

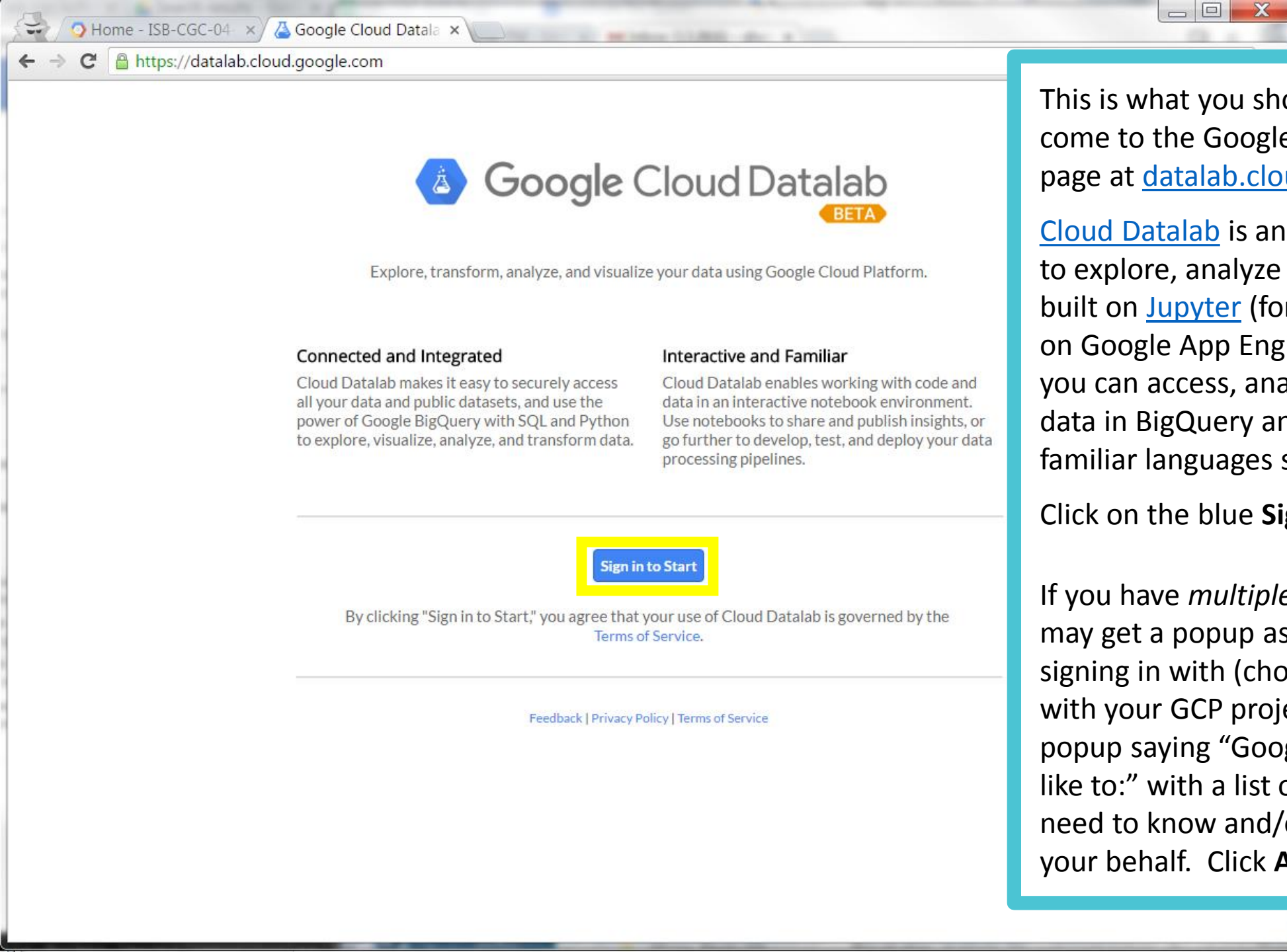
An Introduction to Cloud Datalab (IPython/Jupyter)

(in less than 20 minutes)

brought to you by

The ISB Cancer Genomics Cloud





This is what you should see the first time you come to the Google Cloud Datalab landing page at datalab.cloud.google.com

[Cloud Datalab](#) is an interactive tool created to explore, analyze and visualize data. It is built on [Jupyter](#) (formerly IPython) and runs on Google App Engine. Using Cloud Datalab, you can access, analyze, and manipulate data in BigQuery and Cloud Storage using familiar languages such as Python and SQL.

Click on the blue **Sign in to Start** button.

If you have *multiple* Google identities, you may get a popup asking which one you are signing in with (choose the one associated with your GCP project). You may then see a popup saying “Google Cloud Datalab would like to:” with a list of things Datalab will need to know and/or be allowed to do on your behalf. Click **Allow**.



Explore, transform, analyze, and visualize your data using Google Cloud Platform.

Select Cloud Project:

ISB-CGC workshop
ISB-CGC-04-0003

Cloud Datalab is deployed as a Google App Engine application module in the selected project. The Google Compute Engine and Google BigQuery APIs must be enabled for the project, and you must be authorized to use the project as an owner or editor.

When you first deploy Cloud Datalab, the following additions are made to the selected project:

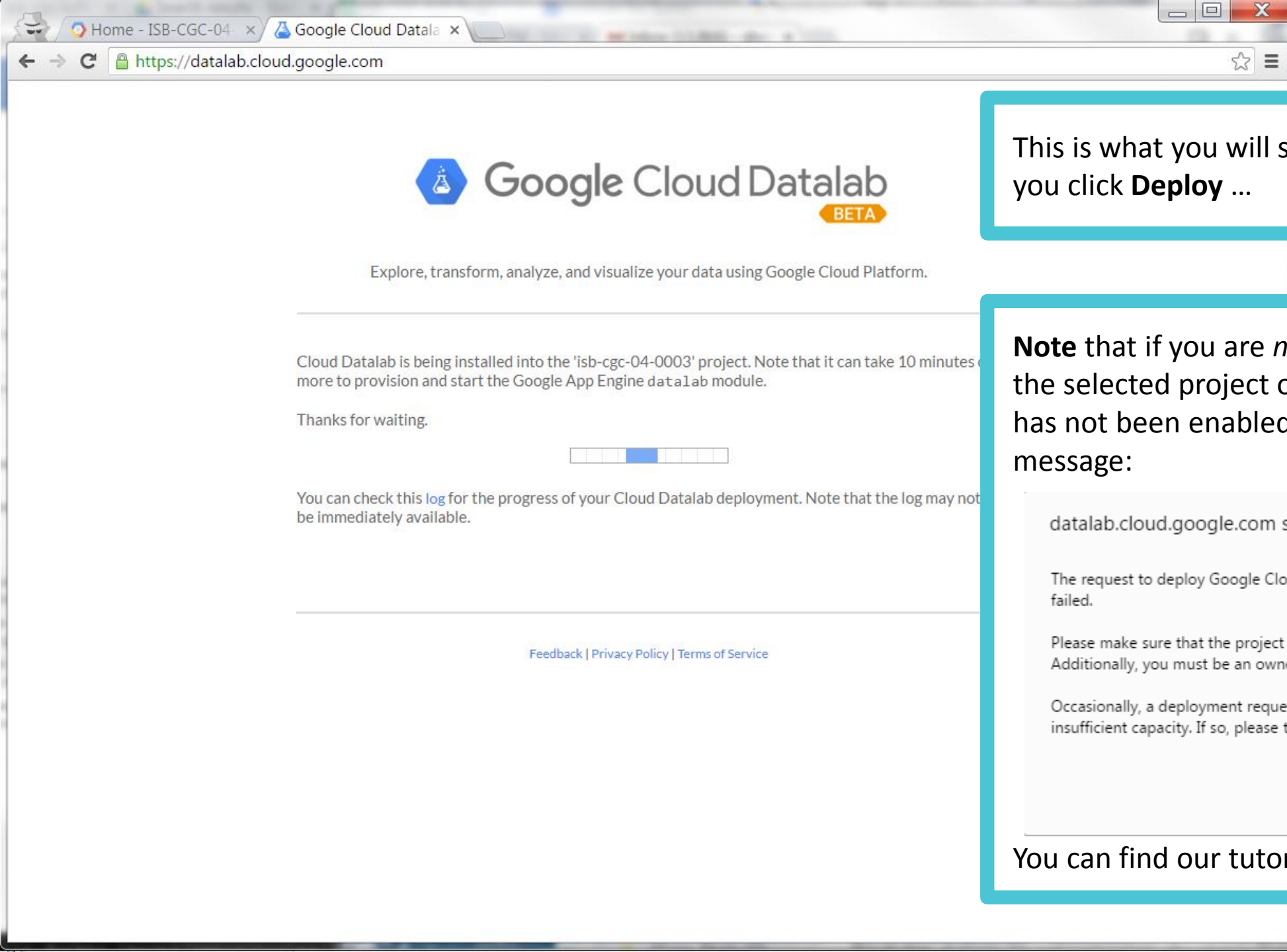
- An App Engine "datalab" module is added. It can be managed from the App Engine section in the Google Developers Console. App Engine charges will accrue.
- A "datalab" Compute Engine network is added.
- A "datalab" branch is added to the Git repository associated with the project. This repository contains pre-installed samples and docs, and will contain any notebooks that you create. This repository is browsable from the Development section in the Google Developers Console.

If you are a member of multiple projects you will now select the cloud project in which you want to deploy Cloud Datalab.

You must be an Editor or Owner of the project and the Compute Engine API must already be enabled.

Cloud Datalab runs on a VM in your project. Multiple members of a project may access a single instance of Datalab, or individuals may prefer to deploy and manage personal instances.

Once you have selected the correct cloud project, click on the blue **Deploy** button. (If the Start and Manage buttons are already blue, then you have an instance of Datalab *already* running – in that case click **Start**.)



This is what you will see for 5-10 minutes after you click **Deploy** ...

Note that if you are *not* an editor or owner on the selected project or the Compute Engine API has not been enabled, you will get this error message:

datalab.cloud.google.com says:

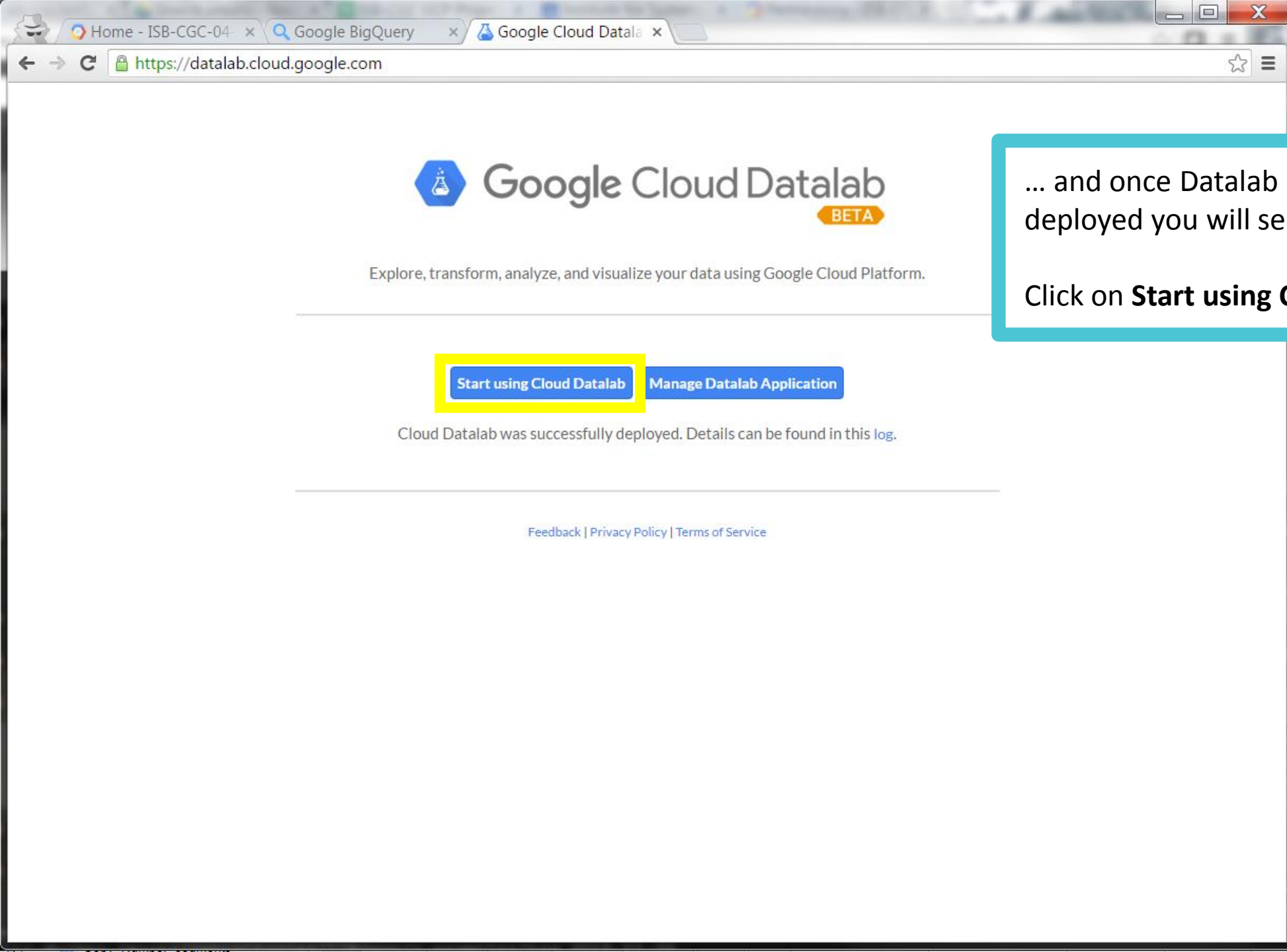
The request to deploy Google Cloud Datalab into project isb-cgc-04-0030 failed.

Please make sure that the project is enabled for billing. Additionally, you must be an owner or editor within the project.

Occasionally, a deployment request may fail due to transient errors or insufficient capacity. If so, please try again.

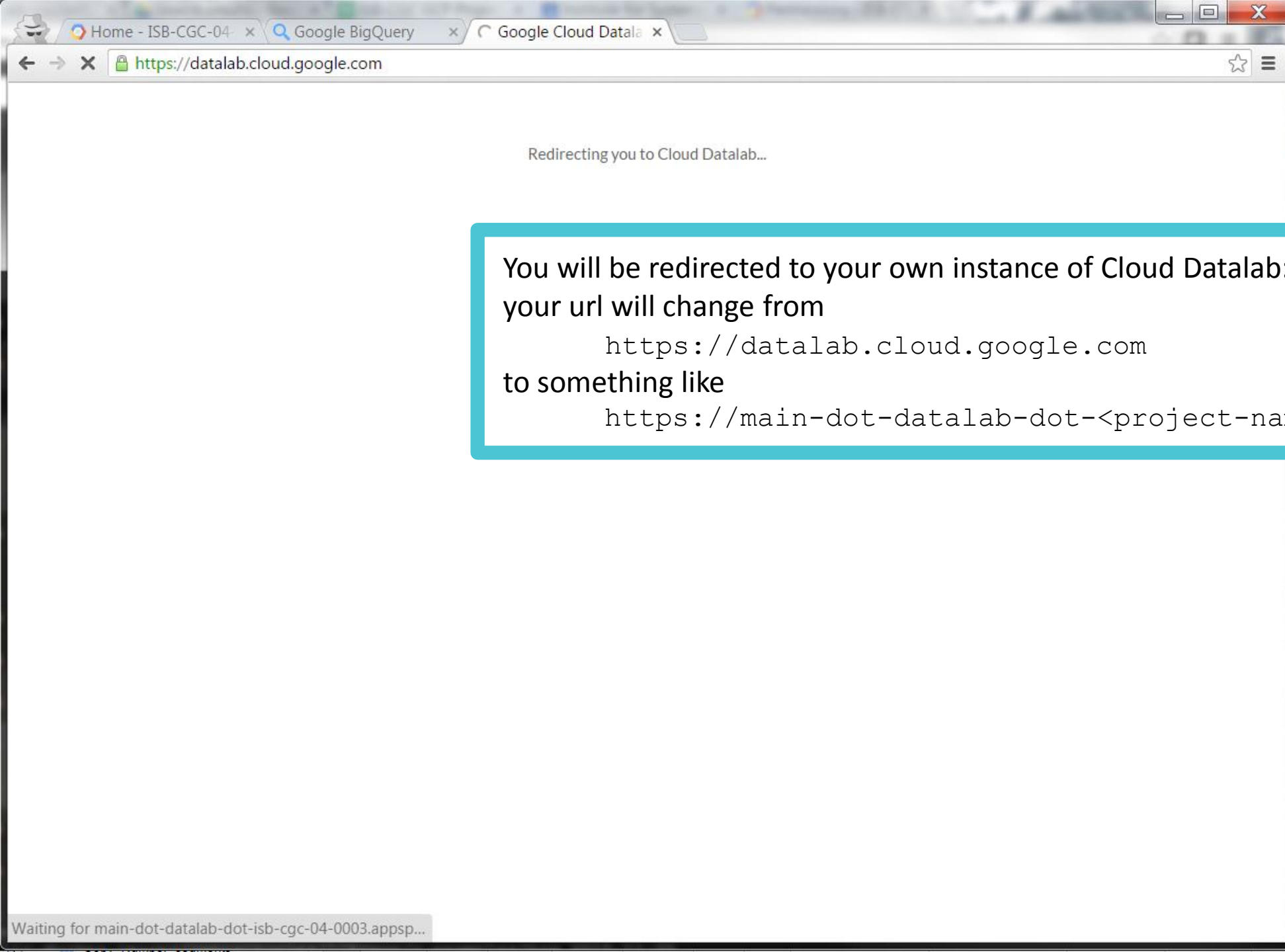
OK

You can find our tutorial on enabling APIs [here](#).



... and once Datalab has been successfully deployed you will see these two new options.

Click on **Start using Cloud Datalab**.

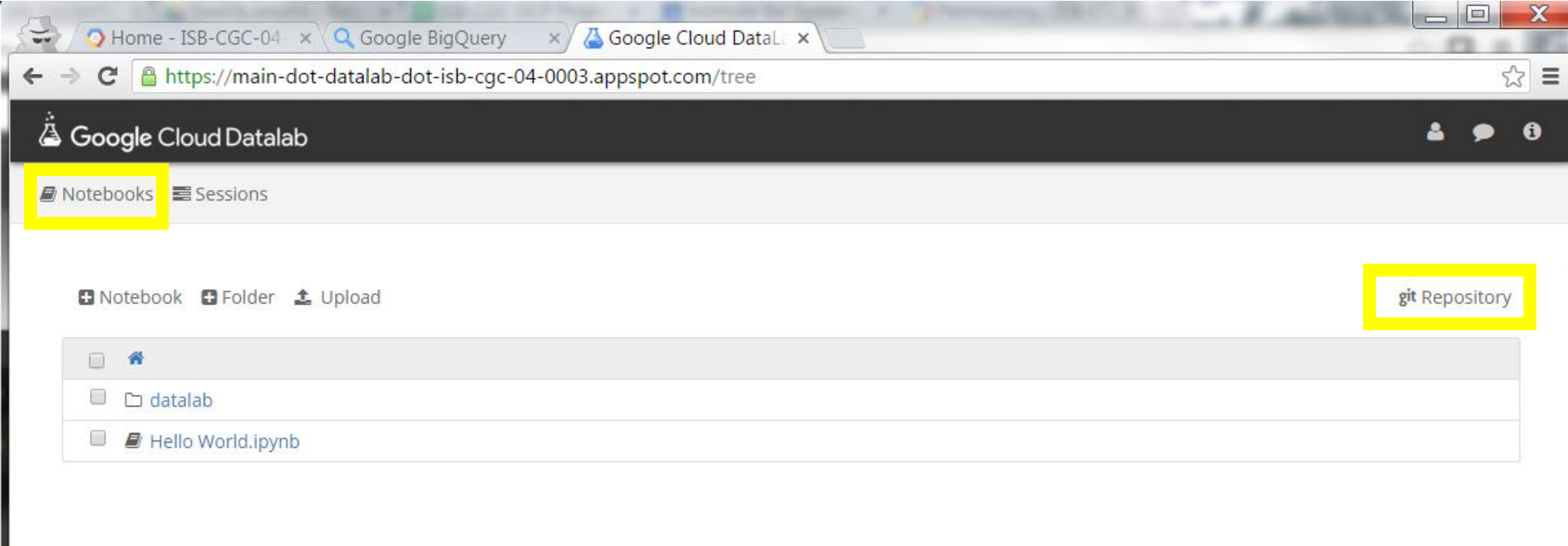


You will be redirected to your own instance of Cloud Datalab:
your url will change from

`https://datalab.cloud.google.com`

to something like

`https://main-dot-datalab-dot-<project-name>.appspot.com/tree`

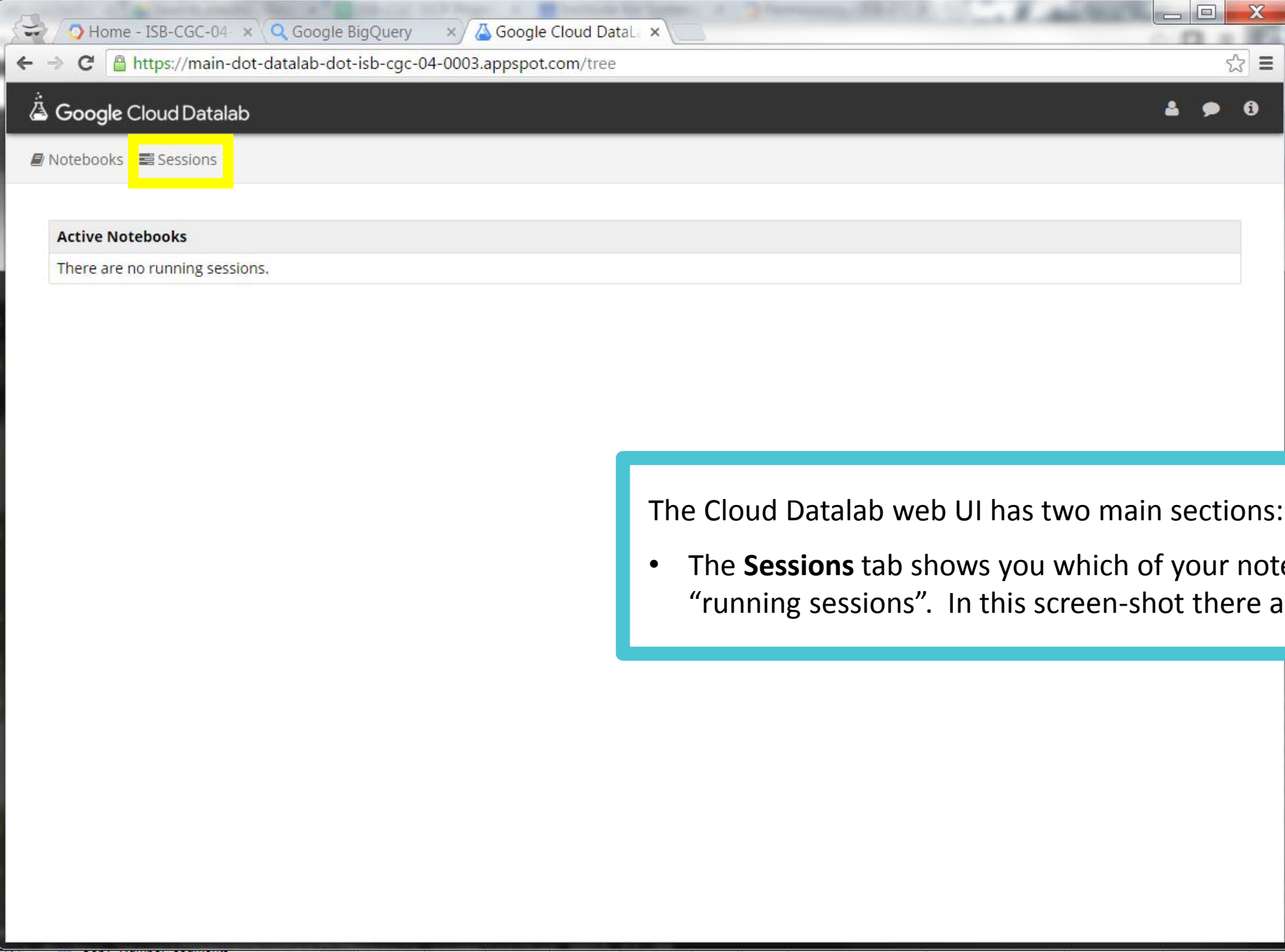


The Cloud Datalab web UI has two main sections: **Notebooks** and **Sessions**.

- The **Notebooks** tab is a file/folder browser connected to your [Google Cloud git Repository](#) which you can access directly from this page, and also from the [Console](#) under [Development](#).

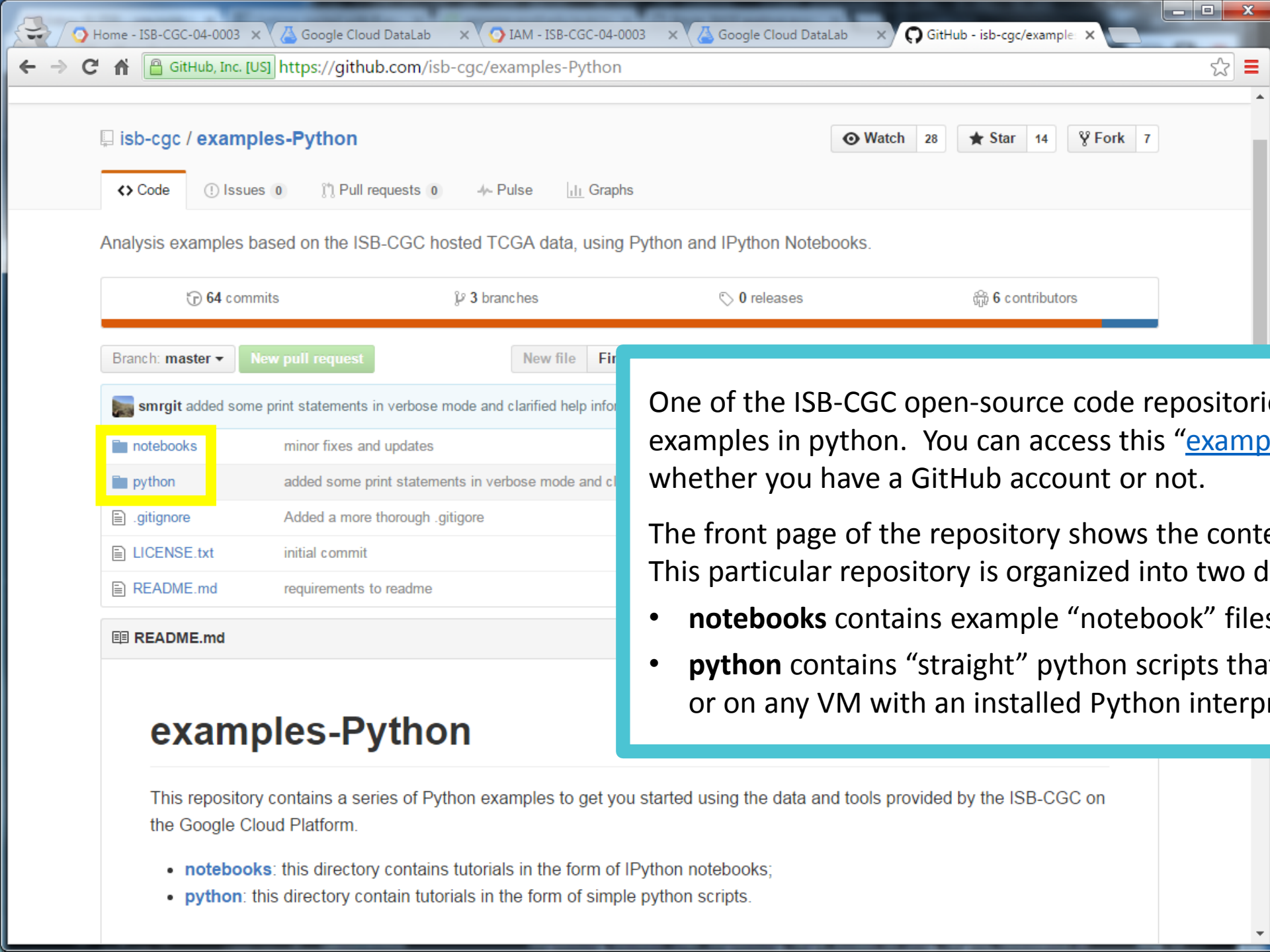
Your Cloud git repo will automatically be populated with a set of example and tutorial IPython notebooks to get you started. These notebooks are in the “datalab” folder, with the exception of the “Hello World” notebook.

Note: IPython notebook files end with the “ipynb” extension.



The Cloud Datalab web UI has two main sections: **Notebooks** and **Sessions**.

- The **Sessions** tab shows you which of your notebooks are “active” in “running sessions”. In this screen-shot there are no running sessions.



One of the ISB-CGC open-source code repositories on GitHub contains examples in python. You can access this “[examples-Python](https://github.com/isb-cgc/examples-Python)” repository whether you have a GitHub account or not.

The front page of the repository shows the contents and the README. This particular repository is organized into two directories:

- **notebooks** contains example “notebook” files for use in Cloud Datalab;
- **python** contains “straight” python scripts that can be run in [Cloud Shell](#) or on any VM with an installed Python interpreter.

Home - ISB-CGC-04-0003 x Google Cloud DataLab x IAM - ISB-CGC-04-0003 x Google Cloud DataLab x examples-Python/notebooks x

GitHub, Inc. [US] https://github.com/isb-cgc/examples-Python/tree/master/notebooks

Branch: master examples-Python / notebooks New file Find file History

smrgit minor fixes and updates Latest commit 54bcca6 19 days ago

..

BCGSC microRNA expression.ipynb	minor updates, this time with outputs (where possible)	6 months ago
BRAF-V600 study using CCLE data.ipynb	minor fixes and updates	19 days ago
Copy Number segments.ipynb	updates	2 months ago
Creating TCGA cohorts -- part 1.ipynb	minor fixes and updates	
Creating TCGA cohorts -- part 2.ipynb	Fix a few links and table nam	
Creating TCGA cohorts -- part 3.ipynb	new example for creating coh	
DNA Methylation.ipynb	changed title	
Protein expression.ipynb	minor updates, this time with	
README.md	Update README.md	
Somatic Mutations.ipynb	minor updates, this time with	
TCGA Annotations.ipynb	minor updates, this time with	
The ISB-CGC open-access TCGA tables...	updates	
UNC HiSeq mRNAseq gene expression.ip...	changed title	6 months ago

README.md

examples-Python/notebooks

The **notebooks** subdirectory of this repository contains a series of IPython notebooks that are intended to help you get started working with the ISB-CGC hosted [TCGA data](#) in BigQuery.

From the **examples-Python** repository home page, click on the **notebooks** folder name to move into that folder -- now you will see a list of the individual ipynb files, and a new **README**.

From here you can click on an individual ipynb file to see its contents.

For example, click on the one titled "**The ISB-CGC open-access TCGA tables**" (the second one from the bottom), which is an introductory notebook.

Home - ISB-CGC-04-0003 x Google Cloud DataLab x IAM - ISB-CGC-04-0003 x Google Cloud DataLab x examples-Python/The ISB-CGC open-access TCGA tables in BigQuery.ipynb

Branch: master examples-Python / notebooks / The ISB-CGC open-access TCGA tables in BigQuery.ipynb Find file Copy path

smrgit updates b5d4679 on Mar 3

2 contributors

149 lines (148 sloc) 12.3 KB

Raw Blame History

The ISB-CGC open-access TCGA tables in Big-Query

The goal of this notebook is to introduce you to a new publicly-available, open-access dataset in BigQuery. This set of BigQuery tables was produced by the [ISB-CGC](#) project, based on the open-access [TCGA](#) data available at the TCGA [Data Portal](#). You will need to have access to a Google Cloud Platform (GCP) project in order to use BigQuery. If you don't already have one, you can sign up for a [free-trial](#) or contact [us](#) and become part of the community evaluation phase of our Cancer Genomics Cloud pilot. (You can find more information about this NCI-funded program [here](#).)

We are not attempting to provide a thorough BigQuery tutorial. Here are links to some resources that you might find useful:

- [BigQuery](#).
- the BigQuery [web UI](#) where you can run queries interactively.
- [IPython](#) (now known as [Jupyter](#)), and
- [Cloud Datalab](#) the recently announced interactive cloud IDE.

There are also many tutorials and samples available on the [Google Cloud](#) website.

In order to work with BigQuery, the first thing you need to do is create a GCP project and enable the BigQuery API.

```
In [1]: import gcp.bigquery as bq
```

The next thing you need to know is how to access the [BigQuery](#) datasets, and datasets are owned by a specific GCP project. The [TCGA](#) dataset, `tcga_201510_alpha`, owned by the `isb-cgc` project. A full table identifier is of the form `<project_id>:<dataset_id>.<table_id>`. Let's start by getting some basic information about the tables in this dataset:

```
In [2]: d = bq.DataSet('isb-cgc:tcga_201510_alpha')
for t in d.tables():
    print('%10d rows %10d bytes' % (t.num_rows, t.size))
```

When you click on an ipynb file in GitHub, you see it *rendered* (as HTML) much as it looks within the Jupyter (IPython) interactive computing environment. The *raw* file is actually a [JSON document](#) which can contain a mix of text, source code, metadata, and rich media output.

The easiest way to bring one of these example notebooks from GitHub into your running instance of Cloud Datalab is a two-step process: 1) save the ipynb file locally, and 2) upload it to Datalab. We will walk you through this process in the next few slides.

Home - ISB-CGC-04-0003 x Google Cloud DataLab x IAM - ISB-CGC-04-0003 x Google Cloud DataLab x examples-Python/The ISB-CGC open-access TCGA tables in BigQuery.ipynb

isb-cgc / examples-Python

Watch 28 Star 14 Fork 7

Code Issues 0 Pull requests 0 Pulse Graphs

Branch: master examples-Python / notebooks / The ISB-CGC open-access TCGA tables in BigQuery.ipynb Find file Copy path

smrgit updates b5d4679 on Mar 3

2 contributors

149 lines (148 sloc) 12.3 KB

The ISB-CGC open-access TCGA tables in Big-Query

The goal of this notebook is to introduce you to a new publicly-available, open-access dataset of TCGA tables was produced by the [ISB-CGC](#) project, based on the open-access [TCGA](#) data available at the [NCI](#). You will need to have access to a Google Cloud Platform (GCP) project in order to use BigQuery. If you do not have a GCP project, you can sign up for a [free-trial](#) or contact [us](#) and become part of the community evaluation phase of our Cloud DataLab project. You can find more information about this NCI-funded program [here](#).)

We are not attempting to provide a thorough BigQuery or IPython tutorial here. Here are links to some resources that you might find useful:

- [BigQuery](#),
- the BigQuery [web UI](#) where you can run queries interactively,
- [IPython](#) (now known as [Jupyter](#)), and
- [Cloud DataLab](#) the recently announced interactive cloud-based platform that we are building.

There are also many tutorials and samples available on github (see, in particular, the [Cloud DataLab](#) project).

In order to work with BigQuery, the first thing you need to do is import the [gcp.bigquery](#) package:

```
In [1]: import gcp.bigquery as bq
```

The next thing you need to know is how to access the specific tables you are interested in. BigQuery tables are organized into datasets, and datasets are owned by a specific GCP project. The tables we are introducing in this notebook are in a dataset called

Raw

- Open link in new tab
- Open link in new window
- Open link in incognito window
- Save link as...
- Copy link address
- Inspect Ctrl+Shift+I

You will need the “raw” file rather than the rendered HTML, so right-click on the **Raw** button (highlighted in yellow above), select “**Save link as...**” and save the ipynb file to your local machine.

Home - ISB-CGC-04-0003 x Google Cloud DataLab x IAM - ISB-CGC-04-0003 x Google Cloud DataLab x examples-Python/notebooks x

GitHub, Inc. [US] https://github.com/isb-cgc/examples-Python/tree/master/notebooks

isb-cgc / examples-Python

Watch 28 Star 14 Fork 7

Code Issues 0 Pull requests 0 Pulse Graphs

Branch: master examples-Python / notebooks /

New file Find file History

smrgit minor fixes and updates Latest commit 54bcca6 19 days ago

..

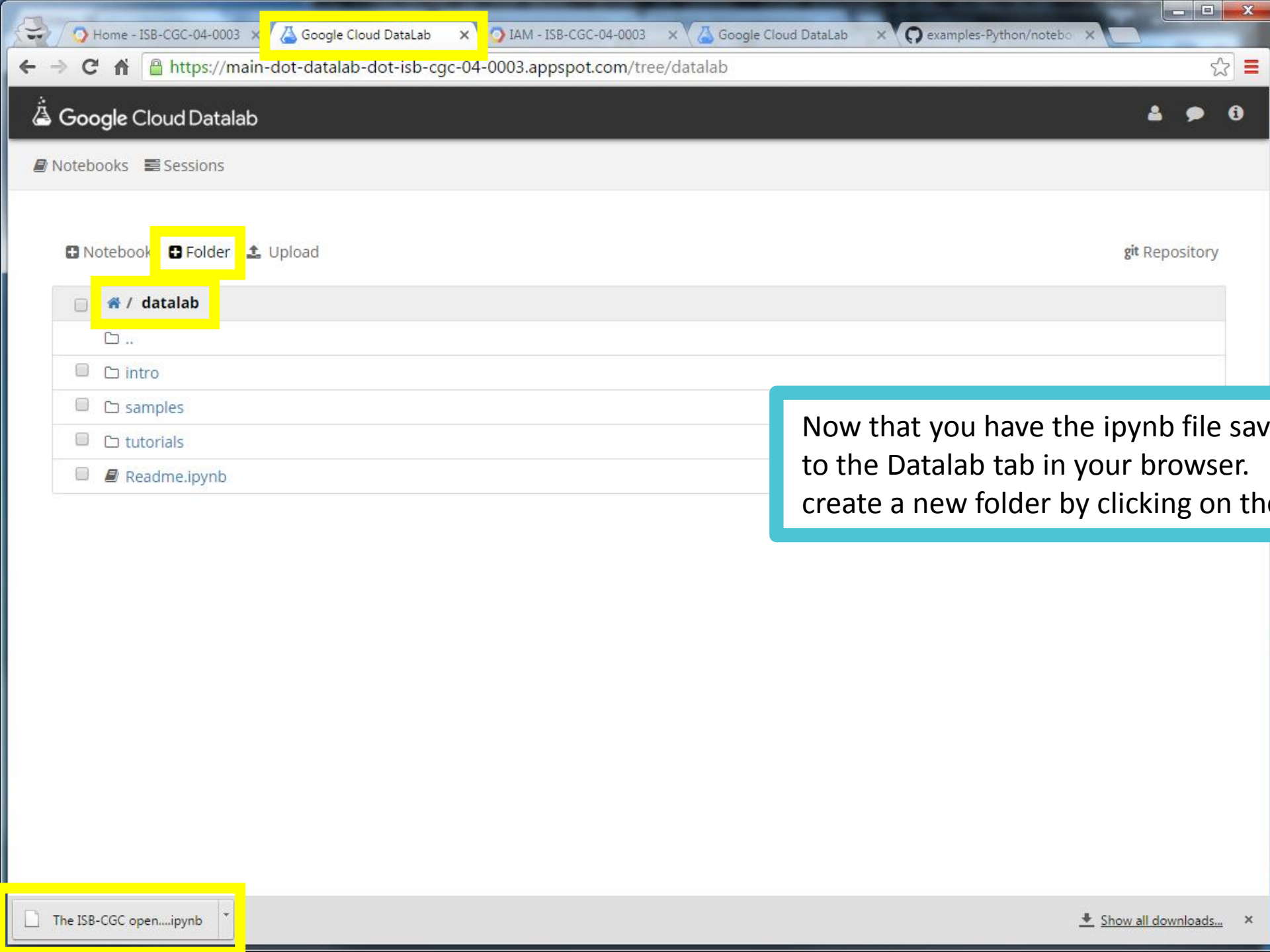
BCGSC microRNA expression.ipynb	minor updates, this time with outputs (where possible)	6 months ago
BRAF-V600 study using CCLE data.ipynb	minor fixes and updates	19 days ago
Copy Number segments.ipynb	updates	
Creating TCGA cohorts -- part 1.ipynb	minor fixes and updates	
Creating TCGA cohorts -- part 2.ipynb	Fix a few links and table names.	
Creating TCGA cohorts -- part 3.ipynb	new example for creating cohorts -- this is the code that is used	
DNA Methylation.ipynb	changed title	
Protein expression.ipynb	minor updates, this time with outputs (where possible)	6 months ago
README.md	Update README.md	2 months ago
Somatic Mutations.ipynb	minor updates, this time with outputs (where possible)	6 months ago
TCGA Annotations.ipynb	minor updates, this time with outputs (where possible)	6 months ago
The ISB-CGC open...ipynb		2 months ago
UNC HiSeq m...		6 months ago
README.m...		

The ISB-CGC open...ipynb

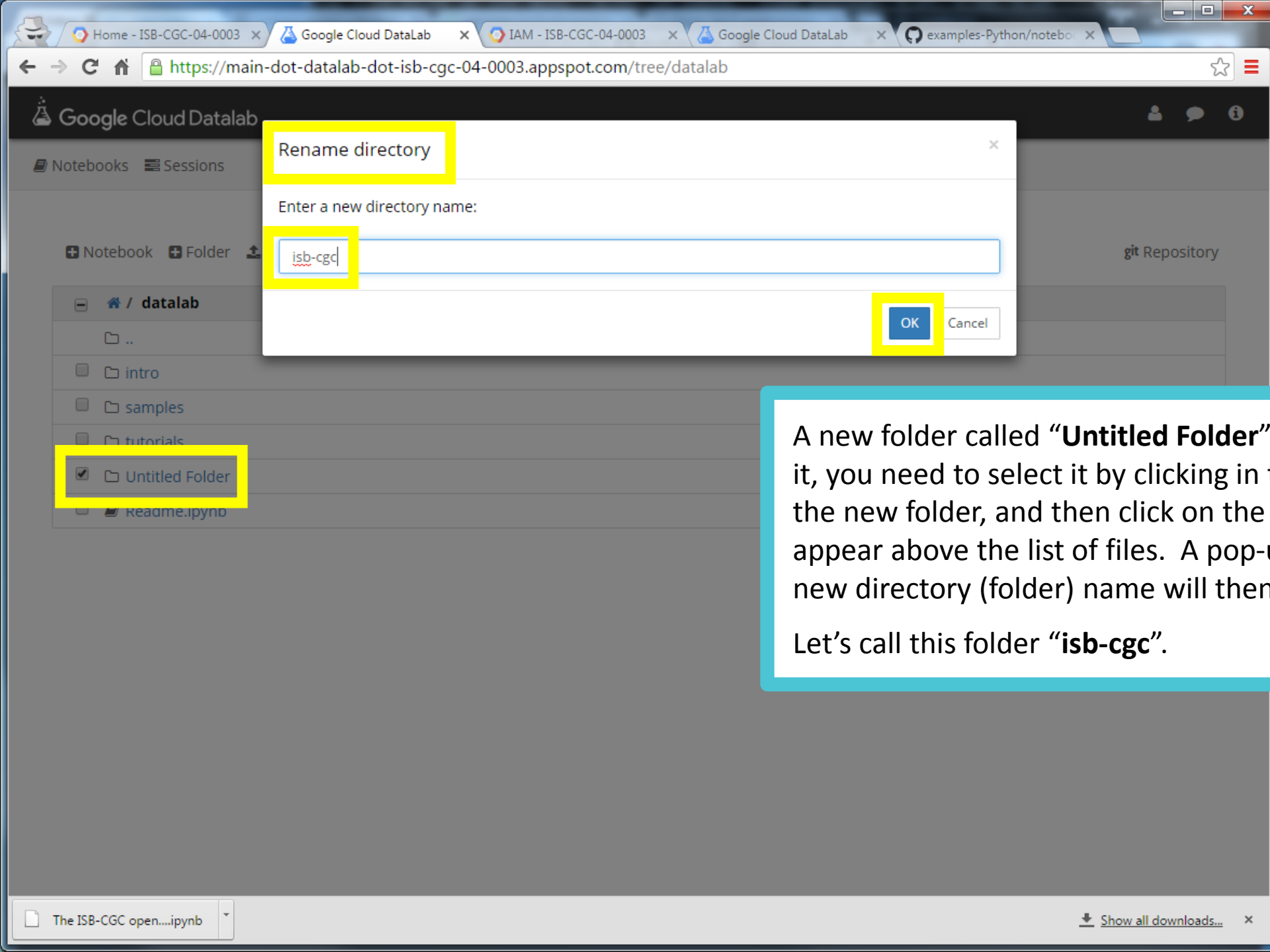
Inspect Ctrl+Shift+I

Show all downloads...

NOTE: if you right-click on the file name as shown in this screen-shot, and select "Save link as..." you will be saving the *rendered HTML* (rather than the IPython JSON document) which you will **not** be able to import into Datalab.

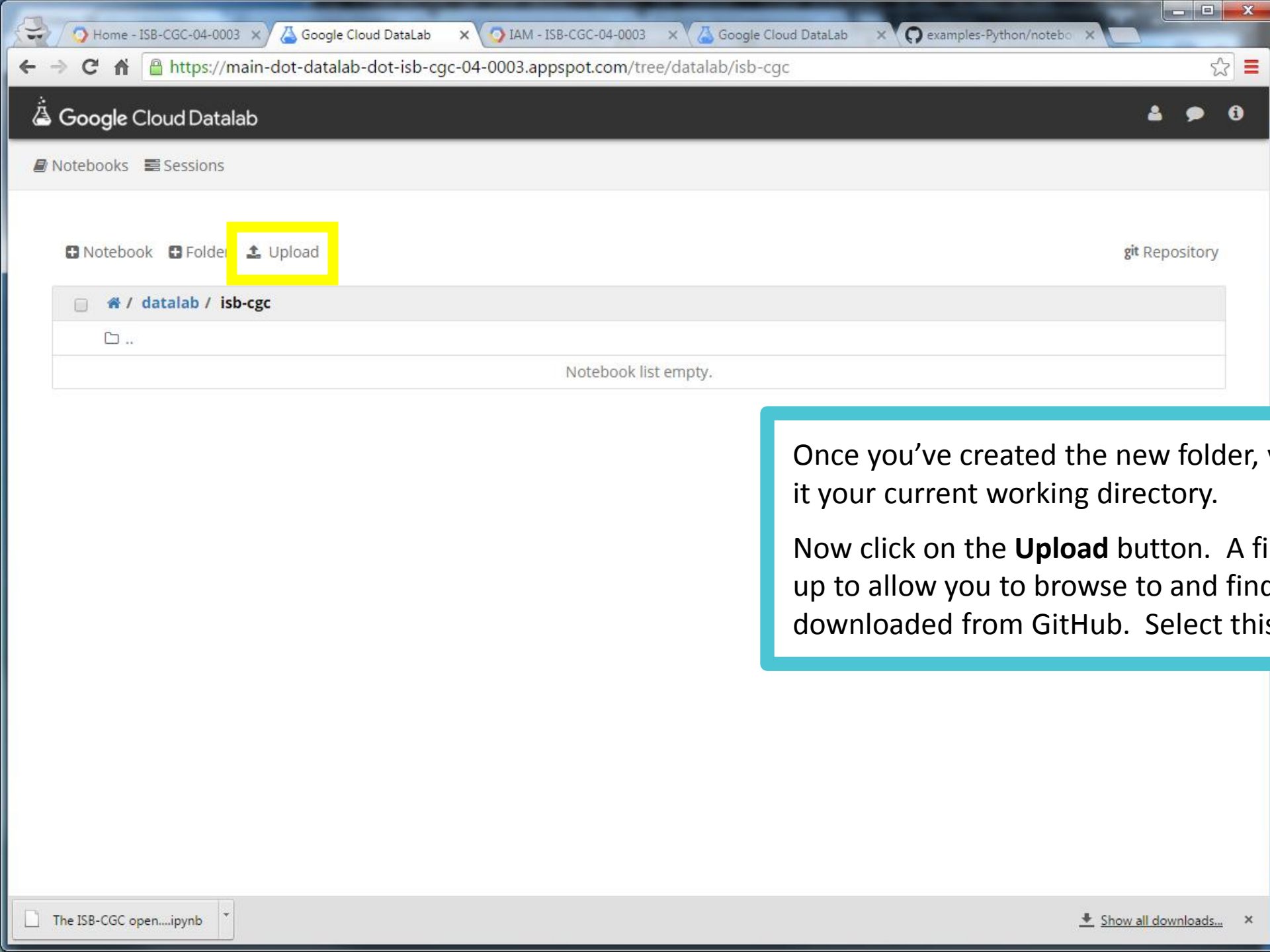


Now that you have the ipynb file saved locally, you can return to the Datalab tab in your browser. In the “datalab” folder let’s create a new folder by clicking on the “**Add Folder**” button.



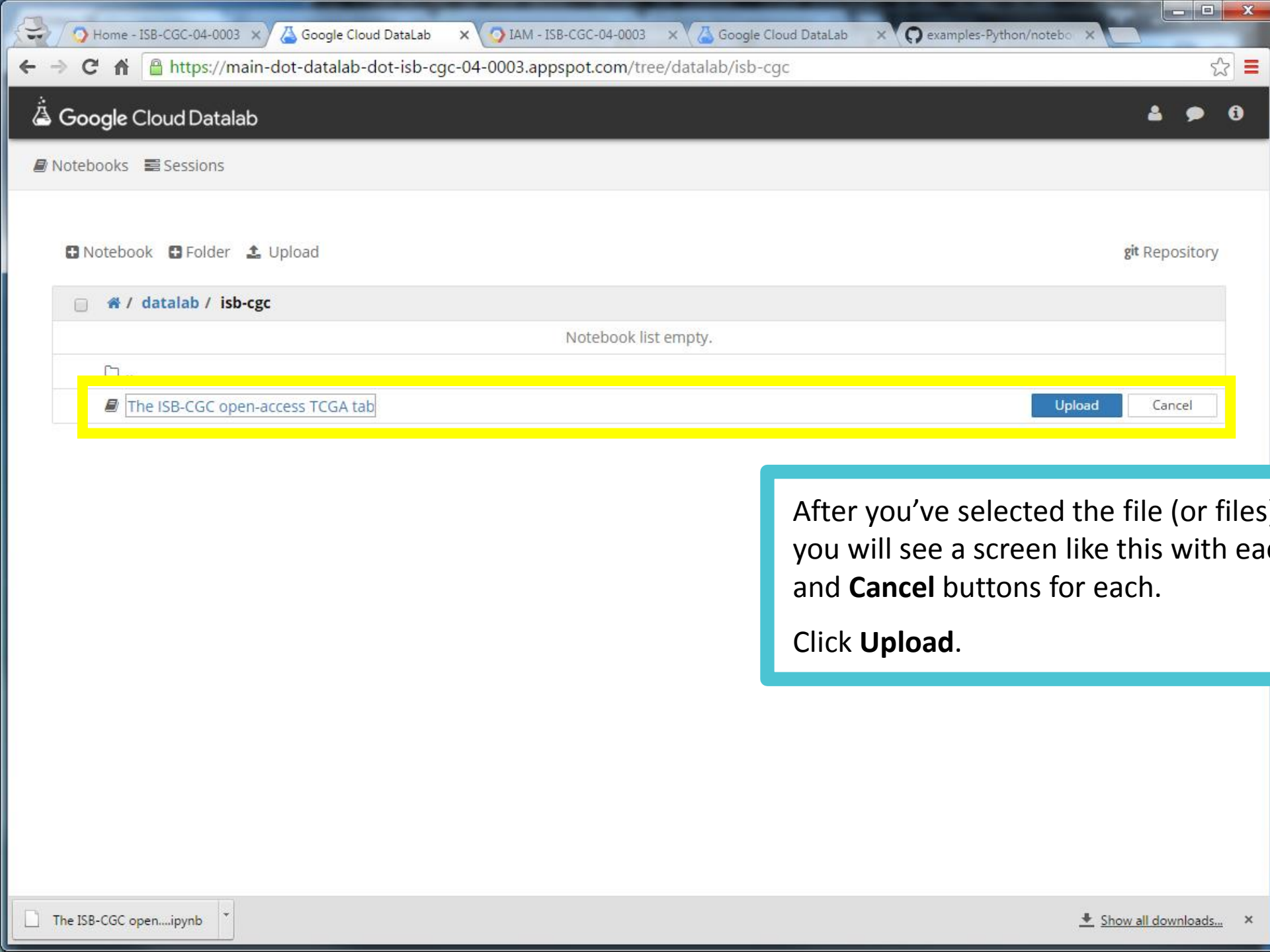
A new folder called “**Untitled Folder**” will be added. To rename it, you need to select it by clicking in the **checkbox** to the left of the new folder, and then click on the **Rename** button that will appear above the list of files. A pop-up prompting you for the new directory (folder) name will then appear.

Let's call this folder “**isb-cgc**”.



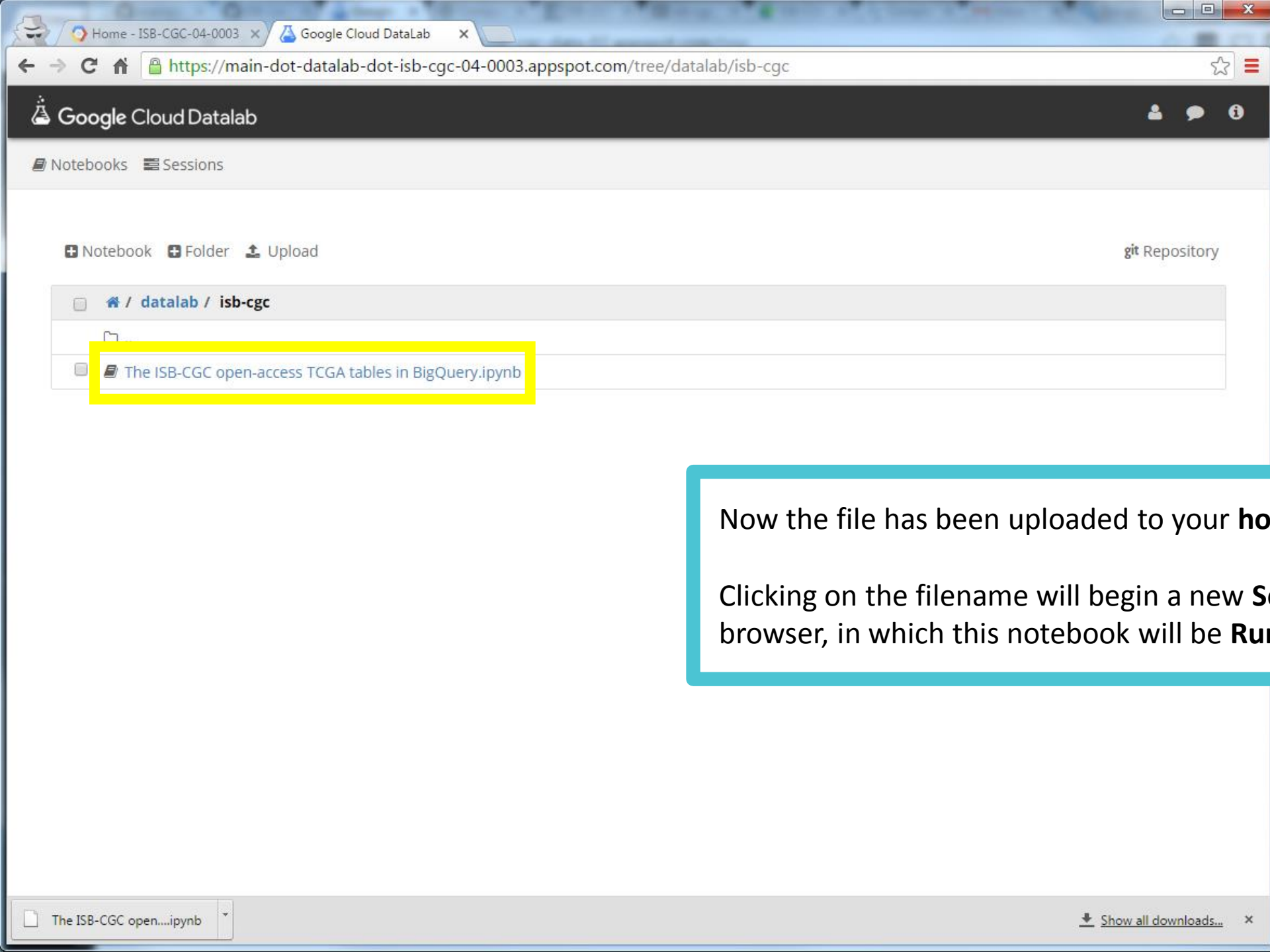
Once you've created the new folder, you can click on it to make it your current working directory.

Now click on the **Upload** button. A file-selection box will open up to allow you to browse to and find the ipynb file that you downloaded from GitHub. Select this file and click **Open**.



After you've selected the file (or files) that you want to upload, you will see a screen like this with each file listed and **Upload** and **Cancel** buttons for each.

Click **Upload**.



Now the file has been uploaded to your **home/datalab/isb-cgc** folder.

Clicking on the filename will begin a new **Session** in a new tab of your browser, in which this notebook will be **Running**.

Home - ISB-CGC-04-0003 x Google Cloud DataLab x The ISB-CGC open-access x

https://main-dot-datalab-dot-isb-cgc-04-0003.appspot.com/notebooks/datalab/isb-cgc/The%20ISB-CGC%20open-access%20TCGA%20

Google Cloud Datalab The ISB-CGC open-access TCGA tables in BigQuery (autosaved)

Notebook ▾

➕ Add Code ➕ Add Markdown 🗑 Delete ⬆ Move Up ⬇ Move Down ▶ Run ⌵ Clear ⌵ ○ Reset Session

Navigation Help

The ISB-CGC open-access TCGA tables in Big-Query

The goal of this notebook is to introduce you to a new publicly-available, open-access dataset in BigQuery. This set of BigQuery tables was produced by the [ISB-CGC](#) project, based on the open [Data Portal](#). You will need to have access to a Google Cloud Platform (GCP) account. If you don't already have one, you can sign up for a [free-trial](#) or contact [us](#) and we'll get you set up. This is the first phase of our Cancer Genomics Cloud pilot. (You can find more information [here](#).)

We are not attempting to provide a thorough BigQuery or IPython tutorial, as a good one already exists. Here are links to some resources that you might find useful:

- [BigQuery](#),
- the BigQuery [web UI](#) where you can run queries interactively,
- [IPython](#) (now known as [Jupyter](#)), and
- [Cloud Datalab](#) the recently announced interactive cloud-based platform developed on.

There are also many tutorials and samples available on github (see, in particular, the [Genomics](#) project).

In order to work with BigQuery, the first thing you need to do is import the `gcp.bigquery` module:

```
import gcp.bigquery as bq
```

The next thing you need to know is how to access the specific tables you want. Tables are organized into datasets, and datasets are owned by a specific GCP project. The tables in this notebook are in a dataset called `tcga_201510_alpha`, owned by the `isb-cgc` project. The form `<project_id>.<dataset_id>.<table_id>`. Let's start by getting information about this dataset:

The ISB-CGC open...ipynb

This is what a “Running Notebook Session” page looks like.

Take a look at the buttons across the top: you can

- **Add Code** (this will add a new “code cell” either at the bottom of the notebook, or below whichever cell your cursor is in)
- **Add Markdown** (this will add a new “markdown cell”)
- **Delete** (this will delete the current cell)
- **Move Up/Down** (you can also move around using the mouse)
- **Run** (clicking on Run will “run” your current cell, or you can use the pull-down to access three additional Run options)
- **Clear** (clicking on Clear will “clear” the outputs only of your current code cell, or you can use the pull-down to access Clear all Cells)
- **Reset Session** (this allows you to restart the current kernel – essentially you can “reboot” this notebook if you’re having problems).

To re-run or test a notebook, try “**Clear all Cells**” and then “**Run all Cells**”

Google Cloud Datalab The ISB-CGC open-access TCGA tables in BigQuery (unsaved changes)

Notebook

- Save
- Save copy
- Rename...
- Download
- Convert to HTML
- Convert to Python

CGC open-access TCGA tables in Big-Query

This notebook is to introduce you to a new publicly-available, open-access dataset in BigQuery. This set of tables was produced by the [ISB-CGC](#) project, based on the [TCGA](#) dataset. You will need to have access to a Google Cloud Platform project. If you already have one, you can sign up for a [free-trial](#) or contact your account manager. (You can find more information about our Cancer Genomics Cloud pilot.)

We are not attempting to provide a thorough BigQuery or IPython tutorial, but if you already have one, you can find it here.

- [BigQuery](#),
- the BigQuery [web UI](#) where you can run queries interactively,
- [IPython](#) (now known as [Jupyter](#)), and
- [Cloud Datalab](#) the recently announced interactive cloud-based environment developed on.

There are also many tutorials and samples available on github (see, for example, the [Genomics](#) project).

In order to work with BigQuery, the first thing you need to do is import the `gcp.bigquery` module:

```
import gcp.bigquery as bq
```

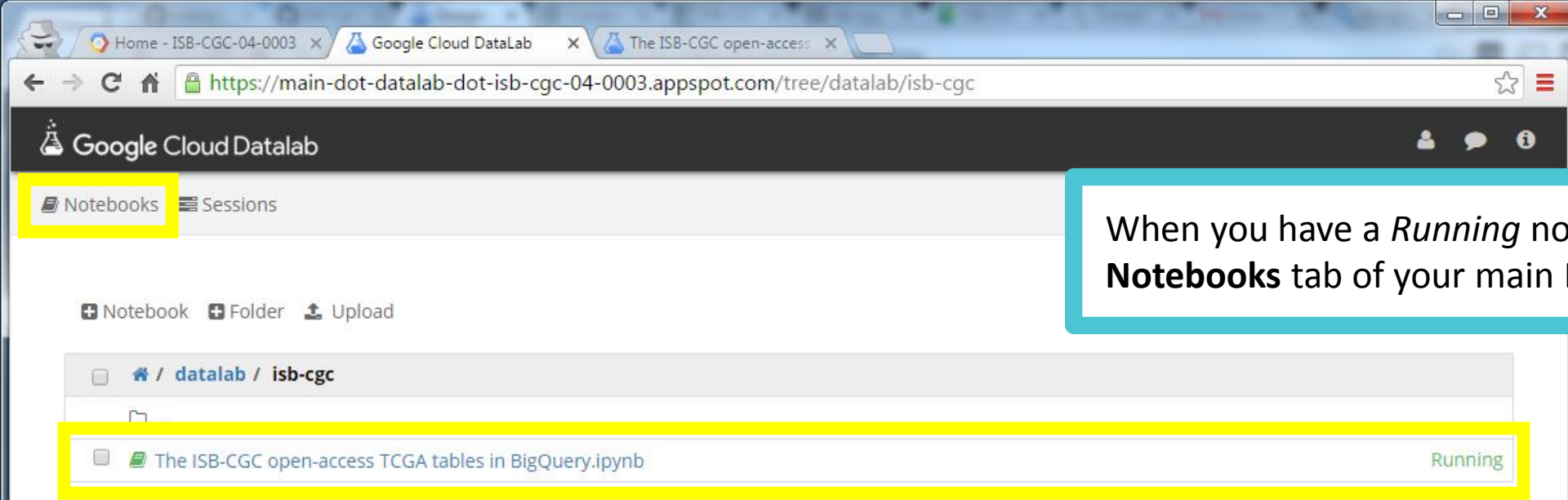
The next thing you need to know is how to access the specific tables you are interested in. BigQuery tables are organized into datasets, and datasets are owned by a specific GCP project. The tables we are introducing in this notebook are in a dataset called `tcga_201510_alpha`, owned by the `isb-cgc` project. A full table identifier is of the form `<project_id>:<dataset_id>.<table_id>`. Let's start by getting some basic information about the tables in this dataset:

documentation and sample notebooks is also a great way to check out how you can use Cloud Datalab.

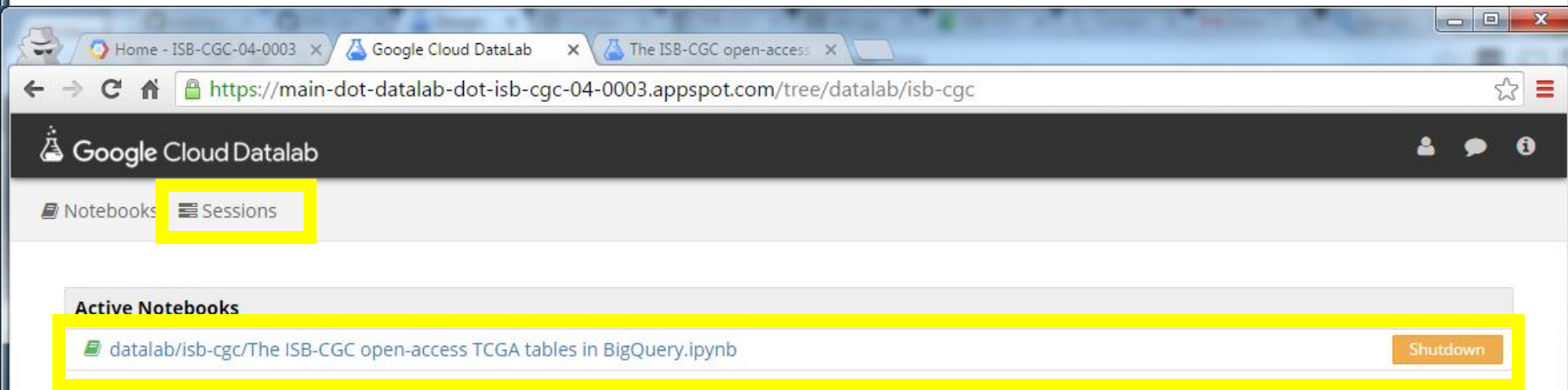
Cloud Datalab will automatically save your work every few minutes, but it's a good idea to double-check whether you have any unsaved changes before you leave this page, shutdown a session, or delete the Datalab VM.

In the top-most bar, next to the name of your current notebook, you will either see **(unsaved changes)** or **(autosaved)**. If you have unsaved changes, go to the **Notebook** pull-down, and select **Save**. This will save the current state of your notebook to your project's git Repository.

Also note the other options available to you in that pull-down: Save copy, Rename, Download, Convert to HTML, and Convert to Python.



When you have a *Running* notebook, this is what the **Notebooks** tab of your main Datalab page will look like.



And this is what the **Sessions** tab will look like.

Google Cloud Platform

App Engine Versions - ISB x Google Cloud Datalab x App Engine Versions - ISB x Google Cloud Datalab x The ISB-CGC open-access x

https://console.cloud.google.com/appengine/versions?project=isb-cgc-04-0003&moduleId=datalab

Google Cloud Platform

App Engine

Versions

REFRESH DELETE STOP START MIGRATE TRAFFIC SPLIT TRAFFIC

Service: datalab More Columns

<input checked="" type="checkbox"/> Version	Status	Traffic Allocation	Instances	Runtime	Environment	Size	Deployed
<input checked="" type="checkbox"/> main	Serving	100%	1 (Google-managed)	custom	Flexible		May 3, 2016, 9:24:41 AM by 901200507895-compute@developer.gservice

IMPORTANT!

Before you go away, it's important to **Delete** your instance of Datalab to avoid incurring further charges for an idle VM. As long as you have made sure that your work has been saved, you can delete the Datalab VM, and simply redeploy Datalab the next time you come back.

These instructions on how to Delete your Datalab VM instance are taken from the Datalab [quickstart](#) documentation:

Go to the [App Engine Versions](#) page in your project's **Cloud Platform Console**. Select datalab from the Service pull-down, then click the check-box next to Version main, and then click **DELETE**.

Home - ISB-CGC-04-0003 x

https://console.cloud.google.com/home/dashboard?p

Google Cloud Platform

Home

Dashboard

Activity

Project: ISB-CGC-04-0003

ID: isb-cgc-04-0003 (#901200507895)

Resources

App Engine
0 instances

Cloud Storage
4 buckets

Trace

Most frequent URIs

URI	50% latency	90% latency
/socket.io/	7 ms	25,049 ms

APIs

Requests (requests/sec)

0.1
0.08

Billing

\$13.48

Approximate charges so far this month

View detailed charges

News

Field of dreams: this week on Google Cloud Platform

2 days ago

Go to cloud status dashboard

Go to the App Engine dashboard

As you learn to use the Google Cloud please make a habit of shutting down or deleting VMs that you are not actively using. An idle VM costs as much per minute as one that is busy analyzing your data.

You can confirm that you are only using the Resources that you expect and intend to be using by checking your **Console Dashboard** page daily.

In particular, keep an eye on your **Resources** and **Billing** details. The Resources box will give you a total count of running VM instances and Storage buckets.

Cancer Genomics Cloud

What Next?

There are a wealth of additional resources available to you online, including for example this [Notebook Gallery](#) with links to the best IPython and Jupyter Notebooks.

The ISB-CGC platform includes an interactive [Web App](#), over a Petabyte of TCGA data in Google Genomics and Cloud Storage, and tutorials and code examples on [GitHub](#) to get you started.

Documentation for the [ISB-CGC](#) platform and [Google Genomics](#) can be found on readthedocs.