Google Cloud Software Development Kit (SDK) for .NET

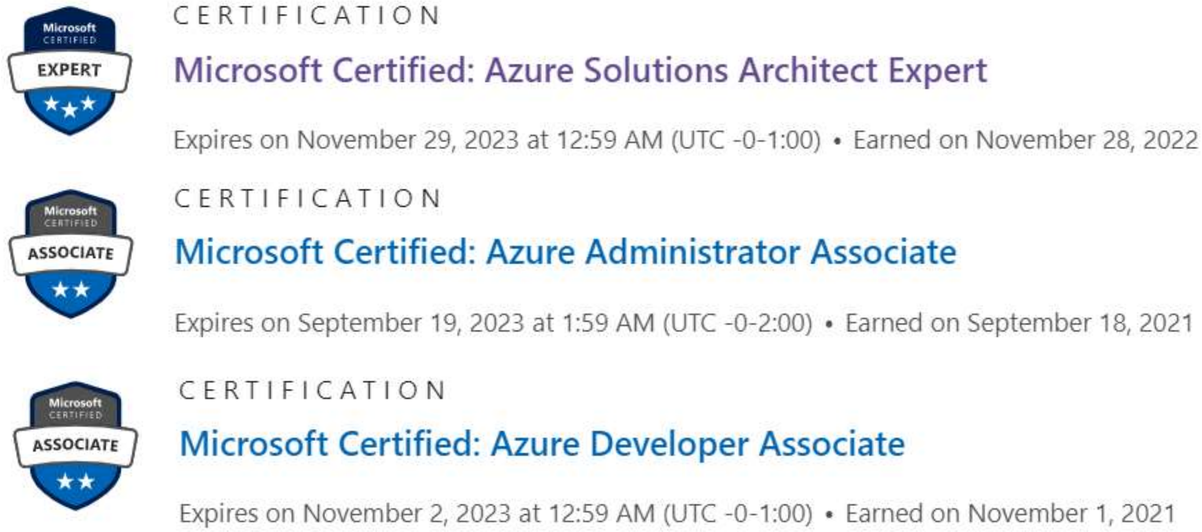
Author: Luis Coco Enríquez

Personal experience

Senior .Net Software Engineer

Microservices, SQL Server SSMS, SSRS, Oracle SQL Developer, ASP.Net WebForms, ASP.Net MVC5, ASP.Net Core 3.1, .Net 6

https://www.linkedin.com/in/luis-coco-enriquez-44a28a29/



CERTIFICATION

Microsoft Certified: Azure Solutions Architect Expert

Expires on November 29, 2023 at 12:59 AM (UTC -0-1:00) • Earned on November 28, 2022

CERTIFICATION

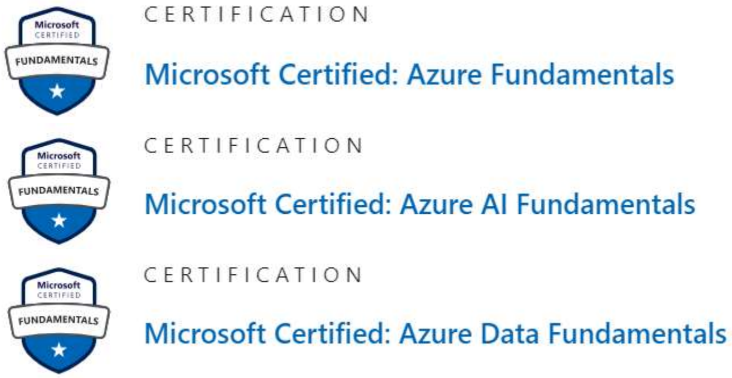
Microsoft Certified: Azure Administrator Associate

Expires on September 19, 2023 at 1:59 AM (UTC -0-2:00) • Earned on September 18, 2021

CERTIFICATION

Microsoft Certified: Azure Developer Associate

Expires on November 2, 2023 at 12:59 AM (UTC -0-1:00) • Earned on November 1, 2021



CERTIFICATION

Microsoft Certified: Azure Fundamentals

CERTIFICATION

Microsoft Certified: Azure AI Fundamentals

CERTIFICATION

Microsoft Certified: Azure Data Fundamentals

Agenda

Topic
What is Google Cloud SDK for .NET?
Key features and components of the Google Cloud SDK or .NET
Create a Google Cloud free tier account
Setting up a .NET development environment
GitHub official samples

Agenda

Topic
Google Cloud guided samples
Create a Bucket, a first simple sample
API reference
Samples in C#

Luxoft
A DXC Technology Company

What is Google Cloud SDK for .NET?

The Google Cloud Software Development Kit (SDK) for .NET is a **collection of libraries, tools, and APIs** that enable developers to integrate their .NET applications with various Google Cloud services.

By using the SDK, developers can leverage the power and scalability of Google Cloud services while writing their applications in .NET.

This opens up a wide range of possibilities for building robust and scalable cloud-native applications, leveraging Google's infrastructure and services.

https://cloud.google.com/dotnet

https://cloud.google.com/dotnet/docs/getting-started

https://cloud.google.com/dotnet/docs/reference

Key features and components

1. **Google Cloud Client Libraries**: provides idiomatic .NET client libraries for different Google Cloud services. These libraries abstract the low-level details of interacting with Google Cloud APIs.

2. **Cloud Tools for Visual Studio**: to manage Google Cloud resources, deploy applications to the cloud, and debug their applications directly within the IDE.

3. **Authentication and Authorization**: The SDK provides authentication and authorization mechanisms to securely access Google Cloud services.

4. **Deployment and Management**: It offers tools and APIs to package and deploy your applications, manage your cloud resources, and monitor their performance.

5. **Testing and Emulation**: for testing and emulating Google Cloud services locally, allowing you to develop and test your applications without incurring costs or relying on a live cloud environment. This helps streamline the development and testing process.

Create a Google Cloud free tier account

1. Create a **gmail account**. https://support.google.com/mail/answer/56256?hl=en

2. Set your **payment info** (credit card data, paypal,… or other).

3. Open an **incognito Chrome window** and login in Google Cloud.

4. **Login** in **Google Cloud**. https://console.cloud.google.com/

5. Activate the **free tier subscription**.

Setting up a .NET development environment

1. **Install** your **IDE**: **Visual Studio, VS Code** or **IntelliJ IDEA Community Edition**.
https://visualstudio.microsoft.com/
https://code.visualstudio.com/download
https://www.jetbrains.com/idea/download/?section=windows

2. **Create a Google Cloud project**. Create a Google Cloud project to run your apps. Google Cloud projects form the basis for creating, enabling, and using all Google Cloud services.

3. **Authentication**. Your .NET app must authenticate itself to use Google Cloud APIs. You use Application Default Credentials (ADC), which let you provide credentials for either local development or in a production environment.

For information about setting up ADC, see Provide credentials to Application Default Credentials. For general information about authentication, see Authentication at Google.

https://cloud.google.com/dotnet/docs/setup?hl=en

Set up Application Default Credentials (ADC)

https://cloud.google.com/docs/authentication/provide-credentials-adc#local-dev

1. **Install "gcloud CLI"**: https://cloud.google.com/sdk/docs/install?hl=en

2. **Create your credential file** (run the command):

```
gcloud auth application-default login
```

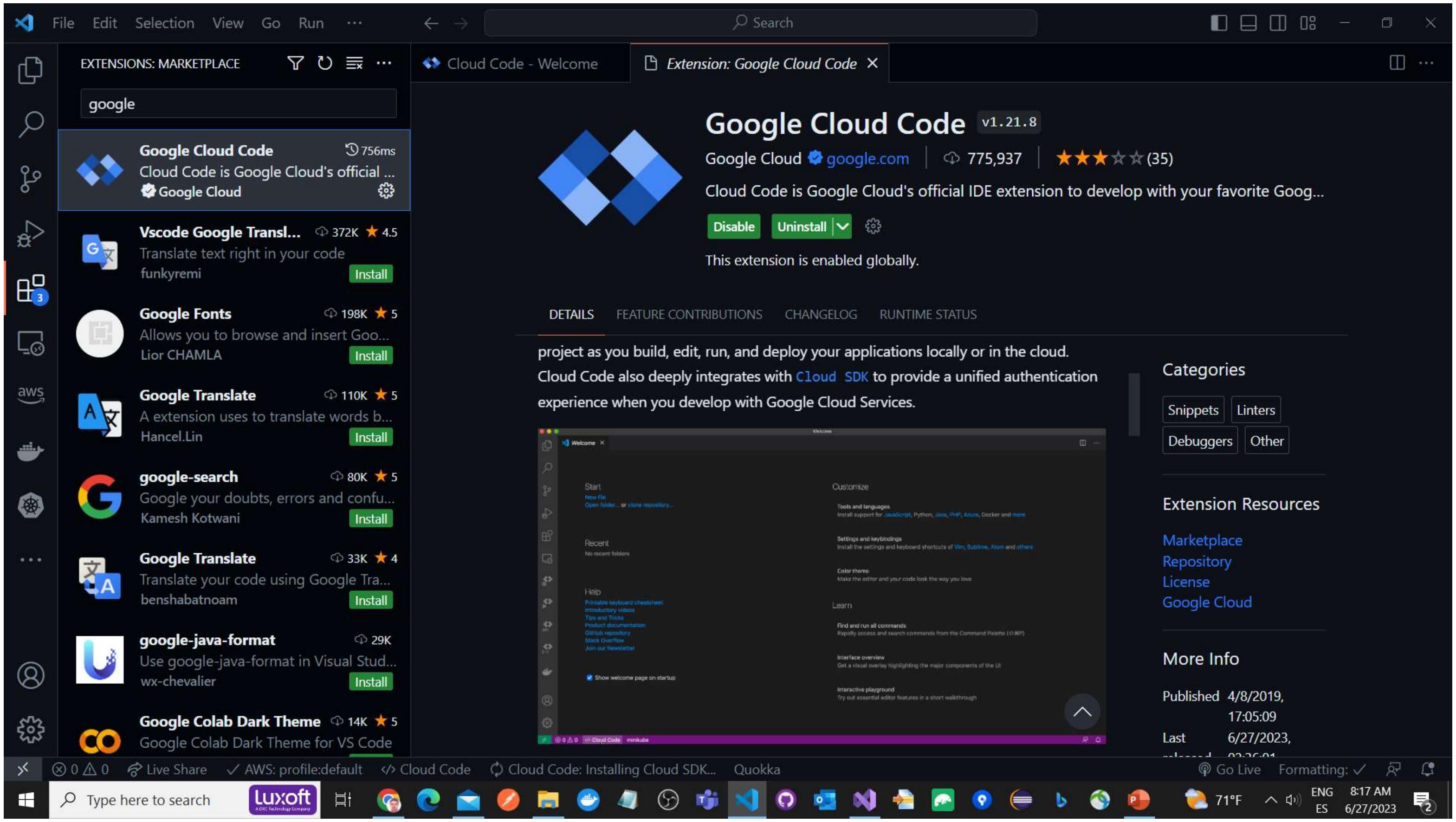
This method stores your credentials in a file on your file system. Any user with access to your file system can use those credentials. When you no longer need these credentials, you should revoke them:

```
gcloud auth application-default revoke
```

User credentials might not work for some methods and APIs, such as the *Cloud Translation API* or the *Cloud Vision API*, without extra parameters or configuration. See Troubleshooting your ADC setup.

https://cloud.google.com/dotnet/docs/setup?hl=en

Google Cloud Code for VSCode



Google Cloud Code 756ms
Cloud Code is Google Cloud's official ...
Google Cloud

Vscode Google Transl... 372K ★ 4.5
Translate text right in your code
funkyremi
Install

Google Fonts 198K ★ 5
Allows you to browse and insert Goo...
Lior CHAMLA
Install

Google Translate 110K ★ 5
A extension uses to translate words b...
Hancel.Lin
Install

google-search 80K ★ 5
Google your doubts, errors and confu...
Kamesh Kotwani
Install

Google Translate 33K ★ 4
Translate your code using Google Tra...
benshabatnoam
Install

google-java-format 29K
Use google-java-format in Visual Stud...
wx-chevalier
Install

Google Colab Dark Theme 14K ★ 5
Google Colab Dark Theme for VS Code

Google Cloud Code v1.21.8

Google Cloud google.com | 775,937 | ★★★☆☆ (35)

Cloud Code is Google Cloud's official IDE extension to develop with your favorite Goog...

This extension is enabled globally.

project as you build, edit, run, and deploy your applications locally or in the cloud. Cloud Code also deeply integrates with `Cloud SDK` to provide a unified authentication experience when you develop with Google Cloud Services.

Categories

Snippets | Linters | Debuggers | Other

Extension Resources

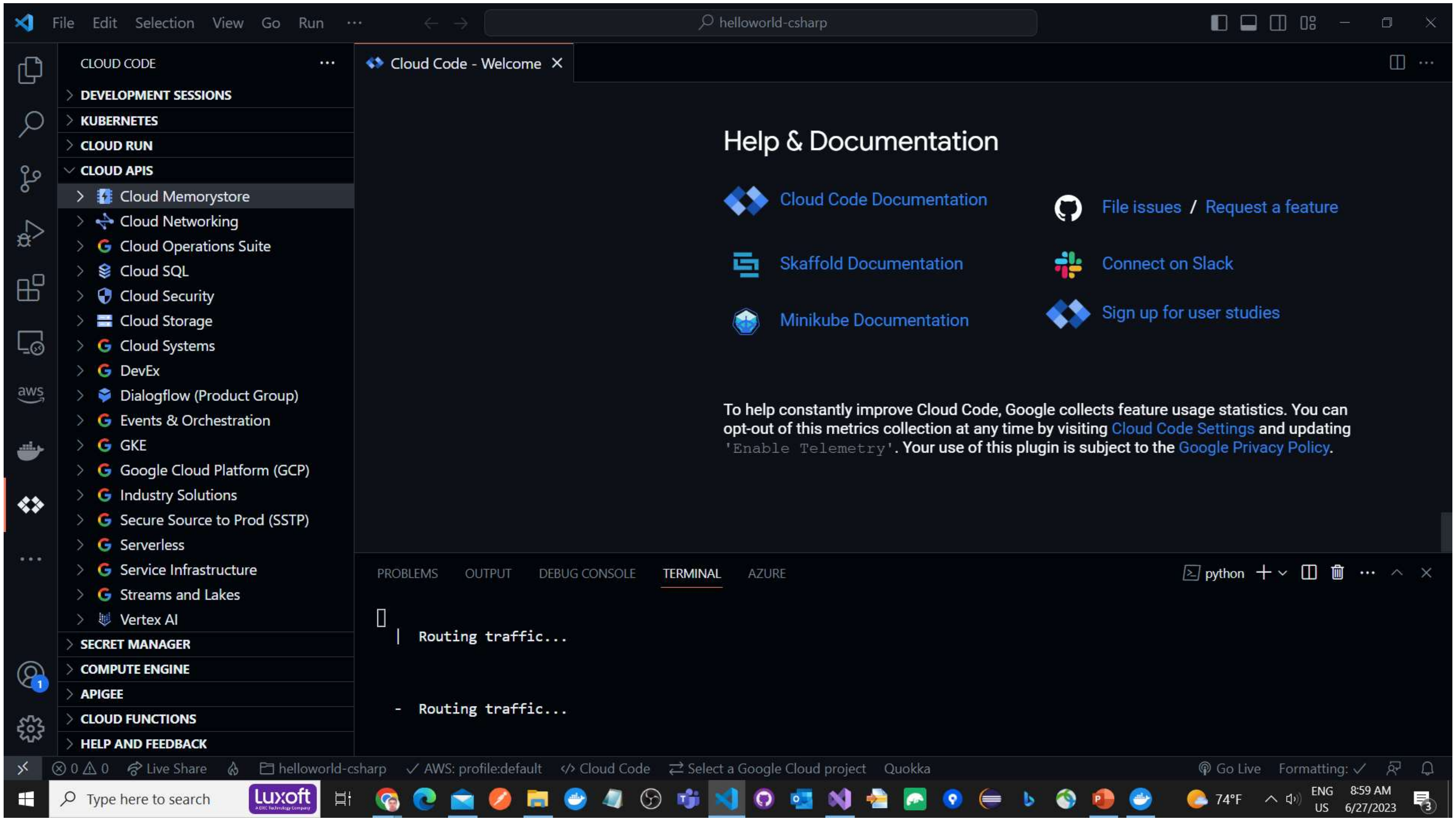
Marketplace
Repository
License
Google Cloud

More Info

Published 4/8/2019, 17:05:09
Last 6/27/2023,

© 2023 Luxoft, A DXC Technology Company. All rights reserved.

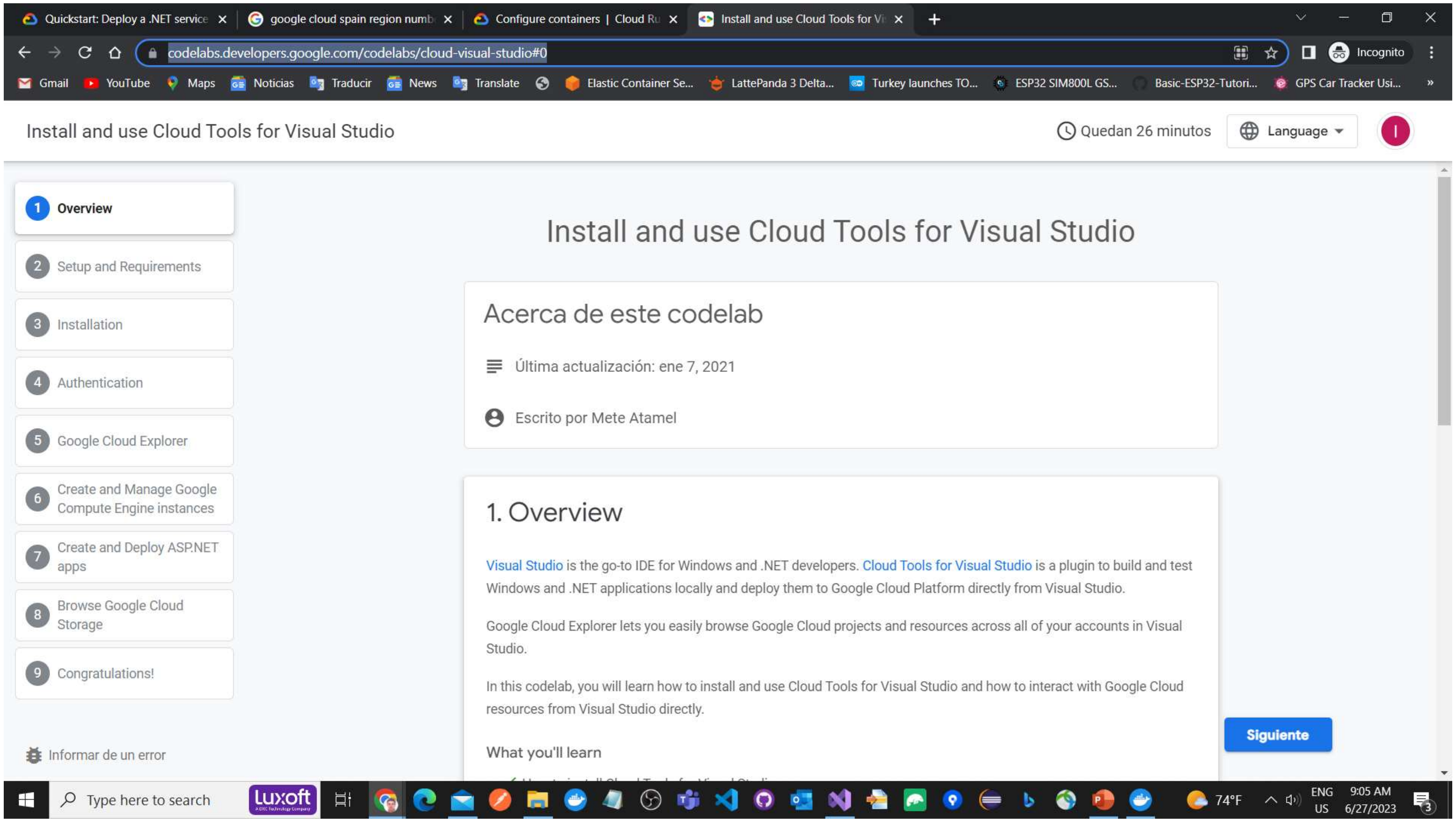
Google Cloud Code for VSCode



To help constantly improve Cloud Code, Google collects feature usage statistics. You can opt-out of this metrics collection at any time by visiting Cloud Code Settings and updating 'Enable Telemetry'. Your use of this plugin is subject to the Google Privacy Policy.

```
| Routing traffic...


- Routing traffic...
```

Google Cloud Code for Visual Studio 2022



Install and use Cloud Tools for Visual Studio

1 Overview
2 Setup and Requirements
3 Installation
4 Authentication
5 Google Cloud Explorer
6 Create and Manage Google Compute Engine instances
7 Create and Deploy ASP.NET apps
8 Browse Google Cloud Storage
9 Congratulations!

Install and use Cloud Tools for Visual Studio

Acerca de este codelab

Última actualización: ene 7, 2021

Escrito por Mete Atamel

1. Overview

Visual Studio is the go-to IDE for Windows and .NET developers. Cloud Tools for Visual Studio is a plugin to build and test Windows and .NET applications locally and deploy them to Google Cloud Platform directly from Visual Studio.

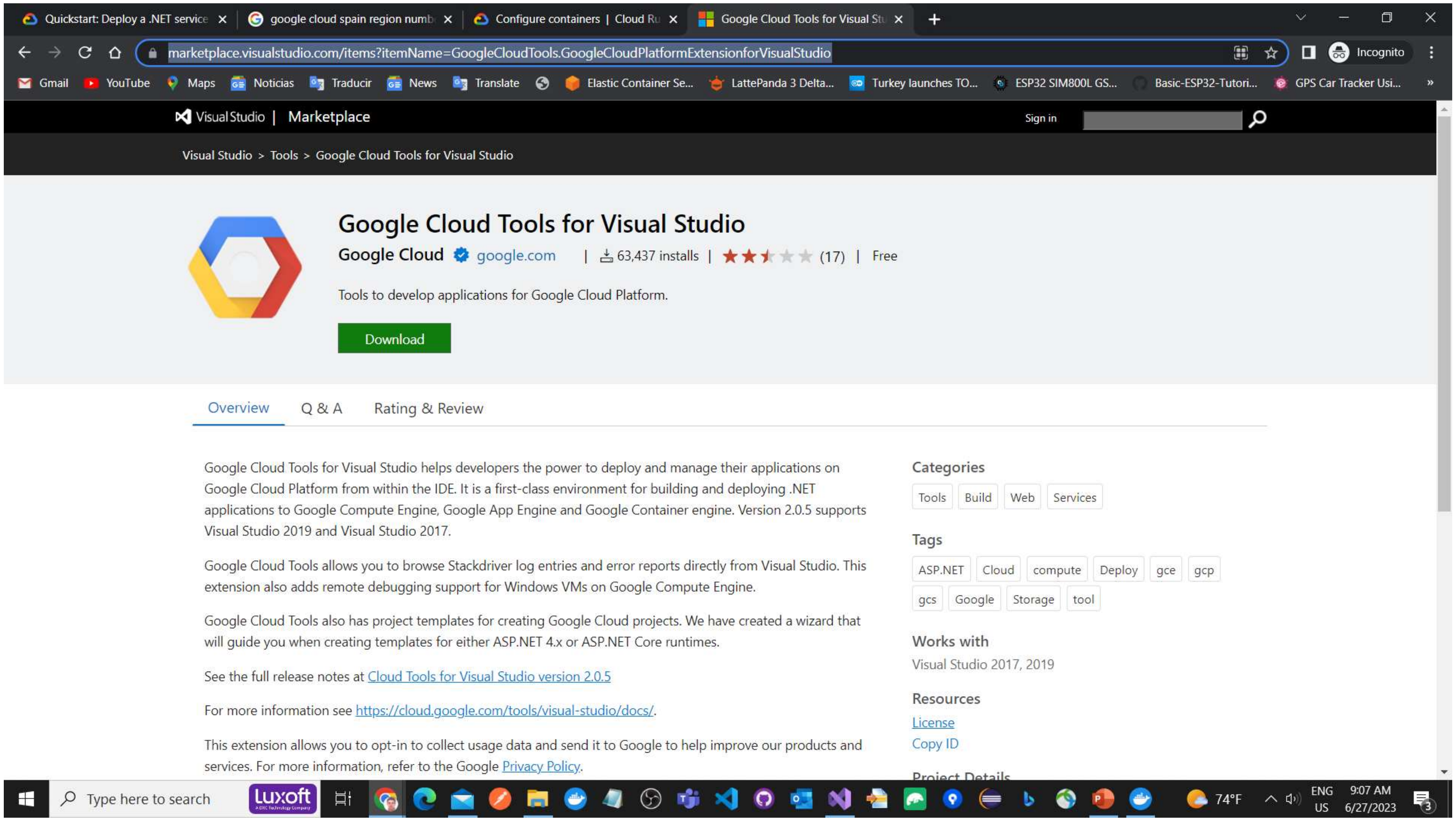
Google Cloud Explorer lets you easily browse Google Cloud projects and resources across all of your accounts in Visual Studio.

In this codelab, you will learn how to install and use Cloud Tools for Visual Studio and how to interact with Google Cloud resources from Visual Studio directly.

What you'll learn

https://codelabs.developers.google.com/codelabs/cloud-visual-studio#0

Google Cloud Code for Visual Studio 2022


Quickstart: Deploy a .NET service
google cloud spain region numb
Configure containers | Cloud Ru
Google Cloud Tools for Visual St
marketplace.visualstudio.com/items?itemName=GoogleCloudTools.GoogleCloudPlatformExtensionforVisualStudio
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
Visual Studio | Marketplace
Sign in
Visual Studio > Tools > Google Cloud Tools for Visual Studio
Download
Overview
Q & A
Rating & Review
Type here to search
luxoft
74°F
ENG
US
9:07 AM
6/27/2023

Google Cloud Tools for Visual Studio

Google Cloud | google.com | 63,437 installs | ★★★☆☆ (17) | Free

Tools to develop applications for Google Cloud Platform.

Google Cloud Tools for Visual Studio helps developers the power to deploy and manage their applications on Google Cloud Platform from within the IDE. It is a first-class environment for building and deploying .NET applications to Google Compute Engine, Google App Engine and Google Container engine. Version 2.0.5 supports Visual Studio 2019 and Visual Studio 2017.

Google Cloud Tools allows you to browse Stackdriver log entries and error reports directly from Visual Studio. This extension also adds remote debugging support for Windows VMs on Google Compute Engine.

Google Cloud Tools also has project templates for creating Google Cloud projects. We have created a wizard that will guide you when creating templates for either ASP.NET 4.x or ASP.NET Core runtimes.

See the full release notes at Cloud Tools for Visual Studio version 2.0.5

For more information see https://cloud.google.com/tools/visual-studio/docs/.

This extension allows you to opt-in to collect usage data and send it to Google to help improve our products and services. For more information, refer to the Google Privacy Policy.

Categories

Tools | Build | Web | Services

Tags

ASP.NET | Cloud | compute | Deploy | gce | gcp | gcs | Google | Storage | tool

Works with

Visual Studio 2017, 2019

Resources

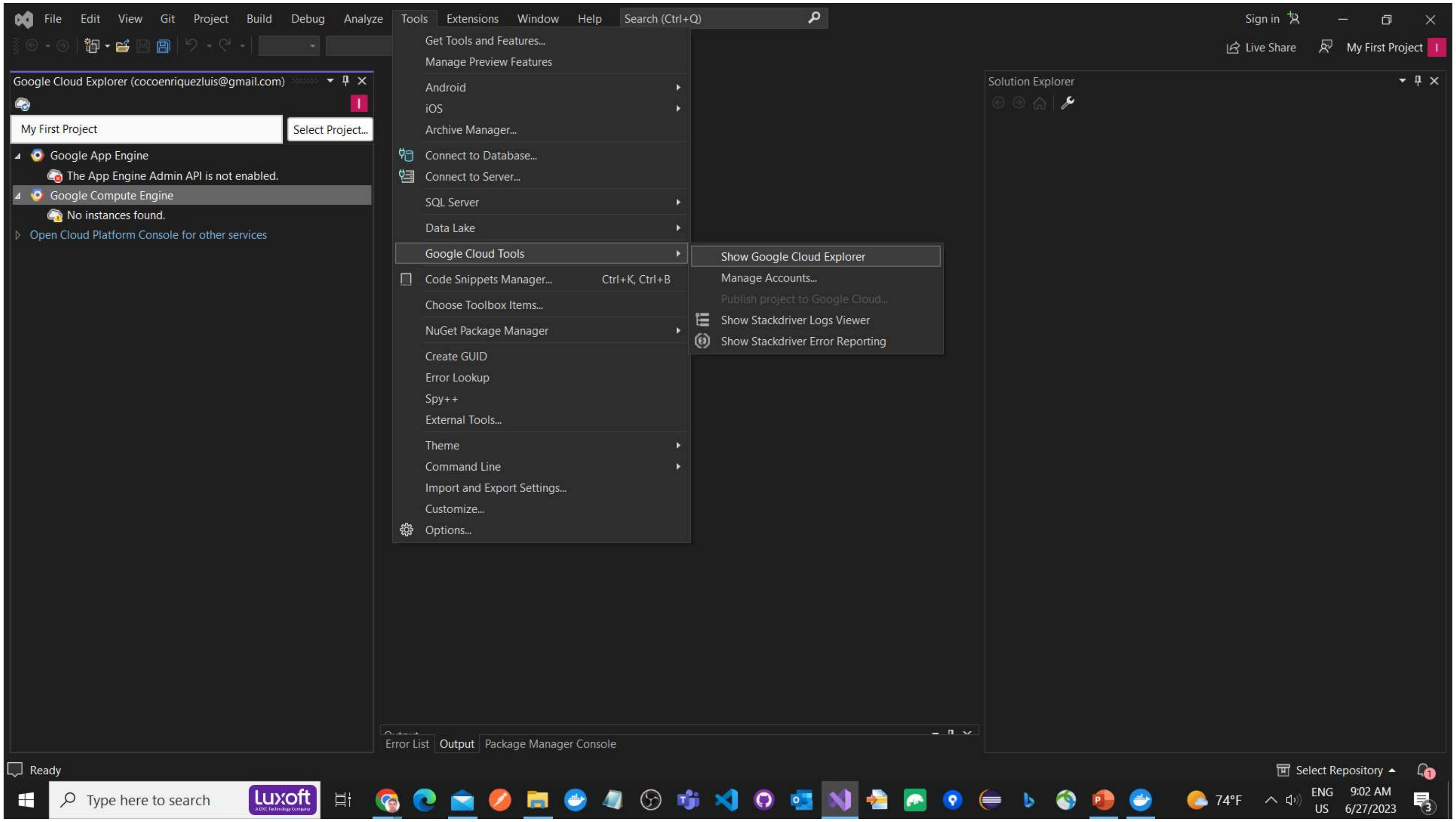
License

Copy ID

Project Details

https://marketplace.visualstudio.com/items?itemName=GoogleCloudTools.GoogleCloudPlatformExtensionforVisualStudio

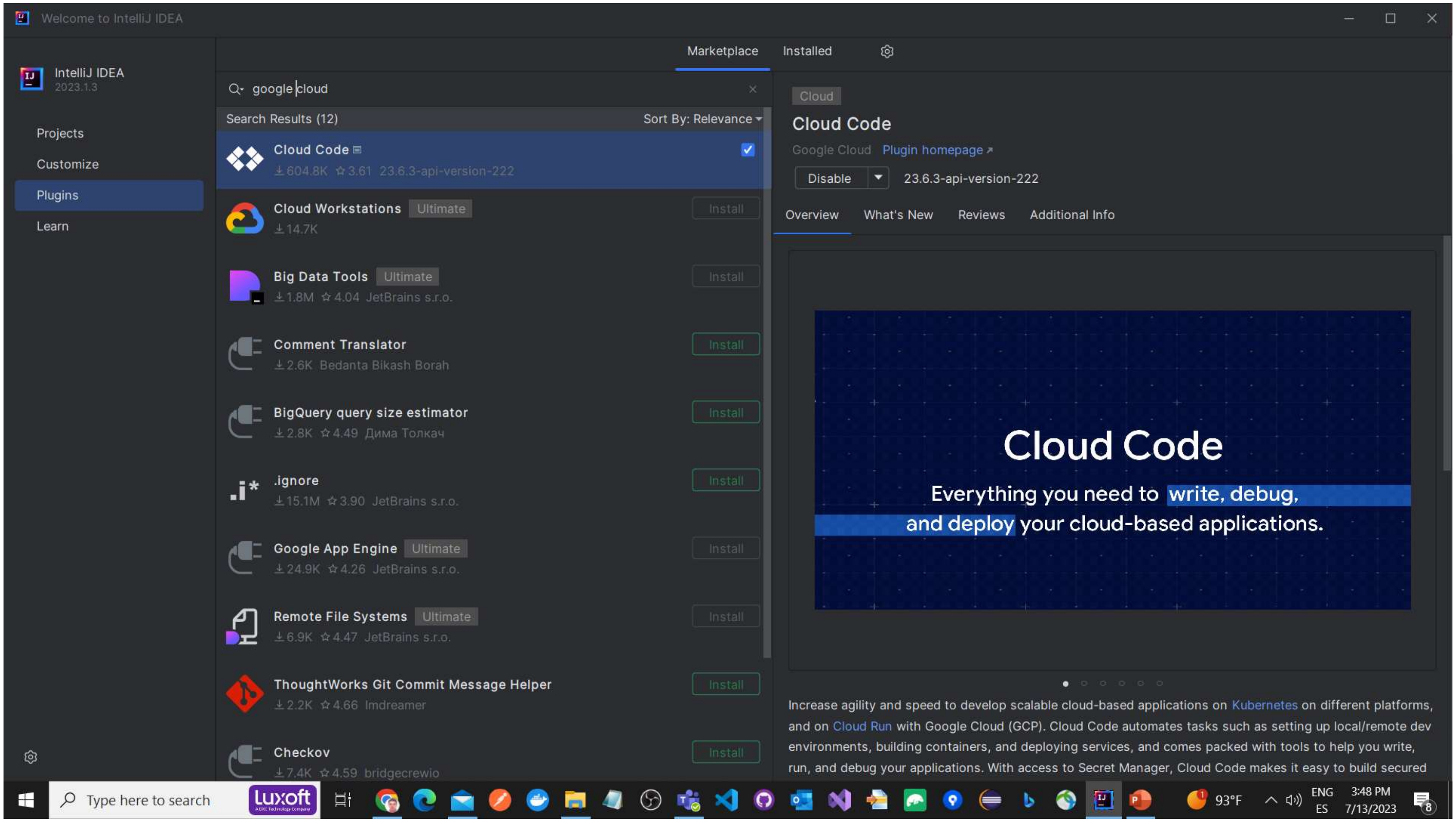
Google Cloud Code for Visual Studio 2022



The App Engine Admin API is not enabled.

No instances found.

IntelliJ IDEA Community Edition



google cloud

Search Results (12) Sort By: Relevance

Cloud Code ⤓604.8K ☆3.61 23.6.3-api-version-222 ☑

Cloud Workstations Ultimate ⤓14.7K Install

Big Data Tools Ultimate ⤓1.8M ☆4.04 JetBrains s.r.o. Install

Comment Translator ⤓2.6K Bedanta Bikash Borah Install

BigQuery query size estimator ⤓2.8K ☆4.49 Дима Толкач Install

.ignore ⤓15.1M ☆3.90 JetBrains s.r.o. Install

Google App Engine Ultimate ⤓24.9K ☆4.26 JetBrains s.r.o. Install

Remote File Systems Ultimate ⤓6.9K ☆4.47 JetBrains s.r.o. Install

ThoughtWorks Git Commit Message Helper ⤓2.2K ☆4.66 lmdreamer Install

Checkov ⤓7.4K ☆4.59 bridgecrewio Install

Cloud

Cloud Code

Google Cloud Plugin homepage ↗

Disable ▾ 23.6.3-api-version-222

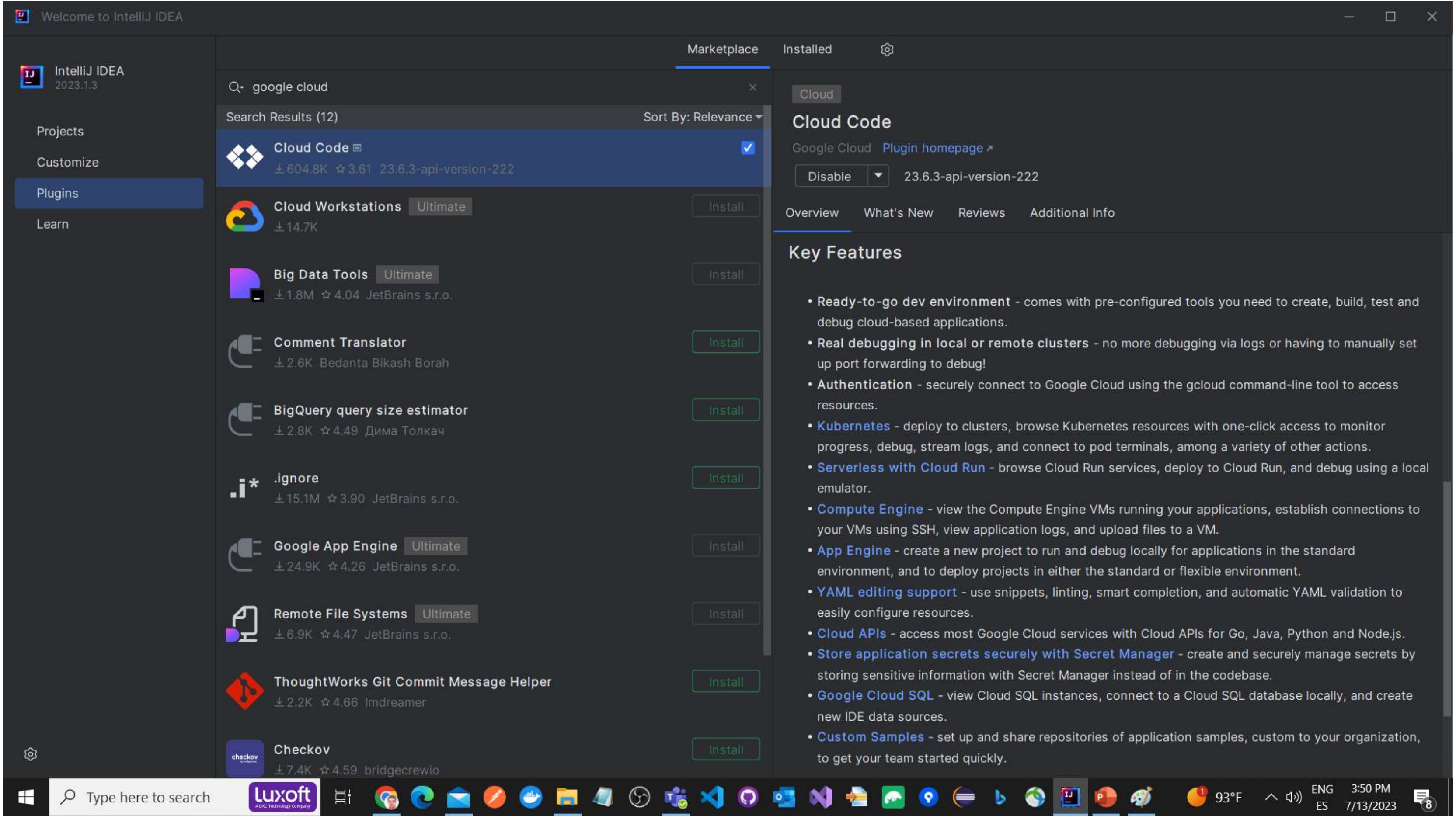
Overview What's New Reviews Additional Info

Cloud Code

Everything you need to write, debug, and deploy your cloud-based applications.

Increase agility and speed to develop scalable cloud-based applications on Kubernetes on different platforms, and on Cloud Run with Google Cloud (GCP). Cloud Code automates tasks such as setting up local/remote dev environments, building containers, and deploying services, and comes packed with tools to help you write, run, and debug your applications. With access to Secret Manager, Cloud Code makes it easy to build secured

IntelliJ IDEA Community Edition



google cloud

Search Results (12) Sort By: Relevance

Cloud Code ☑
⤓604.8K ☆3.61 23.6.3-api-version-222

Cloud Workstations Ultimate Install
⤓14.7K

Big Data Tools Ultimate Install
⤓1.8M ☆4.04 JetBrains s.r.o.

Comment Translator Install
⤓2.6K Bedanta Bikash Borah

BigQuery query size estimator Install
⤓2.8K ☆4.49 Дима Толкач

.ignore Install
⤓15.1M ☆3.90 JetBrains s.r.o.

Google App Engine Ultimate Install
⤓24.9K ☆4.26 JetBrains s.r.o.

Remote File Systems Ultimate Install
⤓6.9K ☆4.47 JetBrains s.r.o.

ThoughtWorks Git Commit Message Helper Install
⤓2.2K ☆4.66 lmdreamer

Checkov Install
⤓7.4K ☆4.59 bridgecrewio

Cloud

Cloud Code

Google Cloud Plugin homepage ↗

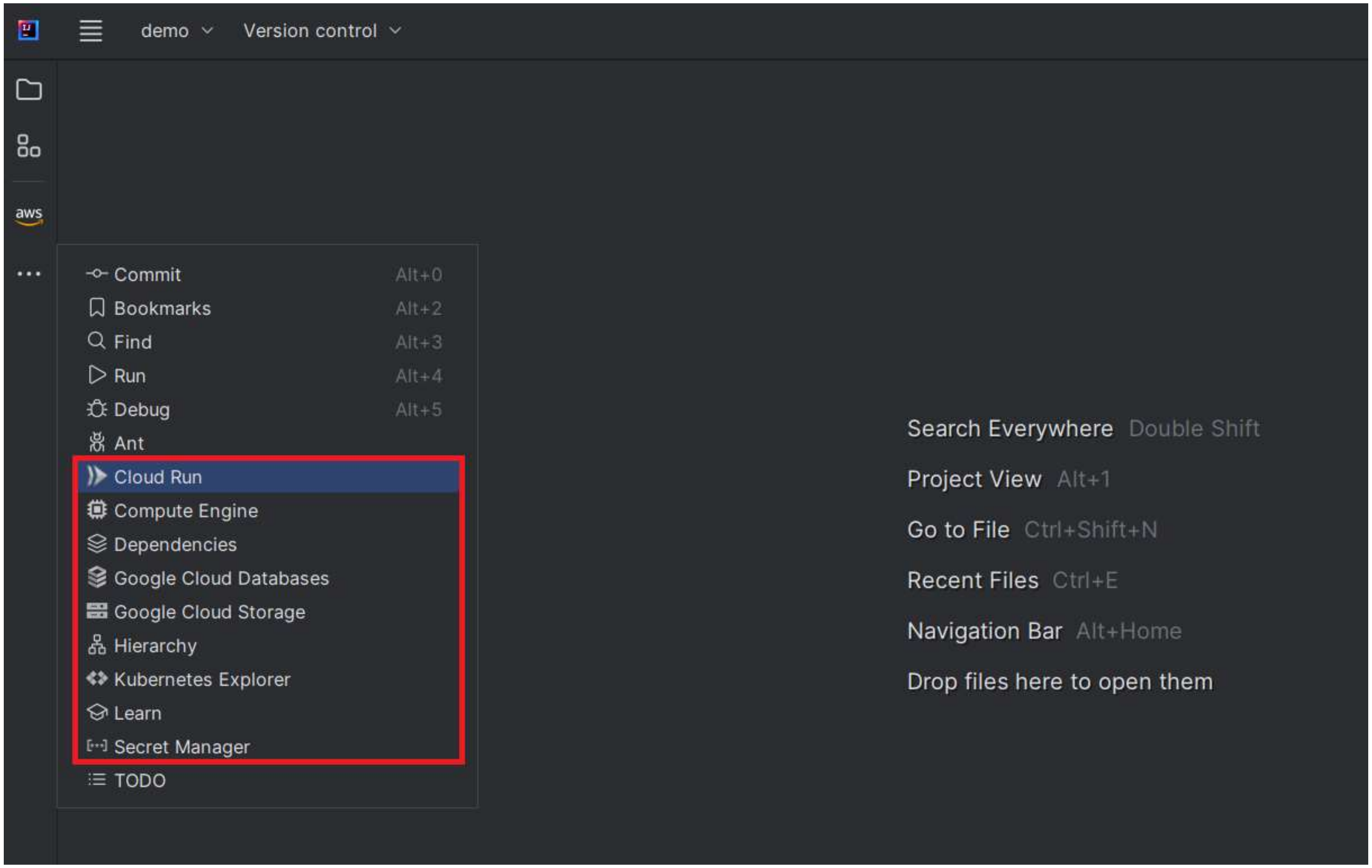
Disable ▾ 23.6.3-api-version-222

Overview What's New Reviews Additional Info

Key Features

- **Ready-to-go dev environment** - comes with pre-configured tools you need to create, build, test and debug cloud-based applications.
- **Real debugging in local or remote clusters** - no more debugging via logs or having to manually set up port forwarding to debug!
- **Authentication** - securely connect to Google Cloud using the gcloud command-line tool to access resources.
- Kubernetes - deploy to clusters, browse Kubernetes resources with one-click access to monitor progress, debug, stream logs, and connect to pod terminals, among a variety of other actions.
- Serverless with Cloud Run - browse Cloud Run services, deploy to Cloud Run, and debug using a local emulator.
- Compute Engine - view the Compute Engine VMs running your applications, establish connections to your VMs using SSH, view application logs, and upload files to a VM.
- App Engine - create a new project to run and debug locally for applications in the standard environment, and to deploy projects in either the standard or flexible environment.
- YAML editing support - use snippets, linting, smart completion, and automatic YAML validation to easily configure resources.
- Cloud APIs - access most Google Cloud services with Cloud APIs for Go, Java, Python and Node.js.
- Store application secrets securely with Secret Manager - create and securely manage secrets by storing sensitive information with Secret Manager instead of in the codebase.
- Google Cloud SQL - view Cloud SQL instances, connect to a Cloud SQL database locally, and create new IDE data sources.
- Custom Samples - set up and share repositories of application samples, custom to your organization, to get your team started quickly.

IntelliJ IDEA Community Edition



Search Everywhere Double Shift

Project View Alt+1

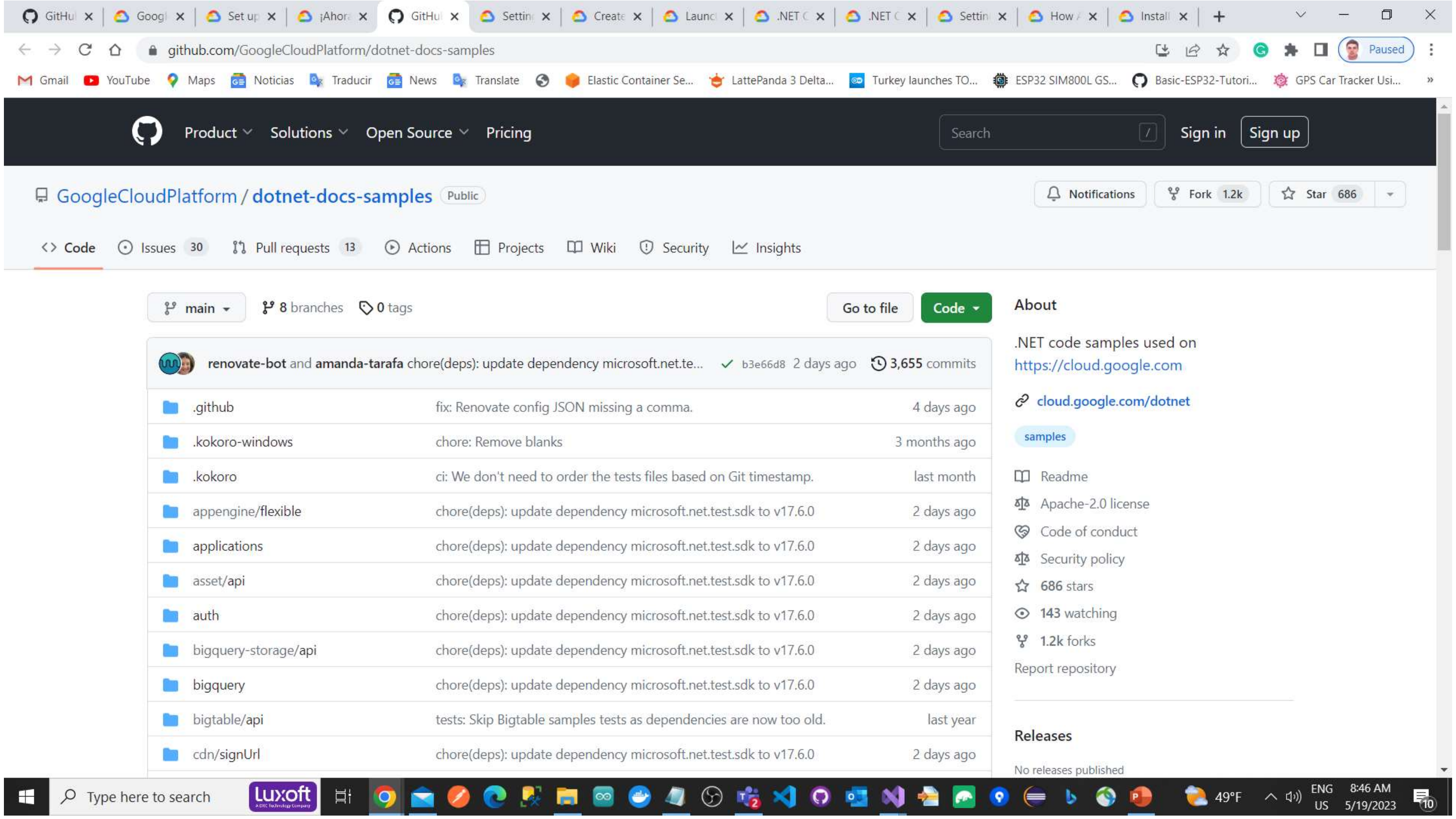
Go to File Ctrl+Shift+N

Recent Files Ctrl+E

Navigation Bar Alt+Home

Drop files here to open them

GitHub official samples



GoogleCloudPlatform / dotnet-docs-samples Public

renovate-bot and amanda-tarafa chore(deps): update dependency microsoft.net.te... ✓ b3e66d8 2 days ago 3,655 commits

.github	fix: Renovate config JSON missing a comma.	4 days ago
.kokoro-windows	chore: Remove blanks	3 months ago
.kokoro	ci: We don't need to order the tests files based on Git timestamp.	last month
appengine/flexible	chore(deps): update dependency microsoft.net.test.sdk to v17.6.0	2 days ago
applications	chore(deps): update dependency microsoft.net.test.sdk to v17.6.0	2 days ago
asset/api	chore(deps): update dependency microsoft.net.test.sdk to v17.6.0	2 days ago
auth	chore(deps): update dependency microsoft.net.test.sdk to v17.6.0	2 days ago
bigquery-storage/api	chore(deps): update dependency microsoft.net.test.sdk to v17.6.0	2 days ago
bigquery	chore(deps): update dependency microsoft.net.test.sdk to v17.6.0	2 days ago
bigtable/api	tests: Skip Bigtable samples tests as dependencies are now too old.	last year
cdn/signUrl	chore(deps): update dependency microsoft.net.test.sdk to v17.6.0	2 days ago

About

.NET code samples used on https://cloud.google.com

cloud.google.com/dotnet

samples

- Readme
- Apache-2.0 license
- Code of conduct
- Security policy
- 686 stars
- 143 watching
- 1.2k forks

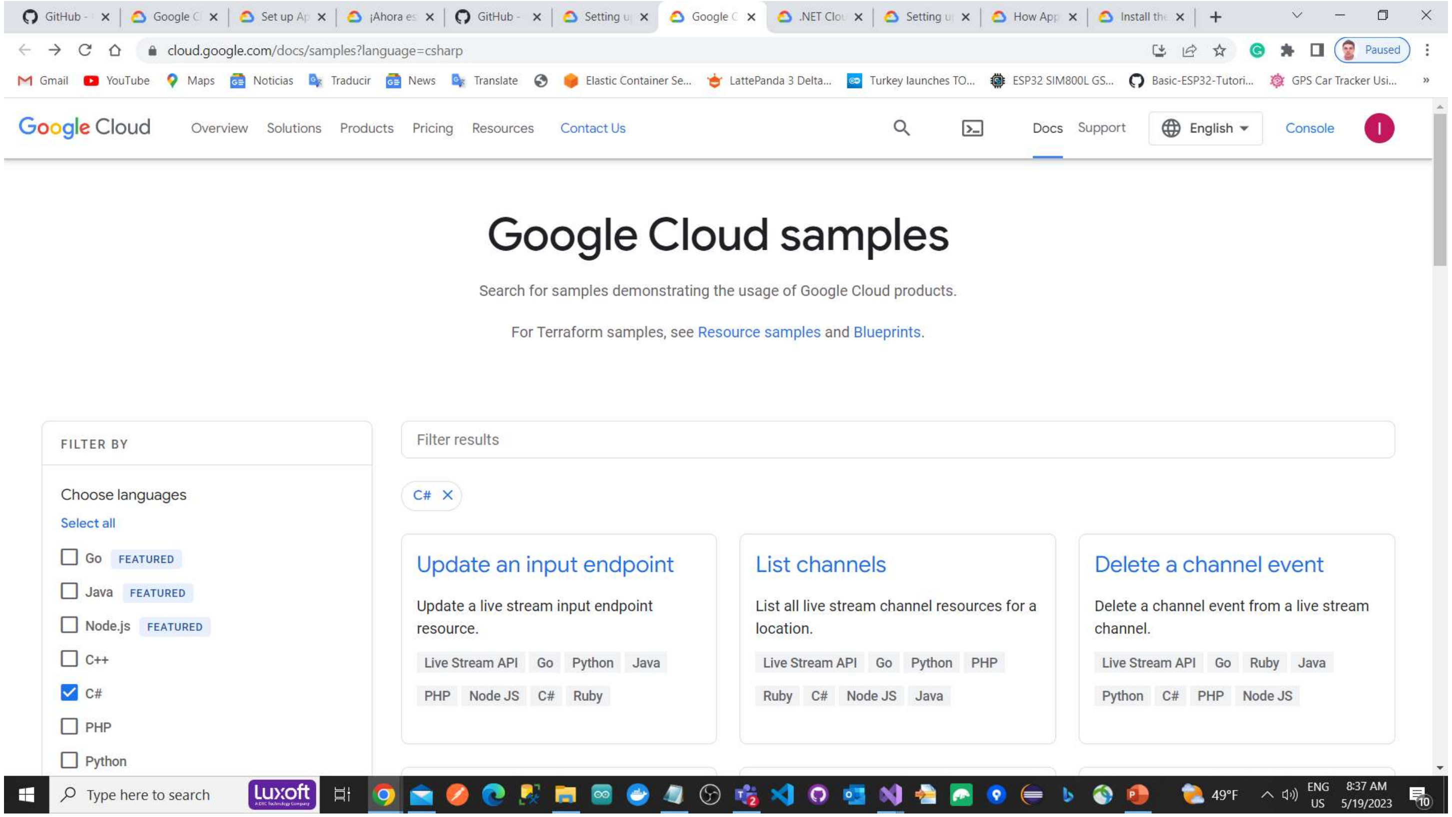
Report repository

Releases

No releases published

https://github.com/GoogleCloudPlatform/dotnet-docs-samples

Google Cloud guided samples



Google Cloud samples

Search for samples demonstrating the usage of Google Cloud products.

For Terraform samples, see Resource samples and Blueprints.

FILTER BY

Choose languages

Select all

- ☐ Go FEATURED
- ☐ Java FEATURED
- ☐ Node.js FEATURED
- ☐ C++
- ☑ C#
- ☐ PHP
- ☐ Python

Filter results

C# ✕

Update an input endpoint

Update a live stream input endpoint resource.

Live Stream API · Go · Python · Java · PHP · Node JS · C# · Ruby

List channels

List all live stream channel resources for a location.

Live Stream API · Go · Python · PHP · Ruby · C# · Node JS · Java

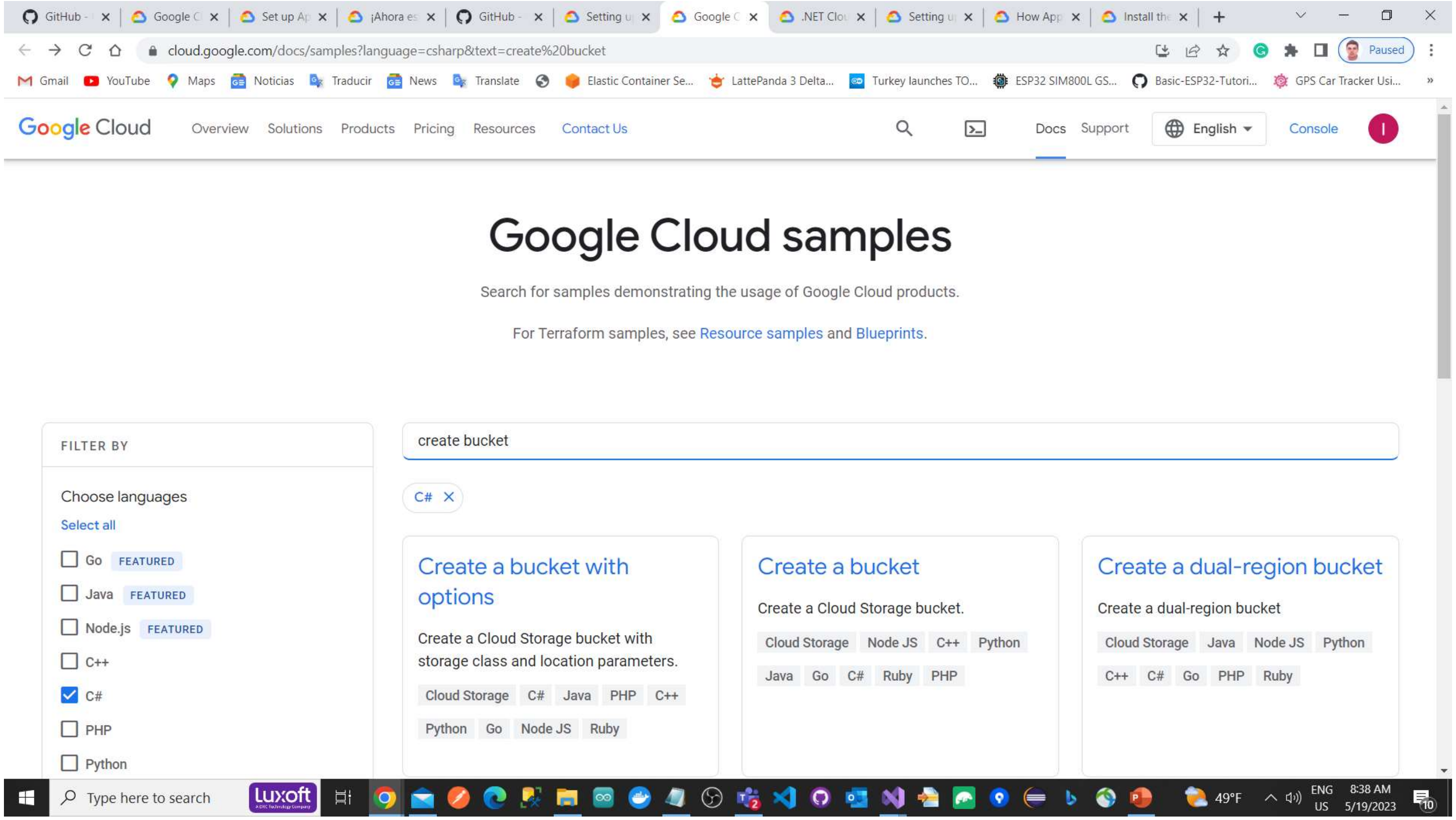
Delete a channel event

Delete a channel event from a live stream channel.

Live Stream API · Go · Ruby · Java · Python · C# · PHP · Node JS

https://cloud.google.com/docs/samples?language=csharp

Create a Bucket, a first simple sample



Google Cloud samples

Search for samples demonstrating the usage of Google Cloud products.

For Terraform samples, see Resource samples and Blueprints.

FILTER BY

Choose languages

Select all

- □ Go FEATURED
- □ Java FEATURED
- □ Node.js FEATURED
- □ C++
- ☑ C#
- □ PHP
- □ Python

create bucket

C# ✕

Create a bucket with options

Create a Cloud Storage bucket with storage class and location parameters.

Cloud Storage · C# · Java · PHP · C++ · Python · Go · Node JS · Ruby

Create a bucket

Create a Cloud Storage bucket.

Cloud Storage · Node JS · C++ · Python · Java · Go · C# · Ruby · PHP

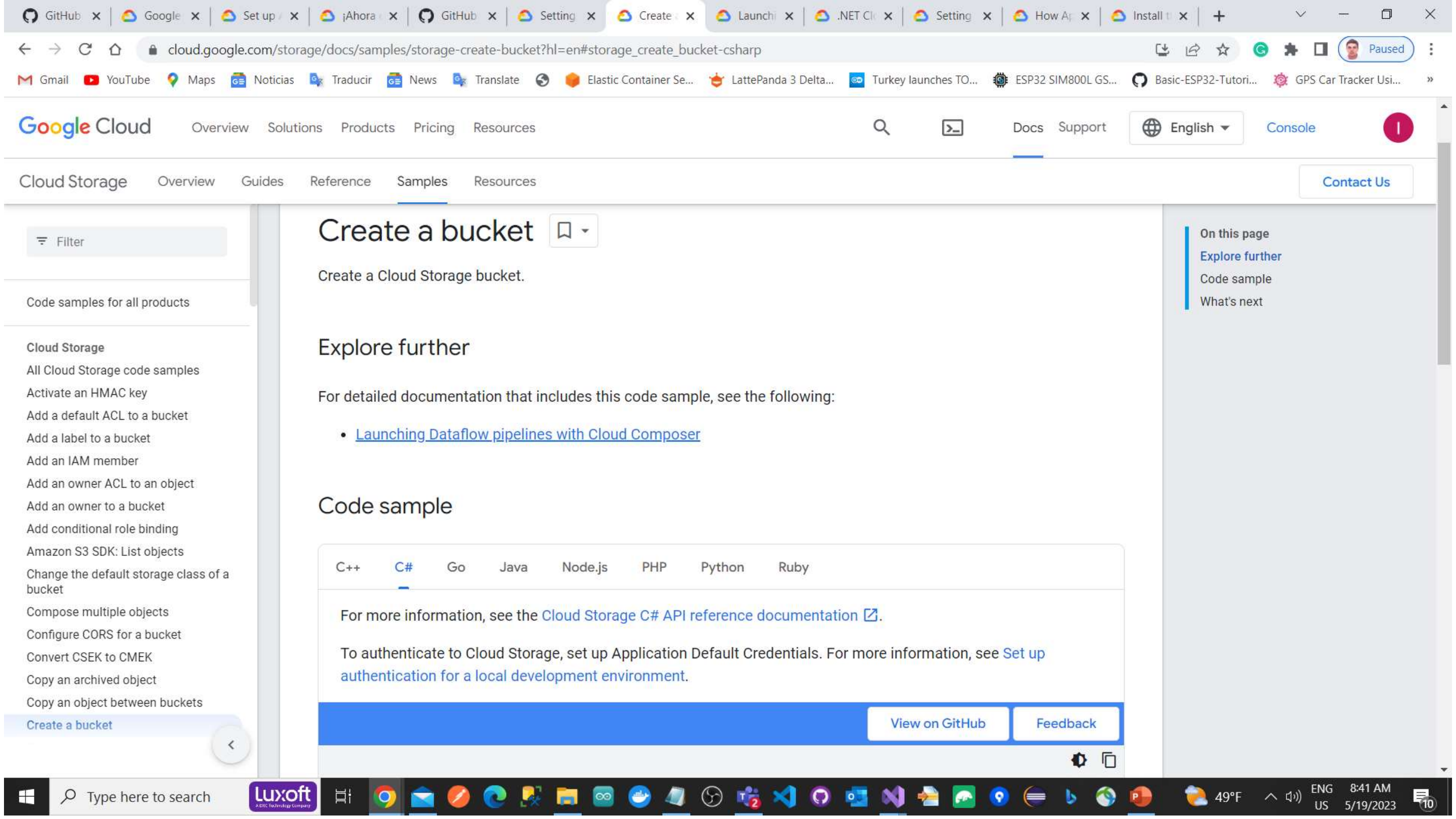
Create a dual-region bucket

Create a dual-region bucket

Cloud Storage · Java · Node JS · Python · C++ · C# · Go · PHP · Ruby

https://cloud.google.com/docs/samples?language=csharp&text=create%20bucket

Create a Bucket, a first simple sample



Create a bucket

Create a Cloud Storage bucket.

Explore further

For detailed documentation that includes this code sample, see the following:

- Launching Dataflow pipelines with Cloud Composer

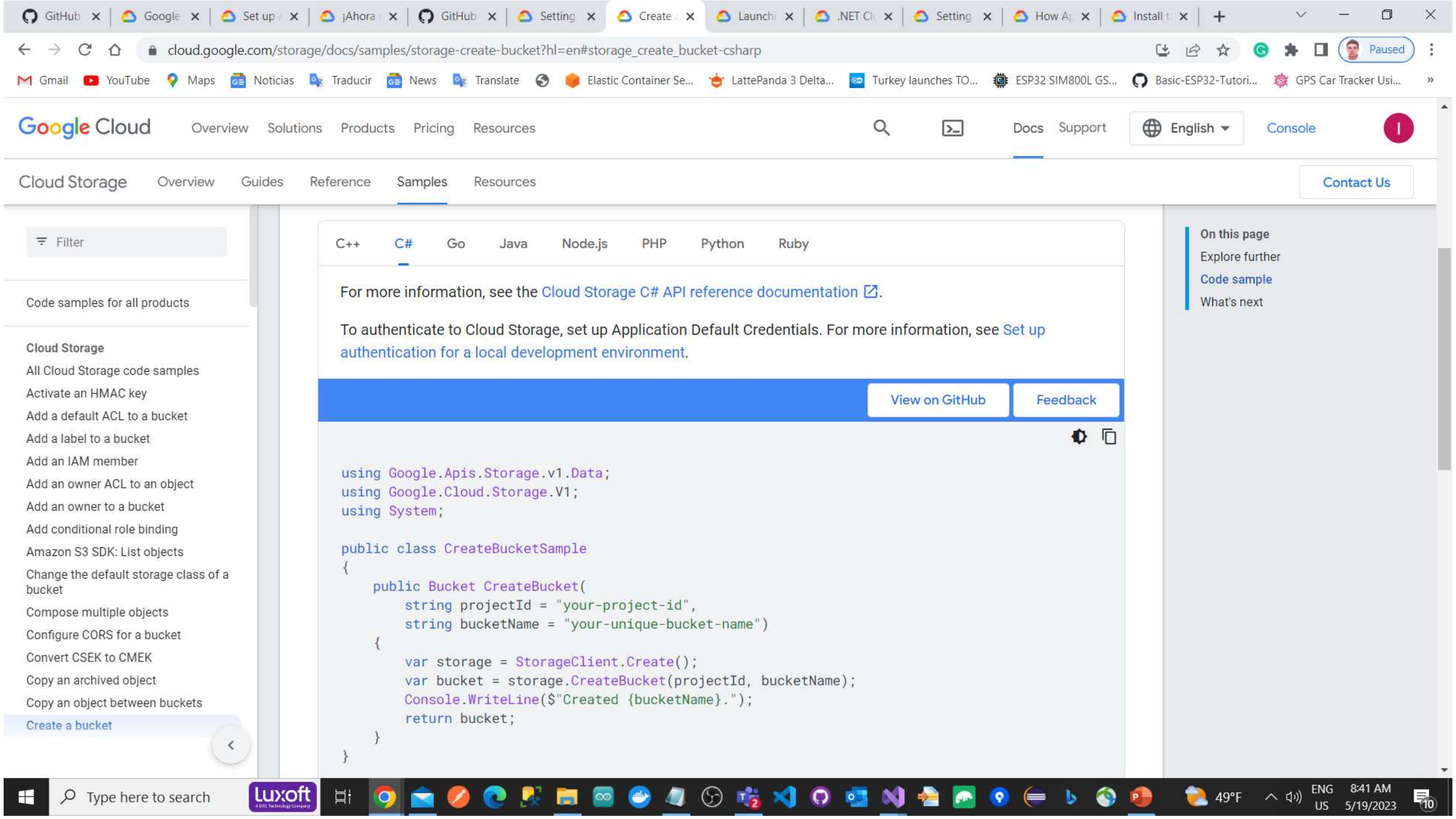
Code sample

For more information, see the Cloud Storage C# API reference documentation.

To authenticate to Cloud Storage, set up Application Default Credentials. For more information, see Set up authentication for a local development environment.

https://cloud.google.com/storage/docs/samples/storage-create-bucket?hl=en#storage_create_bucket-csharp

Create a Bucket, a first simple sample

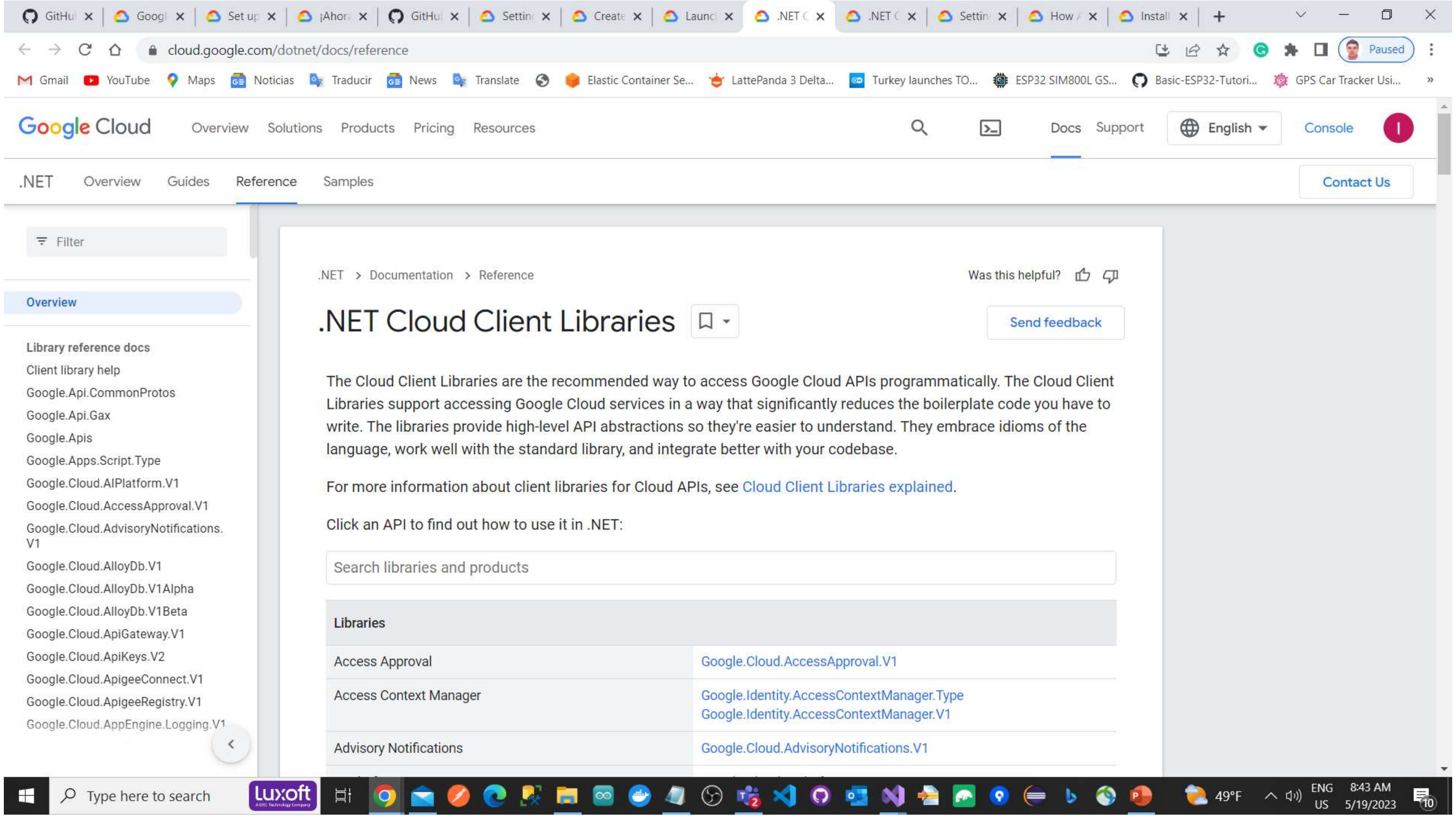


For more information, see the Cloud Storage C# API reference documentation.

To authenticate to Cloud Storage, set up Application Default Credentials. For more information, see Set up authentication for a local development environment.

```
using Google.Apis.Storage.v1.Data;
using Google.Cloud.Storage.V1;
using System;

public class CreateBucketSample
{
    public Bucket CreateBucket(
        string projectId = "your-project-id",
        string bucketName = "your-unique-bucket-name")
    {
        var storage = StorageClient.Create();
        var bucket = storage.CreateBucket(projectId, bucketName);
        Console.WriteLine($"Created {bucketName}.");
        return bucket;
    }
}
```

API reference



.NET Cloud Client Libraries

The Cloud Client Libraries are the recommended way to access Google Cloud APIs programmatically. The Cloud Client Libraries support accessing Google Cloud services in a way that significantly reduces the boilerplate code you have to write. The libraries provide high-level API abstractions so they're easier to understand. They embrace idioms of the language, work well with the standard library, and integrate better with your codebase.

For more information about client libraries for Cloud APIs, see Cloud Client Libraries explained.

Click an API to find out how to use it in .NET:

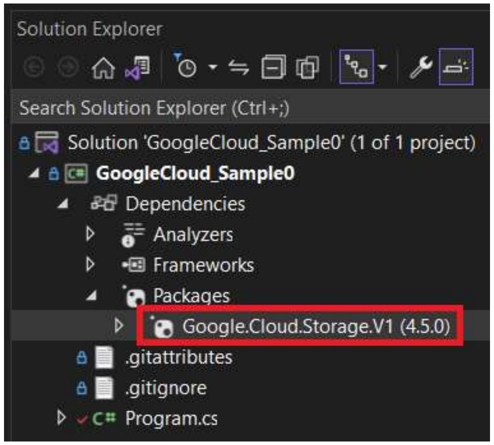
Search libraries and products

Libraries	
Access Approval	Google.Cloud.AccessApproval.V1
Access Context Manager	Google.Identity.AccessContextManager.Type Google.Identity.AccessContextManager.V1
Advisory Notifications	Google.Cloud.AdvisoryNotifications.V1

https://cloud.google.com/dotnet/docs/reference

Sample 1 Buckets (create/list) Files upload/donwload

```
using Google.Apis.Storage.v1.Data;
using Google.Cloud.Storage.V1;

string projectId = "focus-cache-387205";
string bucketName = "luiscocoenriquezsegundo";
string localPath = "C://NetChapterArticles.txt";
string objectName = "NetChapterArticles.txt";

//-------------------------------------------------------------------------------
//Create Bucket
//-------------------------------------------------------------------------------
var storage = StorageClient.Create();
var bucket = storage.CreateBucket(projectId, bucketName);
Console.WriteLine($"Created {bucketName}.");
//-------------------------------------------------------------------------------
//List Buckets
//-------------------------------------------------------------------------------
//var storage = StorageClient.Create();
var bucketsluis = storage.ListBuckets(projectId);
Console.WriteLine("Buckets:");
foreach (var bucketluis in bucketsluis)
{
    Console.WriteLine(bucketluis.Name);
}
//-------------------------------------------------------------------------------
//File Upload
//-------------------------------------------------------------------------------
using var fileStream = File.OpenRead(localPath);
storage.UploadObject(bucketName, objectName, null, fileStream);
Console.WriteLine($"Uploaded {objectName}.");
```



Solution Explorer

Search Solution Explorer (Ctrl+;)

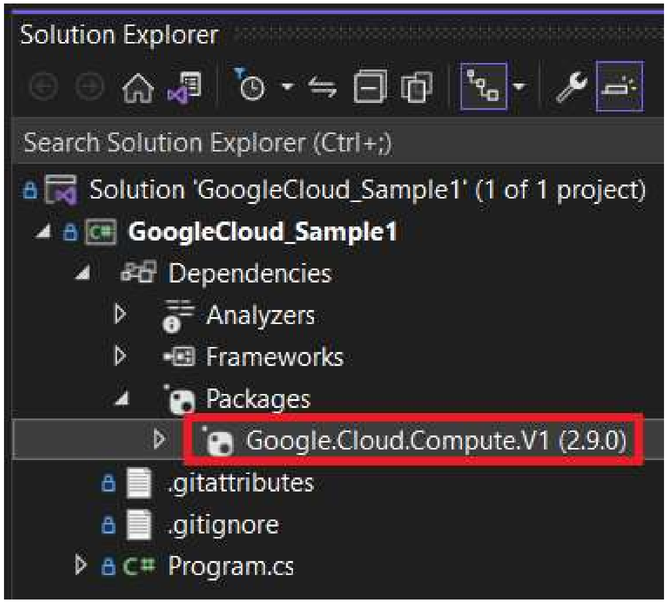
- Solution 'GoogleCloud_Sample0' (1 of 1 project)
  - GoogleCloud_Sample0
    - Dependencies
      - Analyzers
      - Frameworks
      - Packages
        - Google.Cloud.Storage.V1 (4.5.0)
    - .gitattributes
    - .gitignore
    - Program.cs

```
//-------------------------------------------------------------------------------
//File Download
//-------------------------------------------------------------------------------
string downloadlocalpath = "C:\\New folder\\DownloadedFile.txt";
using var outputFile = File.OpenWrite(downloadlocalpath);
storage.DownloadObject(bucketName, objectName, outputFile);
Console.WriteLine($"Downloaded {objectName} to {localPath}.");
//-------------------------------------------------------------------------------
//List Files
//-------------------------------------------------------------------------------
var storageObjects = storage.ListObjects(bucketName);
Console.WriteLine($"Files in bucket {bucketName}:");
foreach (var storageObject in storageObjects)
{
    Console.WriteLine(storageObject.Name);
}
```

Sample 2 Create a Virtual Machine

```
using Google.Cloud.Compute.V1;

string projectId = "focus-cache-387205";
string zone = "europe-southwest1-a";
string machineName = "luis-test-machine";
string machineType = "e2-micro";
string diskImage = "projects/debian-cloud/global/images/family/debian-10";
long diskSizeGb = 10;
string networkName = "default";

Instance instance = new Instance
{
    Name = machineName,
    MachineType = $"zones/{zone}/machineTypes/{machineType}",
    Disks =
            {
                new AttachedDisk
                {
                    AutoDelete = true,
                    Boot = true,
                    Type = ComputeEnumConstants.AttachedDisk.Type.Persistent,
                    InitializeParams = new AttachedDiskInitializeParams
                    {
                        SourceImage = diskImage,
                        DiskSizeGb = diskSizeGb
                    }
                }
            },
    NetworkInterfaces = { new NetworkInterface { Name = networkName } }
};
```

```
InstancesClient client = await InstancesClient.CreateAsync();
var instanceCreation = await client.InsertAsync(projectId, zone, instance);
await instanceCreation.PollUntilCompletedAsync();
```

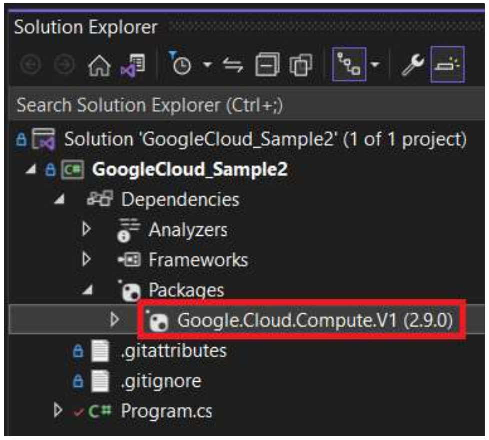


Solution 'GoogleCloud_Sample1' (1 of 1 project)
- GoogleCloud_Sample1
  - Dependencies
    - Analyzers
    - Frameworks
    - Packages
      - Google.Cloud.Compute.V1 (2.9.0)
  - .gitattributes
  - .gitignore
  - Program.cs

Sample 3 Deleting a Virtual Machine

```
using Google.Api;
using Google.Cloud.Compute.V1;

string projectId = "focus-cache-387205";
string zone = "europe-southwest1-a";
string machineName = "luis-test-machine";

InstancesClient client = await InstancesClient.CreateAsync();

// Stop the VM instance before deleting it.
var stopRequest = new StopInstanceRequest
{
    Project = projectId,
    Zone = zone,
    Instance = machineName
};

await client.StopAsync(stopRequest);
```

```
// Start the VM instance before deleting it.
//var startRequest = new StartInstanceRequest
//{
//    Project = projectId,
//    Zone = zone,
//    Instance = machineName
//};

//await client.StartAsync(startRequest);

// Make the request to delete a VM instance.
var instanceDeletion = await client.DeleteAsync(projectId, zone, machineName);

//Wait for the operation to complete using client-side polling.
await instanceDeletion.PollUntilCompletedAsync();
```


Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'GoogleCloud_Sample2' (1 of 1 project)
GoogleCloud_Sample2
Dependencies
Analyzers
Frameworks
Packages
Google.Cloud.Compute.V1 (2.9.0)
.gitattributes
.gitignore
Program.cs

Sample 4 List Virtual Machines and Virtual Machine Types

Listing Virtual Machines

```
using Google.Cloud.Compute.V1;

string projectId = "focus-cache-387205";

InstancesClient client = await InstancesClient.CreateAsync();
IList<Instance> allInstances = new List<Instance>();

// Make the request to list all VM instances in a project.
await foreach (var instancesByZone in client.AggregatedListAsync(projectId))
{
    Console.WriteLine($"Instances for zone: {instancesByZone.Key}");
    foreach (var instance in instancesByZone.Value.Instances)
    {
        Console.WriteLine($"-- Name: {instance.Name}");
        allInstances.Add(instance);
    }
}
```

Listing Virtual Machines Types

```
using Google.Cloud.Compute.V1;

MachineTypesClient machineTypesClient = MachineTypesClient.Create();
ImagesClient imagesClient = ImagesClient.Create();

// List machine types
var machineTypesList = machineTypesClient.List(new ListMachineTypesRequest
{
    Project = "focus-cache-387205",
    Zone = "europe-southwest1-a"
});

Console.WriteLine("Machine Types:");
foreach (MachineType machineType in machineTypesList)
{
    Console.WriteLine($"- {machineType.Name}");
}

Console.WriteLine();
```

Google Cloud Datastore vs Google Cloud Firestore

Google Cloud Datastore and Google Cloud Firestore are both **NoSQL databases** offered by Google Cloud Platform. However, there are some differences between them.

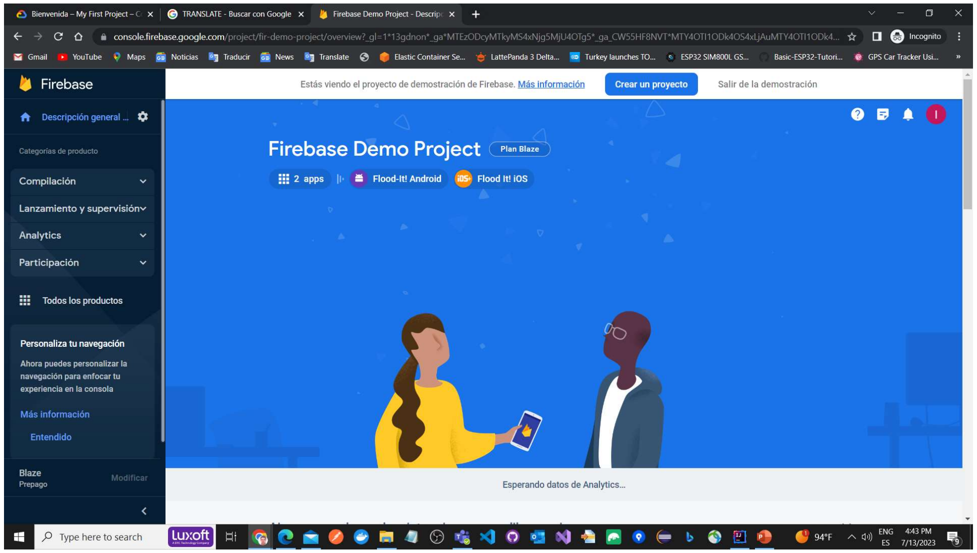
Cloud Firestore is a document-oriented database storing key-value pairs, while **Cloud Datastore** is a document database built for automatic scaling, high performance, and ease of use.

Cloud Firestore is optimized for small documents and easy to use with mobile applications.

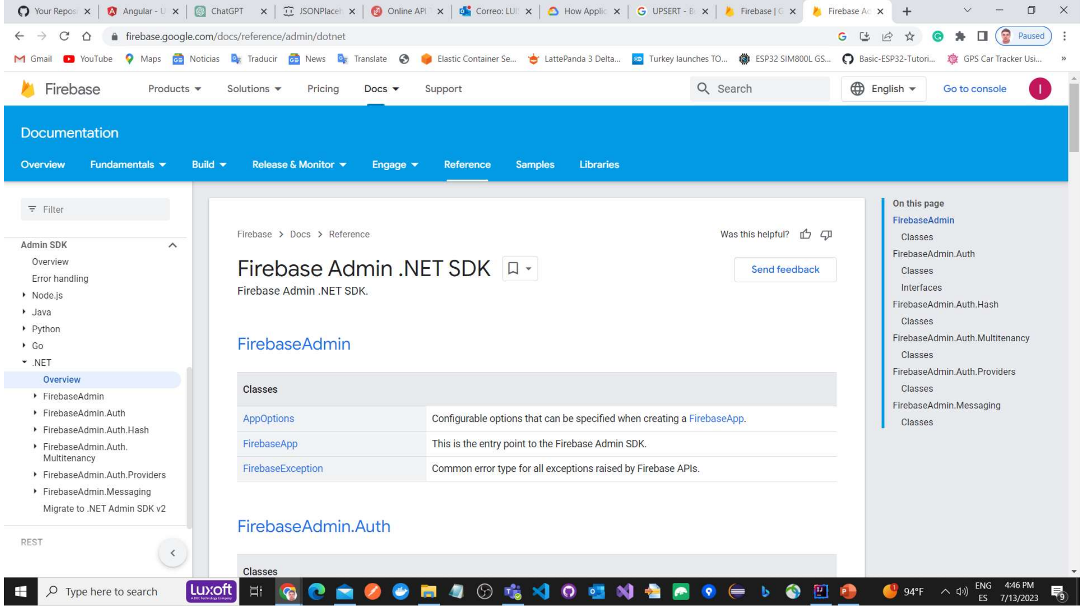
Cloud Firestore has improved write throughput per entity group, transactions, and query consistency over Cloud Datastore

Firestore is a part of the Google **Firebase** app development platform. It is a cloud-hosted **NoSQL database** option for the storage and synchronization of data. Users can directly access Firestore from their web and mobile applications with native SDKs.

Google Cloud Firebase and Firebase Admin .NET SDK



https://firebase.google.com/



Firebase Admin .NET SDK

Firebase Admin .NET SDK.

FirebaseAdmin

Classes	
AppOptions	Configurable options that can be specified when creating a FirebaseApp.
FirebaseApp	This is the entry point to the Firebase Admin SDK.
FirebaseException	Common error type for all exceptions raised by Firebase APIs.

FirebaseAdmin.Auth

Classes

https://firebase.google.com/docs/reference/admin/dotnet

Firebase and Google Cloud share three products: **Cloud Firestore**, **Cloud Functions**, and **Cloud Storage**. These are the same products that exist in Google Cloud, simply exposed for client-side developers via Firebase

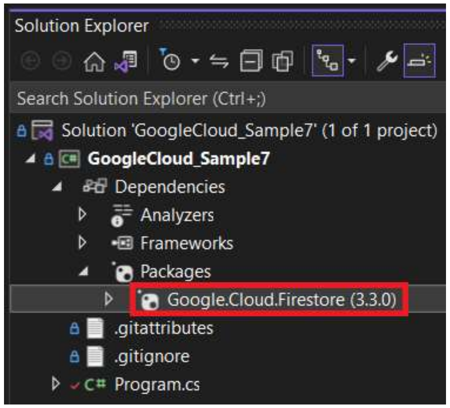
Sample: Firebase Admin .NET SDK

```
using Google.Apis.Auth.OAuth2;
using Google.Cloud.Firestore;
using Google.Cloud.Storage.V1;
using System;

FirestoreDb db = FirestoreDb.Create("<PROJECT_ID>", new GoogleCredential());
Bucket bucket = StorageClient.Create(new 
GoogleCredential()).GetBucket("<BUCKET_NAME>");

// Create a document with a random ID in the "users" collection.
DocumentReference docRef = db.Collection("users").Document();
Dictionary<string, object> user = new Dictionary<string, object>
{
    { "first", "Ada" },
    { "last", "Lovelace" },
    { "born", 1815 }
};
await docRef.SetAsync(user);

// Upload a local file to the bucket.
using (var f = File.OpenRead("path/to/local/file"))
{
    var uploadObject = bucket.UploadObject("<OBJECT_NAME>", f);
    Console.WriteLine($"Uploaded {uploadObject.Name}");
}
```

Sample 5 Firestore

```
using Google.Cloud.Firestore;

string project = "focus-cache-387205";

AddData1(project).Wait();
RetrieveAllDocuments(project).Wait();

void InitializeProjectId(string project)
{
    FirestoreDb db = FirestoreDb.Create(project);
    Console.WriteLine("Created Cloud Firestore client with project ID: {0}",
project);
}

async Task AddData1(string project)
{
    FirestoreDb db = FirestoreDb.Create(project);
    DocumentReference docRef = db.Collection("users").Document("alovelace");
    Dictionary<string, object> user = new Dictionary<string, object>
            {
                { "First", "Ada" },
                { "Last", "Lovelace" },
                { "Born", 1815 }
            };
    await docRef.SetAsync(user);
    Console.WriteLine("Added data to the alovelace document in the users
collection.");
}
```

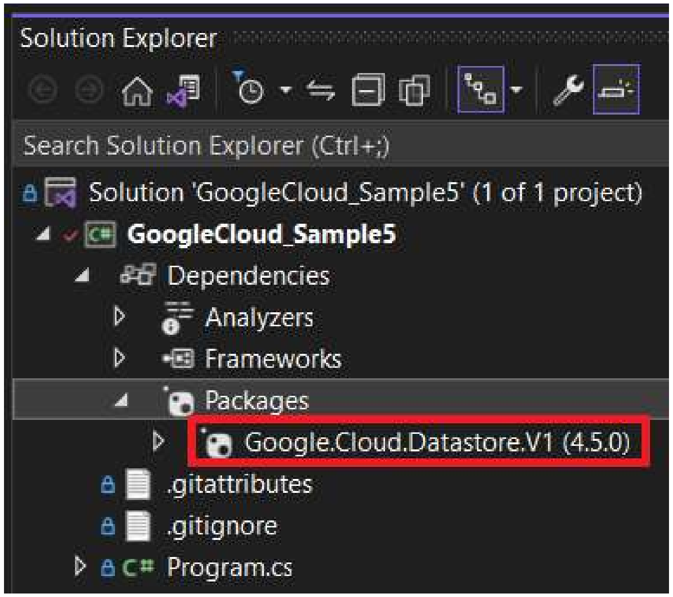
```
async Task RetrieveAllDocuments(string project)
{
    FirestoreDb db = FirestoreDb.Create(project);
    CollectionReference usersRef = db.Collection("users");
    QuerySnapshot snapshot = await usersRef.GetSnapshotAsync();
    foreach (DocumentSnapshot document in snapshot.Documents)
    {
        Console.WriteLine("User: {0}", document.Id);
        Dictionary<string, object> documentDictionary = document.ToDictionary();
        Console.WriteLine("First: {0}", documentDictionary["First"]);
        if (documentDictionary.ContainsKey("Middle"))
        {
            Console.WriteLine("Middle: {0}", documentDictionary["Middle"]);
        }
        Console.WriteLine("Last: {0}", documentDictionary["Last"]);
        Console.WriteLine("Born: {0}", documentDictionary["Born"]);
        Console.WriteLine();
    }
}
```



Solution 'GoogleCloud_Sample7' (1 of 1 project)
GoogleCloud_Sample7
Dependencies
Analyzers
Frameworks
Packages
Google.Cloud.Firestore (3.3.0)
.gitattributes
.gitignore
Program.cs

Sample 6 Datastore

```
using Google.Cloud.Datastore.V1;

string projectId = "focus-cache-387205";

DatastoreDb db = DatastoreDb.Create(projectId);

string kind = "Task";
string name = "sampletask1";

KeyFactory keyFactory = db.CreateKeyFactory(kind);
Key key = keyFactory.CreateKey(name);

var task = new Entity
{
    Key = key,
    ["description"] = "Buy milk"
};
using (DatastoreTransaction transaction = db.BeginTransaction())
{
    // Saves the task
    transaction.Upsert(task);
    transaction.Commit();

    Console.WriteLine($"Saved {task.Key.Path[0].Name}: {(string)task["description"]}");
}
```

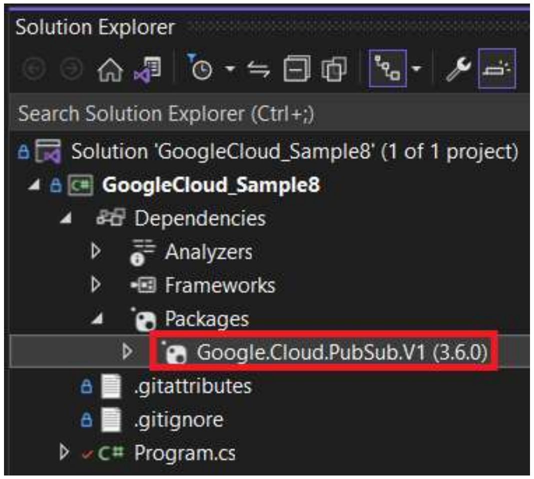

Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'GoogleCloud_Sample5' (1 of 1 project)
GoogleCloud_Sample5
Dependencies
Analyzers
Frameworks
Packages
Google.Cloud.Datastore.V1 (4.5.0)
.gitattributes
.gitignore
Program.cs

Upsert directly applies the changes to the entity. If record exists: All record data is overwritten with entity data. There is no update event.
If the record does not exist: a new record is created.

Sample 7 Pub/Sub

```
using Google.Cloud.PubSub.V1;
using Grpc.Core;

string projectId = "focus-cache-387205";
string topicId = "luis-topic-1";
string subscriptionId = "subscription-first";

//Create a Topic
PublisherServiceApiClient publisher = PublisherServiceApiClient.Create();
var topicName = TopicName.FromProjectTopic(projectId, topicId);
Topic topic = null;

try
{
    topic = publisher.CreateTopic(topicName);
    Console.WriteLine($"Topic {topic.Name} created.");
}
catch (RpcException e) when (e.Status.StatusCode == StatusCode.AlreadyExists)
{
    Console.WriteLine($"Topic {topicName} already exists.");
}
```

```
//Create a Subscription for the topic
SubscriberServiceApiClient subscriber = SubscriberServiceApiClient.Create();

SubscriptionName subscriptionName =
SubscriptionName.FromProjectSubscription(projectId, subscriptionId);
Subscription subscription = null;

try
{
    subscription = subscriber.CreateSubscription(subscriptionName, topicName,
                   pushConfig: null, ackDeadlineSeconds: 60);
}
catch (RpcException e) when (e.Status.StatusCode == StatusCode.AlreadyExists)
{
    // Already exists.  That's fine.
}
```


Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'GoogleCloud_Sample8' (1 of 1 project)
GoogleCloud_Sample8
Dependencies
Analyzers
Frameworks
Packages
Google.Cloud.PubSub.V1 (3.6.0)
.gitattributes
.gitignore
Program.cs

Sample 7 Pub/Sub

Pub/Sub

```
//Publish messages to the above created topic

PublisherClient publisher1 = await PublisherClient.CreateAsync(topicName);

int publishedMessageCount = 0;

List<string> messageTexts = new List<string>();
messageTexts.Add("First message");
messageTexts.Add("Second message");
messageTexts.Add("Third message");
messageTexts.Add("Fourth message");
messageTexts.Add("Fifth message");

var publishTasks = messageTexts.Select(async text =>
{
    try
    {
        string message = await publisher1.PublishAsync(text);
        Console.WriteLine($"Published message {message}");
        Interlocked.Increment(ref publishedMessageCount);
    }
    catch (Exception exception)
    {
        Console.WriteLine($"An error ocurred when publishing message {text}: {exception.Message}");
    }
});
await Task.WhenAll(publishTasks);
```

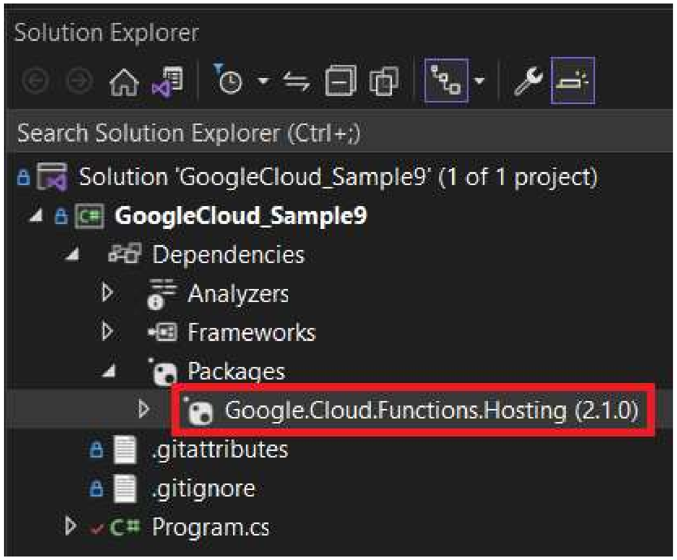
Sample 7 Pub/Sub

Pub/Sub

```
int messages_Count_received = PullMessagesSync(projectId, subscriptionId, true);

int PullMessagesSync(string projectId, string subscriptionId, bool acknowledge)
{
    SubscriptionName subscriptionName = SubscriptionName.FromProjectSubscription(projectId, subscriptionId);
    SubscriberServiceApiClient subscriberClient = SubscriberServiceApiClient.Create();
    int messageCount = 0;
    try
    {
        // Pull messages from server,
        PullResponse response = subscriberClient.Pull(subscriptionName, maxMessages: 20);
        foreach (ReceivedMessage msg in response.ReceivedMessages)
        {
            string text = System.Text.Encoding.UTF8.GetString(msg.Message.Data.ToArray());
            Console.WriteLine($"Message {msg.Message.MessageId}: {text}");
            Interlocked.Increment(ref messageCount);
        }
        if (acknowledge && messageCount > 0)
        {
            subscriberClient.Acknowledge(subscriptionName, response.ReceivedMessages.Select(msg => msg.AckId));
        }
    }
    catch (RpcException ex) when (ex.Status.StatusCode == StatusCode.Unavailable)
    {
        // UNAVAILABLE due to too many concurrent pull requests pending for the given subscription.
    }
    return messageCount;
}
```

Sample 8 Functions

```
using Google.Cloud.Functions.Framework;
using Microsoft.AspNetCore.Http;

namespace HelloWorld;

public class Function : IHttpFunction
{
    public async Task HandleAsync(HttpContext context)
    {
        await context.Response.WriteAsync("Hello World!");
    }
}
```



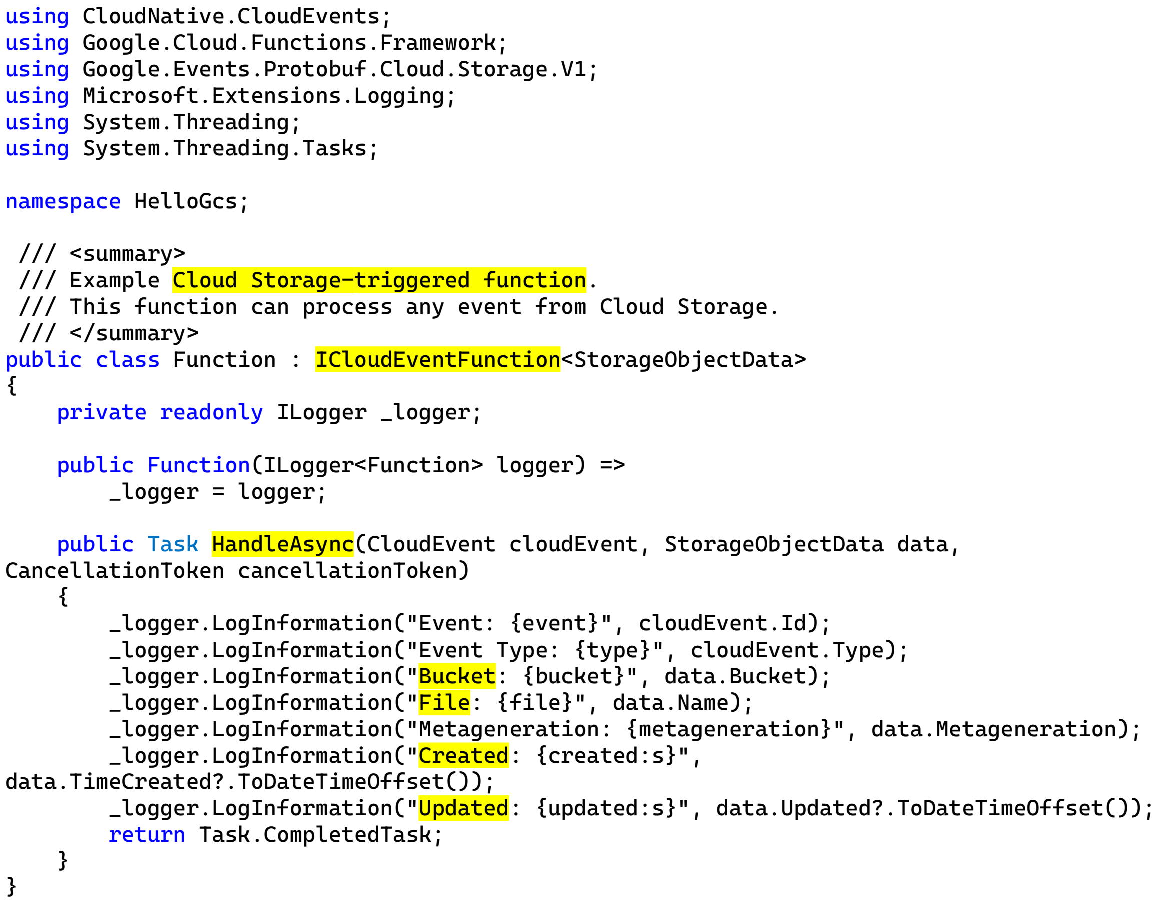
Solution Explorer

Search Solution Explorer (Ctrl+;)

- Solution 'GoogleCloud_Sample9' (1 of 1 project)
  - **GoogleCloud_Sample9**
    - Dependencies
      - Analyzers
      - Frameworks
      - Packages
        - Google.Cloud.Functions.Hosting (2.1.0)
    - .gitattributes
    - .gitignore
    - Program.cs

https://cloud.google.com/functions/docs/concepts/overview

Sample 8 Functions



```
using CloudNative.CloudEvents;
using Google.Cloud.Functions.Framework;
using Google.Events.Protobuf.Cloud.Storage.V1;
using Microsoft.Extensions.Logging;
using System.Threading;
using System.Threading.Tasks;

namespace HelloGcs;

 /// <summary>
 /// Example Cloud Storage-triggered function.
 /// This function can process any event from Cloud Storage.
 /// </summary>
public class Function : ICloudEventFunction<StorageObjectData>
{
    private readonly ILogger _logger;

    public Function(ILogger<Function> logger) =>
        _logger = logger;

    public Task HandleAsync(CloudEvent cloudEvent, StorageObjectData data,
CancellationToken cancellationToken)
    {
        _logger.LogInformation("Event: {event}", cloudEvent.Id);
        _logger.LogInformation("Event Type: {type}", cloudEvent.Type);
        _logger.LogInformation("Bucket: {bucket}", data.Bucket);
        _logger.LogInformation("File: {file}", data.Name);
        _logger.LogInformation("Metageneration: {metageneration}", data.Metageneration);
        _logger.LogInformation("Created: {created:s}",
data.TimeCreated?.ToDateTimeOffset());
        _logger.LogInformation("Updated: {updated:s}", data.Updated?.ToDateTimeOffset());
        return Task.CompletedTask;
    }
}
```

Luxoft

A DXC Technology Company