IoT Core** (highlighted with a red border).
- The **Cloud IoT Core** sends **Updated Config** back to the **Devices**.
- The **Cloud IoT Core** sends **Update Device Config** to **Cloud Functions** and **Cloud Dataflow**.
- Cloud Functions** and **Cloud Pub/Sub** are connected by a bidirectional arrow.
- Cloud Pub/Sub** sends data to **Cloud Dataflow**.
- Cloud Dataflow** sends data to **Cloud Bigtable**.
- Cloud Dataflow** also sends **Serving** data to **Cloud ML**.
- Cloud Bigtable** sends data to **BigQuery**.
- BigQuery** sends **Training** data to **Cloud ML**.

**Data Usage:**

- Cloud ML** sends data to **Cloud Datalab** and **Data Studio**.
- BigQuery** also sends data to **Cloud Datalab** and **Data Studio**.
- Cloud Datalab** and **Data Studio** both send data to **Analytics** (represented by a bar chart icon).

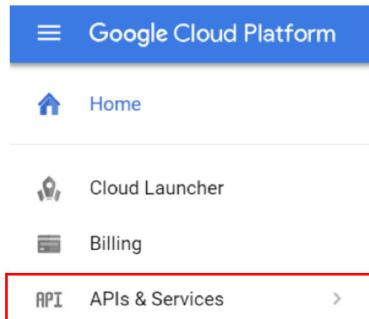
전 세계에 분산된 수백만 대의 장치에서 데이터를 쉽고 안전하게 연결, 관리 및 수집할 수 있는 완벽하게 관리되는 서비스입니다.

이 글에서는 GCP의 IoT 서비스에 대해 알아보고, 실습 과정을 통한 IoT Analytics Pipeline에 대해 설명합니다.

- Cloud IoT Core를 사용하여 MQTT 기반 장치를 연결하고 관리
- Cloud Pub/Sub를 사용하여 Cloud IoT Core의 정보 스트림을 수집
- Cloud DataFlow를 사용하여 IoT 데이터를 처리
- BigQuery를 사용하여 IoT 데이터를 분석

## 실습 절차

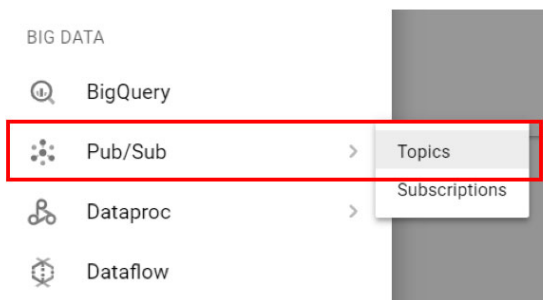
### 1. 수행에 필요한 API활성화 여부 확인



· GCP Console의 왼쪽 메뉴에서 APIs & Services 선택

· 아래의 3가지 서비스의 활성화 여부 확인  
(비 활성화인 경우 ENABLE API버튼을 클릭)

### 2. Cloud Pub/Sub Topic 만들기



· GCP Console의 왼쪽 메뉴에서 Pub/Sub > Topics 선택

#### Create a topic

A topic forwards messages from publishers to subscribers.

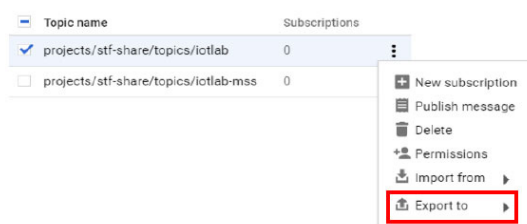
Name

projects/stf-share/topics/iotlab

CANCEL CREATE

· Create Topic을 선택한 후 Topic 이름을 입력

### 3. Topic 생성 후 계정 추가 및 권한 부여



· Pub/Sub > Pub/Sub Publisher 권한 부여

· cloud-iot@system.gserviceaccount.com

## 구글 클라우드 플랫폼에 IoT Analytics Pipeline 구축하기

### 4. BigQuery dataset 생성



· GCP Console의 왼쪽 메뉴에서  
BIG DATA > BigQuery 선택

· 새로운 Dataset 생성을 위해  
Create new dataset을 선택

· iotlab에서 "Add table" 아이콘을 클릭하면,  
Create Table 대화창을 확인 가능

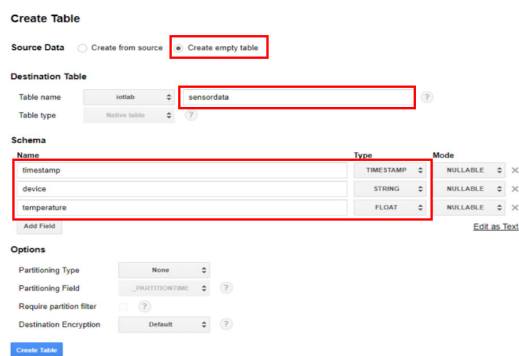


Table 생성

· Source Data : Create empty table

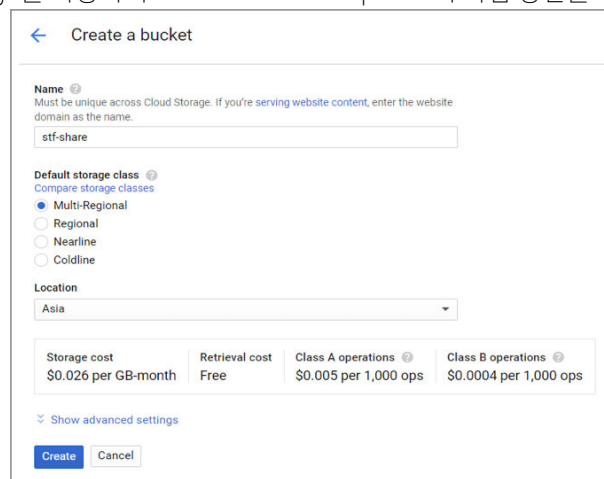
· Table name : sensordata

### 5. Cloud Storage Bucket 생성

Cloud Storage를 사용하면 언제든지 모든 양의 데이터를 전 세계에 저장하고 검색할 수 있습니다.

웹 사이트 콘텐츠 제공, 보관 및 재해 복구를 위한 데이터 저장, 직접 다운로드를 통한 대용량 데이터 객체 배포 등 다양한 시나리오에 Cloud Storage를 사용할 수 있습니다.

이 실습에서는 Cloud Storage를 사용하여 Cloud Dataflow Pipeline의 작업 공간을 제공합니다.



## 구글 클라우드 플랫폼에 IoT Analytics Pipeline 구축하기

### 6. Cloud Dataflow Pipeline 설정

Cloud Dataflow는 서버없이 데이터 분석을 수행 가능합니다.

이 실습에서는 Pub / Sub에서 센서 데이터를 읽고, 최대 온도를 계산하고 이를 BigQuery에 저장하도록 스트리밍 데이터 파이프 라인을 설정합니다.

- GCP Console의 왼쪽 메뉴에서  
BIG DATA > Dataflow 선택

### 7. Compute Engine VM 준비하기

```
sudo apt-get remove google-cloud-sdk -y
curl https://sdk.cloud.google.com | bash
gcloud init
```

```
gcloud components update
gcloud components install beta
sudo apt-get update
sudo apt-get install python-pip openssl git -y
sudo pip install pyjwt paho-mqtt cryptography
git clone http://github.com/GoogleCloudPlatform/training-data-analyst
```

### 8. IoT 장치들을 위한 Registry 생성

시뮬레이션 장치를 등록하기 위한 Registry를 생성해야 합니다.

Registry는 장치에 대한 제어 지점입니다.

## 구글 클라우드 플랫폼에 IoT Analytics Pipeline 구축하기

```
export PROJECT_ID=[ Your Project ID ]
export MY_REGION=[ asia-east1 ]
gcloud beta iot registries create iotlab-registry \
  --project=$PROJECT_ID \
  --region=$MY_REGION \
  --event-notification-config=topic=projects/$PROJECT_ID/topics/iotlab
```

### 9. 암호화 키 쌍 생성

IoT장치를 Cloud IoT Core에 안전하게 연결하기 위한 암호화 키 쌍을 만들어야 합니다.

```
cd $HOME/training-data-analyst/quests/iotlab/
openssl req -x509 -newkey rsa:2048 -keyout rsa_private.pem \
  -nodes -out rsa_cert.pem -subj "/CN=unused"
```

### 10. Registry에 시뮬레이션 장치 등록

장치를 cloud IoT Core에 등록하기 위해서는 반드시 Registry에 우선적으로 등록되어야 합니다.

아래 명령어를 통하여 두 대의 장치를 등록할 수 있습니다.

```
gcloud beta iot devices create temp-sensor-buenos-aires \
  --project=$PROJECT_ID \
  --region=$MY_REGION \
  --registry=iotlab-registry \
  --public-key path=rsa_cert.pem,type=rs256
```

```
gcloud beta iot devices create temp-sensor-istanbul \
  --project=$PROJECT_ID \
  --region=$MY_REGION \
  --registry=iotlab-registry \
  --public-key path=rsa_cert.pem,type=rs256
```

### 11. 시뮬레이션 장치 실행

시뮬레이션 장치에서 생성된 데이터는 Cloud IoT Core를 통해 Cloud Pub/Sub Topic으로 전송됩니다.

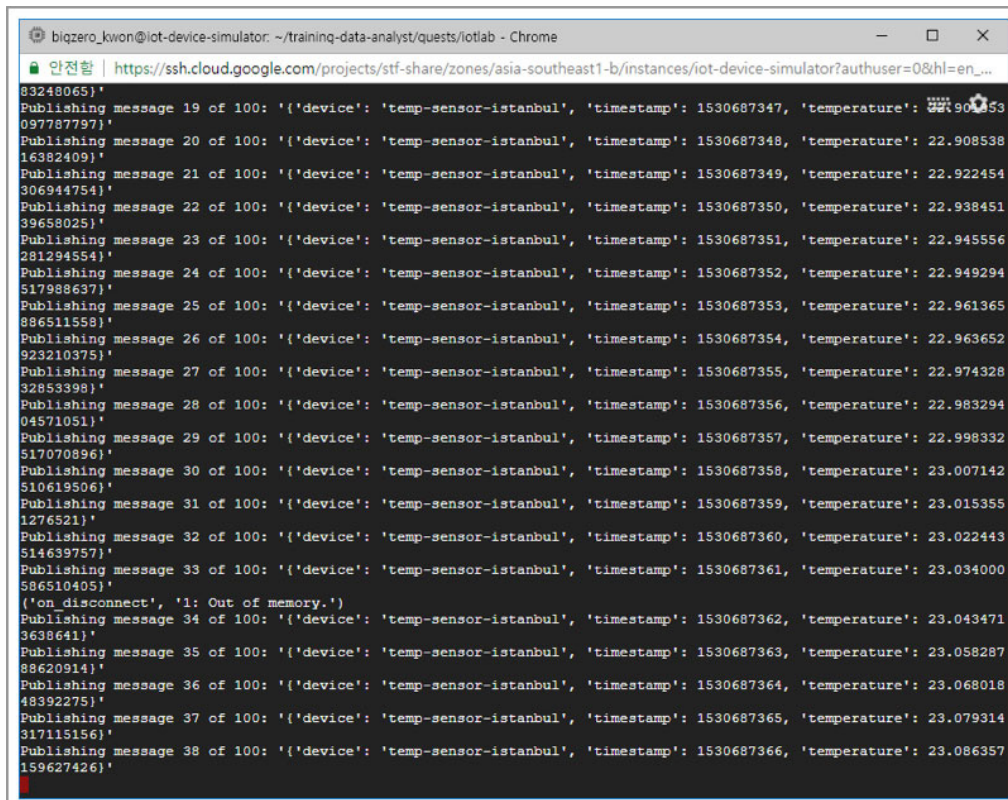
Dataflow job은 Topic으로부터 메시지를 읽어 BigQuery에 데이터를 저장합니다.

## 구글 클라우드 플랫폼에 IoT Analytics Pipeline 구축하기

```
cd $HOME/training-data-analyst/quests/iotlab/
wget https://pki.google.com/roots.pem
```

```
python clouddiot_mqtt_example_json.py W
-project_id=$PROJECT_ID W
-cloud_region=$MY_REGION W
-registry_id=iotlab-registry W
-device_id=temp-sensor-buenos-aires W
-private_key_file=rsa_private.pem W
-message_type=event W
-algorithm=RS256 > buenos-aires-log.txt 2>&1
```

```
python clouddiot_mqtt_example_json.py W
-project_id=$PROJECT_ID W
-cloud_region=$MY_REGION W
-registry_id=iotlab-registry W
-device_id=temp-sensor-istanbul W
-private_key_file=rsa_private.pem W
-message_type=event W
-algorithm=RS256
```



```
bigzero_kwon@iot-device-simulator: ~/training-data-analyst/quests/iotlab - Chrome
안전함 | https://ssh.cloud.google.com/projects/stf-share/zones/asia-southeast1-b/instances/iot-device-simulator?authuser=0&hl=en_...
83248065)'
Publishing message 19 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687347, 'temperature': 22.90853
0977877971}'
Publishing message 20 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687348, 'temperature': 22.908538
163824091}'
Publishing message 21 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687349, 'temperature': 22.922454
3069447541}'
Publishing message 22 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687350, 'temperature': 22.938451
39658025}'
Publishing message 23 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687351, 'temperature': 22.945556
281294554}'
Publishing message 24 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687352, 'temperature': 22.949294
517988637}'
Publishing message 25 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687353, 'temperature': 22.961365
886511558}'
Publishing message 26 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687354, 'temperature': 22.963652
923210375}'
Publishing message 27 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687355, 'temperature': 22.974328
32853398}'
Publishing message 28 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687356, 'temperature': 22.983294
04571051}'
Publishing message 29 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687357, 'temperature': 22.998332
517070896}'
Publishing message 30 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687358, 'temperature': 23.007142
510619506}'
Publishing message 31 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687359, 'temperature': 23.015355
1276521}'
Publishing message 32 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687360, 'temperature': 23.022443
514639757}'
Publishing message 33 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687361, 'temperature': 23.034000
586510405}'
('on_disconnect', '1: Out of memory.')
Publishing message 34 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687362, 'temperature': 23.043471
3638641}'
Publishing message 35 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687363, 'temperature': 23.058287
88620914}'
Publishing message 36 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687364, 'temperature': 23.068018
48392275}'
Publishing message 37 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687365, 'temperature': 23.079314
317115156}'
Publishing message 38 of 100: {'device': 'temp-sensor-istanbul', 'timestamp': 1530687366, 'temperature': 23.086357
159627426}'
```

## 12. BigQuery를 이용한 Sensor Data 분석

아래 쿼리를 이용하여 시뮬레이션 장치를 통해 입력된 센서 데이터 값을 확인 가능합니다.

```
#standardsql
```

```
SELECT timestamp, device, temperature from iotlab.sensordata
```

```
ORDER BY timestamp DESC
```

```
LIMIT 100
```

New Query ?

```

1 #standardsql
2 SELECT timestamp, device, temperature from iotlab.sensordata
3 ORDER BY timestamp DESC
4 LIMIT 100
5

```

RUN QUERY

Save Query

Save View

Format Query

Show Options

Query complete (2.2s elapsed, 3.36 KB processed)

Results

Details

Download as CSV

Row	timestamp	device	temperature
1	2078-10-30 08:09:03 UTC	ID14104437	2.0
2	2078-10-30 08:09:03 UTC	ID14104437	9999.0
3	2018-07-04 09:13:51 UTC	temp-sensor-istanbul	16.989718969595128
4	2018-07-04 09:13:50 UTC	temp-sensor-istanbul	16.998693455331075
5	2018-07-04 09:13:48 UTC	temp-sensor-istanbul	17.01341910151476



구글 클라우드 플랫폼(GCP)에 대해 더 알고 싶으세요?

지금 바로 베스핀글로벌 전문 컨설턴트에게 문의하세요. 베스핀글로벌의 GCP 전문 엔지니어가 답해드립니다.

▶ [GCP 서비스 바로가기](#)

▶ [Contact us](#)

베스핀글로벌에 대해 더 알고 싶다면 아래 링크를 클릭해주세요. 베스핀글로벌을 자세히 알려드립니다.

▶ [About us](#)

## “Helping You Adopt Cloud.”

클라우드 전문 인력 및 클라우드 관리 어떻게 시작해야 하나요?

클라우드 IT를 대한민국에서 가장 잘 아는 400여 명의 클라우드 전문가를 베스핀글로벌에서 만나실 수 있습니다.

[Meet 베스핀글로벌 전문 컨설팅팀](#)

클라우드의 도입을 고민하신다면 베스핀글로벌 컨설팅팀이 최적의 클라우드 컨설팅을 제공해 드립니다.

[Meet 아시아 최고의 클라우드 운영팀, 베스핀글로벌 매니지드 서비스팀](#)

클라우드 운영에 대한 노하우가 부족해 걱정이라면 클라우드 서비스의 베테랑이 여러분의 클라우드를 관리해드립니다.

[Meet OpsNow](#)

클라우드 비용을 50% 절감하고, 장애 대응 시간을 절반으로 줄일 수 있는 멀티-클라우드 관리 플랫폼 OpsNow를 만나보세요.

## About Bespin Global

[국내 최다 클라우드 인증 자격을 보유한 MSP](#)

[국내 유일 ISO 인증, ISMS 인증을 확보한 MSP](#)

[가트너가 인정한 한중일 유일한 MSP](#)

베스핀글로벌은 클라우드 IT를 위해 태어난 클라우드 매니지드 서비스 기업입니다. 클라우드 도입을 위한 전략컨설팅, 클라우드 상에서 수많은 고객의 IT 자산을 안정적으로 운영하고 관리할 수 있는 서비스와 솔루션, 클라우드를 기반한 머신러닝과 빅데이터와 같은 최신 기술의 빠른 도입까지 클라우드로 할 수 있는 전 영역을 End - to - End로 제공합니다. 2년 연속 가트너 매직 콰드런트 퍼블릭 클라우드 MSP 부문에 한·중·일 최초로 등재되었고, 포브스로부터 한국의 주목할 만한 유니콘으로 선정되기도 했습니다.

클라우드로 가기로 결정하였다면 누구와 함께 갈지를 선택해야 합니다.

처음부터 끝까지 믿을 만한 파트너를 찾는다면 베스핀글로벌이 정답입니다.

클라우드로 가기로 결정했다면  
누구와 함께 갈지를 선택해야 합니다.

처음부터 끝까지 믿을만한 파트너를 찾는다면  
베스핀글로벌이 정답입니다.

 [베스핀글로벌 웹사이트](#)

 [서비스 문의](#)

베스핀글로벌 소셜미디어



**BESPIN GLOBAL**