



COM5940: NEW MEDIA BUSINESS MODEL & INNOVATION REST API, USER AUTHENTICATION, AUTHORISATION & ENCRYPTION

Bernard Suen

Center for Entrepreneurship

Chinese University of Hong Kong



Center for
Entrepreneurship

Today's Agenda

1. Setting Up MySQL Database on PythonAnywhere.
2. What is RESTful API? Why is it so important?
3. Leverage Airtable RESTful API for Data Usage.
4. Understand HTTP Status Codes.
5. Running RESTful API Test in Postman/JavaScript.
6. Authentication, Authorisation and Encryption.
7. Cookies, sessionStorage, and localStorage.
8. Demo of Airtable as tool for SCRUM project management.

MySQL
on
PythonAnywhere

The RESTful API

Airtable
for
Agile PM

Setting Up MySQL on PythonAnywhere

Your MySQL password has been changed.

X

MySQL

Postgres

MySQL settings

Connecting:

Use these settings in your web applications.

Database host address: bernardsys.mysql.pythonanywhere-services.com
Username: bernardsys

Your databases:

Click a database's name to start a MySQL console logged in to it.

Start a console on: bernardsys\$default

Create a database

Your database names always start with your username + "\$". There's no need to type that prefix in below, though: PythonAnywhere will automatically add it.

Database name:

Enter new database name

Create

MySQL password:

This should be different to your main PythonAnywhere password, because it is likely to appear in plain text in any web applications you write.

New password:

Confirm password:

Set MySQL password

Your MySQL password has been changed.

x

MySQL

Postgres

MySQL settings

Connecting:

Use these settings in your web applications.

Database host address: bernardsys.mysql.pythonanywhere-services.com

Username: bernardsys

Your databases:

Click a database's name to start a MySQL console logged in to it.

Start a console on: bernardsys\$default

Create a database

Your database names always start with your username + "\$". There's no need to type that prefix in below, though: PythonAnywhere will automatically add it.

Database name:

demo

Create

MySQL password:

This should be different to your main PythonAnywhere password, because it is likely to appear in plain text in any web applications you write.

New password:

.....

▼

Confirm password:

.....

Set MySQL password

MySQL

Postgres

MySQL settings

Connecting:

Use these settings in your web applications.

Database host address: bernardsys.mysql.pythonanywhere-services.com
Username: bernardsys

Your databases:

Click a database's name to start a MySQL console logged in to it.

Start a console on: bernardsys\$default
Start a console on: bernardsys\$demo

Create a database

Your database names always start with your username + "\$". There's no need to type that prefix in below, though: PythonAnywhere will automatically add it.

Database name:

Create

MySQL password:

This should be different to your main PythonAnywhere password, because it is likely to appear in plain text in any web applications you write.

New password:

Confirm password:

Set MySQL password

MySQL

Postgres

MySQL settings

Connecting:

Use these settings in your web applications.

Database host address: bernardsys.mysql.pythonanywhere-services.com
Username: bernardsys

Your databases:

Click a database's name to start a MySQL console logged in to it.

Start a console on: bernardsys\$default

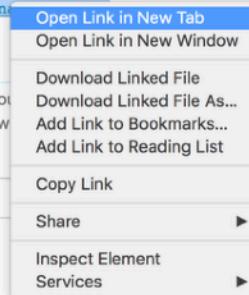
Start a console on: bernard

Create a database

Your database names always start with your prefix in below, though: PythonAnywhere w

Database name:

Enter new database name

Create

I to type that

MySQL password:

This should be different to your main PythonAnywhere password, because it is likely to appear in plain text in any web applications you write.

New password:

Confirm password:

Set MySQL password

 MySQL: bernardsys\$demo

Loading console...
Preparing execution environment... OK
Reticulating splines... OK
Loading Mysql interpreter... |

 Share with others



 MySQL: bernardsys\$demo

 Share with others



Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 13215630

Server version: 5.7.21-log MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> |

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 13215739

Server version: 5.7.21-log MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| bernardsys$default |
| bernardsys$demo   |
+-----+
3 rows in set (0.21 sec)
```

```
mysql> use bernardsys$demo;
```

 MySQL: bernardsys\$demo

 Share with others



Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 13215739

Server version: 5.7.21-log MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| bernardsys$default |
| bernardsys$demo |
+-----+
3 rows in set (0.21 sec)
```

```
mysql> use bernardsys$demo;
Database changed
mysql> |
```

Uploading SQL file to MySQL server on PythonAnywhere

Open Refine Import

HELP ? 🔔



Product

Roll-up

Venues

Users

sw_product



SHARE

BLOCKS

Hide fields

Filter

Group

Sort

Color



...

Rename view

Duplicate view

Copy another view's configuration

Download CSV

Print view

Delete view

	Name	Lat	Lng	url		
1	North Point	22.287111	114.191667	https://en.wikipedia.org/w...		https://dl.airtable.com/.attachments/e685bbc78
2	Mong Kok	22.322500	114.170556	https://en.wikipedia.org/w...		https://dl.airtable.com/.attachments/7fd289e11d
3	Happy Valley	22.266667	114.183333	https://en.wikipedia.org/w...		https://dl.airtable.com/.attachments/018c45cd6
4	Victoria Peak	22.275469	114.143828	https://en.wikipedia.org/w...		https://dl.airtable.com/.attachments/10ffd68809
5	Lan Kwai Fong	22.280972	114.155528	https://en.wikipedia.org/w...		https://dl.airtable.com/.attachments/9efc1776d5
6	Choi Hung	22.334484	114.210024	https://en.wikipedia.org/w...		https://dl.airtable.com/.attachments/15dbc608c
7	HKBU	22.338936	114.181953	https://en.wikipedia.org/w...		https://dl.airtable.com/.attachments/877503bcb
8	CUHK	22.418498	114.204074	https://en.wikipedia.org/w...		https://dl.airtable.com/.attachments/695430430
9	HKU	22.284167	114.137778	https://en.wikipedia.org/w...		https://dl.airtable.com/.attachments/f8e3c9c3ae
10	HK Science Musuem	22.301000	114.177655	https://en.wikipedia.org/w...		https://dl.airtable.com/.attachments/22f383737e
11	HK Cultural Center	22.293850	114.170323	https://en.wikipedia.org/w...		https://dl.airtable.com/.attachments/792c15887
12	Victoria Harbour	22.287753	114.173619	https://en.wikipedia.org/w...		https://dl.airtable.com/.attachments/46d727c49
+						

localhost/phpMyAdmin/tbl_structure.php?db=abc&table=venues&token=ec029d4aa76b525ac4d69abe302b324e&goto=tbl_structure.php&back=tbl_struct...

phpMyAdmin

Server: localhost:3306 » Database: test

Structure SQL Search Query Export Import Operations Privileges Routines More

	Table	Action						Rows	Type	Collation	Size		
<input type="checkbox"/>	book		Browse	Structure	Search	Insert	Empty	Drop	3	InnoDB	latin1_swedish_ci	16 Kib	
<input type="checkbox"/>	data		Browse	Structure	Search	Insert	Empty	Drop	3	InnoDB	latin1_swedish_ci	16 Kib	
<input type="checkbox"/>	users		Browse	Structure	Search	Insert	Empty	Drop	8	MyISAM	utf8_general_ci	2.5 Kib	
3 tables										Sum			
										14	InnoDB	utf8_unicode_ci	34.5 Kib

Check All With selected:

[Print view](#) [Data Dictionary](#)

* **Create table**

Name: Number of columns: Go

Console



Home



Favorites

Recent

Favorites

Filter databases by name

Server: localhost:3306 » Database: test

Structure

SQL

Search

Query

Export

Import

Operations

Privileges

Routines

More

Importing into the database "test"

File to Import:

File may be compressed (gzip, bzip2, zip) or uncompressed.

A compressed file's name must end in **.[format].[compression]**. Example: **.sql.zip**

Browse your computer: No file chosen (Max: 32MiB)

You may also drag and drop a file on any page.

Character set of the file:

Partial Import:

Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. (*This might be a good way to import large files, however it can break transactions.*)

Skip this number of queries (for SQL) or lines (for other formats), starting from the first one:

Format:

Format-Specific Options:

SQL compatibility mode:

[Recent](#)[Favorites](#)Filter databases by name X

suenlab1_6d8

test
 New
 book
 data
 users

users

webzash

willey

wp52a

wp475_c

wp498a

wp498b

wp498c

wp510b

wp510c

wp510d

wp510e

Server: localhost:3306 » Database: test

[Structure](#) [SQL](#) [Search](#) [Query](#) [Export](#) [Import](#) [Operations](#) [Privileges](#) [Routines](#) [More](#)Browse your computer: No file chosen (Max: 32MiB)

You may also drag and drop a file on any page.

Character set of the file: **Partial Import:** Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. *(This might be a good way to import large files, however it can break transactions.)*Skip this number of queries (for SQL) or lines (for other formats), starting from the first one: **Format:** SQL

ESRI Shape File

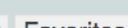
OpenDocument Spreadsheet

XML

 CSV

MediaWiki Table

 Do not use INCREMENT for zero values Read as multibytes



Recent

Favorites

Filter databases by name X

- suenlab1_6d8
 - test
 - New
 - book
 - data
 - users
 - users
 - webzash
 - willey
 - wp52a
 - wp475_c
 - wp498a
 - wp498b
 - wp498c
 - wp510b
 - wp510c
 - wp510d
 - wp510e

Skip this number of queries (for SQL) or lines (for other formats), starting from the first one: **Format:**

CSV

*Note: If the file contains multiple tables, they will be combined into one.***Format-Specific Options:** Update data when duplicate keys found on import (add ON DUPLICATE KEY UPDATE),

Columns separated with:

,

Columns enclosed with:

,

Columns escaped with:

,

Lines terminated with:

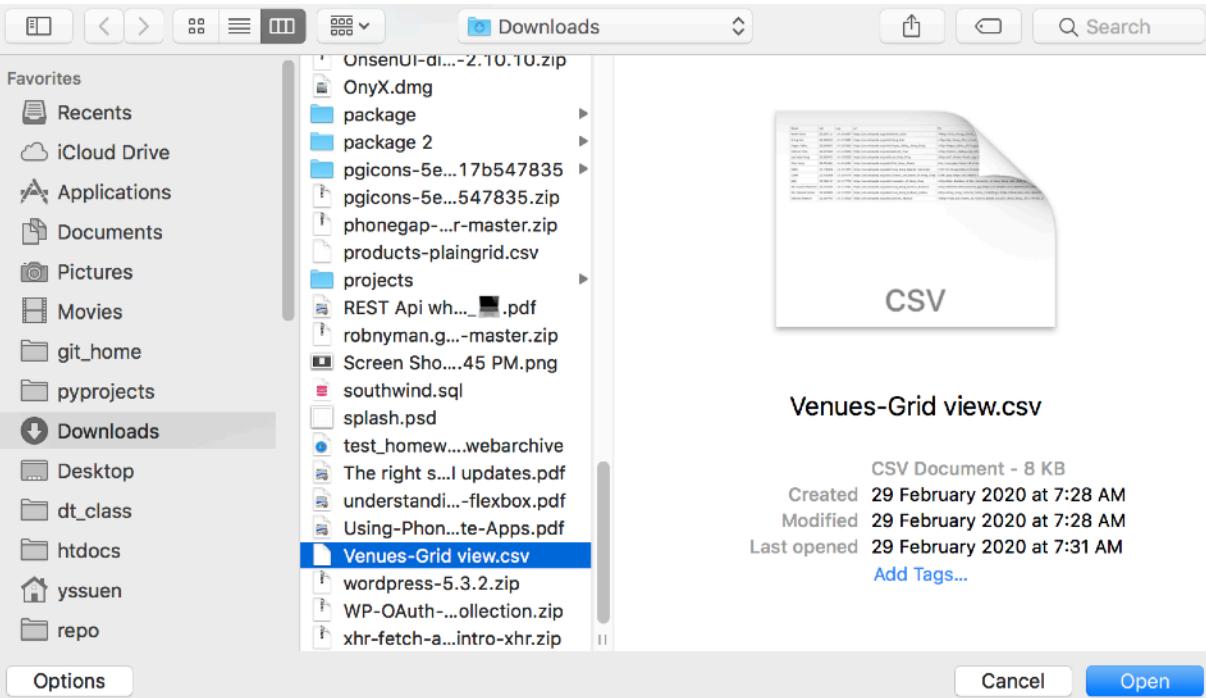
 The first line of the file contains the table column names (*if this is unchecked, the first line will become part of the data*) Do not abort on INSERT error

phpMyAdmin

Recent Favorites

Filter databases by name X

- invpl
- kickstart
- learn
- mgmt6821
- myerp
- mysql
- october_d1
- performance_schema
- qcdemo
- quandary
- seo
- service-leadership
- southwind
- suenlab1_6d8
- test
 - New
 - book
 - data



Venues-Grid view.csv

CSV Document - 8 KB

Created 29 February 2020 at 7:28 AM

Modified 29 February 2020 at 7:28 AM

Last opened 29 February 2020 at 7:31 AM

Add Tags...

Format:

CSV

Note: If the file contains multiple tables, they will be combined into one.

Format-Specific Options:

Console

Update data when duplicate keys found on import (add ON DUPLICATE KEY UPDATE)

Recent Favorites

Filter databases by name X

- invpl
- kickstart
- learn
- mgmt6821
- myerp
- mysql
- october_d1
- performance_schema
- qcdemo
- quandary
- seo
- service-leadership
- southwind
- suenlab1_6d8
- test
 - New
 - book
 - data

Importing into the table "venues"

File to Import:

File may be compressed (gzip, bzip2, zip) or uncompressed.

A compressed file's name must end in **.[format].[compression]**. Example: **.sql.zip**

Browse your computer: Venues-Grid view.csv (Max: 32MiB)

You may also drag and drop a file on any page.

Character set of the file:

Venues-Grid view.csv

Partial Import:

Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. (*This might be a good way to import large files, however it can break transactions.*)

Skip this number of queries (for SQL) or lines (for other formats), starting from the first one:

Format:

Note: If the file contains multiple tables, they will be combined into one.

Format-Specific Options:

Console Update data when duplicate keys found on import (add ON DUPLICATE KEY UPDATE)

phpMyAdmin

Recent Favorites

Filter databases by name X

- suenlab1_6d8
- test
 - New
 - book
 - data
- TABLE 4
- users

- users
- webzash
- willey
- wp52a
- wp475_c
- wp498a
- wp498b
- wp498c
- wp510b
- wp510c

Server: localhost:3306 » Database: test » Table: TABLE 4

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	me	varchar(18)	utf8_general_ci		Yes	NULL		Change Drop More
2	Lat	decimal(8,6)			Yes	NULL		Change Drop More
3	Lng	decimal(9,6)			Yes	NULL		Change Drop More
4	url	varchar(61)	utf8_general_ci		Yes	NULL		Change Drop More
5	Pic	varchar(245)	utf8_general_ci		Yes	NULL		Change Drop More
6	img_url	varchar(158)	utf8_general_ci		Yes	NULL		Change Drop More
7	desc	varchar(431)	utf8_general_ci		Yes	NULL		Change Drop More

Check All With selected: Browse Change Drop Primary Unique Index

Print view Propose table structure Move columns Improve table structure

Add 1 column(s) after desc

+ Indexes

Information

Space usage

Row statistics

Data	0 B
Index	0 B
Total	0 B

Server: localhost:3306 » Database: test » Table: TABLE 4

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Alter table order by: me (singly) Ascending Go

Move table to (database.table): test . TABLE 4 Add AUTO_INCREMENT value Go

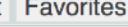
Table options: Rename table to venues Table comments Storage Engine InnoDB Collation Go

Copy table to (database.table): test . TABLE 4 Structure and data Go

Table maintenance: Flush the table (FLUSH)

Delete data or table: Empty the table (TRUNCATE) Delete the table (DROP)

Console



Recent

Favorites

Filter databases by name X

suenlab1_6d8

test

New

book

data

users

venues

users

webzash

willey

wp52a

wp475_c

wp498a

wp498b

wp498c

wp510b

wp510c

Table structure Relation view

	#	Name	Type	Collation	Attributes	Null	Default	Extra	Action		
	1	me	varchar(18)	utf8_general_ci		Yes	NULL		Change	Drop	More
	2	Lat	decimal(8,6)			Yes	NULL		Change	Drop	More
	3	Lng	decimal(9,6)			Yes	NULL		Change	Drop	More
	4	url	varchar(61)	utf8_general_ci		Yes	NULL		Change	Drop	More
	5	Pic	varchar(245)	utf8_general_ci		Yes	NULL		Change	Drop	More
	6	img_url	varchar(158)	utf8_general_ci		Yes	NULL		Change	Drop	More
	7	desc	varchar(431)	utf8_general_ci		Yes	NULL		Change	Drop	More

Check All

With selected:



Change



Drop



Primary



Unique



Index

Print view

Propose table structure

Move columns

Improve table structure

Add 1

column(s) after desc



+ Indexes

Information

Space usage

Row statistics

Data 16 KiB

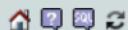
Format Compact

Index 0 B

Collation utf8_general_ci

Total 16 KiB

Creation Feb 29, 2020 at 07:48 AM



Recent | Favorites

Filter databases by name X

suenlab1_6d8

test
 New
book
data
users
venuesusers
webzash
willey
wp52a
wp475_cwp498a
wp498b
wp498c
wp510b
wp510c
wp510d

← Server: localhost:3306 » Database: test » Table: venues

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	me	varchar(18)	utf8_general_ci		Yes	NULL		Change Drop More
2	Lat	decimal(8,6)			Yes	NULL		Change Drop More
3	Lng	decimal(9,6)			Yes	NULL		Change Drop More
4	url	varchar(61)	utf8_general_ci		Yes	NULL		Change Drop More
5	Pic	varchar(255)			NULL			Change Drop More
6	img_url	varchar(100)			NULL			Change Drop More
7	desc	varchar(400)			NULL			Change Drop More

Do you really want to execute
"ALTER TABLE `venues` DROP
'Pic' ;"?

OK

Cancel

Print view Propose tab

Add 1

+ Indexes

Improve table structure

Go

Information

Space usage

Row statistics

Data 16 KiB

Format Compact

Index 0 B

Collation utf8_general_ci

Total 16 KiB

Creation Feb 29, 2020 at 07:48 AM

Console



Recent Favorites

Filter databases by name X

- october_d1
- performance_schema
- qcdemo
- quandary
- seo
- service-leadership
- southwind
- suenlab1_6d8
- test
 - New
 - book
 - data
 - users
 - venues
- users
- webzash
- willey

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Privileges](#) [Operations](#) [Triggers](#)[Table structure](#) [Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	me	varchar(18)	utf8_general_ci		Yes	NULL		Change Drop More
2	Lat	decimal(8,6)			Yes	NULL		Change Drop More
3	Lng	decimal(9,6)			Yes	NULL		Change Drop More
4	url	varchar(61)	utf8_general_ci		Yes	NULL		Change Drop More
5	img_url	varchar(158)	utf8_general_ci		Yes	NULL		Change Drop More
6	desc	varchar(431)	utf8_general_ci		Yes	NULL		Change Drop More

 Check All

With selected:

[Browse](#)[Change](#)[Drop](#)[Primary](#)[Unique](#)[Index](#)[Print view](#)[Propose table structure](#)[Move columns](#)[Improve table structure](#)[Add 1](#)

column(s) after desc

[Go](#)[+ Indexes](#)

Information

Space usage

Data 16 KiB

Index 0 B

Total 16 KiB

Row statistics

Format Compact

Collation utf8_general_ci

Creation Feb 29, 2020 at 07:48 AM

localhost x flask_e x lesson x MySQL x Album x Bare-b x Album x Open x csv to x Batteri x mysql x +

localhost/phpMyAdmin/tbl_structure.php?db=test&table=venues&token=ec029d4aa76b525ac4d69abe302b324e&goto=tbl_structure.php&back=tbl_struct... ☆

phpMyAdmin

Recent Favorites Filter databases by name X

Structure

Name	Type	Length/Values	Default	Collation	Attributes
me	VARCHAR	18	NULL	utf8_general_ci	

Preview SQL Save

invpl kickstart learn mgmt6821 myerp mysql october_d1 performance_schema qcdemo quandary seo service-leadership southwind suenlab1_6d8 test New book data

Console

phpMyAdmin

Server: localhost:3306 » Database: test » Table: venues

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Structure

Name	Type	Length/Values	Default	Collation	Attributes
name	VARCHAR	18	NULL	utf8_general_ci	

Preview SQL Save

Recent Favorites Filter databases by name X

- invpl
- kickstart
- learn
- mgmt6821
- myerp
- mysql
- october_d1
- performance_schema
- qcdemo
- quandary
- seo
- service-leadership
- southwind
- suenlab1_6d8
- test
 - New
 - book
 - data

Console

phpMyAdmin

Server: localhost:3306 » Database: test » Table: venues

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 11 (12 total, Query took 0.0002 seconds.)

SELECT * FROM `venues`

Show all | Number of rows: 25 Filter rows: Search this table

+ Options

name	Lat	Lng	url	img_url
North Point	22.287111	114.191667	https://en.wikipedia.org/wiki/North_Point	https://dl.airtable.com/.attachments/e685
Mong Kok	22.322500	114.170556	https://en.wikipedia.org/wiki/Mong_Kok	https://dl.airtable.com/.attachments/7fd2
Happy Valley	22.266667	114.183333	https://en.wikipedia.org/wiki/Happy_Valley,_Hong_K...	https://dl.airtable.com/.attachments/018c

Console

Server: localhost:3306 » Database: test » Table: venues

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Export Method:

Quick - display only the minimal options
 Custom - display all possible options

Rows:

Dump some row(s)
Number of rows: 12
Row to begin at: 0
 Dump all rows

Output:

Rename exported databases/tables/columns
 Save output to a file

File name template: use this for future exports
Character set of the file:
Compression:

View output as text

Skip tables larger than MiB
Console

Recent Favorites Filter databases by name X

- invpl
- kickstart
- learn
- mgmt6821
- myerp
- mysql
- october_d1
- performance_schema
- qcdemo
- quandary
- seo
- service-leadership
- southwind
- suenlab1_6d8
- test
- New
- book
- data

Server: localhost:3306 » Database: test » Table: venues

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Privileges](#) [Operations](#) [Triggers](#)

Recent Favorites Filter databases by name X

invpl
kickstart
learn
mgmt6821
myerp
mysql
october_d1
performance_schema
qcdemo
quandary
seo
service-leadership
southwind
suenlab1_6d8
test
New
book
data

INSERT DELAYED statements
 INSERT IGNORE statements

Function to use when dumping data: **INSERT**

Syntax to use when inserting data:

- include column names in every `INSERT` statement
Example: `INSERT INTO tbl_name (col_A,col_B,col_C) VALUES (1,2,3)`
- insert multiple rows in every `INSERT` statement
Example: `INSERT INTO tbl_name VALUES (1,2,3), (4,5,6), (7,8,9)`
- both of the above
Example: `INSERT INTO tbl_name (col_A,col_B,col_C) VALUES (1,2,3), (4,5,6), (7,8,9)`
- neither of the above
Example: `INSERT INTO tbl_name VALUES (1,2,3)`

Maximal length of created query **50000**

Dump binary columns in hexadecimal notation (*for example, "abc" becomes 0x616263*)
 Dump TIMESTAMP columns in UTC (*enables TIMESTAMP columns to be dumped and reloaded between servers in different time zones*)

Go

Console

venues.sql

```
--  
26 -- Table structure for table `venues`--  
27 --  
28 --  
29 CREATE TABLE `venues` (←  
30   `name` varchar(18) DEFAULT NULL,←  
31   `Lat` decimal(8,6) DEFAULT NULL,←  
32   `Lng` decimal(9,6) DEFAULT NULL,←  
33   `url` varchar(61) DEFAULT NULL,←  
34   `img_url` varchar(158) DEFAULT NULL,←  
35   `desc` varchar(431) DEFAULT NULL←  
36 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;←  
37 --  
38 --  
39 -- Dumping data for table `venues`--  
40 --  
41 --  
42 INSERT INTO `venues` (`name`, `Lat`, `Lng`, `url`, `img_url`, `desc`) VALUES--  
43 ('North Point', '22.287111', '114.191667', 'https://en.wikipedia.org/wiki/North_Point', 'https://dl.airtable.com/.attachments/  
• e685bbc78ed305ba5c19d9341209c75f/1ddc090f/440px-Chun_Yeung_Street_Tram_140.jpg', 'North Point is a an urban area in the Eastern  
• District of Hong Kong. It is in the northeastern part of Hong Kong Island, between Causeway Bay and Quarry Bay, and projects  
• toward Kowloon Bay.'),  
44 ('Mong Kok', '22.322500', '114.170556', 'https://en.wikipedia.org/wiki/Mong_Kok', 'https://dl.airtable.com/.attachments/  
~/Downloads/venues.sql 1:1
```

LF UTF-8 SQL GitHub Git (0)



Dashboard Consoles **Files** Web Tasks Databases

/home/ bernardsys

Open Bash console here 0% full – 296.0 KB of your 512.0 MB quota

Directories

Enter new directory name

New directory

.cache/
.ipython/
.local/
.virtualenvs/
mysite/



Files

Enter new file name, eg hello.py

New file

.bash_history	2019-09-18 12:32 56 bytes
.bashrc	2019-01-13 07:53 559 bytes
.gitconfig	2019-01-13 07:53 266 bytes
.my.cnf	2020-02-28 23:24 40 bytes
.mysql_history	2020-02-28 23:26 3.3 KB
.profile	2019-01-13 07:53 79 bytes
.python_history	2019-09-18 12:33 1.9 KB
.pythonstartup.py	2019-01-13 07:53 77 bytes
.vimrc	2019-01-13 07:53 4.6 KB
README.txt	2019-01-13 07:53 232 bytes

Upload a file

100MiB maximum size

www.pythonanywhere.com/user/bernardsys/files/home/bernardsys

Forums Help Blog Account Log out

pythonanywhere

/home/ bernardsys

Directories

Enter new directory name

.cache/ .ipython/ .local/ .virtualenvs/ mysite/

git_home pyprojects Downloads Desktop dt_class htdocs yssuen

Favorites Recents iCloud Drive Applications Documents Pictures Movies

package package 2 pgicons-5e... 17b547835 pgicons-5e...547835.zip phonegap-...r-master.zip products-plaingrid.csv projects REST Api wh..._.pdf robnyman.g...-master.zip Screen Sho...45 PM.png southwind.sql splash.psd test_homew...webarchive The right s...l updates.pdf understandi...-flexbox.pdf Using-Phon...te-Apps.pdf Venues-Grid view.csv

venues.sql

7 KB

Created 29 February 2020 at 8:13 AM
Modified 29 February 2020 at 8:13 AM
Last opened --
[Add Tags...](#)

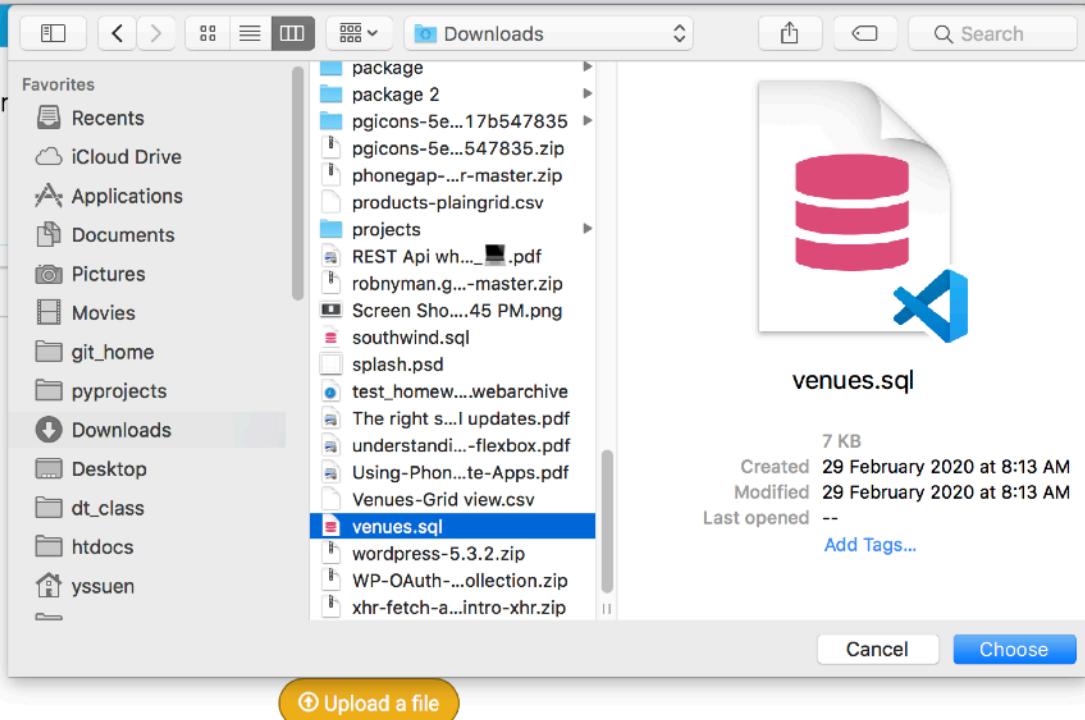
Cancel Choose

Upload a file 100MiB maximum size

Copyright © 2011-2020 PythonAnywhere LLP — [Terms](#) — [Privacy & Cookies](#)

Web Tasks Databases
296.0 KB of your 512.0 MB quota

New file





/home/ bernardsys

Dashboard Consoles **Files** Web Tasks Databases

Open Bash console here

0% full – 292.0 KB of your 512.0 MB quota

Directories

Enter new directory name

New directory

.cache/
.ipython/
.local/
.virtualenvs/
mysite/



Files

Enter new file name, eg hello.py

New file

.bash_history	2019-09-18 12:32 56 bytes
.bashrc	2019-01-13 07:53 559 bytes
.gitconfig	2019-01-13 07:53 266 bytes
.my.cnf	2020-02-28 23:24 40 bytes
.mysql_history	2020-02-28 23:26 3.3 KB
.profile	2019-01-13 07:53 79 bytes
.python_history	2019-09-18 12:33 1.9 KB
.pythonstartup.py	2019-01-13 07:53 77 bytes
.vimrc	2019-01-13 07:53 4.6 KB
README.txt	2019-01-13 07:53 232 bytes
venues.sql	2020-02-29 00:18 7.2 KB

Upload a file

100MiB maximum size

MySQL

Postgres

MySQL settings

Connecting:

Use these settings in your web applications.

Database host address: `bernardsys.mysql.pythonanywhere-services.com`

Username: `bernardsys`

Your databases:

Click a database's name to start a MySQL console logged in to it.

Start a console on: [`bernardsys\$default`](#)

Start a console on: [`bernardsys\$demo`](#)

Start a console on: [`bernardsys\$demo2`](#)

Create a database

Your database names always start with your username + "\$". There's no need to type that prefix in below, though: PythonAnywhere will automatically add it.

Database name:

Create

MySQL password:

This should be different to your main PythonAnywhere password, because it is likely to

MySQL: bernardsys\$demo

Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 11669073
Server version: 5.7.21-log MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| bernardsys$default |
| bernardsys$demo   |
| bernardsys$demo2  |
+-----+
4 rows in set (1.59 sec)

mysql> use bernardsys$demo;
```



MySQL: bernardsys\$demo

Share with others



Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 11669073

Server version: 5.7.21-log MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| bernardsys$default |  
| bernardsys$demo |  
| bernardsys$demo2 |  
+-----+  
4 rows in set (1.59 sec)
```

```
mysql> use bernardsys$demo;  
Database changed  
mysql> █
```

MySQL: bernardsys\$demo

Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 11669073
Server version: 5.7.21-log MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| bernardsys$default |
| bernardsys$demo   |
| bernardsys$demo2  |
+-----+
4 rows in set (1.59 sec)

mysql> use bernardsys$demo;
Database changed
mysql> show tables;
```

MySQL: bernardsys\$demo

Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 11669073
Server version: 5.7.21-log MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| bernardsys$default |
| bernardsys$demo   |
| bernardsys$demo2  |
+-----+
4 rows in set (1.59 sec)

mysql> use bernardsys$demo;
Database changed
mysql> show tables;
+-----+
| Tables_in_bernardsys$demo |
+-----+
| users                      |
+-----+
1 row in set (0.00 sec)

mysql> source venues.sql;
```



MySQL: bernardsys\$demo

[Share with others](#)

```
mysql> use bernardsys$demo;
Database changed
mysql> show tables;
+-----+
| Tables_in_bernardsys$demo |
+-----+
| users
+-----+
1 row in set (0.00 sec)
```

```
mysql> source venues.sql;
Query OK, 0 rows affected (0.00 sec)
```

```
Query OK, 0 rows affected (0.00 sec)
```

```
Query OK, 0 rows affected (0.00 sec)
```

```
Query OK, 0 rows affected (0.00 sec)
```

```
Query OK, 0 rows affected (0.00 sec)
```

```
Query OK, 0 rows affected (0.00 sec)
```

```
Query OK, 0 rows affected (0.00 sec)
```

```
Query OK, 12 rows affected (0.01 sec)
Records: 12  Duplicates: 0  Warnings: 0
```

```
Query OK, 0 rows affected (0.00 sec)
```

```
Query OK, 0 rows affected (0.00 sec)
```

```
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> █
```

MySQL: bernardsys\$demo

```
mysql> source venues.sql;
Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.02 sec)

Query OK, 12 rows affected (0.01 sec)
Records: 12  Duplicates: 0  Warnings: 0

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

mysql> show tables;
+-----+
| Tables_in_bernardsys$demo |
+-----+
| users
| venues
+-----+
2 rows in set (0.00 sec)

mysql> select * from venues;
```

```
MySQL: bernardsys$demo
+-----+
| colleges - Chung Chi College, New Asia College and United College - the oldest of which was founded in 1949.
| HKU | 22.284167 | 114.137778 | https://en.wikipedia.org/wiki/University_of_Hong_Kong | https://dl.airtable.com/.attachments/f8e3c9c3ae07744c86d98f16361f7d7f/acfba70e/440px-Main_Building_of_the_University_of_Hong_Kong_and_clock_tower.jpeg | The University of Hong Kong (HKU) is a public research university in Hong Kong. Founded in 1911, its origins trace back to the Hong Kong College of Medicine for Chinese, which was founded in 1887. It is the oldest tertiary institution in Hong Kong.[6] Also, HKU is the first university established by the British Empire in East Asia. HKU has always been the best university in Hong Kong and one of the best universities in Asia.
| HK Science Musuem | 22.301000 | 114.177655 | https://en.wikipedia.org/wiki/Hong_Kong_Science_Museum | https://dl.airtable.com/.attachments/22f383737e8fb434bdff596329a7016a/42678f91/440px-HKScienceMuseumview.jpg | The Hong Kong Science Museum was first conceived by the Urban Council in 1976. The council hired American firm E. Verner Johnson and Associates in 1984 to help plan the museum.
| HK Cultural Center | 22.293850 | 114.170323 | https://en.wikipedia.org/wiki/Hong_Kong_Cultural_Centre | https://dl.airtable.com/.attachments/792c15887963b8bf9e75075b21a91d0d/dc57215f/480px-Hong_Kong_Cultural_Centre_201408.jpg | The centre is located on the southwestern tip of Tsim Sha Tsui, on the former location of the Kowloon Station of the Kowloon-Canton Railway.
| Victoria Harbour | 22.287753 | 114.173619 | https://en.wikipedia.org/wiki/Victoria_Harbour | https://dl.airtable.com/.attachments/46d727c499e74ade15065af197e45ea6/e9570ba3/440px-Vista_del_Puerto_de_Victoria_desde_Sky100_Hong_Kong_2013-08-09_DD_10.jpeg | Victoria Harbour is a natural landform harbour separating Hong Kong Island in the south from the Kowloon Peninsula to the north.
+-----+
12 rows in set (0.00 sec)

mysql> 
```

**Using the MySQLdb module instead of
pymysql.cursors module (not supported on)
PythonAnywhere) for MySQL access and
CRUD operations.**



"Batteries Included"

The following modules are available as standard on PythonAnywhere. Just `import` them, they're ready to go!

The system image

Your system image determines which package versions you have installed by default. You have the "dangermouse" image; [contact us](#) to change it.

If you want to view what packages a different system image has, select it from this dropdown: dangermouse

Python version	Module name	Version	Description
Any ▾	Filter by name	Filter by version	Filter by description
3.6	alabaster	0.7.9	A configurable sidebar-enabled Sphinx theme
3.6	alembic	0.8.10	A database migration tool for SQLAlchemy.

Show All

https://www.pythonanywhere.com/batteries_included/

MySQL Data Manipulation Using Either Pymysql or MySQLdb Module

Retrieve MySQL Records

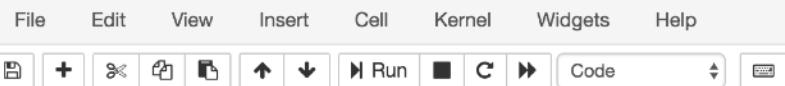
```
In [9]: 1 from flask import Flask, render_template, jsonify
2 import pandas as pd # initialise pandas module ****
3 import pymysql.cursors
4 connection = pymysql.connect(host="localhost",
5                               user="root",
6                               passwd="root",
7                               db="southwind",
8                               charset='utf8mb4',
9                               cursorclass=pymysql.cursors.DictCursor)
10
11 with connection.cursor() as cursor:
12     # Execute SQL Select command in Python using PyMysql built-in methods
13     sql = "SELECT * FROM products"
14     cursor.execute(sql)
15     # Display results
16     result = cursor.fetchall()
17     print(result)
18     print(pd.DataFrame(result)) # ****
19     print('\n')
20 connection.close()

[{'productID': 1001, 'productCode': 'PEN', 'name': 'Pen Red', 'quantity': 5000, 'price': Decimal('1.23'), 'supplierID': 501}, {'productID': 1002, 'productCode': 'PEN', 'name': 'Pen Blue', 'quantity': 8000, 'price': Decimal('1.25'), 'supplierID': 501}, {'productID': 1003, 'productCode': 'PEN', 'name': 'Pen Black', 'quantity': 2000, 'price': Decimal
```



```
In [ ]: 1 pip install mysqlclient # install MySQLdb module for substituting pymysql
```

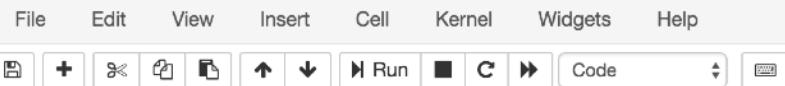
```
In [2]: 1 from flask import Flask, render_template, jsonify
2 # import pymysql.cursors #initialise PyMysql module
3 import MySQLdb
4 import pandas as pd # initialise pandas module *****
5 # connection = pymysql.connect(host="localhost",
6 connection = MySQLdb.connect(host="localhost",
7                         user="root",
8                         passwd="root",
9                         db="southwind",
10                        charset='utf8mb4')
11 # cursorclass=pymysql.cursors.DictCursor)
12
13 with connection.cursor() as cursor:
14     # Execute SQL Select command in Python using PyMysql built-in methods
15     sql = "SELECT * FROM products"
16     cursor.execute(sql)
17     # Display results
18     result = cursor.fetchall()
19 connection.close()
20 print(result)
21 print('\n')
22 final_result = [list(i) for i in result]
23 dataset=[]
24 dict={}
25 for i in final_result:
26     dict['productID'] = i[0]
27     dict['productCode'] = i[1]
28     dict['name'] = i[2]
29     dict['quantity'] = i[3]
30     dict['price'] = float(i[4])
31     dataset.append(dict)
```



```
13 with connection.cursor() as cursor:
14     # Execute SQL Select command in Python using PyMysql built-in methods
15     sql = "SELECT * FROM products"
16     cursor.execute(sql)
17     # Display results
18     result = cursor.fetchall()
19 connection.close()
20 print(result)
21 print('\n')
22 final_result = [list(i) for i in result]
23 dataset=[]
24 dict={}
25 for i in final_result:
26     dict['productID'] = i[0]
27     dict['productCode'] = i[1]
28     dict['name'] = i[2]
29     dict['quantity'] = i[3]
30     dict['price'] = float(i[4])
31     dict['supplierID'] = i[5]
32     # print(i)
33     # print(dict)
34     dataset.append(dict.copy()) #markers.append(fld.copy())
35 print(dataset)
```

```
((1001, 'PEN', 'Pen Red', 5000, Decimal('1.23'), 501), (1002, 'PEN', 'Pen Blue', 8000, Decimal('1.25'), 501), (1003, 'PEN', 'Pen Black', 2000, Decimal('1.25'), 501), (1004, 'PEC', 'Pencil 2B', 10000, Decimal('0.48'), 501), (1005, 'PEC', 'Pencil 2H', 8000, Decimal('0.49'), 502), (1006, 'PEC', 'Pencil HB', 0, Decimal('99999.99'), 501), (2001, 'PEC', 'Pencil 3B', 500, Decimal('0.52'), 503), (2002, 'PEC', 'Pencil 4B', 200, Decimal('0.62'), 501), (2003, 'PEC', 'Pencil 5B', 100, Decimal('0.73'), 503), (2004, 'PEC', 'Pencil 6B', 500, Decimal('0.47'), 502))
```

```
[{'productID': 1001, 'productCode': 'PEN', 'name': 'Pen Red', 'quantity': 5000, 'price': 1.23, 'supplierID': 501}, {'productID': 1002, 'productCode': 'PEN', 'name': 'Pen Blue', 'quantity': 8000, 'price': 1.25, 'supplierID': 501}, {'productID': 1003, 'productCode': 'PEN', 'name': 'Pen Black', 'quantity': 2000, 'price': 1.25, 'supplierID': 501}, {'productID': 1004, 'productCode': 'PEC', 'name': 'Pencil 2B', 'quantity': 10000, 'price': 0.48, 'supplierID': 501},
```



```
1 / aer name():
2     user = {"name": "Bernard"}
3     with connection.cursor() as cursor:
4         # Execute SQL Select command in Python using PyMysql built-in methods
5         sql = "SELECT * FROM products"
6         cursor.execute(sql)
7         # Display results
8         result = cursor.fetchall()
9         connection.close()
10        # print(result)
11        # print('\n')
12        final_result = [list(i) for i in result]
13        dataset=[]
14        dict={}
15        for i in final_result:
16            dict['productID'] = i[0]
17            dict['productCode'] = i[1]
18            dict['name'] = i[2]
19            dict['quantity'] = i[3]
20            dict['price'] = float(i[4])
21            dict['supplierID'] = i[5]
22            # print(i)
23            # print(dict)
24            dataset.append(dict.copy()) #markers.append(fld.copy())
25
26        return render_template('index3.html', album_user=user, dataset=dataset)
27
28 if __name__ == '__main__':
29     from werkzeug.serving import run_simple
30     run_simple('localhost', 9029, app)
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
```

* Running on <http://localhost:9029/> (Press CTRL+C to quit)
127.0.0.1 - - [29/Feb/2020 09:12:23] "GET / HTTP/1.1" 200 -

Main call to action

Secondary action

Pen Red

Col#:1

Product Code:PEN

Quantity:5000

Price:1.23

Pen Blue

Col#:2

Product Code:PEN

Quantity:8000

Price:1.25

Pen Black

Col#:3

Product Code:PEN

Quantity:2000

Price:1.25

**Reproducing the same effect on
PythonAnywhere.**



/home/bernardsys/mysite/flask_app.py

Keyboard shortcuts:

Normal ▾

Share

Save

Save as...

>>> Run



```
1  from flask import Flask, render_template, url_for
2  # import pymysql.cursors #initialise PyMysql module
3  import MySQLdb
4
5  app = Flask(__name__)
6  # CORS(app)
7
8  # connection = pymysql.connect(host="localhost",
9  connection = MySQLdb.connect(host="bernardsys.mysql.pythonanywhere-services.com",
10                         user="bernardsys",
11                         passwd=[REDACTED],
12                         db="bernardsys$demo",
13                         charset='utf8mb4')
14  # cursorclass=pymysql.cursors.DictCursor)
15
16 @app.route("/")
17 def home():
18     return render_template('index.html')
19
20 @app.route("/venues")
21 def venues():
22     user = {"name": "Bernard"}
23     with connection.cursor() as cursor:
24         # Execute SQL Select command in Python using PyMysql built-in methods
25         sql = "SELECT * FROM venues"
26         cursor.execute(sql)
27         # Display results
28         result = cursor.fetchall()
```

[MySQL](#)[Postgres](#)

MySQL settings

Connecting:

Use these settings in your web applications.

Database host address: `bernardsys.mysql.pythonanywhere-services.com`
Username: `bernardsys`

Your databases:

Click a database's name to start a MySQL console logged in to it.

Start a console on: [`bernardsys\$default`](#)
Start a console on: [`bernardsys\$demo`](#)
Start a console on: [`bernardsys\$demo2`](#)

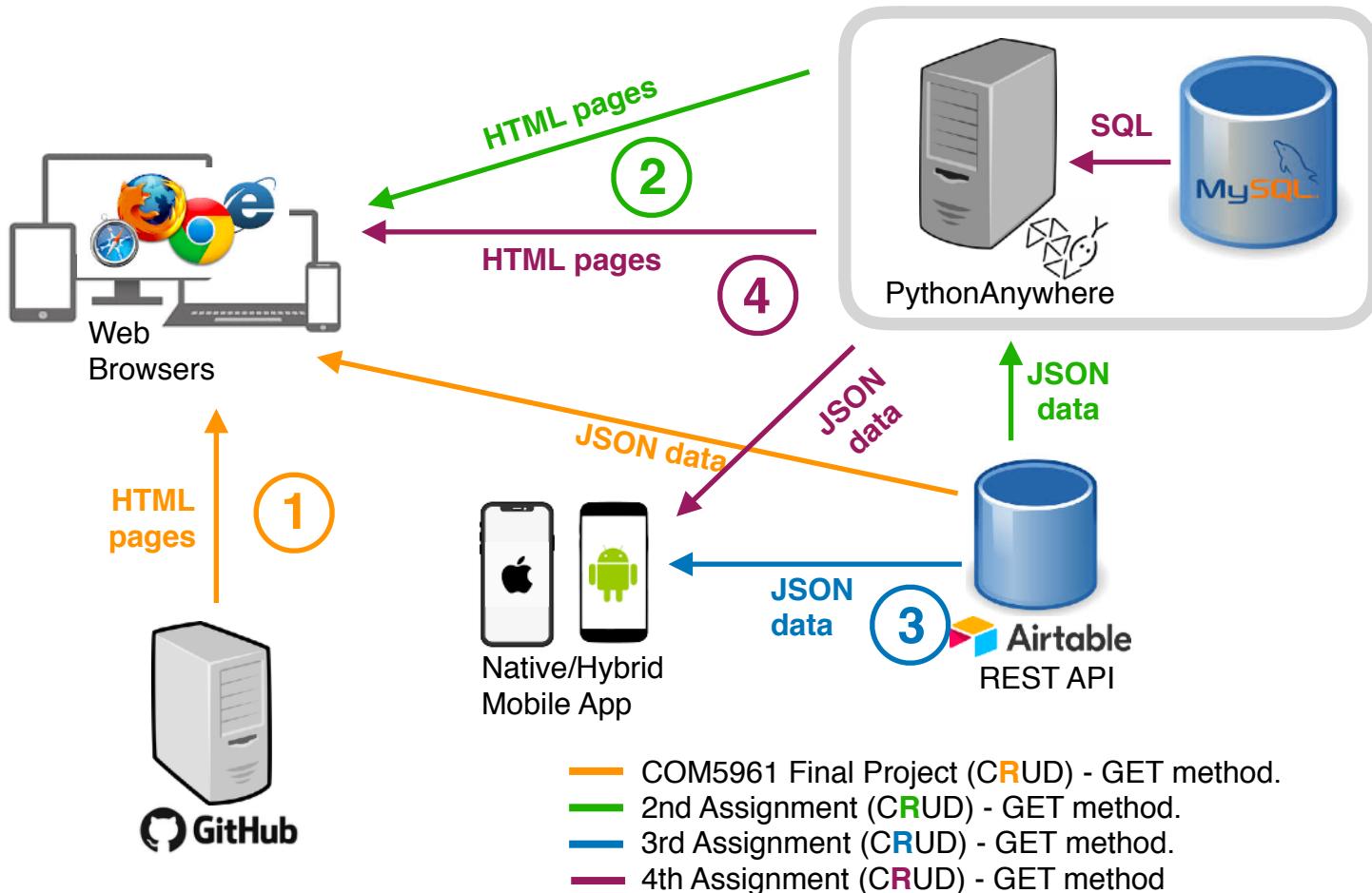
**From gaining access to data through API
to granting access to data through API.**

MySQL
on
PythonAnywhere

The RESTful API

Airtable
for
Agile PM

Different Ways of Data Retrieval



To go beyond the “R” in **CRUD, we need a mechanism to capture request and data on the front-end (web, mobile, or IoT device) for **submitting** to the server back-end.**

3 Ways of Submitting Request

- 1. Form method**
- 2. Query-string method**
- 3. JSON method (method used by RESTful API and can be combined with the Form method for submitting data to the server)**

Form Method (e.g. HTML Form)



HTML

CSS

JAVASCRIPT

SQL

PYTHON

PHP

BOOTSTRAP

MORE ▾

EXERCISES ▾

- HTML Layout
- HTML Responsive
- HTML Computercode
- HTML Semantics
- HTML Entities
- HTML Symbols
- HTML Charset
- HTML URL Encode
- HTML XHTML
- HTML Style Guide

HTML Forms

- HTML Forms
- HTML Form Elements
- HTML Input Types
- HTML Input Attributes
- HTML Input Form Attributes

HTML Graphics

- HTML Canvas
- HTML SVG

HTML Media

- HTML Media

HTML Forms

[◀ Previous](#)[Next ▶](#)

HTML Form Example

First name:

Last name:

[Try it Yourself »](#)

The <form> Element

The HTML `<form>` element defines a form that is used to collect user input:

https://www.w3schools.com/html/html_forms.asp

BS Jumbotron

BS Wells

BS Alerts

BS Buttons

BS Button Groups

BS Glyphicons

BS Badges/Labels

BS Progress Bars

BS Pagination

BS Pager

BS List Groups

BS Panels

BS Dropdowns

BS Collapse

BS Tabs/Pills

BS Navbar

BS Forms

BS Inputs

BS Inputs 2

BS Input Sizing

BS Media Objects

BS Carousel

BS Modal

BS Tooltip

Bootstrap Vertical Form (default)

Email:

Enter email

Password:

Enter password

 Remember me

Submit

The following example creates a vertical form with two input fields, one checkbox, and a submit button:

Example

```
<form action="/action_page.php">
  <div class="form-group">
    <label for="email">Email address:</label>
    <input type="email" class="form-control" id="email">
  </div>
  <div class="form-group">
    <label for="pwd">Password:</label>
```

https://www.w3schools.com/bootstrap/bootstrap_forms.asp

Result Size: 1256 x 24

Run »

```
<div class="form-group">
  <label for="pwd">Password:</label>
  <input type="password" class="form-control" id="pwd" placeholder="Enter password" name="pwd">
</div>
<div class="checkbox">
  <label><input type="checkbox" name="remember"> Remember me</label>
</div>
<button type="submit" class="btn btn-default">Submit</button>
</form>
</div>

</body>
</html>
```

Vertical (basic) form

Email:

Password:

Remember me

Combine Form with User Authentication Using Flask-login Module

Please login in first.

Vertical (basic) form

Email:

Password:

Remember me

Wrong password!

Vertical (basic) form

Email:

Password:

Remember me

Home Page

Dear bernard@cuhk.edu.hk, welcome to Bernard's pages. Your login has been granted.

Col#:1
Product Code:PEN
Quantity:5000
Price:1.23

[View](#)

Col#:2
Product Code:PEN
Quantity:8000
Price:1.25

[View](#)

Col#:3
Product Code:PEN
Quantity:2000
Price:1.25

[View](#)

Col#:4
Product Code:PEC
Quantity:10000
Price:0.48

[View](#)

Col#:5
Product Code:PEC
Quantity:8000
Price:0.49

[View](#)

Col#:6
Product Code:PEC
Quantity:0
Price:99999.99

[View](#)

Col#:7
Product Code:PEC
Quantity:500
Price:0.52

[View](#)

Col#:8
Product Code:PEC
Quantity:200
Price:0.62

[View](#)

Col#:9
Product Code:PEC
Quantity:100
Price:0.73

[View](#)

Col#:10
Product Code:PEC
Quantity:500
Price:0.47

[View](#)

You're already login.

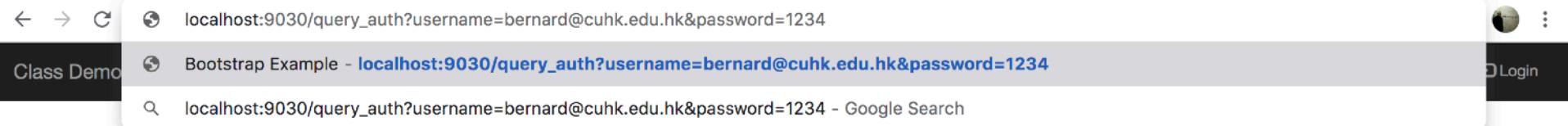
```
1 ##### Flask Modules Setup #####
2
3 from flask import Flask, render_template, url_for, redirect, request, make_response, Response, jsonify
4 from flask_login import LoginManager, login_user, logout_user, login_required, UserMixin
5 import MySQLdb
6
7 ##### Initialize Flask App #####
8
9 app = Flask(__name__)
10
11 ##### MySQL Database Setup #####
12
13 connection = MySQLdb.connect(host="localhost",
14                             user="root",
15                             passwd="root",
16                             db="southwind",
17                             charset='utf8mb4')
18
19 ##### Login Manager Setup #####
20
21 login_manager = LoginManager()
22 login_manager.init_app(app)
23 login_manager.login_view = 'login'
24 app.config['SECRET_KEY'] = "lkkajdghdadkgla jkgah"
25
26
27 @login_manager.user_loader
28 def load_user(user_id):
29     return User(user_id)
30
31 class User(UserMixin):
32     def __init__(self,id):
33         self.id = id
```

```
-->
35 ##### Web Page Routes Setup #####
36
37 @app.route("/")
38 @login_required
39 def home():
40     return render_template('home.html')
41
42 @app.route("/login")
43 def login():
44     message = 'Please login in first.'
45     return render_template('login.html', message=message)
46
47 @app.route('/query_auth')
48 def query_auth():
49     username = request.args.get("username")
50     password = request.args.get("password")
51     if username == "bernard@cuhk.edu.hk" and password == "1234":
52         login_user(User(1))
53         message = "Dear " + username + ", welcome to Bernard's pages. Your login has been granted."
54         return make_response(message, 200, )
55     else:
56         message = 'Wrong password!'
57         return make_response( message, 401, {'WWW-Authenticate': 'Basic realm="Login Required"'})
58
59 @app.route("/form_auth",methods=['POST'])
60 def form_auth():
61     username = request.form['email']
62     password = request.form['pwd']
63     if username == "bernard@cuhk.edu.hk" and password == "1234":
64         login_user(User(1))
65         message = "Dear " + username + ", welcome to Bernard's pages. Your login has been granted."
66         return render_template('home.html', message=message)
67     else:
68         message = 'Wrong password!'
69         return render_template('login.html', message=message)
70
```

```
71 @app.route('/album')
72 @login_required
73 def album():
74     user = {"name": "Bernard"}
75     with connection.cursor() as cursor:
76         sql = "SELECT * FROM products"
77         cursor.execute(sql)
78         result = cursor.fetchall()
79     final_result = [list(i) for i in result]
80     dataset = []
81     dict = {}
82     for i in final_result:
83         dict['productID'] = i[0]
84         dict['productCode'] = i[1]
85         dict['name'] = i[2]
86         dict['quantity'] = i[3]
87         dict['price'] = float(i[4])
88         dict['supplierID'] = i[5]
89         dataset.append(dict.copy()) #markers.append(fld.copy())
90
91     return render_template('product2.html', album_user=user, dataset=dataset)
92
93 @app.route('/logout/')
94 @login_required
95 def logout():
96     logout_user()
97     message = 'Thanks for logging out.'
98     return render_template('login.html', message=message)
99
100 @app.errorhandler(500)
101 def internal_error(error):
102     message = 'Wrong password!'
103     return render_template('login.html', message=message), 500
104
```

Query-string Method

```
47 @app.route('/query_auth')
48 def query_auth():
49     username = request.args.get("username")
50     password = request.args.get("password")
51     if username == "bernard@cuhk.edu.hk" and password == "1234":
52         login_user(User(1))
53         message = "Dear " + username + ", welcome to Bernard's pages. Your login has been granted."
54         return make_response(message, 200, )
55     else:
56         message = 'Wrong password!'
57         return make_response( message, 401, { 'WWW-Authenticate': 'Basic realm="Login Required"' } )
58
```



Thanks for logging out.

Vertical (basic) form

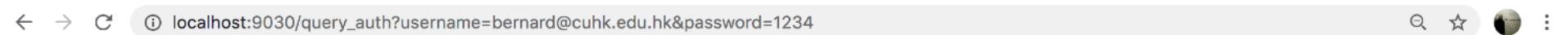
Email:

Password:

Remember me

http://localhost:9030/query_auth?username=bernard@cuhk.edu.hk&password=1234

**This is just an example. Never expose password as a query string in your code.
Use a form instead (Make sure it is transmitted through https so the password will
be encrypted.)**



Dear bernard@cuhk.edu.hk, welcome to Bernard's pages. Your login has been granted.

JSON Method (RESTful API)

**Before doing that, we need a special tool for testing
the JSON API. This tool is called Postman.**

Home > Apps > Postman



Postman

Offered by: www.getpostman.com

★★★★★ 9,096 | [Extensions](#) | 3,876,124 users

Runs offline

[Launch app](#)

Overview

Reviews

Support

Related



Cloud SWF Player with ...

★★★★★ 2,741



Text

★★★★★ 1,229



Chrome Apps & Extensi...

★★★★★ 1,387



Advanced REST client

★★★★★ 12,153



Vysor

★★★★★ 5,661



Secure Shell App

★★★★★ 3,142



Caret

★★★★★ 1,250



BrowserStack Local

★★★★★ 272



Web Server for Chrome

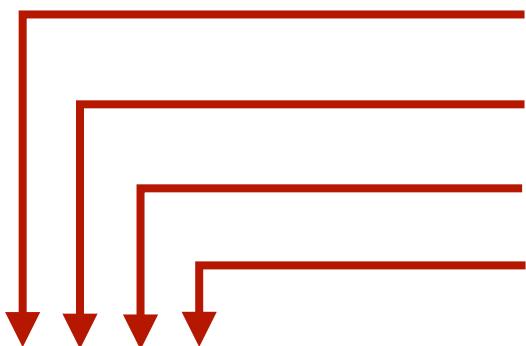
★★★★★ 1,353



Robot Mesh Connect App

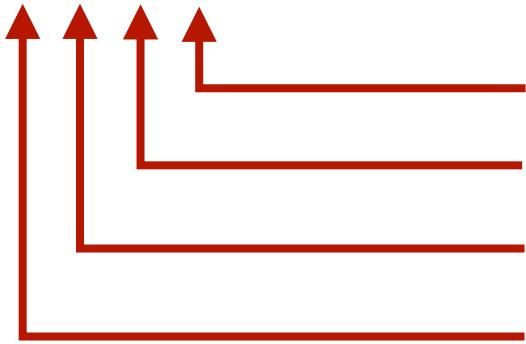
69,129 users

<https://chrome.google.com/webstore/detail/postman/fhbjgbiflinjbdfgehcddcbnccccdomop?hl=en>



Create
Retrieve
Update
Delete

CRUD Operations in RESTful API via Postman



Delete
Put
Get
Post

GET (LIST Records) Method

PUT https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo

POST https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo

POST localhost:9007/addblog

▼ August 4

POST 127.0.0.1:5000/addblog

POST 127.0.0.1:5000/

POST localhost:9007/addblog

Request

Builder Team Library

Sync Off

Sign In

Filter

127. http http http http http http No Environment

GET https://api.airtable.com/v0/appM38HXIEVhxmnqx... Send Save

Headers (1)

Key	Value
Authorization	Bearer key

Body Cookies Headers (11) Test Results Status: 200 OK Time: 518 ms

Pretty Raw Preview JSON

Response

```
{ "records": [ { "id": "rec8vF2VbIg0Tdx3F", "fields": { "lname": "Lu", "student_id": "99999", "date_of_birth": "2001-08-22", "fname": "Mary", "full_name": "Mary Lu", "record_id": "rec8vF2VbIg0Tdx3F" }, "createdTime": "2019-09-19T14:32:08.000Z" }, { "id": "recb1ZFR8ZL0H89ti", "fields": { "lname": "Lu", "student_id": "99999", "date_of_birth": "2001-08-22", "fname": "Mary", "full_name": "Mary Lu", "record_id": "recb1ZFR8ZL0H89ti" } } ] }
```

GET (Specific Record) Method

VblgOTdx3F

GET https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo/recpOHXSrfRmlvQia

GET https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo/rec8vF2VbIgOTdx3F

GET https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo

DEL https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo/recpOHXSrfRmlvQia

PUT https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo

PUT https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo

PUT https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo

PUT https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo

Builder Team Library SYNC OFF Sign In Filter http http http http http http No Environment GET https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo/rec8vF2VbIgOTdx3F Params Send Save Headers (1) Authorization Key Value Description Bulk Presets Headers (11) Body Cookies Test Results Status: 200 OK Time: 631 ms Request Response

```
{  
  "id": "rec8vF2VbIgOTdx3F",  
  "fields": {  
    "lname": "Lu",  
    "student_id": "99999",  
    "date_of_birth": "2001-08-22",  
    "fname": "Mary",  
    "full_name": "Mary Lu",  
    "record_id": "rec8vF2VbIgOTdx3F"  
  },  
  "createdTime": "2019-09-19T14:32:08.000Z"  
}
```

POST Method

ble.com/v0/appM3
8HXIEVhxmnqx/flaskdemo

PUT https://api.airtable.com/v0/appM3
8HXIEVhxmnqx/flaskdemo

PUT https://api.airtable.com/v0/appM3
8HXIEVhxmnqx/flaskdemo

POST https://api.airtable.com/v0/appM3
8HXIEVhxmnqx/flaskdemo

POST localhost:9007/addblog

▼ August 4

POST 127.0.0.1:5000/addblog

POST 127.0.0.1:5000/

POST localhost:9007/addblog

POST localhost:9007/addblog

POST localhost:9007/addblog

POST localhost:9007/addblog

POST localhost:9007/addblog

Filter Collections Clear all

Builder Team Library

Sync Off Sign In

Chrome apps are being deprecated. Download our free native apps for continued support and better performance. [Learn more](#)

POST https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo

Params Send Save

Headers (2)

Key	Value	Description	...	Bulk	Presets ▾
Authorization	Bearer key				
Content-Type	application/json				
New key	Value	Description			

Response

Hit the Send button to get a response.

Do more with requests

Share Mock Monitor Document



Chrome apps are being deprecated. [Download our free native apps](#) for continued support and better performance. [Learn more](#)

X

POST Method

Collections

[Clear all](#)

PUT https://api.airtable.com/v0/appM38HXIEVhxmnx/flaskdemo

PUT https://api.airtable.com/v0/appM38HXIEVhxmnx/flaskdemo

POST https://api.airtable.com/v0/appM38HXIEVhxmnx/flaskdemo

POST localhost:9007/addblog

August 4

POST 127.0.0.1:5000/addblog

POST 127.0.0.1:5000/

POST localhost:9007/addblog

POST localhost:9007/addblog

POST localhost:9007/addblog

POST localhost:9007/addblog

POST localhost:9007/addblog

POST localhost:9007/addblog

http http http http http http http No Environment

POST https://api.airtable.com/v0/appM38HXIEVhxmnx... Params Send Save

Authorization Headers (2) Body (selected) Pre-request Script Tests Code

form-data x-www-form-urlencoded raw binary JSON (application/json)

```
1 {  
2   "records": [  
3     {  
4       "fields": {  
5         "fname": "Bernard",  
6         "lname": "Suen",  
7         "student_id": "121212",  
8         "date_of_birth": "1997-7-1"  
9       }  
10      }  
11    ]  
12 }
```

Request

Status: 200 OK Time: 436 ms

Body Cookies Headers (11) Test Results

Pretty Raw Preview JSON ↗

```
1 {  
2   "records": [  
3     {  
4       "id": "recC41RXQdItIVvDM",  
5       "fields": {  
6         "lname": "Suen",  
7         "student_id": "121212"  
8       }  
9     }  
10   ]  
11 }
```

Response

PUT Method

ble.com/v0/appM3
8HXIEVhxmnqx/flaskdemo/rec8vF2
VblgOTdx3F

GET https://api.airtable.com/v0/appM3
8HXIEVhxmnqx/flaskdemo

DEL https://api.airtable.com/v0/appM3
8HXIEVhxmnqx/flaskdemo/recpOH
XSrfRmlvQia

PUT https://api.airtable.com/v0/appM3
8HXIEVhxmnqx/flaskdemo

POST https://api.airtable.com/v0/appM3
8HXIEVhxmnqx/flaskdemo

POST localhost:9007/addblog

Builder Team Library SYNC OFF Sign In

Chrome apps are being deprecated. [Download](#) our free native apps for continued support and better performance. [Learn more](#)

Filter Collections Clear all

PUT https://api.airtable.com/v0/appM38HXIEVhxmnqx... Params Send Save

Headers (2) Authorization: Bearer key Content-Type: application/json

Request

Status: 200 OK Time: 727 ms

Pretty Raw Preview JSON

```
[{"records": [{"id": "rect6erpDSPNty1TN", "fields": {"lname": "Trump", "student_id": "5354", "date_of_birth": "1948-02-21", "fname": "Donald", "full_name": "Donald Trump", "record_id": "rect6erpDSPNty1TN"}, "createdTime": "2019-09-19T15:18:41.000Z"}]}
```

PUT Method

table.com/v0/appM3
8HXIEVhxmnqx/flaskdemo/rec8vF2
VblgOTdx3F

GET https://api.airtable.com/v0/appM3
8HXIEVhxmnqx/flaskdemo

DEL https://api.airtable.com/v0/appM3
8HXIEVhxmnqx/flaskdemo/recpOH
XSrfRmlvQia

PUT https://api.airtable.com/v0/appM3
8HXIEVhxmnqx/flaskdemo

POST https://api.airtable.com/v0/appM3
8HXIEVhxmnqx/flaskdemo

POST localhost:9007/addblog

Builder Team Library SYNC OFF Sign In No Environment Filter Collections Clear all PUT https://api.airtable.com/v0/appM38HXIEVhxmnq... Params Send Save Authorization Headers (2) Body (raw JSON (application/json)) Tests Code

Request

```
{
  "records": [
    {
      "id": "rect6erpDSPNty1TN",
      "fields": {
        "fname": "Donald",
        "lname": "Trump",
        "student_id": "5354",
        "date_of_birth": "1948-02-21"
      }
    }
  ]
}
```

Body Cookies Headers (11) Test Results Status: 200 OK Time: 727 ms

Response

```
{
  "records": [
    {
      "id": "rect6erpDSPNty1TN",
      "fields": {
        "lname": "Trump"
      }
    }
  ]
}
```

DELETE Method

_collections

Request

Hit the Send button to get a response.

Do more with requests

Share Mock Monitor Document

The screenshot shows the Postman application interface. At the top, there are buttons for NEW, Runner, Import, and a plus sign icon. To the right are icons for Builder, Team Library, a gear, SYNC OFF, Sign In, and notifications. A yellow banner at the top states: "Chrome apps are being deprecated. Download our free native apps for continued support and better performance. Learn more". Below the banner, there's a search bar labeled "Filter" and a "Collections" section with a "Clear all" link. On the left, a list of API endpoints is shown with various methods (GET, POST, PUT, DEL) and URLs. The main workspace shows a DELETE request being configured. The URL is set to "https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo/recb1ZFR8ZL0HB9ti". The "Headers (2)" tab is selected, showing two headers: "Authorization" with value "Bearer key" and "Content-Type" with value "application/x-www-form-urlencoded". The "Send" button is highlighted in blue. Below the configuration, a message says "Hit the Send button to get a response." and "Do more with requests". At the bottom, there are buttons for Share, Mock, Monitor, and Document.

DELETE Method

Request

Response

The screenshot shows the Postman application interface. On the left, a sidebar lists various API endpoints with their methods and URLs. A specific endpoint for a DELETE method is highlighted. The main workspace shows a DELETE request being sent to the URL `https://api.airtable.com/v0/appM38HXIEVhxmnqx/flaskdemo`. The 'Body' tab is selected, showing the content type as `x-www-form-urlencoded`. The response body is displayed as a JSON object:

```
1 {  
2   "deleted": true,  
3   "id": "recb1ZFR8ZL0HB9ti"  
4 }
```

**JSON Method (RESTful API) to check
out our login form and album.**

Please login in first.

Vertical (basic) form

Email:

Password:

Remember me

```
109 ###### API Endpoints #####
110
111
112 @app.route('/api_auth', methods=['POST'])
113 def api_auth():
114     username = request.json['username']
115     password = request.json['password']
116     if username == "bernard@cuhk.edu.hk" and password == "1234":
117         login_user(User(1))
118         message = "Dear " + username + ", welcome to Bernard's pages. Your login has been granted."
119         return jsonify({'response':'ok!', 'message':message})
120     else:
121         message = 'Wrong password!'
122         return jsonify({'response':'Invalid!', 'message':'Cannot authenticate.'})
123
```

NEW  Runner  Import  Builder Team Library   SYNC OFF  Sign In    

⚠ Chrome apps are being deprecated. [Download](#) our free native apps for continued support and better performance. [Learn more](#) 

Filter  History Collections  

All Me Team

JS Ajax REST Get/Post 
2 requests

POST http://localhost:9009/api/post

GET http://localhost:9009/api/get

RESTful Test
14 requests

DEL https://api.airtable.com/v0/appM3...

POST 127.0.0.1:5000/addblog

GET https://api.airtable.com/v0/appM3...

GET https://api.airtable.com/v0/appM3...

POST https://api.airtable.com/v0/appM3...

PUT https://api.airtable.com/v0/appM3...

DEL https://api.airtable.com/v0/appM3...

GET https://api.airtable.com/v0/appM3...

GET https://api.airtable.com/v0/appM3...

localhost:  localhost:  http://loc:  localhost:  localhost:  + ... No Environment   

POST  localhost:9030/api_auth Params  

Authorization Headers (1) Body Pre-request Script Tests 

Key	Value	Description	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> Content-Type	application/json				
New key	Value	Description			

Body Cookies Headers (5) Test Results Status: 405 METHOD NOT ALLOWED Time: 60 ms

allow → POST, OPTIONS

content-length → 178

content-type → text/html

date → Mon, 02 Mar 2020 07:52:46 GMT

server → Werkzeug/0.16.0 Python/3.7.4

NEW  Runner  Import 

Builder

Team Library



SYNC OFF

Sign In



 Chrome apps are being deprecated. [Download](#) our free native apps for continued support and better performance. [Learn more](#) 

 Filter

History  Collections

All Me Team  

 JS Ajax REST Get/Post 
2 requests

 http://localhost:9009/api/post

 http://localhost:9009/api/get

 RESTful Test
14 requests

 https://api.airtable.com/v0/appM3...

 127.0.0.1:5000/addblog

 https://api.airtable.com/v0/appM3...

 https://api.airtable.com/v0/appM3...

 https://api.airtable.com/v0/appM3...

 https://api.airtable.com/v0/appM3...

 https://api.airtable.com/v0/appM3...

 https://api.airtable.com/v0/appM3...

 https://api.airtable.com/v0/appM3...

localhost:  localhost:  http://loc...  localhost:  localhost:  + ... No Environment

POST  localhost:9030/api_auth

Params

Send 

Save 

Authorization Headers (1) Body  Pre-request Script Tests

Code

form-data x-www-form-urlencoded raw binary JSON (application/json) 

1 {"username": "bernard@cuhk.edu.hk", "password": "1234"}|

Body Cookies Headers (5) Test Results

Status: 405 METHOD NOT ALLOWED

Time: 60 ms

allow → POST, OPTIONS

```
124 @app.route('/api_album', methods=['GET'])
125 @login_required
126 def api_album():
127     user = {"name": "Bernard"}
128     with connection.cursor() as cursor:
129         sql = "SELECT * FROM products"
130         cursor.execute(sql)
131         result = cursor.fetchall()
132     final_result = [list(i) for i in result]
133     dataset=[]
134     dict={}
135     for i in final_result:
136         dict['productID'] = i[0]
137         dict['productCode'] = i[1]
138         dict['name'] = i[2]
139         dict['quantity'] = i[3]
140         dict['price'] = float(i[4])
141         dict['supplierID'] = i[5]
142         dataset.append(dict.copy()) #markers.append(fld.copy())
143     return jsonify({'Album': dataset})
144
```

NEW Runner Import C+ Builder Team Library

SYNC OFF Sign In

Chrome apps are being deprecated. Download our free native apps for continued support and better performance. [Learn more](#)

Filter History Collections

All Me Team

JS Ajax REST Get/Post ★
2 requests

POST http://localhost:9009/api/post

GET http://localhost:9009/api/get

RESTful Test
14 requests

DEL https://api.airtable.com/v0/appM3...

POST 127.0.0.1:5000/addblog

GET https://api.airtable.com/v0/appM3...

GET https://api.airtable.com/v0/appM3...

POST https://api.airtable.com/v0/appM3...

PUT https://api.airtable.com/v0/appM3...

DEL https://api.airtable.com/v0/appM3...

GET https://api.airtable.com/v0/appM3...

GET https://api.airtable.com/v0/appM3...

localhost: localhost: localhost: http://loc... localhost: localhost: + ... No Environment

GET localhost:9030/api_album Params Send Save

Authorization Headers (1) Body Pre-request Script Tests Code

Key	Value	Description	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> Content-Type	application/json				
New key	Value	Description			

Body Cookies Headers (5) Test Results Status: 200 OK Time: 60 ms

Pretty Raw Preview

```
{"Album": [{"name": "Pen Red", "price": 1.23, "productCode": "PEN", "productID": 1001, "quantity": 5000, "supplierID": 501}, {"name": "Pen Blue", "price": 1.25, "productCode": "PEN", "productID": 1002, "quantity": 8000, "supplierID": 501}, {"name": "Pen Black", "price": 1.25, "productCode": "PEN", "productID": 1003, "quantity": 2000, "supplierID": 501}, {"name": "Pencil 2B", "price": 0.48, "productCode": "PEC", "productID": 1004, "quantity": 10000, "supplierID": 501}, {"name": "Pencil 2H", "price": 0.49, "productCode": "PEC", "productID": 1005, "quantity": 8000, "supplierID": 502}, {"name": "Pencil HB", "price": 99999.99, "productCode": "PEC", "productID": 1006, "quantity": 0, "supplierID": 501}, {"name": "Pencil 3B", "price": 0.52, "productCode": "PEC", "productID": 2001, "quantity": 500, "supplierID": 503}, {"name": "Pencil 4B", "price": 0.62, "productCode": "PEC", "productID": 2002, "quantity": 200, "supplierID": 501}, {"name": "Pencil 5B", "price": 0.73, "productCode": "PEC", "productID": 2003, "quantity": 100, "supplierID": 503}, {"name": "Pencil 6B", "price": 0.47, "productCode": "PEC", "productID": 2004, "quantity": 500, "supplierID": 502}]} 
```

Why is API so important?

GROWTH STRATEGY

The Strategic Value of APIs

by Bala Iyer and Mohan Subramaniam

JANUARY 07, 2015

McKinsey&Company
Digital McKinsey

How We Help Clients Our Insights Careers Our People Contact Us

Article
October 2017

Management's next frontier: Making the most of the ecosystem economy

By Jürgen Metzert and Anand Swaminathan

API經濟來了：從開放創新到Open API

by 《數位時代》整合行銷部 2014.11.04



McKinsey&Company
Digital McKinsey

How We Help Clients Our Insights Careers Our People Contact Us

Article
September 2017

What it really takes to capture the value of APIs

≡ Deloitte.
Insights

Article

API economy

From systems to business services

IBM Cloud 雲端平台 > 瞭解雲端

API 經濟

在您的營運模式中加入 API，
藉此催化創新並觸角伸向新對象

[觀賞影片 \(01:25\)](#)[閱讀 API 入門電子書](#)

關於 IBM

參加評估



An Excerpt From

Creating a Successful API Program to Drive Digital Transformation

Ed Murphy
API Practice Lead, Perficient

<https://www.youtube.com/watch?v=DtC3fR48Pvs>



Product Blocks Templates Universe Pricing Sign in [Sign up](#)

Create, your way

Part spreadsheet, part database, and entirely flexible, teams use Airtable to organize their work, their way.

Enter your email address

[Get started](#)

Already using Airtable? [Sign in](#)



[INTRODUCTION](#)[RATE LIMITS](#)[AUTHENTICATION](#)[PRODUCT TABLE](#)[ROLL-UP TABLE](#)[VENUES TABLE](#)[USERS TABLE](#)[SW_PRODUCT TABLE](#)[ERRORS](#)

AUTHENTICATION

Airtable uses simple token-based authentication. To generate or manage your API key, visit your [account](#) page. **Your API key carries the same privileges as your user account, so be sure to keep it secret!**

You can authenticate to the API by providing your API key in the HTTP authorization bearer token header. Alternatively, a slightly lower-security approach is to provide your API key with the [api_key](#) query parameter.

All API requests must be authenticated and made over HTTPS.

PRODUCT TABLE

Fields

Each record in the [Product](#) table contains the following fields:

FIELD NAME	TYPE	DESCRIPTION
company	Text	string A single line of text.

[curl](#)[JavaScript](#) show API key

EXAMPLE USING BEARER TOKEN (RECOMMENDED)

```
$ curl https://api.airtable.com/v0/appKIU0zkdHt3AVTL/Product \
-H "Authorization: Bearer key[REDACTED]"
```

EXAMPLE USING QUERY PARAMETER

```
$ curl https://api.airtable.com/v0/appKIU0zkdHt3AVTL \
/Product?api_key=key[REDACTED]
```

EXAMPLE VALUES

```
"Phillips"
"Phillips"
"Phillips"
"Phillips"
```

指南



开发指南

小程序提供了一个简单、高效的应用开发框架和丰富的组件及API，帮助开发者在微信中开发具有原生APP体验的服务。

本章分主题的介绍了小程序的开发语言、框架、能力、调试等内容，帮助开发者快速全面的了解小程序开发的方方面面。

想要更具体了解关于框架、组件、API的详细内容，请参考对应的参考文档：

- [小程序框架参考文档](#)
- [小程序组件参考文档](#)
- [小程序 API 参考文档](#)
- [小程序服务端 API 参考文档](#)



Mini Programs Providers

Partner with WeChat and millions of developers to enable the Mini Programs ecosystem

[Become a Provider](#)



Benefits for Providers

Benefits for Providers



WeChat official licensing

Help to implement and standardize services



Efficient Development

No need to figure out complicated parameters



WeChat Services

Grow at different stages by leveraging our Growth Plan

Provider Growth Plan



WeChat Academy

WeChat Mini Program ecosystem knowledge sharing platform.
Helps providers improve by providing a platform for sharing
technical know-how and experience.



APIs for Basic Functionality

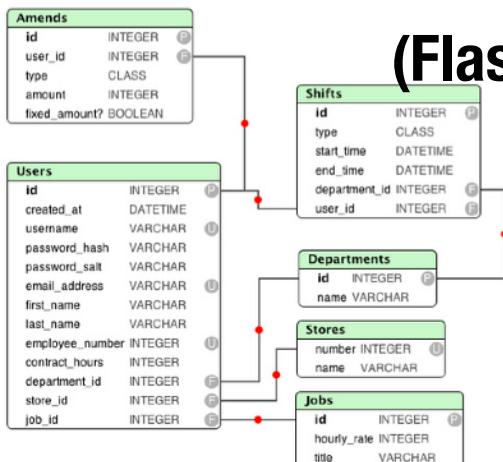
A full range of APIs are available for developing basic functions

API is an important part of the
MVC Framework.

Source: commons.wikimedia.org



Source: [Caius Durling \(Flickr\)](https://flic.kr)

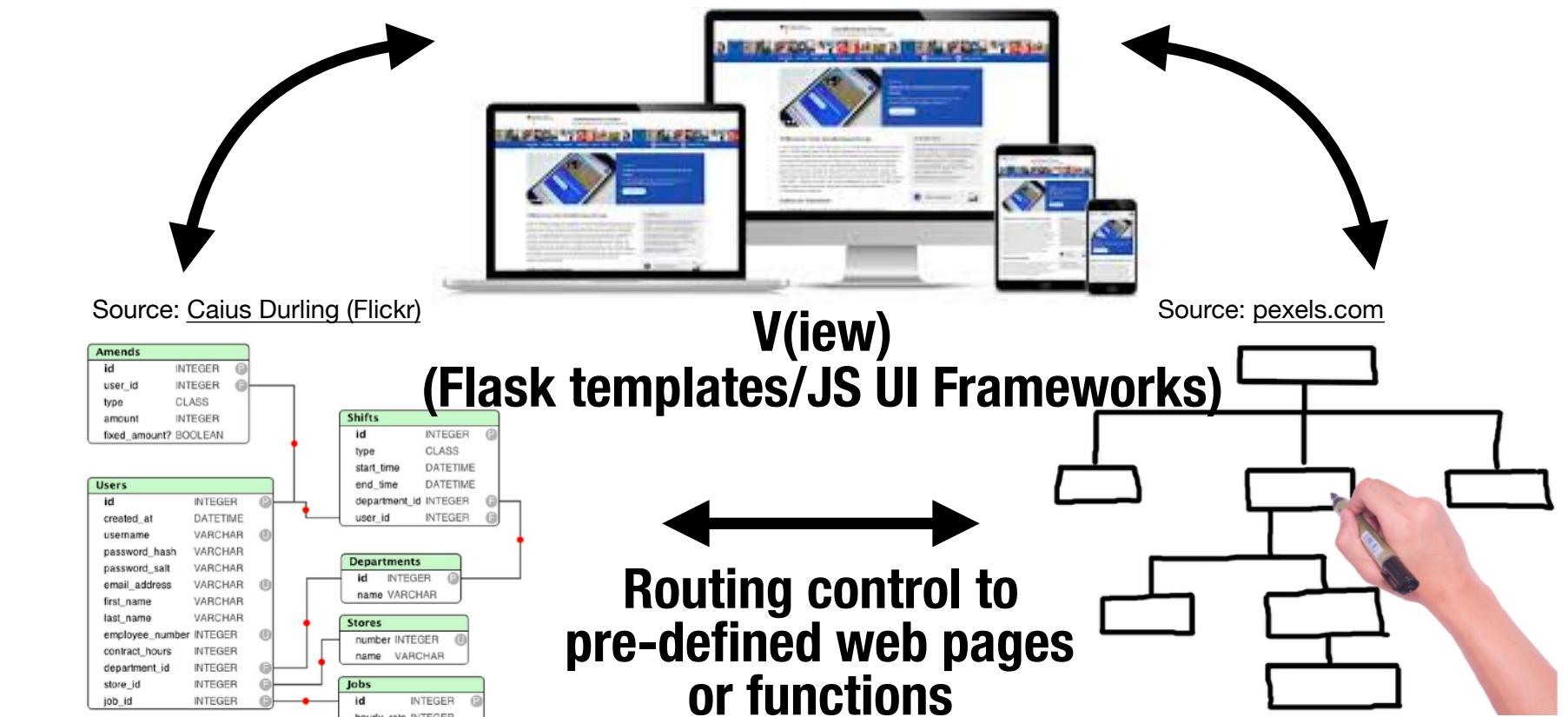


M(odel)
(data and algorithmic models)

V(iew) (Flask templates/JS UI Frameworks)

Routing control to
pre-defined web pages
or functions

C(ontroller)
(e.g. Flask Routes/API Endpoints)



Source: pexels.com

URI Routing in Flask & REST Endpoints

API (e.g. REST)



BACK-END TECHNOLOGY

后台

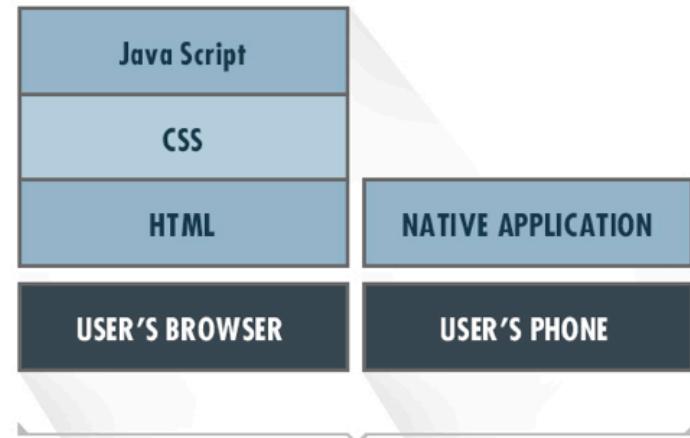
WHAT IS YOUR “CLOUD” AND “STACK” STRATEGY?

云架构及前端和后端的全栈策略

{ JSON }



THE INTERNET



CLIENT-SIDE

FRONT-END TECHNOLOGY

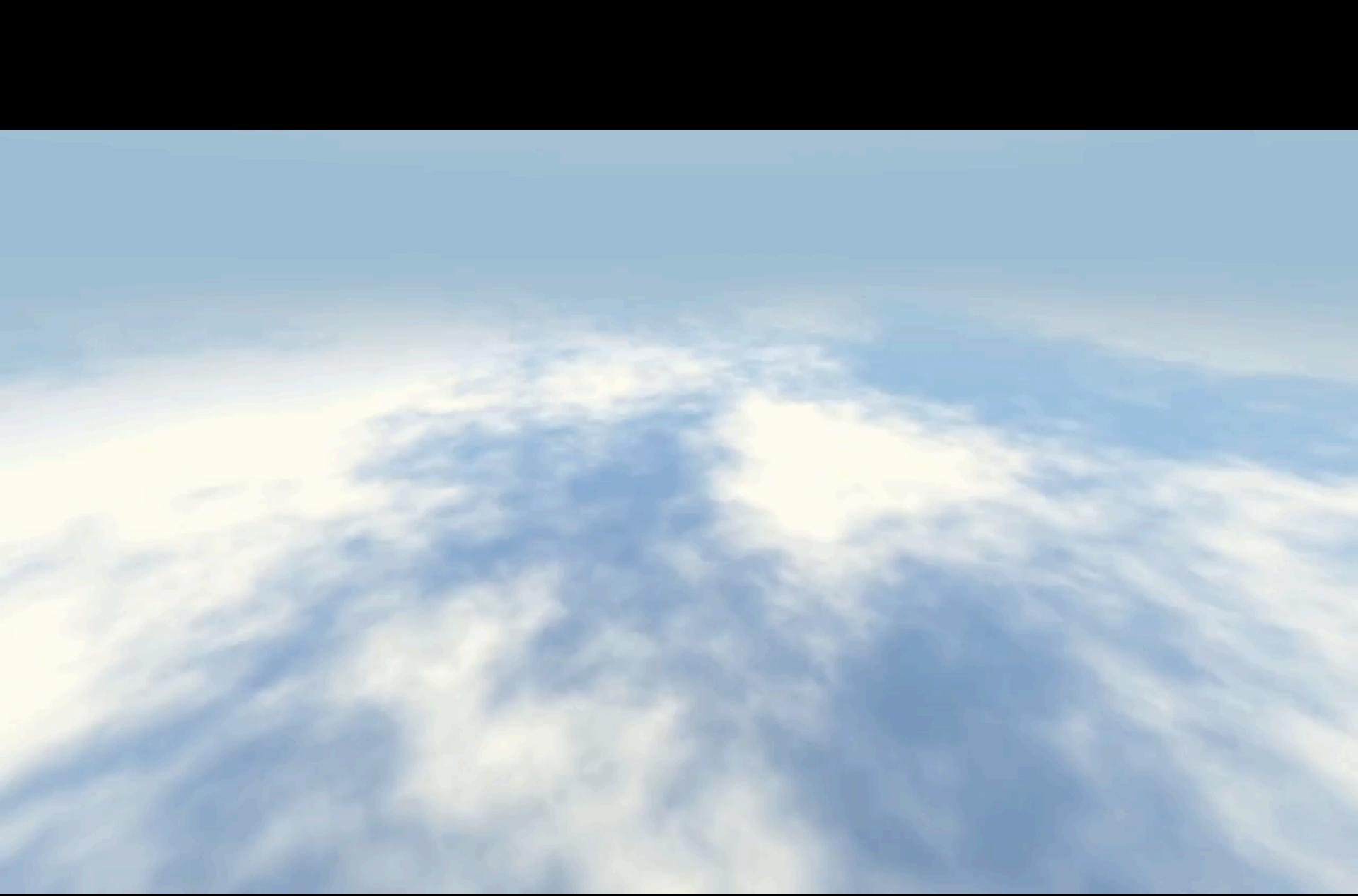
前台



{World of Internet}

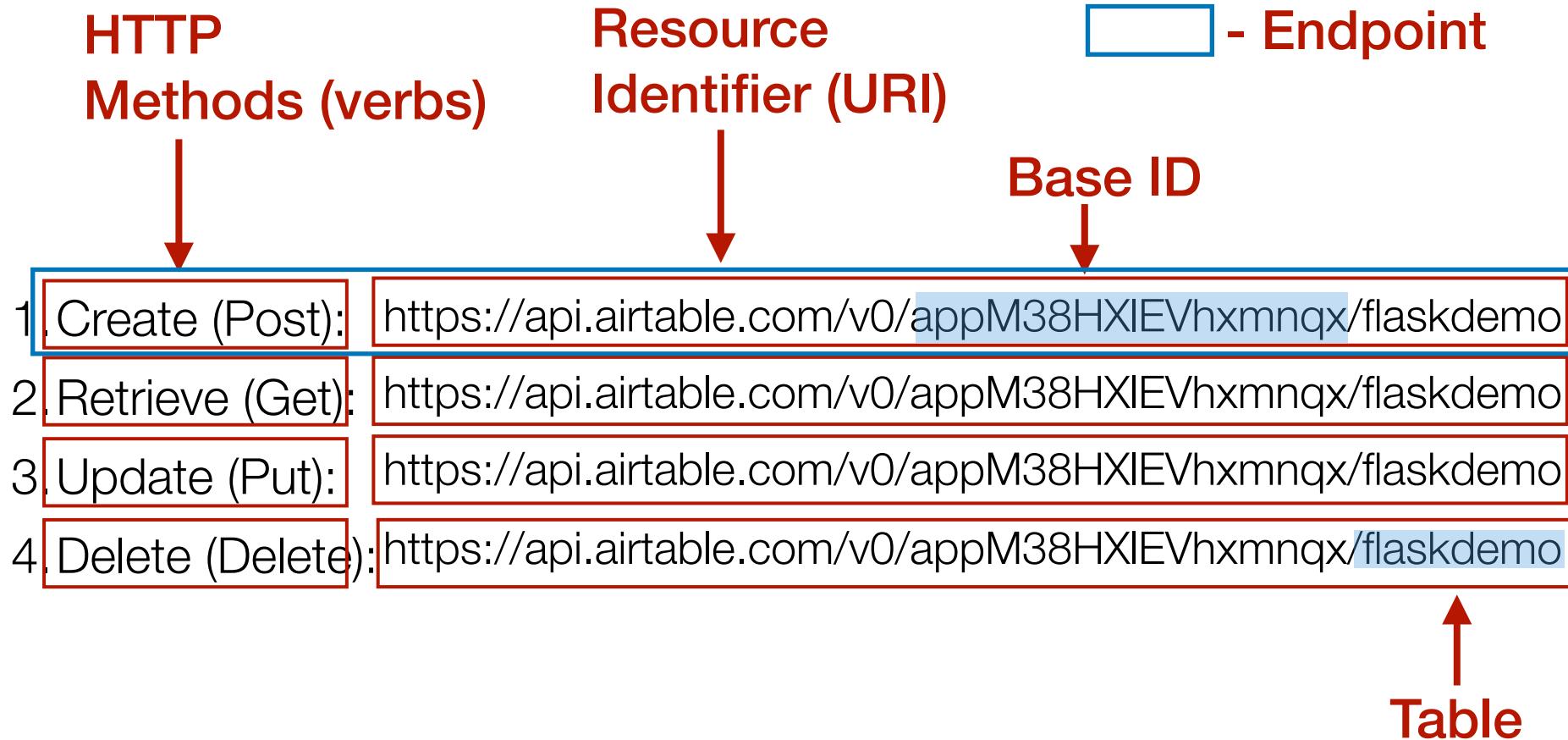


https://www.youtube.com/watch?v=lvlnf9HW_Jw



<https://www.youtube.com/watch?v=u-RnFs9dby4>

Basic RESTful Methods and Structure for Airtable



Examples of endpoints

- Endpoint

Here's an example of the endpoints for the Relationships resource in the Instagram API:

The screenshot shows a web browser displaying the Instagram API developer documentation at <https://www.instagram.com/developer/endpoints/relationships/>. The left sidebar lists various endpoints under the 'Endpoints' section, with 'Relationships' currently selected. The main content area displays two API endpoints for the 'Relationships' resource.

Endpoint 1: /users/self/follows

Method: GET
URL: https://api.instagram.com/v1/users/self/Follows?access_token=ACCESS-TOKEN
Response: [button]
Description: Get the list of users this user follows.
Requirements: Scope: follower_list
Parameters: ACCESS_TOKEN: A valid access token.

Endpoint 2: /users/self/followed-by

Method: GET
URL: https://api.instagram.com/v1/users/self/Followed-by?access_token=ACCESS-TOKEN
Response: [button]
Description: Get the list of users this user is followed by.
Requirements: Scope: follower_list
Parameters: ACCESS_TOKEN: A valid access token.



Airtable RESTful API

INTRODUCTION

The [Project Tracker](#) API provides an easy way to integrate your [Project Tracker](#) data in Airtable with any external system. The API closely follows REST semantics, uses JSON to encode objects, and relies on standard HTTP codes to signal operation outcomes.

The API documentation below is specifically generated for your base. We recommend that you use the [graphical Airtable interface](#) to add a few records of example data for each table. These records will be displayed in the documentation examples generated below.

The ID of this base is [appM38HX1EVhxmnqx](#).

Please note: if you make changes to a field (column) name or type, the API interface for those fields will change correspondingly. Therefore, please make sure to update your API implementation accordingly whenever you make changes to your Airtable schema from the graphical interface.

Official API client:

- JavaScript: [airtable.js](#) (Node.js + browser)

Community-built API clients:

- Ruby: [airrecord](#)
- .NET: [airtable.net](#)

RATE LIMITS

The API is limited to 5 requests per second. If you exceed this rate, you will receive a 429 status code and will need to wait 30 seconds before subsequent requests will succeed.

The [official JavaScript client](#) has built-in retry logic.

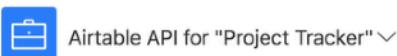
If you anticipate a higher read volume, we recommend using a caching proxy. This rate limit is the same for all plans and increased limits are not currently available.

curl

JavaScript

 show API key

Airtable RESTful API



Open base...

INTRODUCTION	List flaskdemo records
RATE LIMITS	To list records in <code>flaskdemo</code> , issue a GET request to the <code>flaskdemo</code> endpoint.
AUTHENTICATION	Returned records do not include any fields with "empty" values, e.g. "", [], or <code>false</code> .
DESIGN PROJECTS TABLE	You can use the following parameters to filter, sort, and format the results:
TASKS TABLE	fields <code>array of strings</code> optional Only data for fields whose names are in this list will be included in the result. If you don't need every field, you can use this parameter to reduce the amount of data transferred.
CLIENTS TABLE	For example, to only return data from <code>fame</code> and <code>lname</code> , send these two query parameters:
STAGE TABLE	<code>fields= fname</code> <code>fields= lname</code>
VENUES TABLE	
IMPORTED TABLE TABLE	
USERS TABLE	
FLASKDEMO TABLE Fields	filterByFormula <code>string</code> optional A formula used to filter records. The formula will be evaluated for each record, and if the result is not <code>0</code> , <code>false</code> , "", <code>NaN</code> , [], or <code>#Error!</code> the record will be included in the response.
List records	If combined with <code>view</code> , only records in that view which satisfy the formula will be returned.
Retrieve a record	
Create records	
Update records	
Delete records	
ERRORS	

curl JavaScript

EXAMPLE REQUEST

```
curl "https://api.airtable.com/v0/appM38HXLEVhxmnqx/flaskdemo?maxRecords=3&view=Grid%20view" \
-H "Authorization: Bearer keyTcsTzckqyBT1k8"
```

EXAMPLE RESPONSE

```
{
  "records": [
    {
      "id": "recp0HXSrfrMIVQia",
      "fields": {
        "fame": "John",
        "lname": "Chan",
        "student_id": "12349",
        "date_of_birth": "2009-02-21"
      },
      "createdTime": "2019-09-19T14:04:24.000Z"
    },
    {
      "id": "recASxs4aPaHLejj3",
      "fields": {
        "fame": "Emily",
        "lname": "Wong",
        "student_id": "67891",
        "date_of_birth": "2008-04-03"
      },
      "createdTime": "2019-09-19T14:04:24.000Z"
    },
    {
      "id": "rec8vF2VbIg0Tdx3F",
      "fields": {
        "fame": "Mary",
        "lname": "Lu",
        "student_id": "99999",
        "date_of_birth": "2001-08-22"
      },
      "createdTime": "2019-09-19T14:32:08.000Z"
    }
  ]
}
```

show API key



The Wikimedia Foundation wants to make talk pages accessible to all, to improve the depth, quality and the health on the wikis.
We invite all Wikimedians to consult on how to achieve this goal. Join us!



[Help with translations!]

List of HTTP status codes

From Wikipedia, the free encyclopedia

This is a list of [Hypertext Transfer Protocol](#) (HTTP) response status codes. Status codes are issued by a server in response to a client's request made to the server. It includes codes from IETF [Request for Comments](#) (RFCs), other specifications, and some additional codes used in some common applications of the [Hypertext Transfer Protocol](#) (HTTP). The first digit of the status code specifies one of five standard classes of responses. The message phrases shown are typical, but any human-readable alternative may be provided. Unless otherwise stated, the status code is part of the HTTP/1.1 standard ([RFC 7231](#)).^[1]

The [Internet Assigned Numbers Authority](#) (IANA) maintains the official registry of HTTP status codes.^[2]

Microsoft [Internet Information Services](#) (IIS) sometimes uses additional decimal sub-codes for more specific information,^[3] however these sub-codes only appear in the response payload and in documentation, not in the place of an actual HTTP status code.

All HTTP response status codes are separated into five classes (or categories). The first digit of the status code defines the class of response. The last two digits do not have any class or categorization role. There are five values for the first digit:

HTTP

[Persistence](#) · [Compression](#) · [HTTPS](#) · [QUIC](#)

Request methods

[OPTIONS](#) · [GET](#) · [HEAD](#) · [POST](#) · [PUT](#) ·
[DELETE](#) · [TRACE](#) · [CONNECT](#) · [PATCH](#)

Header fields

[Cookie](#) · [ETag](#) · [Location](#) · [HTTP referer](#) · [DNT](#) ·
· [X-Forwarded-For](#)

Status codes

[301 Moved Permanently](#) · [302 Found](#) ·
[303 See Other](#) · [403 Forbidden](#) ·
[404 Not Found](#) ·
[451 Unavailable For Legal Reasons](#)

Security access control methods

[Basic access authentication](#) ·
[Digest access authentication](#)

V · T · E

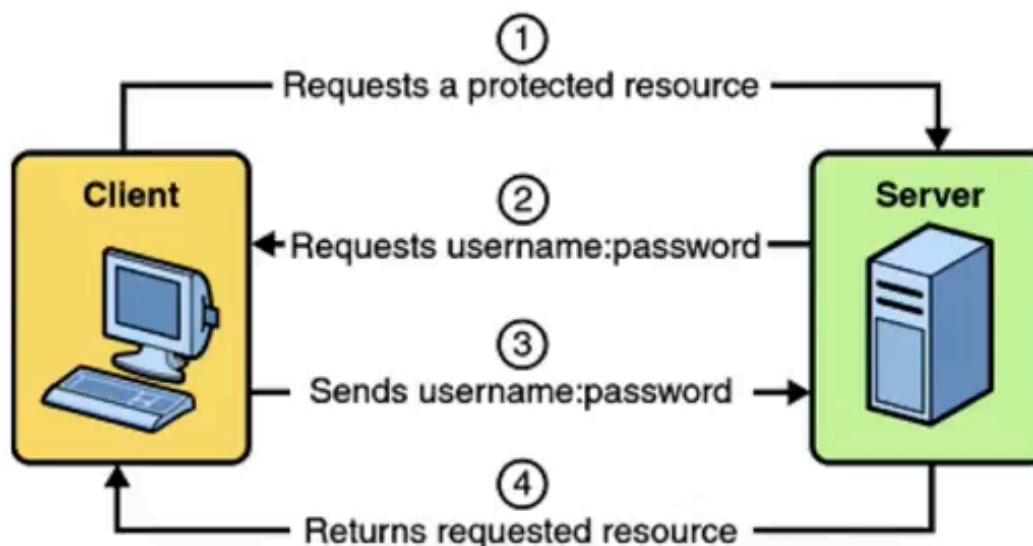
- **200 - OK** : Means that the request has been completed successfully and the server has returned the response. Usually returned after a successful **GET** request.
- **201 - Created** : Usually returned after a successful **POST** request. Summarizes that the resource has been created.
- **400 - Bad Request** : It's returned from the server when a request was sent with some missing or invalid parameters. Usually returned in response to **POST** or **PUT** requests.
- **401 - Unauthorized** : Means that user was not authorized to perform certain action. For example, a user tried to create or delete a resource without providing authentication credentials.
- **403 - Forbidden** : Means that server understood the request but refused to complete it due to authentication. It happens when a user provides authentication credentials but it doesn't have sufficient rights to perform the action.
- **404 - Not Found** : The most (in)famous of all the status codes. Summarizes that a resource, the user was looking for, was not found.
- **405 - Method not Allowed** : Means that an HTTP verb supplied in the request was not supported by the resource. An example might be a user trying to update a read-only resource.
- **410 - Gone** : Means that a resource has been moved to another location. An example might be trying to delete an already deleted resource that has been moved to trash.
- **500 - Internal Server Error** : It's returned when a server encounters an unexpected condition and doesn't complete the request.

How to limit API access to authenticated and authorised people?

To grab data through **RESTful API**, you need:

- URI (Resources Identifier)
- HTTP method
- Headers (Authorization, Content-Type)
- Body (JSON, form-urlencoded)
- Authentication methods (e.g. Basic, OAuth 2, JWT)

Basic Authentication



Source: <http://docs.oracle.com/cd/E19226-01/820-7627/bnc1>





ScienceDirect

Journals & Books



Register

Sign in

Basic Authentication

Related terms:

[Authentication Method](#),
[Digest Authentication](#)

[View all Topics >](#)

[Download as PDF](#) [Set alert](#)

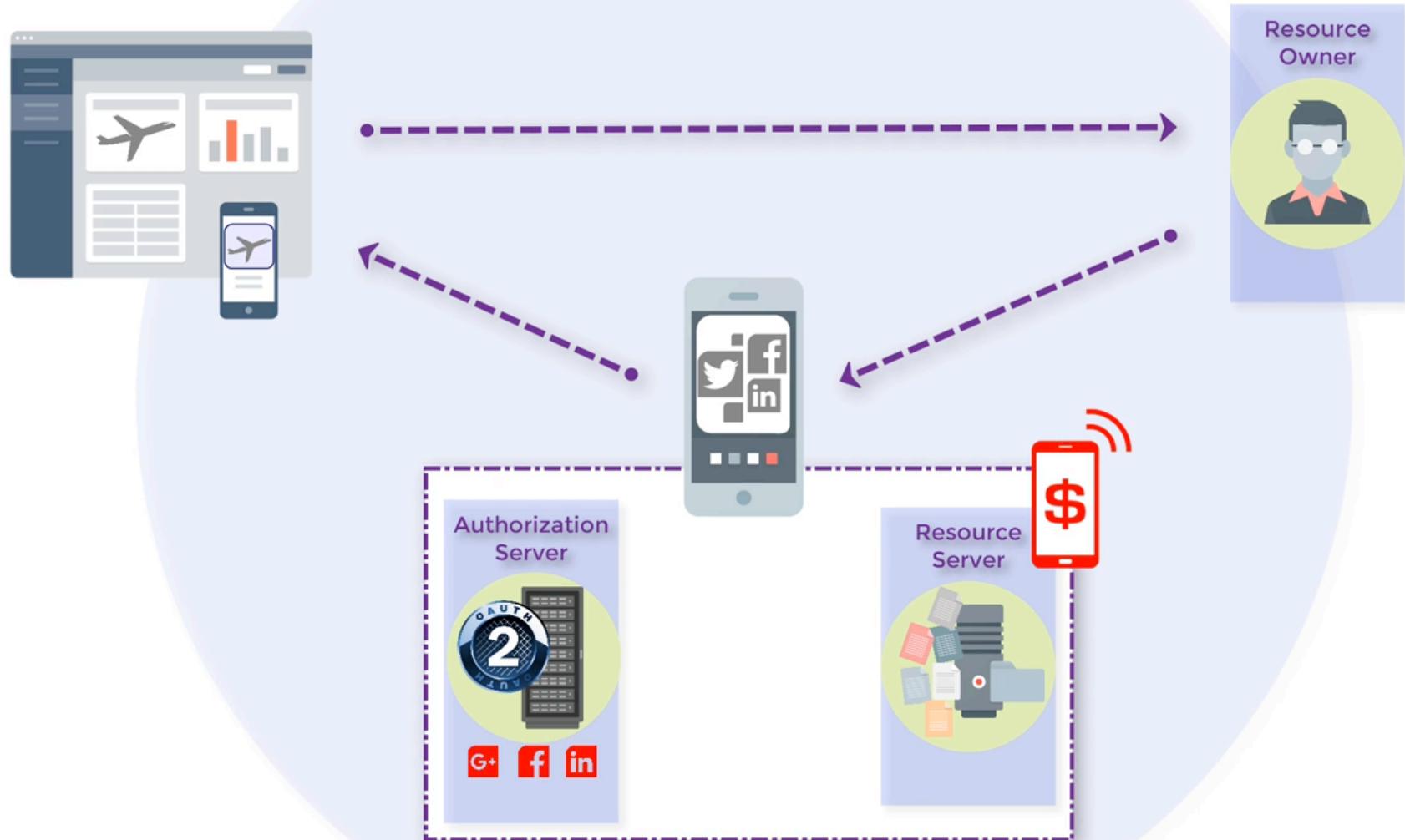
[About this page](#)

Authentication and Granular
Access

Administration of an IIS 7.0 Web
Server

<https://www.sciencedirect.com/topics/computer-science/basic-authentication>

OAuth 2.0



<https://www.youtube.com/watch?v=zEysfglbqlg>



Language: EN ▾

Contents

[OAuth Roles](#)[Abstract Protocol Flow](#)[Application Registration](#)[Authorization Grant](#)[Grant Type: Authorization Code](#)[Grant Type: Implicit](#)[Grant Type: Resource Owner Password Credentials](#)[Grant Type: Client Credentials](#)

An Introduction to OAuth 2

Posted July 21, 2014 ① 1.6m

SECURITY

API

CONCEPTUAL

By [Mitchell Anicas](#)
[Become an author](#)

Introduction

OAuth 2 is an authorization framework that enables applications to obtain limited access to user accounts on an HTTP service, such as Facebook, GitHub, and DigitalOcean. It works by delegating user authentication to the service that hosts the user account, and authorizing third-party applications to access the user account. OAuth 2 provides authorization flows for web and desktop applications, and mobile devices.

This informational guide is geared towards application developers, and provides an overview of OAuth 2 roles, authorization grant types, use cases, and flows.

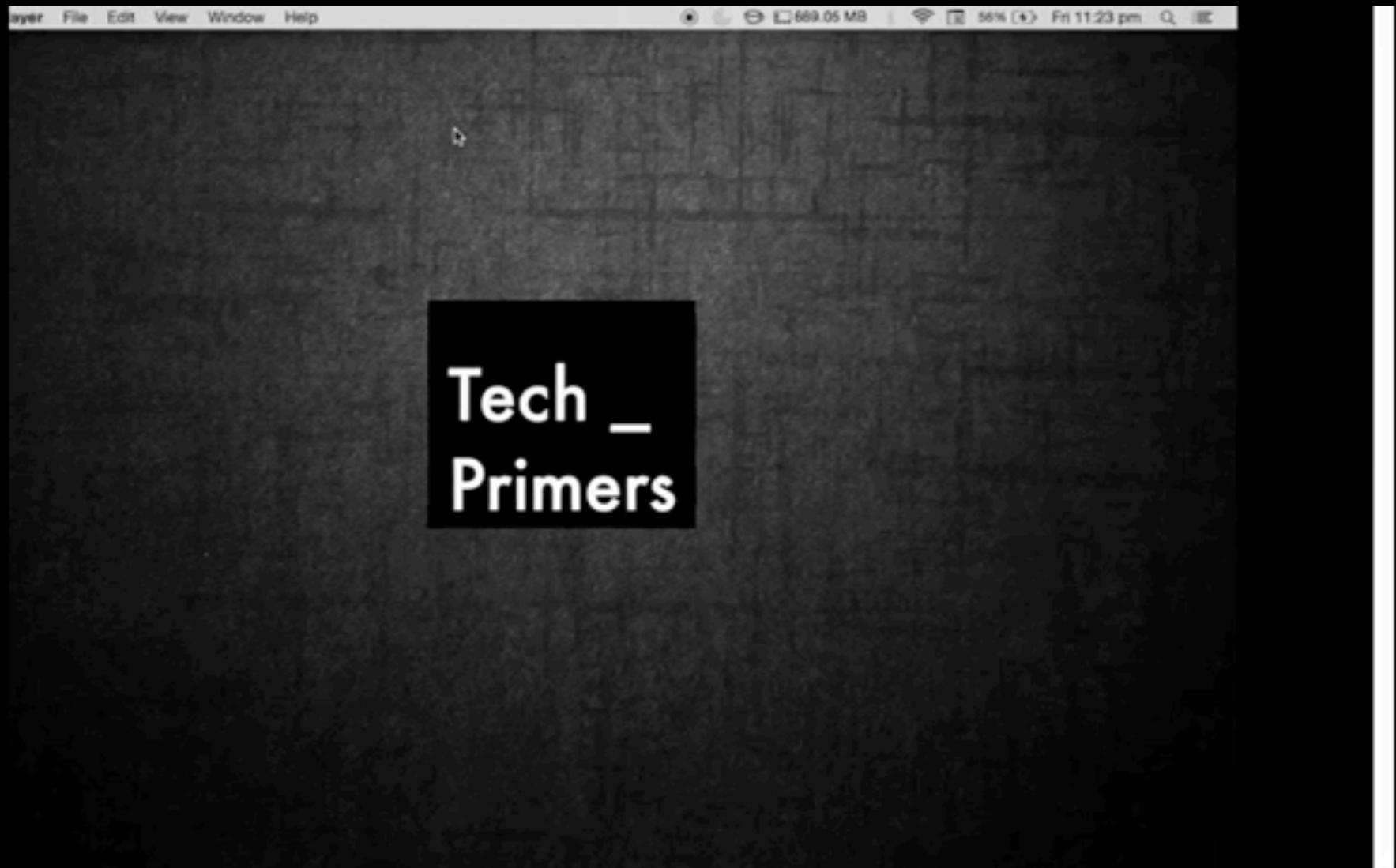
Let's get started with OAuth Roles!

Related

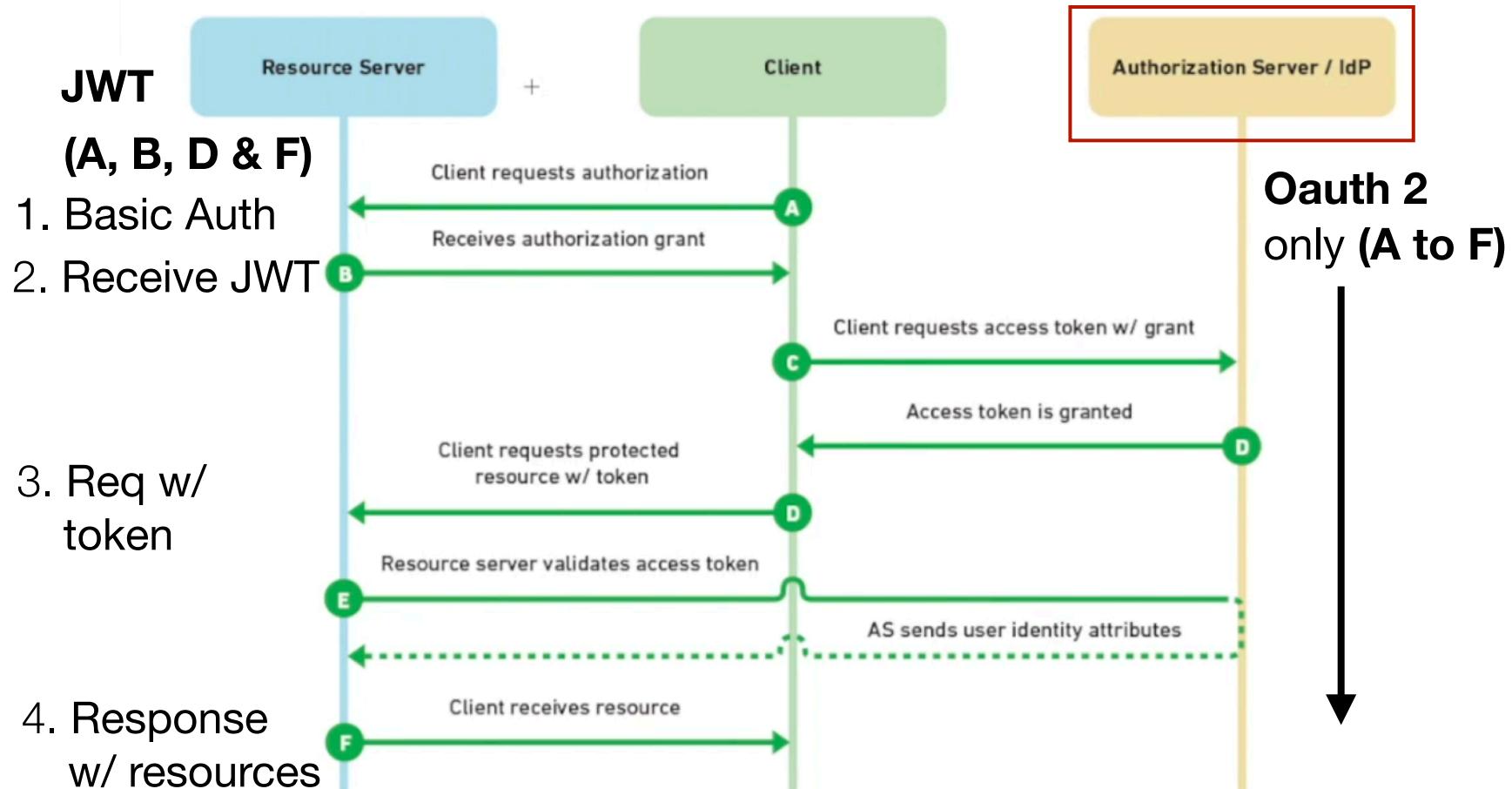
[Recommended Steps To Harden Apache HTTP on FreeBSD 12.0](#) [Tutorial](#)[How To Set Up SSH Keys on CentOS 8](#) [Tutorial](#)

<https://www.digitalocean.com/community/tutorials/an-introduction-to-oauth-2>

JWT vs. OAuth 2



https://www.youtube.com/watch?time_continue=55&v=a9R3Gq1BKxI&feature=emb_logo



Introduction to JWT

👤 Avi Aryan - ⏰ July 2, 2016 - 📁 Event Management / Open Event

In this post, I will try to explain what is JWT, what are its advantages and why you should be using it.

JWT stands for JSON Web Tokens. Let me explain what each word means.

1. **Tokens** – Token is in tech terms a piece of data (claim) which gives access to certain piece of information and allows certain actions.
2. **Web** – Web here means that it was designed to be used on the web i.e. web projects.
3. **JSON** – JSON means that the token can contain json data. In JWT, the json is first serialized and then **Base64 encoded**.

A JWT looks like a random sequence of strings separated by 2 dots. The **yyyyy** part which you see below has the Base64 encoded form of json data mentioned earlier.

<https://blog.fossasia.org/introduction-to-jwt/>

Summary

3 Common API Authentication Methods

	BASIC AUTH (e.g. Flask-login)	OAUTH 2 (e.g. Wechat, FB)	JWT (e.g. Airtable)
Encryption	No (Rely on https)	Yes	Yes
Authentication	User Name/ Password	Token (from resource provider)	Token (stored locally)
Authorization	NA Rely on server- side programs.	Yes	Yes
Multiple Parties	NA	Yes	NA

**MySQL
on
PythonAnywhere**

The RESTful API

**Airtable
for
Agile PM**

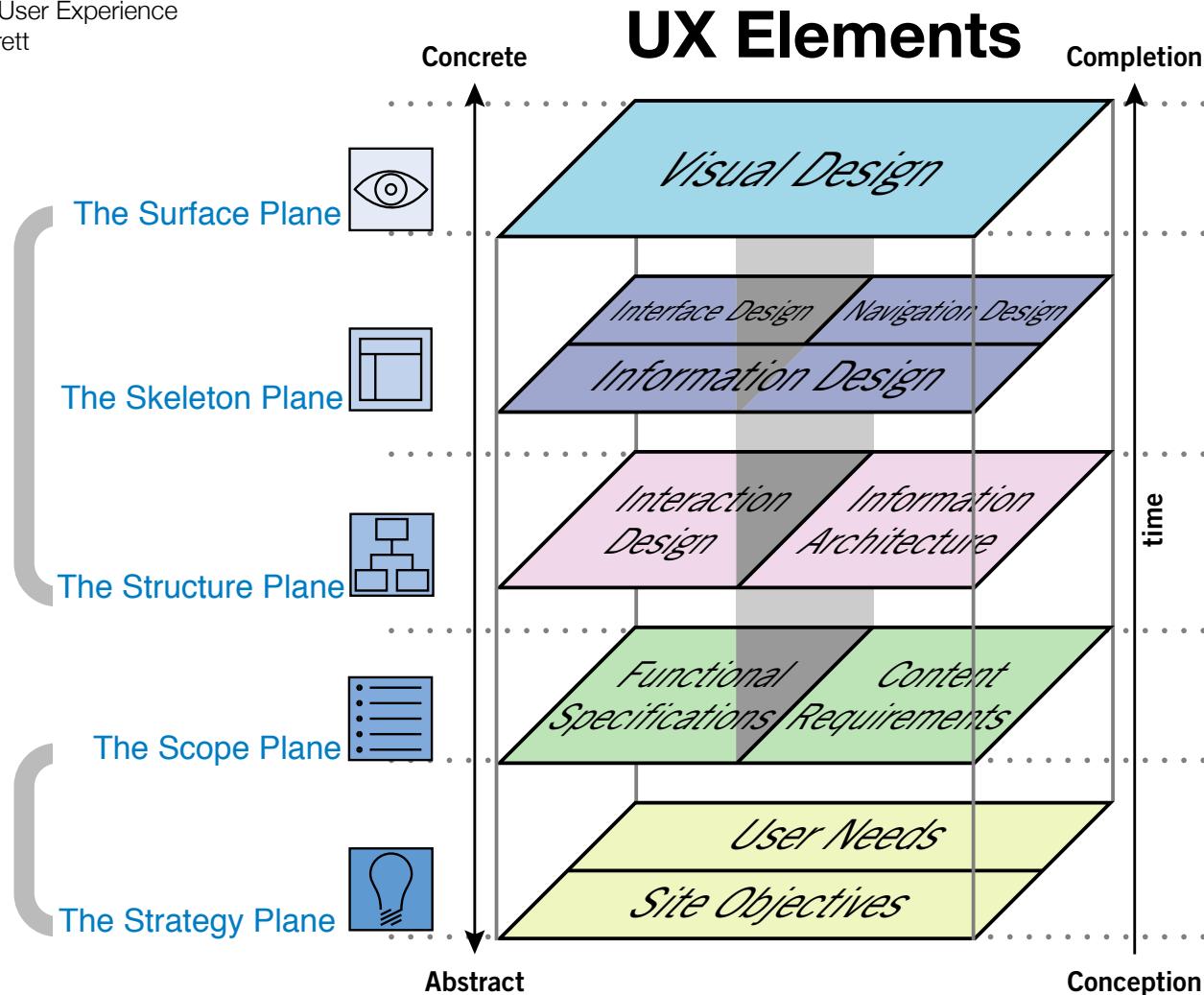
Source: Elements of User Experience
by Jesse James Garrett

Solution Space

how and how much

Problem Space

who, what, and why



 com5940 2020

[Students](#) [Scores](#) [Teams](#)
[Backlog](#) ▾

[Sprints](#)
[Strategic_Journey_Stages](#)
[Lessons](#)
[Tasks](#) +

[SHARE](#)

▼ [Grid view](#)

1 hidden field

[Filter](#)

[Group](#)

[Sort](#)

[Color](#)


	Story	Notes	Link_to_Journey	Status
1	Understand Lean Method Approach	Students get familiar with lean method basic co...	Learn full stack MVC SCRUM	Research
2	MAMP MySQL Administration Setup	Students learn to setup MAMP MySQL admin t...	Learn full stack MVC SCRUM	Problem Set 1
3	Jupyter Notebook Setup	Students learn to set up Jupyter Notebook for t...	Cross-platform Development	
4	Differences between Web, PWA, Hybrid, and Native App.	Students can tell the major differences betwee...	Learn full stack MVC SCRUM	Cross-platform
5	PhoneGap Setup	Students learn to use PhoneGap desktop and P...	Learn full stack MVC SCRUM	Master REST
6	PythonAnywhere Setup	Students learn to setup a PythonAnywhere acc...	Master REST API	Learn full stack MVC SCR
7	Adobe PhoneGap Build account	Students get to open up an account at build.ph...	Master REST API	Learn full stack MVC SCR
8	MySQL and CRUD	Students master basic MySQL and CRUD to pr...	Learn full stack MVC SCRUM	
9	Full-stack MVC concepts	Students can distinguish front-end from back-...	Learn full stack MVC SCRUM	
10	Flask and Jinja2 render_template engine	Students learn to use the Flask and Jinja2 rend...	Master REST API	Cross-platform Developm
11	SCRUM terms	Understand the SCRUM method and the key te...	Learn full stack MVC SCRUM	
12	DT, Lean, and Agile for SCRUM	Students watch online videos and class lecture...	Learn full stack MVC SCRUM	
13	Product Requirement Plan	Students will use Elements of UX as a framewo...	Learn full stack MVC SCRUM	
14	Product Release Plan	Students will use the SCRUM method to define ...	Learn full stack MVC SCRUM	

 com5940 2020

[Students](#) [Scores](#) [Teams](#) [Backlog](#) [Sprints](#) [Strategic_Journey_Stages](#) [Lessons](#) [Tasks](#) [+ Create](#) [trash](#) [refresh](#) [SHARE](#) [grid](#)
 Grid view

 1 hidden field

 Filter

 Group

 Sort

 Color


...

	A Story	Notes	Link_to_Journey	Status
1	Understand Lean Method Approach	Students get familiar with lean method basic co...	Learn full stack MVC SCRUM	Research
2	MAMP MySQL Administration Setup	Students learn to setup MAMP MySQL admin t...	Learn full stack MVC SCRUM	Problem Set 1
3	Jupyter Notebook Setup	Students learn to set up Jupyter Notebook for t...	Cross-platform Development	
4	Differences between Web, PWA, Hybrid, and Native App.	Students can tell the major differences betwee...	Learn full stack MVC SCRUM	Cross-platform
5	PhoneGap Setup	Students learn to use PhoneGap desktop and P...	Learn full stack MVC SCRUM	Master REST
6	PythonAnywhere Setup	Students learn to setup a PythonAnywhere acc...	Master REST API	Learn full stack MVC SCR
7	Adobe PhoneGap Build account	Students get to open up an account at build.ph...	Master REST API	Learn full stack MVC SCR
8	MySQL and CRUD	Students master basic MySQL and CRUD to pr...	Learn full stack MVC SCRUM	
9	Full-stack MVC concepts	Students can distinguish front-end from back-...	Learn full stack MVC SCRUM	
10	Flask and Jinja2 render_template engine	Students learn to use the Flask and Jinja2 rend...	Master REST API	Cross-platform Developm
11	SCRUM terms	Understand the SCRUM method and the key te...	Learn full stack MVC SCRUM	
12	DT, Lean, and Agile for SCRUM	Students watch online videos and class lecture...	Learn full stack MVC SCRUM	
13	Product Requirement Plan	Students will use Elements of UX as a framewo...	Learn full stack MVC SCRUM	
14	Product Release Plan	Students will use the SCRUM method to define ...	Learn full stack MVC SCRUM	
15	Customer Discovery: From early adopters to crossing the chasm	Students will learn about the importance of ide...	Cross-platform Development	

com5940 2020

HELP ? 🔍 🔔 🔑

SHARE 🔍 🔑



Students Scores Teams Backlog Sprints Strategic_Journey_Stages Lessons Tasks 🔍 🔑

	A Release	Notes	Link_to_lesson	release date	+	
1	Problem Set 1	Jupyter Notebook database connection	Python Workshop Lesson 2 - Intro	17/2/2020		
2	Problem Set 2	Flask connection to Airtable	Lesson 3 - Agile Project Management	24/2/2020		
3	Problem Set 3	PhoneGap Prototype Zip (Airtable)	Lesson 4 - Mobile App Development	2/3/2020		
4	Problem Set 4	Flask connection to MySQL	Lesson 5 - Introduction to REST API,	9/3/2020		
5	Team Report 1	PRD (Product Requirement Document)...	Lesson 8 - Cloud Infrastructure for C	16/3/2020		
6	Problem Set 5	PhoneGap Prototype Zip (MySQL)	Lesson 6 - Integrate JS Front-end wi	23/3/2020		
7	Team Report 2	PRD (Product Requirement Document)...	Lesson 10 - Overview of Machine Le	6/4/2020		
8	Final Presentation	Product, Business Model and Growth ...	Lesson 11 - Regression and Classific	4/5/2020		
9	Final Report	Product, Business Model and Growth ...	Lesson 13 - Final Presentation Less	18/5/2020		

com5940 2020

HELP ? 🔔 🚙



Students Scores Teams

Backlog

Sprints

Strategic_Journey_Stages

Lessons

Tas



SHARE

BLOCKS

Grid view 1 hidden field

Filter

Group

Sort

Color



...



	A Name	B Description	C Category
1	Learn full stack MVC SCRUM	Provide students with an overview of full stack MVC application development.	Connecting Flask to MySQL database Product Release Plan Product Management
2	Cross-platform Development	Evaluate the pros and cons in cross-platform application development using various frameworks.	Different ways to submit data to the back-end Customer Discovery: Functional Requirements
3	Master REST API	Cover essential programming concepts for full stack REST API based MVC applications.	Different ways to authenticate and authorise API access. Connecting IOT to REST API
4	Cloud and ML based CRISP-DM Infrastructure	Understand the importance of choosing and building the right digital infrastructure.	CRISP-DM for data driven product development Type of machine learning models
5	3C based growth hacking	Perform growth hacking using the 2A3R model by integrating content personalization.	Build growth metrics to measure conversion progress Conversion funnel analysis

+

com5940 2020

HELP ?  



Students

Scores

Teams

Backlog

Sprints

Strategic_Journey_Stages

Lessons

Task



SHARE

BLOCKS

Grid view



2 hidden fields

Filter

Group

Sort

Color



...

Q

	A Name	Sprints	
1	Python Workshop	Problem Set 1	
2	Lesson 1 - Overview: Developing Entrepreneurial Mindset for Digital Innovation	Team Report 1	
3	Lesson 2 - Introduction to SQL and MySQL Database for Web Development	Problem Set 2 Problem Set 1	
4	Lesson 3 - Agile Project Management for Achieving Product-Market Fit	Team Report 2 Team Report 1 Problem Set 2	
5	Lesson 4 - Mobile App Development within a MVC Context	Problem Set 3	
6	Lesson 5 - Introduction to REST API, User Authentication, Authorization and Encr...	Problem Set 4	
7	Lesson 6 - Integrate JS Front-end with Flask RESTful API Back-end Services	Problem Set 5	
8	Lesson 7 - Business Goals and Conversion Funnel Design	Final Presentation Team Report 2	
	Lesson 8 - Cloud Infrastructure for Data Preparation, Analytics and Modelling	Final Report Final Presentation Team Report 2 Team Report 1	
10	Lesson 9 - Data Preparation and Exploratory Data Analysis	Final Report Final Presentation Team Report 2	
11	Lesson 10 - Overview of Machine Learning and Predictive Analytics	Final Report Final Presentation Team Report 2	
12	Lesson 11 - Regression and Classification Models in Supervised Learning	Final Report Final Presentation	
13	Lesson 12 - Performance Metrics in Google Analytics and Dashboard Display	Final Report Final Presentation	
14	Lesson 13 - Final Presentation	Final Report	
+			

THANK YOU FOR YOUR TIME!