## SQL

## SQLite

SQL server pyodbc

SQLite sqlite3

SQLalchemy

oracle cx\_oracle

mysql cx\_oracle

steps:

1. establish connection
2. create a cursor object
3. send receive data (queries)

cur.execute()

cur.executemany()

cur.fetchone()

cur.fetchall()

## json

javascript object notation

json

json.load

json.dump

python JSON

dict object

list, tuple, set… array

str string

int, float number

True/False true/false

None null

## serialisation:

pickle

load, dump

## virtual env:

virtualenv

sudo apt install python-pip

sudo pip install virtualenv

virtualenv env

activate & deactivate virtual env:

being in the same directory, run: (linux & mac)

source env/bin/activate

deactivate

being in the same directory, run: (windows)

env\Scripts\activate

## flask:

being in the virtual env:

pip install flask

created

\_\_init\_\_.py

routes.py

virtual.py

export FLASK\_APP=virtual.py

flask run

…

…

<http://localhost:5000/>

to end the server

ctrl+C

first set of codes:

**routes.py:**

from app import app

@app.route('/')

@app.route('/index')

# view function

def index():

return "let the force be with you"

**\_\_init\_\_.py**

from flask import Flask

app = Flask(\_\_name\_\_)

from app import routes

**virtual.py**

from app import app

### jinja

templating language for python

looks similar to django’s templates

{{ }} expressions (output)

{% %} statements

{# #} comments

# line statements

flask-iniconfig