

DAY 1: EXPERIMENT

Not just coding it up, but all workflow stuff up to running it

1. Background: replicability and proper procedure
2. Workflow and organisation
3. Experiment design
4. Coding experiment
5. Ethics and pre-registration
6. Hosting experiment on a server
7. Running experiment
8. Downloading data

ETHICS

Not that much to say here, since everyone's ethics committees are different and the MTurk issues will be discussed later

Main points are simply:

1. Remember to get ethics approval before running anything!
2. Save it and your ethics documents
3. Trick of the trade (for minimal risk): describe the experiment abstractly enough to accommodate additional conditions without having to get additional approval
4. It's an annoying hoop but it's there for a reason, so as much as possible try to view it as an opportunity to make sure you are doing right by your participants as much as possible

PRE-REGISTRATION

Should you pre-register? Always?
How do you pre-register?

The “should” is more complicated, but really good to have a handle on because your reasons for doing it will shape what it is you are doing.

(WHEN) SHOULD YOU PRE-REGISTER?

Split up into groups of 2 or 3. See if you can generate some pros and cons (or caveats, or concerns) to pre-registration, which we'll then discuss as a group

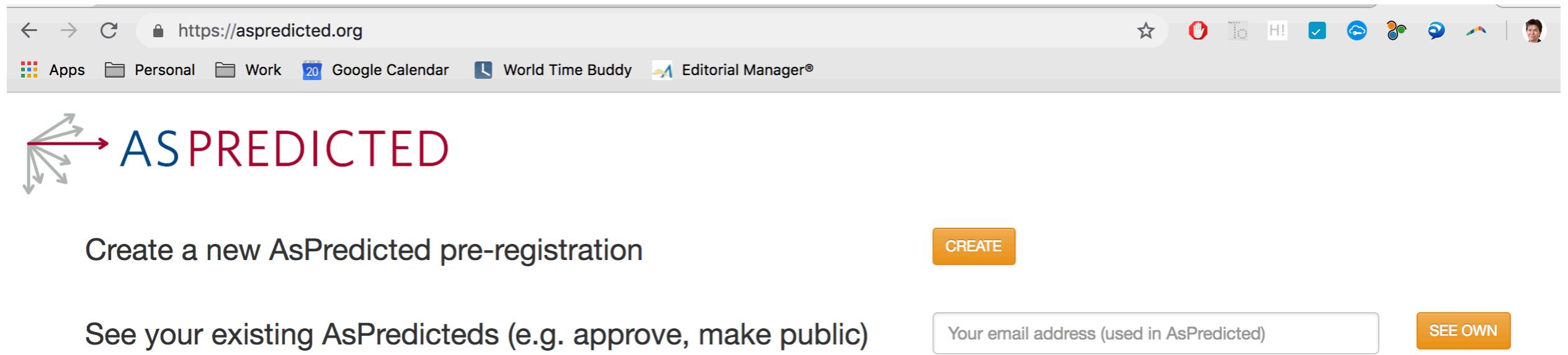
HOW DO YOU PRE-REGISTER?

Main goal: eliminate worries about p-hacking by removing researcher degrees of freedom (in the experimental design and the analysis): **not** to make a hypothesis that you end up supporting with your data

All of the details of pre-registration are oriented around achieving that aim

HOW DO YOU PRE-REGISTER?

<https://aspredicted.org>



The screenshot shows a web browser window with the URL <https://aspredicted.org> in the address bar. The browser's toolbar includes icons for back, forward, search, and various extensions. Below the address bar, the browser menu bar shows "Apps", "Personal", "Work", "Google Calendar", "World Time Buddy", and "Editorial Manager®". The main content area features the AsPredicted logo with a stylized arrow icon and the word "AS PREDICTED". Below the logo, there are two main calls-to-action: "Create a new AsPredicted pre-registration" with an orange "CREATE" button, and "See your existing AsPredicteds (e.g. approve, make public)" with a text input field for "Your email address (used in AsPredicted)" and an orange "SEE OWN" button.

AS PREDICTED

Create a new AsPredicted pre-registration

CREATE

See your existing AsPredicteds (e.g. approve, make public)

Your email address (used in AsPredicted)

SEE OWN

HOW DO YOU PRE-REGISTER?

How does it work?

- One author briefly answers 9 questions.
- All participating authors receive an email asking for approval.
- If everyone approves, it is saved and stays private until an author acts to make it public, or it remains private forever. ([Why?](#))
- Authors may share anonymous .pdf with reviewers.
- If made public, a single-page .pdf is generated. That document can be used as a supplement. ([See sample](#))
- The .pdf contains a unique URL that allows for one-click verification. That URL can be included in the paper.
- The .pdf is automatically stored in the web-archive. ([See sample](#))
- There are no accounts, userids, or passwords.

What if things don't go "as predicted"

You can just say so in the paper:

- "Contrary to expectations, we found that..."
- "Unexpectedly, we also found that..."
- "In addition to the analyses we pre-registered we also ran..."
- "We encountered an unexpected situation, and followed our Standard Operating Procedure" (.pdf)

THE 9 QUESTIONS

1. Data collection. Have any data been collected for this study already?
2. Hypothesis. What's the main question being asked or hypothesis being tested in the study?
3. Dependent variable. Describe the key dependent variable(s) specifying how they will be measured.
4. Conditions. How many and which conditions will participants be assigned to?
5. Analyses. Specify exactly which analyses you will conduct to examine the main question/hypotheses.
6. Outliers and exclusions. Describe exactly how outliers will be defined and handled, and your precise rule(s) for excluding observations.
7. Sample size. How many observations will be collected or what will determine sample size.
8. Other. Anything else you would like to pre-register?
9. Name. Give a title for this AsPredicted registration.

TIPS FOR PRE-REGISTERING

1. Don't provide too little information

Specify anything you need to specify so that a skeptic won't be able to point to that as a place that you could have been p-hacking: specific measures, sample size (or explanation), specific variables, specific tests, exclusion criteria

2. Don't provide too much information

Don't specify things that would go into a paper that are irrelevant to p-hacking concerns, like the theoretical justification of the study

TIPS FOR PRE-REGISTERING

Item in preregistration	Bad answer	What's wrong with it?	Good answer
Research Question or Hypothesis	Building on the work of Picasso (1901-1904), we hypothesized that....	You don't need reasons for asking the research question because they do not inform possible p-hacking. Just state the question or hypothesis of interest.	Question: Does sadness increase preference for the color blue?
Dependent variable	Preference for the color blue	This preference can be measured in many different ways so this statement underspecifies how it will be measured.	Participants will rate their liking for red, blue, orange, and purple on 7-point scales (1 = not at all; 7 = an extreme amount). Preference for blue will be defined as the difference between a participant's rating for blue and their average rating of the three non-blue colors.
Manipulations/Conditions	We will manipulate mood by having participants watch different videos.	This leaves room for cherry-picking from amongst a larger set of conditions. Specify the exact conditions and the exact manipulations.	Before rating their color preferences, participants will be randomly assigned to one of three conditions in which they watch a clip from either a sad video (<i>My Dog Skip</i>), a happy video (<i>Pitch Perfect</i>), or a neutral video (<i>Gone Curling</i>) .

TIPS FOR PRE-REGISTERING

Item in preregistration	<i>Bad answer</i>	<i>What's wrong with it?</i>	<i>Good answer</i>
Analyses	We will regress preference for the color blue on mood condition	There are many ways to run these analyses. For example, are you including covariates? How will "mood condition" be coded? If applicable, how will the standard errors be computed?	We will run an OLS regression predicting preference for the color blue with condition (coded 1 = sad video; 0 = happy or neutral video). We will control for gender (1 = male; 0 = female) in this analysis.
Outliers & Exclusions	We will exclude participants who are inattentive, and those who show an extreme preference for the color orange.	What counts as "inattentive"? What counts as "extreme preference for the color orange"? You must define these things.	We will exclude participants who fail at least two out of the three attention checks that we will include at the beginning of our study (before the manipulation). We will also exclude participants whose rating of orange is higher than 5 on the 7-point scale.
Sample size	We conducted a power analysis that showed that ... And so we decided to collect between 100 and 200 observations.	Your power analysis is irrelevant to whether you p-hacked; leave it out. Also, any sample size between 100 and 200 is consistent with this preregistration.	We will stop data collection once 150 participants have submitted a response on MTurk. Deviations from this goal are entirely due to MTurk software and outside of our control.

PRE-REGISTRATION

My opinion: It's almost always worth doing, and imposes a good discipline for figuring out exactly what your analyses will be before you do them. That said, I sometimes don't pre-register if it is completely exploratory research, and am just up front that it is exploratory. (And usually aim to replicate before publishing)

I also personally tend to think that small sample sizes and lack of replication are bigger problems in many ways than lack of pre-registration, and worry sometimes that the focus on pre-registration takes the focus off of these things. Pre-registration is a great idea usually, but it is *not* a panacea!

OKAY.. I'VE PRE-
REGISTERED. NOW
HOW DO I PUT IT UP?

THIS IS WHERE YOU NEED GOOGLE APP ENGINE

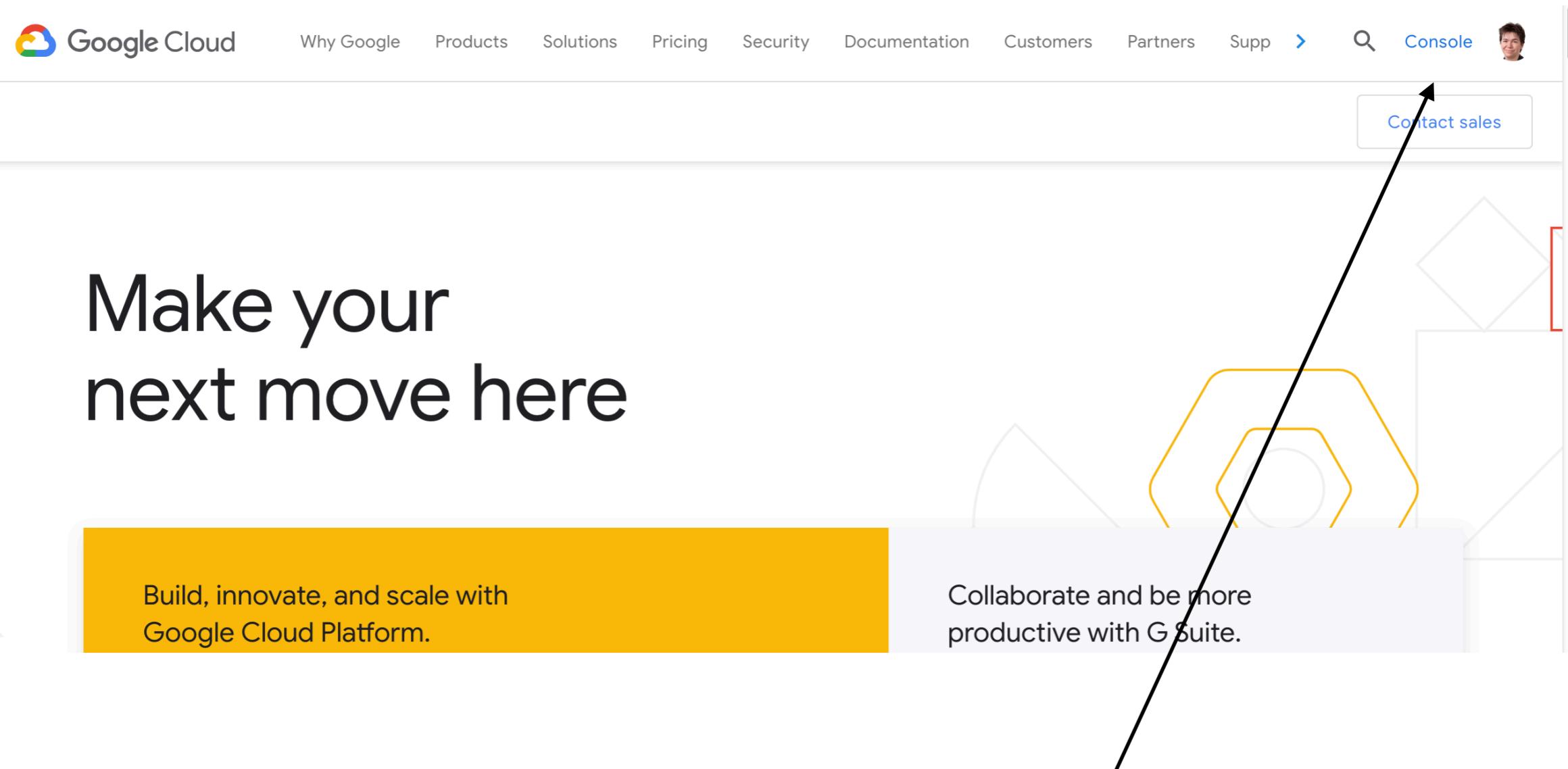
I hope you have succeeded in installing the Google App Engine SDK. If you have not, raise your hand.
(The rest of you, have fun playing around with javascript)

OUTLINE OF STEPS

1. Go to google cloud
2. Make a new project (or go into an existing one if you're just modifying an old one)
3. Open a terminal and go to the command line in the directory with the experiment
4. Initialise google cloud in that directory
5. Deploy the experiment so it shows up online.
6. Go to your project url.

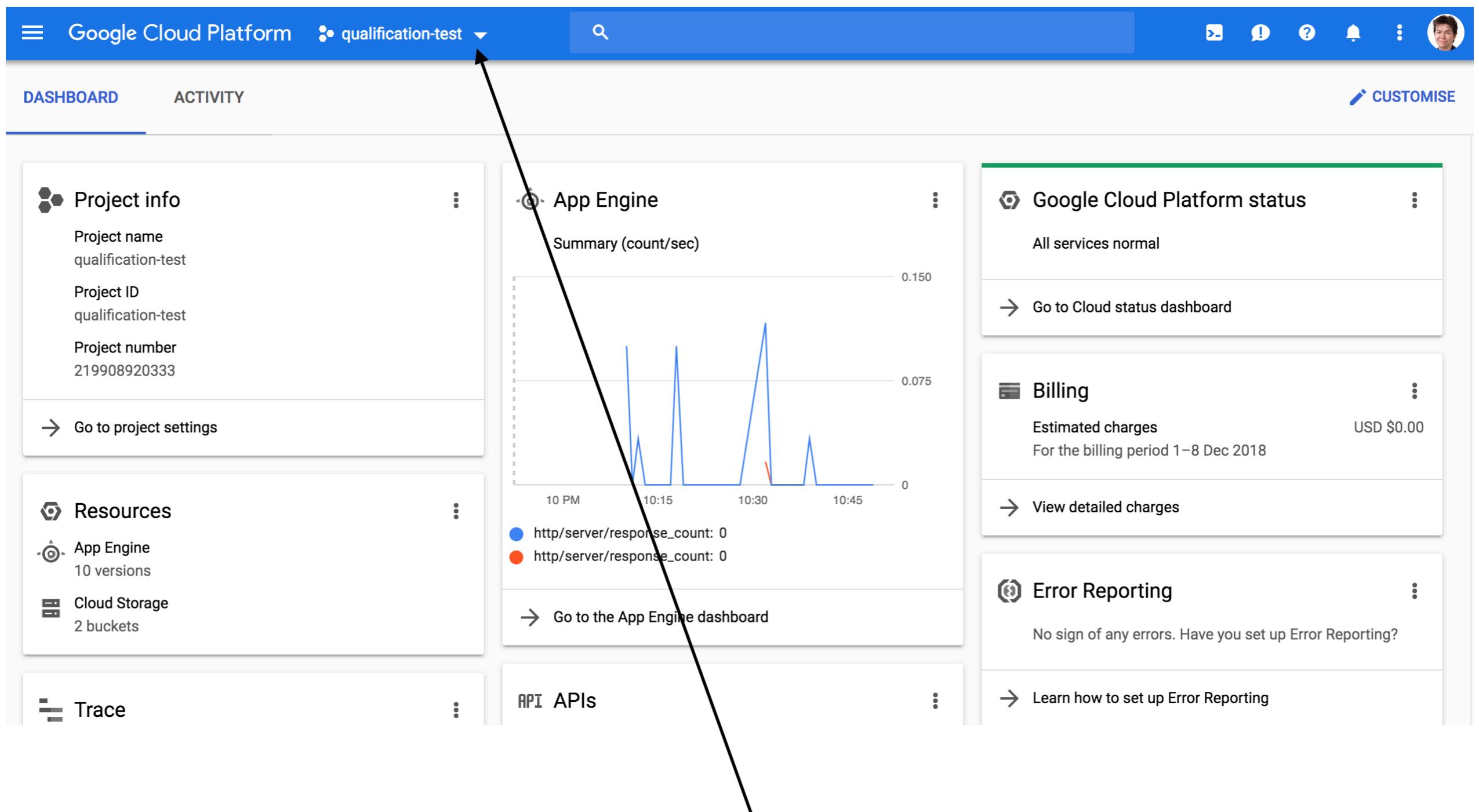
1. GO TO GOOGLE CLOUD

cloud.google.com



Go to console

2. CREATE PROJECT IN GOOGLE CLOUD



The screenshot shows the Google Cloud Platform dashboard for the project "qualification-test". A black arrow points from the text "This will list your projects" down to the project selection dropdown in the top navigation bar.

Google Cloud Platform qualification-test

DASHBOARD ACTIVITY CUSTOMISE

Project info

- Project name qualification-test
- Project ID qualification-test
- Project number 219908920333

[Go to project settings](#)

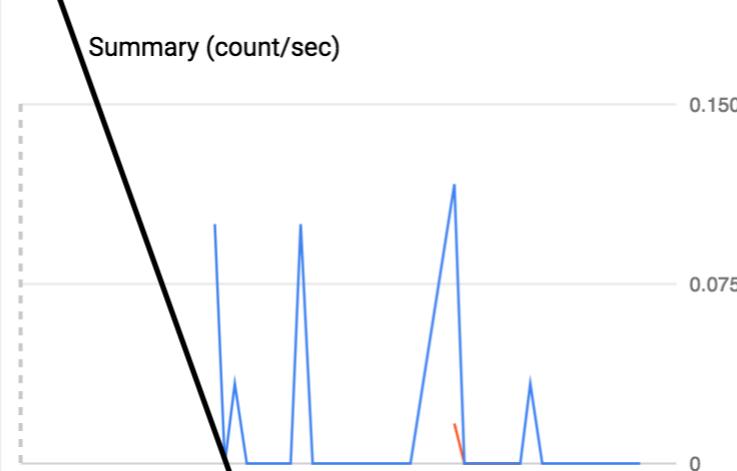
Resources

- App Engine 10 versions
- Cloud Storage 2 buckets

Trace

App Engine

Summary (count/sec)



- http/server/response_count: 0
- http/server/response_count: 0

[Go to the App Engine dashboard](#)

API APIs

Google Cloud Platform status

All services normal

[Go to Cloud status dashboard](#)

Billing

Estimated charges USD \$0.00
For the billing period 1–8 Dec 2018

[View detailed charges](#)

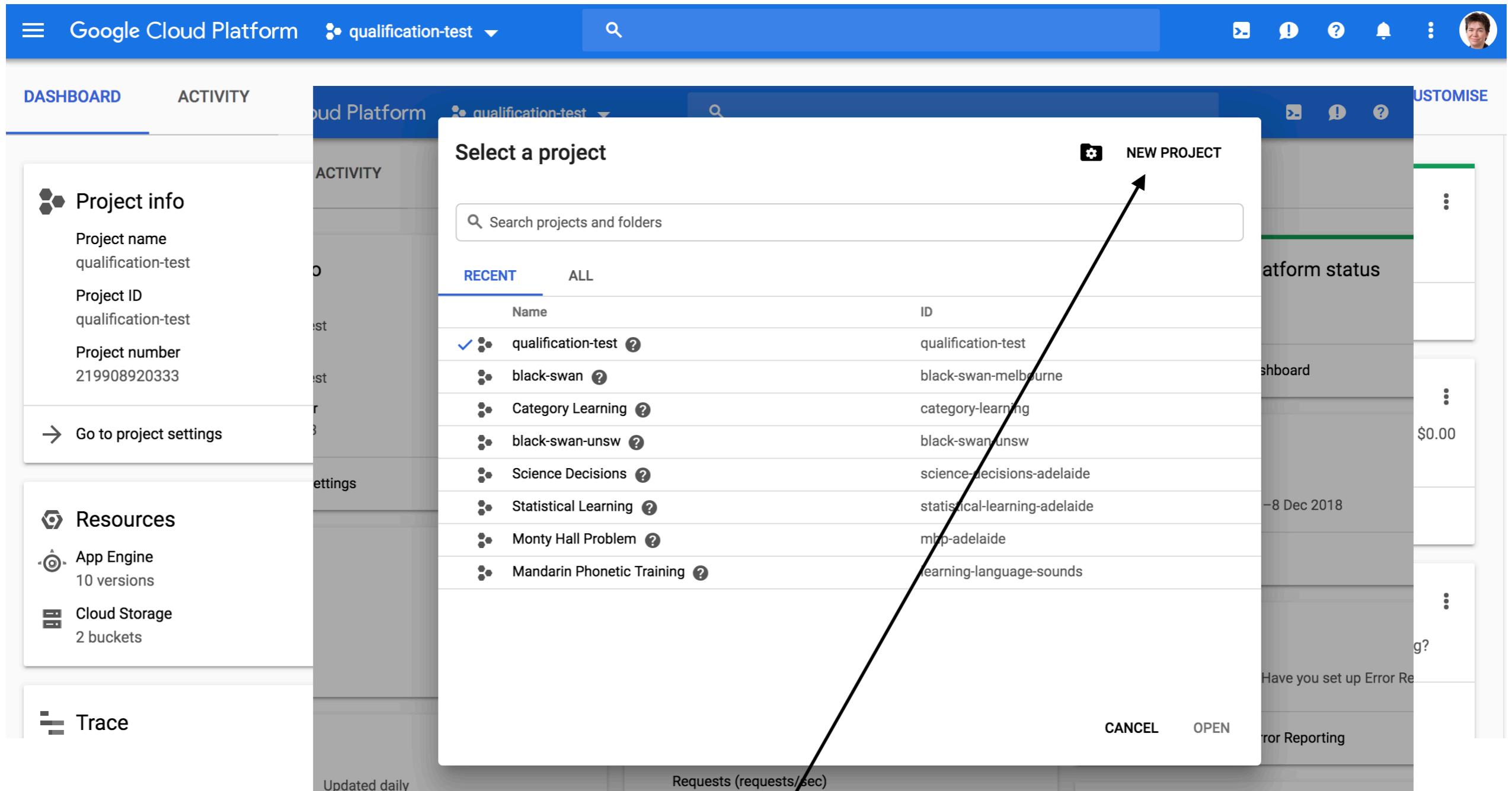
Error Reporting

No sign of any errors. Have you set up Error Reporting?

[Learn how to set up Error Reporting](#)

This will list your projects

2. CREATE PROJECT IN GOOGLE CLOUD



You can select “new project”

2. CREATE PROJECT IN GOOGLE CLOUD

Google Cloud Platform

New Project

You have 6 projects remaining in your quota. Request an increase or delete projects.

[Learn more](#)

[MANAGE QUOTAS](#)

Project Name *
My Project 86837

Project ID: linen-age-224911. It cannot be changed later. [EDIT](#)

Location *
No organisation [BROWSE](#)

Parent organisation or folder

CREATE **CANCEL**

This is what is going to show up in your url so try to name it something descriptive (for you) but that doesn't give away details you don't want to give away to the participants

2. CREATE PROJECT IN GOOGLE CLOUD

The screenshot shows the 'New Project' page in the Google Cloud Platform. At the top, there's a blue header bar with the 'Google Cloud Platform' logo and a search icon. Below the header, the title 'New Project' is displayed. A warning message in a light gray box says: '⚠ You have 6 projects remaining in your quota. Request an increase or delete projects.' It includes a 'Learn more' link and a 'MANAGE QUOTAS' button. The main form area starts with a 'Project Name *' field containing 'My Project 86837'. To the right of this field is a question mark icon. Below the name, it says 'Project ID: linen-age-224911. It cannot be changed later.' with an 'EDIT' link. Under 'Location *', the option 'No organisation' is selected, with a 'BROWSE' button to its right. A note below says 'Parent organisation or folder'. At the bottom left are 'CREATE' and 'CANCEL' buttons. On the right side, a 'Notifications' section shows a message: 'Create Project: chdssprojecttest1' followed by 'in 1 minute'. At the very bottom, there's a 'SEE ALL ACTIVITIES' link.

When you press
create you
should see
something like
this

2. CREATE PROJECT IN GOOGLE CLOUD

Select a project



NEW PROJECT

RECENT		ALL
Name		ID
✓	chdssprojecttest1	chdssprojecttest1
•	qualification-test	qualification-test
•	black-swan	black-swan-melbourne
•	Category Learning	category-learning
•	black-swan-unsw	black-swan-unsw
•	Science Decisions	science-decisions-adelaide
•	Statistical Learning	statistical-learning-adelaide
•	Monty Hall Problem	mhp-adelaide
•	Mandarin Phonetic Training	learning-language-sounds

Now select it
from your list

CANCEL OPEN

3. OPEN TERMINAL AND GO TO COMMAND LINE DIRECTORY WITH YOUR EXPERIMENT

Mac

1. Open terminal (in Applications - Utilities)
2. Go to your folder using `cd` command (`ls` to show contents of directory).
This folder is the one with your python scripts and `index.html` in it.
`cd Documents/teaching/2018/.../experiment/code/`

Windows

1. Open terminal (Start - then type `cmd` at the Search/Run line)
2. Unlike Mac, Windows must pass the full path of the script to the Python interpreter. If your interpreter is in the `C:\Python27` folder you would type:
`C:\Python27\python.exe C:\Users\Username\Desktop\...\experiment\code\serveit.py 8000`

4. INITIALISE GOOGLE CLOUD IN THAT DIRECTORY

```
Amys-MBP-3:code amy$ gcloud init
```

Welcome! This command will take you through the configuration of gcloud.

Settings from your current configuration [default] are:

core:

```
account: perfors@gmail.com
disable_usage_reporting: 'True'
project: black-swan-melbourne
```

Pick configuration to use:

- [1] Re-initialize this configuration [default] with new settings
 - [2] Create a new configuration
 - [3] Switch to and re-initialize existing configuration: [qualification-test]
- Please enter your numeric choice: 1

4. INITIALISE GOOGLE CLOUD IN THAT DIRECTORY

Your current configuration has been set to: [default]

You can skip diagnostics next time by using the following flag:

```
gcloud init --skip-diagnostics
```

Network diagnostic detects and fixes local network connection issues.

Checking network connection...done.

Reachability Check passed.

Network diagnostic (1/1 checks) passed.

Choose the account you would like to use to perform operations for this configuration:

[1] perfors@gmail.com

[2] Log in with a new account

Please enter your numeric choice: 1

4. INITIALISE GOOGLE CLOUD IN THAT DIRECTORY

Pick cloud project to use:

- [1] adelaide-expert-study
- [2] adelaide-expert-studya
- [3] adelaide-expert-studyb
- [4] adelaide-label-learning
- [5] black-swan-melbourne
- [6] black-swan-unsw
- [7] category-learning
- [8] chdssprojecttest1
- [9] choices-experiment
- [10] choices-inference
- [11] expert-choices-expt
- [12] hazel-scenarios
- [13] label-learning-adelaide
- [14] language-study-adelaide
- [15] learn-new-language-adelaide
- [16] learning-language-sounds
- [17] mhp-adelaide
- [18] qualification-test
- [19] science-decisions-adelaide
- [20] statistical-learning-adelaide
- [21] Create a new project

Please enter numeric choice or text value (must exactly match list item): 8

4. INITIALISE GOOGLE CLOUD IN THAT DIRECTORY

Your current project has been set to: [chdssprojecttest1].

Not setting default zone/region (this feature makes it easier to use [gcloud compute] by setting an appropriate default value for the --zone and --region flag). See <https://cloud.google.com/compute/docs/gcloud-compute> section on how to set default compute region and zone manually. If you would like [gcloud init] to be able to do this for you the next time you run it, make sure the Compute Engine API is enabled for your project on the <https://console.developers.google.com/apis> page.

Your Google Cloud SDK is configured and ready to use!

- * Commands that require authentication will use perfors@gmail.com by default
 - * Commands will reference project `chdssprojecttest1` by default
- Run `gcloud help config` to learn how to change individual settings

This gcloud configuration is called [default]. You can create additional configurations if you work with multiple accounts and/or projects. Run `gcloud topic configurations` to learn more.

Some things to try next:

- * Run `gcloud --help` to see the Cloud Platform services you can interact with. And run `gcloud help COMMAND` to get help on any gcloud command.
- * Run `gcloud topic -h` to learn about advanced features of the SDK like arg files and output formatting

Updates are available for some Cloud SDK components. To install them, please run:

```
$ gcloud components update
```

Amys-MBP-3:code amy\$

5. DEPLOY THE EXPERIMENT

Annoying thing to do first: Remove
the application and version fields
from app.yaml

Download new versions of
backend.py and **backend.pyc**
and write over existing ones

5. DEPLOY THE EXPERIMENT

```
Amys-MBP-3:code amy$ gcloud app deploy
```

You are creating an app for project [chdssprojecttest1]. WARNING: Creating an App Engine application for a project is irreversible and the region cannot be changed. More information about regions is at <<https://cloud.google.com/appengine/docs/locations>>.

Please choose the region where you want your App Engine application located:

- [1] us-west2 (supports standard and flexible)
- [2] us-central (supports standard and flexible)
- [3] europe-west (supports standard and flexible)
- [4] europe-west3 (supports standard and flexible)
- [5] asia-east2 (supports standard and flexible)
- [6] europe-west2 (supports standard and flexible)
- [7] us-east1 (supports standard and flexible)
- [8] us-east4 (supports standard and flexible)
- [9] asia-northeast1 (supports standard and flexible)
- [10] asia-south1 (supports standard and flexible)
- [11] australia-southeast1 (supports standard and flexible)
- [12] southamerica-east1 (supports standard and flexible)
- [13] northamerica-northeast1 (supports standard and flexible)
- [14] cancel

Please enter your numeric choice: 2

I usually choose a region near my participants to minimise lag

5. DEPLOY THE EXPERIMENT

Creating App Engine application in project [chdssprojecttest1] and region [us-central]....done.
Services to deploy:

```
descriptor: [/Users/amy/Documents/teaching/2018/summerschool/chdss2018/day1_experiments/  
experiment/code/app.yaml]  
source: [/Users/amy/Documents/teaching/2018/summerschool/chdss2018/day1_experiments/  
experiment/code]  
target project: [chdssprojecttest1]  
target service: [default]  
target version: [20181208t231304]  
target url: [https://chdssprojecttest1.appspot.com]
```

Do you want to continue (Y/n)? **Y**

5. DEPLOY THE EXPERIMENT

Beginning deployment of service [default]...

Some files were skipped. Pass `--verbosity=info` to see which ones.

You may also view the gcloud log file, found at

[/Users/amy/.config/gcloud/logs/2018.12.08/23.10.17.880726.log].

```
Uploading 89 files to Google Cloud Storage
```



File upload done.

Updating service [default]...done.

Setting traffic split for service [default]...done.

Deployed service [default] to [https://chdssprojecttest1.appspot.com]

You can stream logs from the command line by running:

```
$ gcloud app logs tail -s default
```

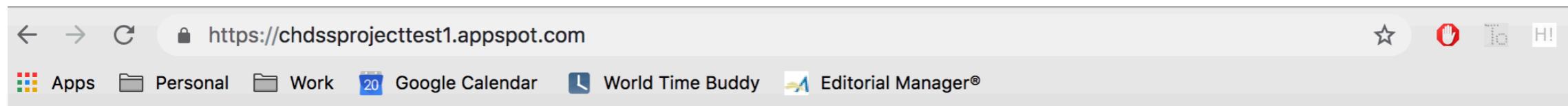
To view your application in the web browser run:

```
$ gcloud app browse
```

```
Amys-MBP-3:code amy$
```

There is your url!

6. GO TO YOUR PROJECT URL



UNSW Computational Cognitive Science

Thanks for accepting the HIT. "**The Spheres of Sodor**" is a short psychological study investigating how people make decisions. It involves the following steps:

1. We ask for demographic information (not connected to your Amazon ID)
2. Because this is a University research project, we ask for your informed consent. (The format of the consent form is a standard university document, so it sometimes looks a little weird on MTurk)
3. The study then explains how to do the task in detail. You will need to pass a short test to check that you understand how the study works.
4. Next comes the experiment itself.
5. At the end, we'll give you the completion code you need to get paid for the HIT.

The total time taken should be about 5 minutes. Please don't use the "back" button on your browser or close the window until you reach the end and receive your completion code. This is very likely to break the experiment and may make it difficult for you to get paid. However, if something does go wrong, please contact us! When you're ready to begin, click on the "start" button below.

[Start!](#)

WHAT IF YOU MAKE CHANGES?

Just re-initialise and redeploy!

`gcloud init`

`gcloud app deploy`

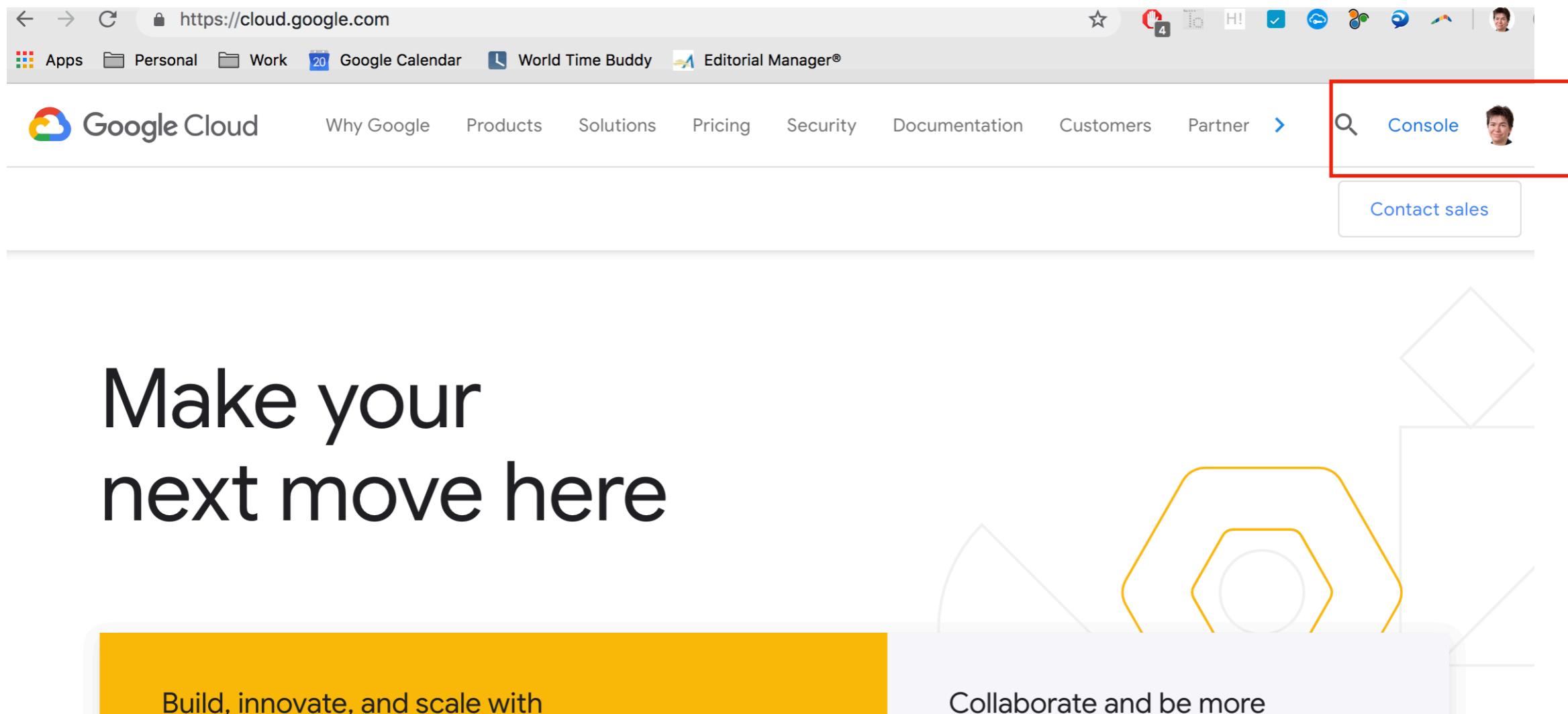
DAY 1: EXPERIMENT

Not just coding it up, but all workflow stuff up to running it

1. Background: replicability and proper procedure
2. Workflow and organisation
3. Experiment design
4. Coding experiment
5. Ethics and pre-registration
6. Hosting experiment on a server
7. Downloading data and wrap-up

VIEW YOUR DATA

You can see your data coming in by looking in the google cloud console online



The screenshot shows the Google Cloud homepage at https://cloud.google.com. The top navigation bar includes links for Apps, Personal, Work, Google Calendar, World Time Buddy, Editorial Manager®, Why Google, Products, Solutions, Pricing, Security, Documentation, Customers, Partner, and a search bar labeled "Console". A red box highlights the "Console" link in the search bar. Below the search bar is a "Contact sales" button. The main content area features the text "Make your next move here" and two buttons: "Build, innovate, and scale with" (yellow background) and "Collaborate and be more" (light gray background). The background of the page features abstract geometric shapes.

VIEW YOUR DATA

The screenshot shows the Google Cloud Platform dashboard for the project "chdssprojecttest1". The left sidebar lists services under "COMPUTE" (App Engine, Compute Engine, Kubernetes Engine, Cloud Functions) and "STORAGE" (Bigtable, Datastore, Firestore, Storage, SQL). A red box highlights the "Datastore" section, which contains three sub-options: "Entities", "Dashboard", and "Indexes". The "Dashboard" option is selected and has a red border around it. A callout box with a dashed border points to this selection with the text: "Pins appear here". On the right side of the dashboard, there are several cards: "App Engine" (Summary chart showing requests/sec), "Billing" (estimated charges for the billing period 1–9), "Error Reporting" (no errors found), and "APIs" (Requests (requests/sec)). A large text box on the right side of the dashboard area contains the message: "The data is kept in your datastore (make sure you're in the right project!)".

https://console.cloud.google.com/datastore/stats?project=chdssprojecttest1

← → ⌂ https://console.cloud.google.com/home/dashboard?project=chdssprojecttest1&_ga=2.192909674.-1936... ☆ 1 H! ↗

Apps Personal Work Google Calendar World Time Buddy Editorial Manager®

≡ Google Cloud Platform chdssprojecttest1 ⚙

Home

Pins appear here ? ×

COMPUTE

- App Engine >
- Compute Engine >
- Kubernetes Engine >
- Cloud Functions >

STORAGE

- Bigtable
- Datastore > Entities
- Firestore > Dashboard
- Storage > Indexes
- SQL > Admin

App Engine

Summary (count/sec)

All services normal

Go to Cloud status dashboard

Billing

Estimated charges
For the billing period 1–9

Error Reporting

No sign of any errors. Have Reporting?

APIs

Requests (requests/sec)

0.0175

The data is kept in your datastore
(make sure you're in the right project!)

Go to the App Engine dashboard

Learn how to set up Error Reporting

News

VIEW YOUR DATA

The screenshot shows the Google Cloud Platform Datastore interface. At the top, there's a blue header bar with the Google Cloud Platform logo, the project name "chdssprojecttest1", a search icon, and three small icons for notifications, help, and more.

The main area has a sidebar on the left with icons for Datastore, Dashboard, Indexes, and Admin. The main content area is titled "Entities". It has two tabs: "QUERY BY KIND" (which is selected) and "QUERY BY GQL". Below these tabs, there's a "Kind" dropdown set to "DataObject" and a "FILTER ENTITIES" button. A table below lists one entity:

	Name/ID ↑	content	date
<input type="checkbox"/>	id=5629499534213120	"rt","stimulus","button_pressed","trial_type","tri...	2018-12-10 (08:55:41.046) AEDT

DOWNLOADING DATA IS EASY

Go to your url and add on /info— it will automatically download a csv file called results

SOB

DOWNLOADING DATA IS EASY

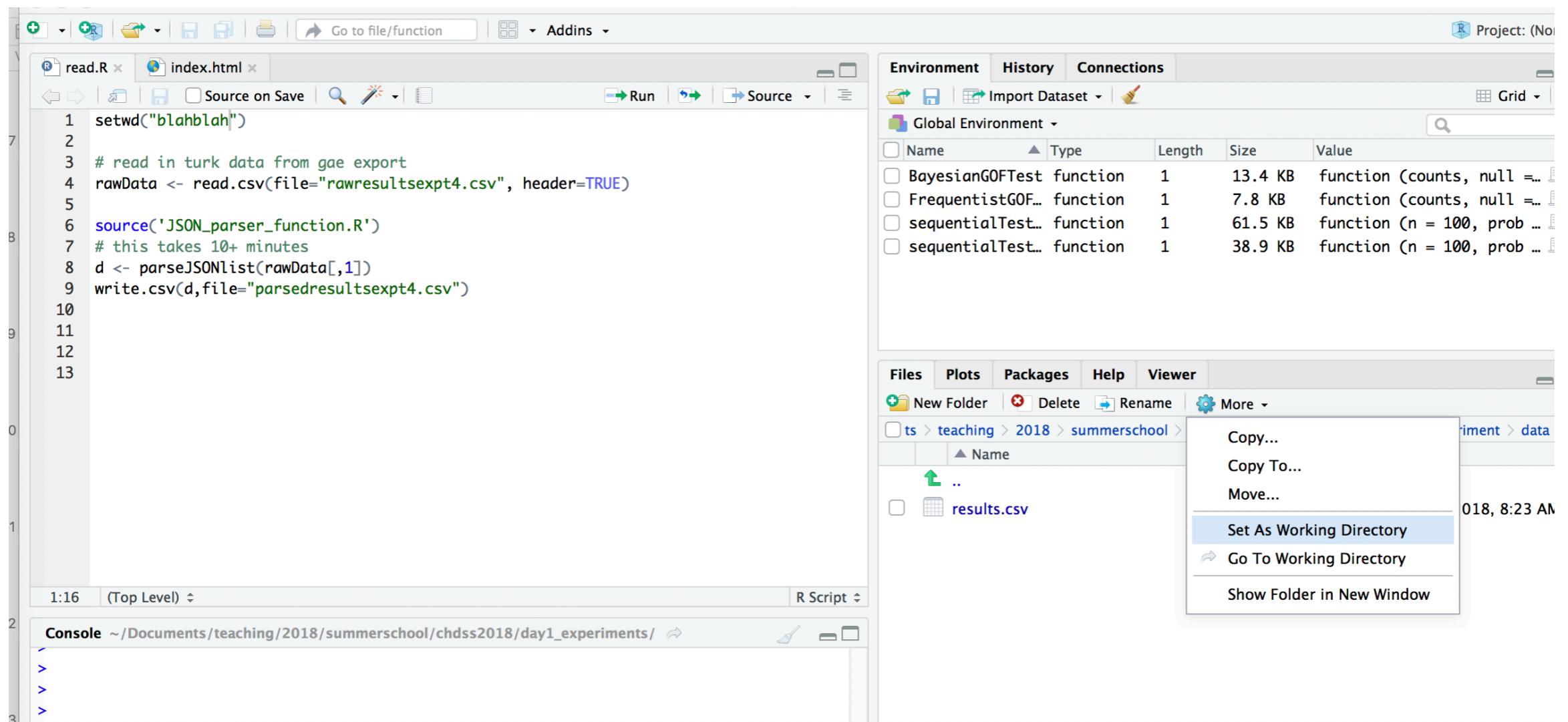
Go to your url and add on /info— it will automatically download a csv file called **results**

This file is in json format, not one that you can use, but the **read.R** script and the **JSON_parser_function.R** script (both of which you can download from the resources section) will convert it to a proper csv file

I put them in the **experiments/data** directory, along with the **results.csv** file.
Then, open **read.R**.

OPENING READ.R

First, set your working directory to the one with read.R and your results file in it



OPENING READ.R

Put that into your script (there is a better way to do this which Dani will cover tomorrow, but for now, let's do this)

The screenshot shows the RStudio interface with the following components:

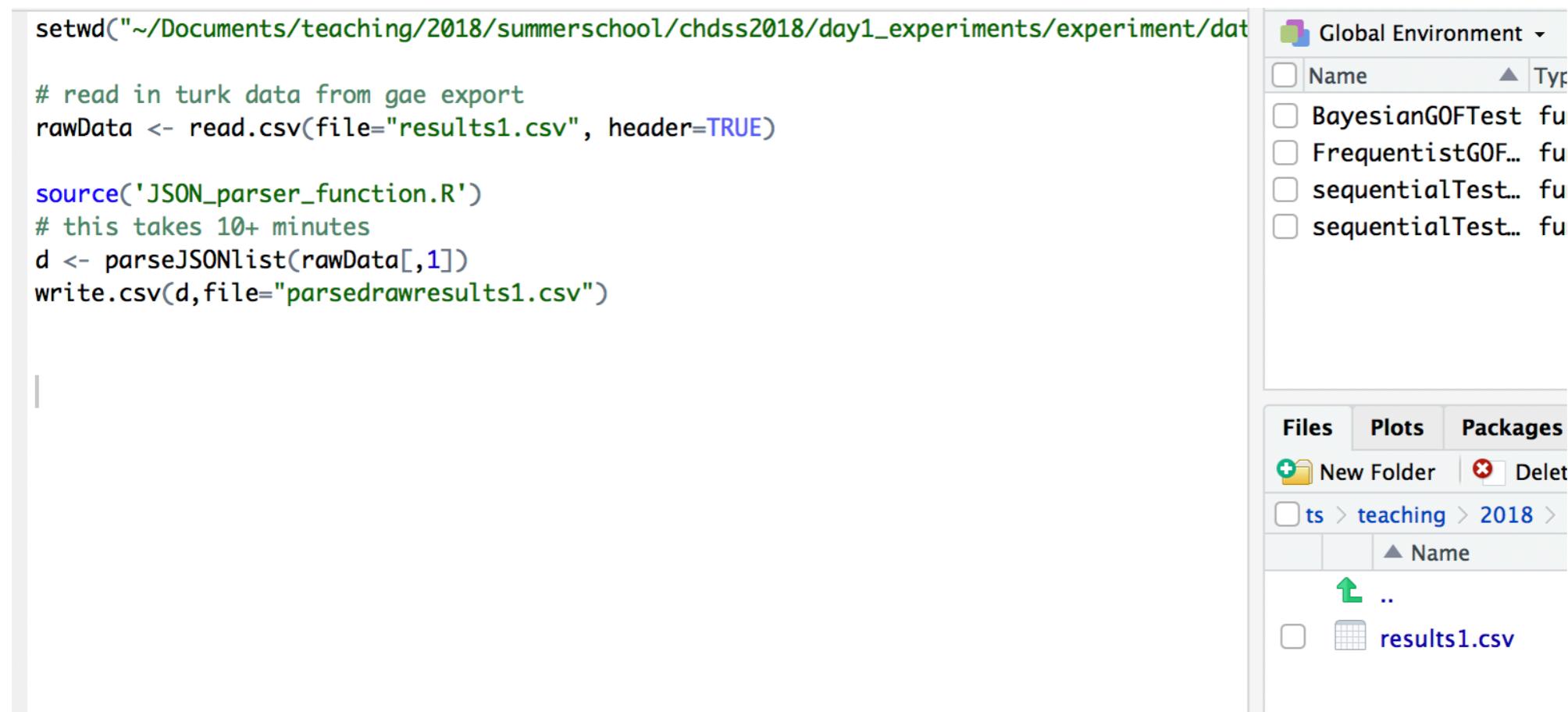
- Top Bar:** Includes tabs for "read.R" and "index.html", a "Go to file/function" search bar, and an "Addins" dropdown.
- Left Panel:** Displays the content of the "read.R" script. The code reads a CSV file, sources a JSON parser function, and writes a new CSV file.
- Environment Tab:** Shows the global environment with four functions listed:

Name	Type	Length	Size	Value
BayesianGOFTest	function	1	13.4 KB	func
FrequentistGOF...	function	1	7.8 KB	func
sequentialTest...	function	1	61.5 KB	func
sequentialTest...	function	1	38.9 KB	func

- Files Tab:** Shows a directory tree under "teaching": "ts > teaching > 2018 > summerschool > chdss2018 > day1_experiments". A file named "results.csv" is listed with a size of 2 B.
- Console Tab:** Displays the command "R Script" and the path "Console ~/Documents/teaching/2018/summerschool/chdss2018/day1_experiments/experiment/data/".

OPENING READ.R

Change the names of your files as appropriate — a good suggestion is to rename `results.csv` to `results1.csv` to cover multiple runs of the same project, and your output to be `parsedrawresults1.csv` or something like that



The screenshot shows the RStudio interface. On the left is the script editor containing R code. On the right are two panes: the Global Environment and the Files browser.

Script Editor (Top Left):

```
setwd("~/Documents/teaching/2018/summerschool/chdss2018/day1_experiments/experiment/data")
# read in turk data from gae export
rawData <- read.csv(file="results1.csv", header=TRUE)

source('JSON_parser_function.R')
# this takes 10+ minutes
d <- parseJSONlist(rawData[,1])
write.csv(d,file="parsedrawresults1.csv")
```

Global Environment (Top Right):

Name	Type
BayesianGOFTest	fu
FrequentistGOF...	fu
sequentialTest..	fu
sequentialTest..	fu

Files Browser (Bottom Right):

Files		Plots	Packages
<input type="checkbox"/>	New Folder	<input checked="" type="checkbox"/>	Delete
ts > teaching > 2018 >			
Name			
<input type="checkbox"/>	...	<input type="checkbox"/>	results1.csv

OPENING READ.R

Source, and you will have a readable csv file
with all of your data!

All ready for the real fun to begin.