上海外国语大学
SHANGHAI INTERNATIONAL STUDIES UNIVERSITY

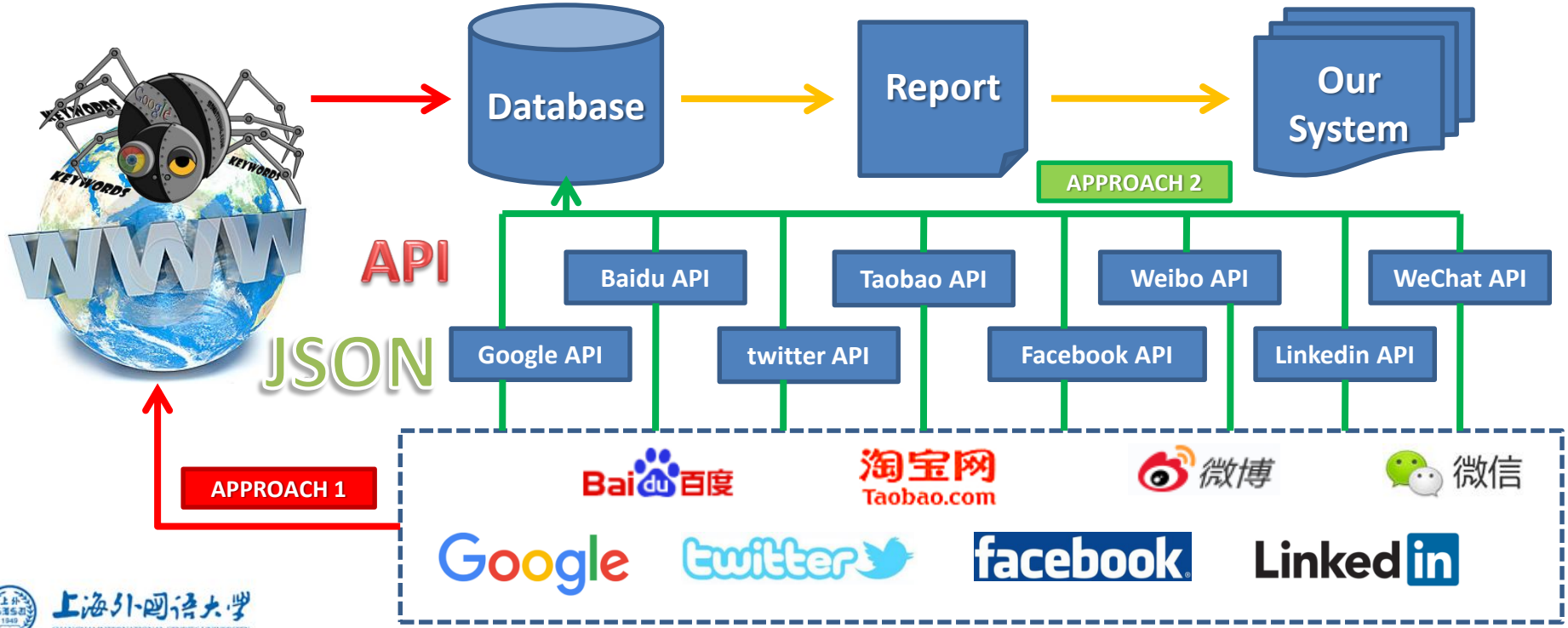
New Media
Data Analytics and Application

Lecture 6: Data Collection via Web API

Ting Wang

Objectives

How to get data from social media?



Outlines

- Web API
- JSON
- Examples of Web API Applications

上海外国语大学
SHANGHAI INTERNATIONAL STUDIES UNIVERSITY

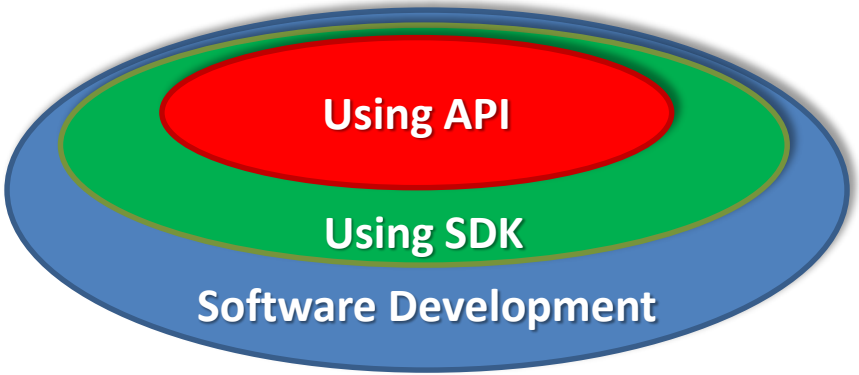
an access to the web site data

Web API

SDK (Software Development Kit)

软件开发工具包

a set of
- software development tools,
- source code,
- instructions,
- examples,
- and …



Web API

How to print a booklet for our sport games?

具体的

2

Using SDK
在人家的套路基础上走

1

Traditional Software Development
自己设计套路走

抽象的

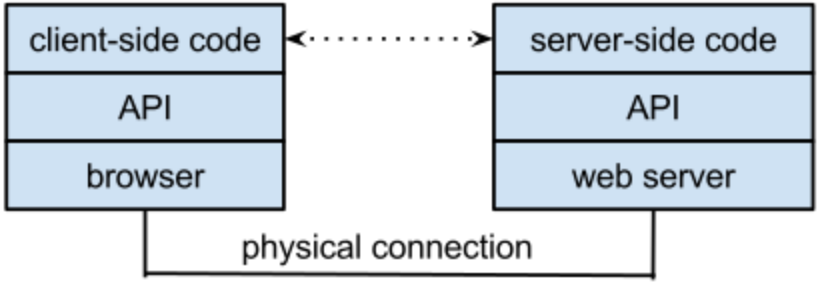
3

Using Web API
自己不动脑子，
完全按人家的套路走

API (Application Programming Interface)

应用程序接口

a set of
subroutine definitions,
protocols,
and tools

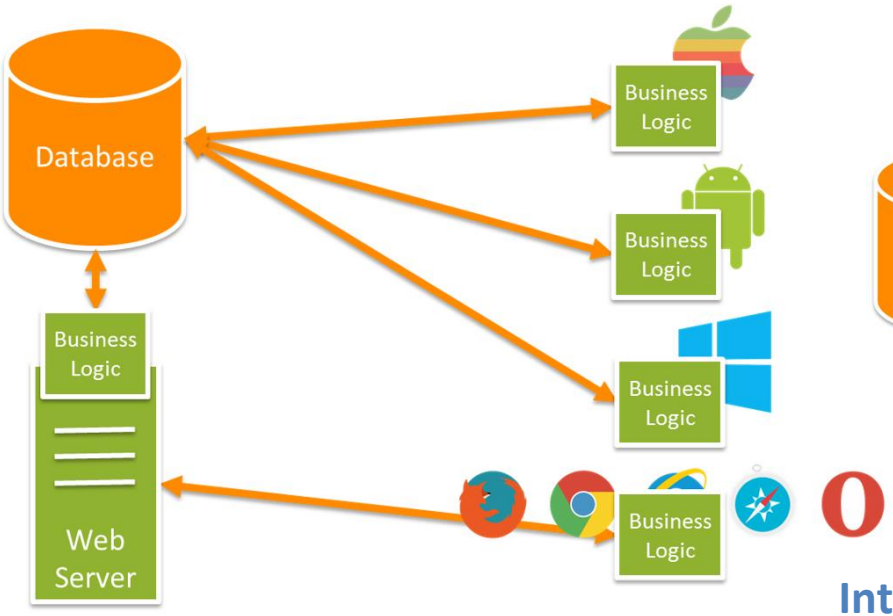


- What is Web API?

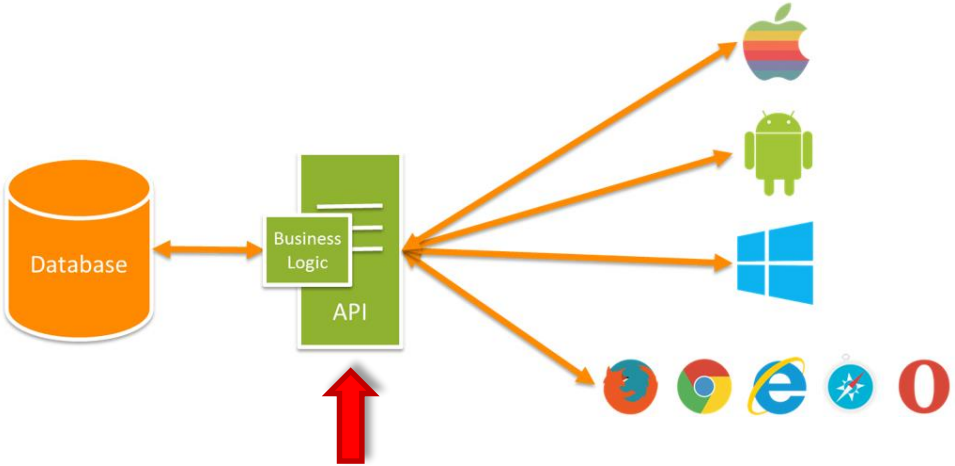
BS/CS

Web API

Short Short Ago



Now



ALL via ONE!!

Integrating all the operations into one package, including “insert”, “delete”, “update”, and “select”.

In Unified Style, two crucial elements:

– Unified Method :

- HTTP (HyperText Transfer Protocol)

超文本传输协议

– Unified Language :

- JSON
- XML

HTTP(HyperText Transfer Protocol)

超文本传输协议

an application protocol for distributed, collaborative, hypermedia information systems, which is the foundation of data communication for the World Wide Web.

- GET
  - Give Me Information
- POST
  - Give Server Information

Do you remember "Login.py" ?

Ref. http://www.w3school.com.cn/tags/html_ref_httpmethods.asp

Web API

```python
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

from flask import Flask
from flask import request
#import pymysql
import pymssql
#import sqlite3

app = Flask(__name__)

@app.route('/', methods=['GET', 'POST'])
def home():
    return '''<h1>Home</h1>
              <p><a href="/login">Login</a></p>'''


@app.route('/login', methods=['GET'])
def login_form():
    return '''<form action="/login" method="POST">
              <p><input name="username"></p>
              <p><input name="password" type="password"></p>
              <p><button type="submit">Log In Now</button></p>
              </form>'''
```

```python
@app.route('/login', methods=['POST'])
def login():
    # get data from database:
    # 1.Connection Open
    conn = pymssql.connect(user='sa', password='123456', database='login')
    # 2.Cursor Creating:
    cursor = conn.cursor()
    # 3.SQL Execution
    sqlstr = 'select * from customer where customer_name=\''+request.form['username']+\
             '\' and customer_pwd=\''+request.form['password']+'\''
    # 4.Cursor Moving
    cursor.execute(sqlstr)
    numrows = len(cursor.fetchall())
    conn.commit()
    # 5.Connection Close
    cursor.close()
    conn.close()
    # get data from request:
    if numrows>0:
        return '<h3>Hello, '+request.form['username']+'</h3>'
    return '<h3>Bad username or password.</h3>'

if __name__ == '__main__':
    app.run()
```

Login.py

Language of Web API

– XML (EXtensible Markup Language)

```
<?xml version="1.0" encoding="UTF-8" ?>
        <employees>
                <firstName>Bill</firstName>
                <lastName>Gates</lastName>
        </employees>
        <employees>
                <firstName>George</firstName>
                <lastName>Bush</lastName>
        </employees>
        <employees>
                <firstName>Thomas</firstName>
                <lastName>Carter</lastName>
        </employees>
```

Ref. http://www.w3school.com.cn/x.asp

Language of Web API

– JSON (JavaScript Object Notation)

```
{ "employees": [
{ "firstName":"Bill" , "lastName":"Gates" },
{ "firstName":"George" , "lastName":"Bush" },
{ "firstName":"Thomas" , "lastName":"Carter" }
] }
```

Ref. http://www.w3school.com.cn/json/

Transfer between XML and JSON

http://www.bejson.com/xml2json/

在线XML、JSON数据互转

如题

XML

```
<?xml version="1.0" encoding="UTF-8" ?>
    <employees>
        <firstName>Bill</firstName>
        <lastName>Gates</lastName>
    </employees>
    <employees>
        <firstName>George</firstName>
        <lastName>Bush</lastName>
    </employees>
    <employees>
        <firstName>Thomas</firstName>
        <lastName>Carter</lastName>
    </employees>
```

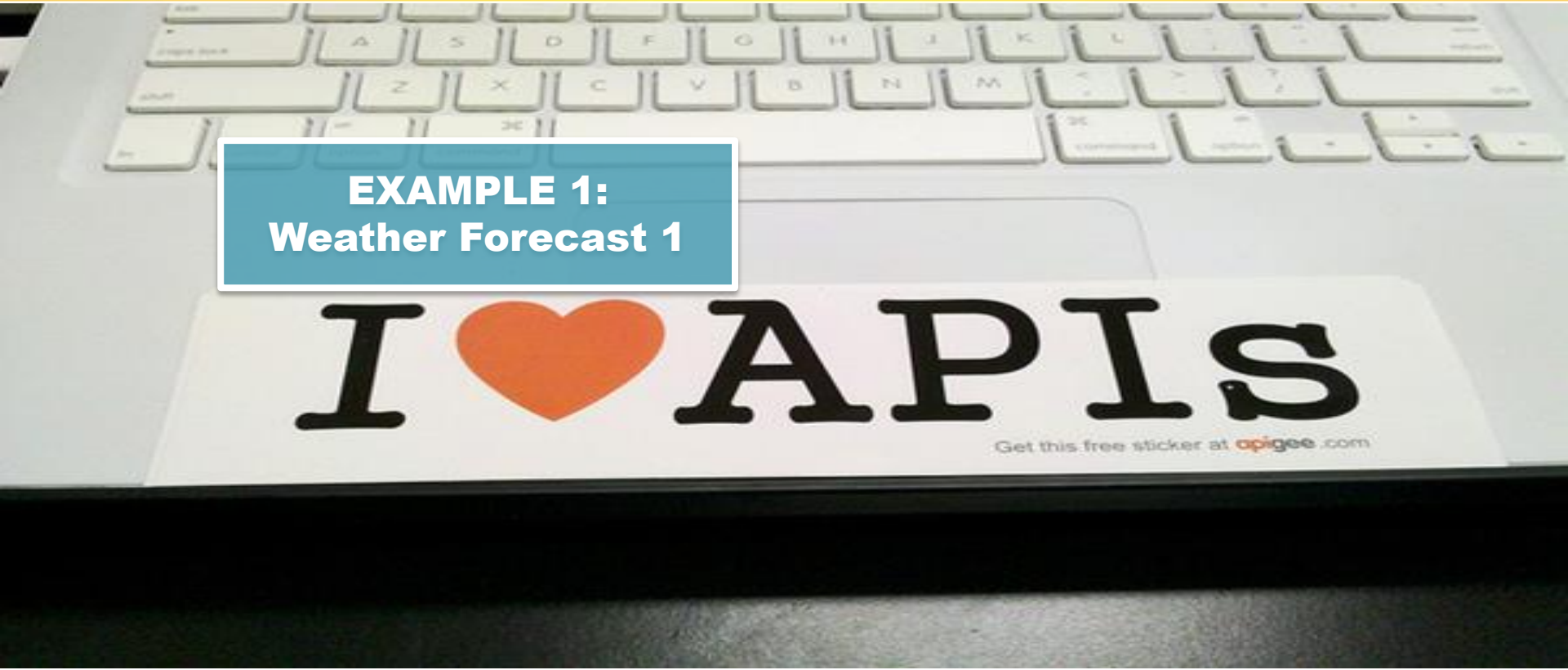
转义编码

清空

JSON

```
{ "employees": [ { "firstName":"Bill" , "lastName":"Gates" }, {
"firstName":"George" , "lastName":"Bush" },{ "firstName":"Thomas" ,
"lastName":"Carter" } ] }
```

Why JSON is more and more popular?

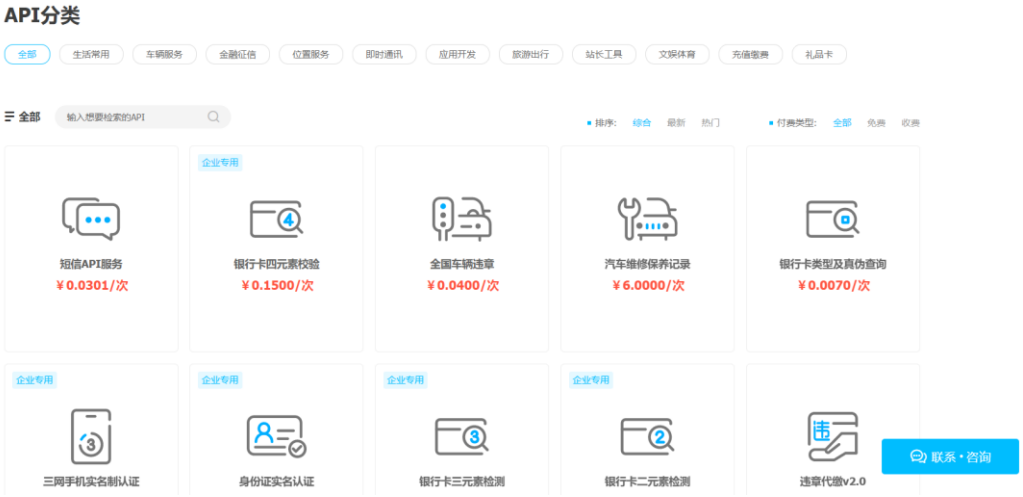


EXAMPLE 1:
Weather Forecast 1

Web API

www.juhe.cn

聚合数据



Juhe is a technology company that offers mobile data services and products in China.

上海外国语大学
SHANGHAI INTERNATIONAL STUDIES UNIVERSITY

Step 1: you should have your own account

- Log in the web: https://www.juhe.cn/register
- You may register your account by cellphone or email.
- You may log in the website after you register it.

Step 2: select an API

- You may select a FREE API for test.
- Do not forget to confirm your information in two months.

Web API

Submit your Information

1 实名认证 — 2 应用待认证 — 3 应用认证中 — 4 应用完成认证

用户名: tingwang1981

聚合名称: 12306火车票查询

APPKEY: [redacted]

创建时间: 2018-04-16 16:26:39

* 应用平台: 其他

* 应用名称: APP名称/网站名称/微信公众号或订阅号名称

应用地址:

应用描述: 简述应用的基本功能或微信公众号，不超过128个字

提交认证

Step 3: apply the new data API

- You may start to use your API after you confirm your information.
- Save the AppKey

Web API

- You may visit the main page of your API to get more detailed information.

- If the API is not free, you should pay for it.

Step 4: write the code

- Examples are important.
- Sometimes, there is no Python examples.

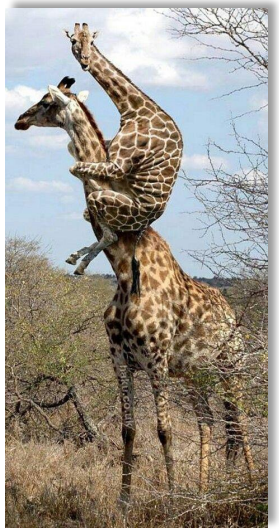
How to write python code,

if we cannot find it in examples?

To look for a similar one!

Web API

- Here is an example which has a code in Python.
  https://www.juhe.cn/docs/api/id/11



Standing on the Shoulders of Giants (SSG)

不要什么都去自己写，要thinking in SSG！

Web API

- Thinking in SSG!

① Analyze Code: https://code.juhe.cn/docs/1018

② Rewrite Code: based on both python examples from other APIs and the codes in your API.

③ Test Code

Some Codes are in Python 2, which should be rewritten with Python 3.

Project

Lecture 6 D:\SISU\Courses\2018新媒体数据分析与应用 (20

- API.py
- CityWeather.py
- CityWeather2.py
- CityWeatherAPI.py

External Libraries

CityWeatherAPI.py

```python
# 全国天气预报调用示例代码 — 聚合数据
# 在线接口文档：https://www.juhe.cn/docs/api/id/39
# ----------------------------------------

def main():
    # 配置您申请的APPKey
    appkey = "[redacted]"

    # 1.城市天气查询
    request1(appkey, "GET")


# 城市天气查询
def request1(appkey, m="GET"):
    url = "http://v.juhe.cn/weather/index"
    params = {
        "cityname": "苏州",  # 需要查询的城市
        "key": appkey,  # 应用APPKEY(应用详细页查询)
        "dtype": "",  # 返回数据的格式,xml或json，默认json
        "format": "",  # //未来6天预报(future)两种返回格式，1或2，默认1
    }
    params = urlencode(params)
    if m == "GET":
```

main()

Run CityWeatherAPI

'wind': '东南风3-5级', 'week': '星期六', 'date': '20180421'}, 'day_20180422': {'temperature': '9℃~21℃', 'weather': '晴', 'weather_id': {'fa': '00', 'fb': '00'}, 'wind': '东南风3-5级', 'week': '星期日', 'date': '20180422'}}}

Web API

- The Code

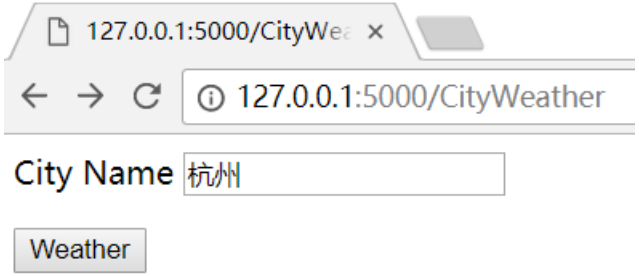
```python
# -*- coding: utf-8 -*-
import json, urllib.request
from urllib.parse import urlencode


# ----------------------------------
# 全国天气预报调用示例代码 — 聚合数据
# 在线接口文档：https://www.juhe.cn/docs/api/id/39
# ----------------------------------

def main():
    # 配置您申请的APPKey
    appkey = “Your APPKEY”

    # 1.城市天气查询
    request1(appkey, "GET")
```

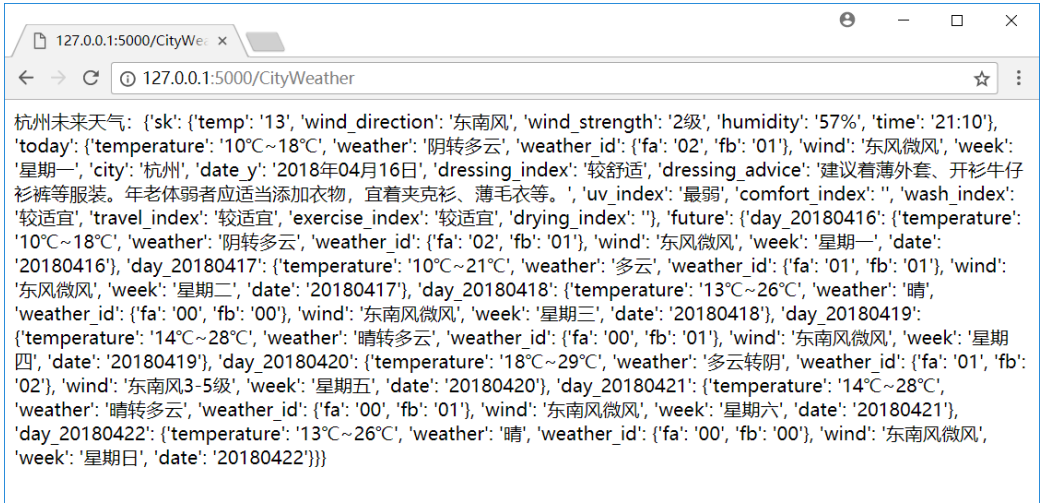
```python
# 城市天气查询
def request1(appkey, m="GET"):
    url = "http://v.juhe.cn/weather/index"
    params = {
        "cityname": "苏州",  # 需要查询的城市
        "key": appkey,  # 应用APPKEY(应用详细页查询)
        "dtype": "",  # 返回数据的格式,xml或json，默认json
        "format": "",  # //未来6天预报(future)两种返回格式，1或2，
默认1
    }
    params = urlencode(params)
    if m == "GET":
        f = urllib.request.urlopen("%s?%s" % (url, params))
    else:
        f = urllib.request.urlopen(url, params)

    content = f.read()
    res = json.loads(content)
    if res:
        error_code = res["error_code"]
        if error_code == 0:
            # 成功请求
            print(res["result"])

        else:
            print("%s:%s" % (res["error_code"], res["reason"]))

    else:
        print("request api error")


if __name__ == '__main__':
    main()
```

Step 5: embed your API code to your system

- Question:
  Flask can provide us a web interface. Can we embed the code to Flask website?

Results：





杭州未来天气：{'sk': {'temp': '13', 'wind_direction': '东南风', 'wind_strength': '2级', 'humidity': '57%', 'time': '21:10'}, 'today': {'temperature': '10℃~18℃', 'weather': '阴转多云', 'weather_id': {'fa': '02', 'fb': '01'}, 'wind': '东风微风', 'week': '星期一', 'city': '杭州', 'date_y': '2018年04月16日', 'dressing_index': '较舒适', 'dressing_advice': '建议着薄外套、开衫牛仔衫裤等服装。年老体弱者应适当添加衣物，宜着夹克衫、薄毛衣等。', 'uv_index': '最弱', 'comfort_index': '', 'wash_index': '较适宜', 'travel_index': '较适宜', 'exercise_index': '较适宜', 'drying_index': ''}, 'future': {'day_20180416': {'temperature': '10℃~18℃', 'weather': '阴转多云', 'weather_id': {'fa': '02', 'fb': '01'}, 'wind': '东风微风', 'week': '星期一', 'date': '20180416'}, 'day_20180417': {'temperature': '10℃~21℃', 'weather': '多云', 'weather_id': {'fa': '01', 'fb': '01'}, 'wind': '东风微风', 'week': '星期二', 'date': '20180417'}, 'day_20180418': {'temperature': '13℃~26℃', 'weather': '晴', 'weather_id': {'fa': '00', 'fb': '00'}, 'wind': '东南风微风', 'week': '星期三', 'date': '20180418'}, 'day_20180419': {'temperature': '14℃~28℃', 'weather': '晴转多云', 'weather_id': {'fa': '00', 'fb': '01'}, 'wind': '东南风微风', 'week': '星期四', 'date': '20180419'}, 'day_20180420': {'temperature': '18℃~29℃', 'weather': '多云转阴', 'weather_id': {'fa': '01', 'fb': '02'}, 'wind': '东南风3-5级', 'week': '星期五', 'date': '20180420'}, 'day_20180421': {'temperature': '14℃~28℃', 'weather': '晴转多云', 'weather_id': {'fa': '00', 'fb': '01'}, 'wind': '东南风微风', 'week': '星期六', 'date': '20180421'}, 'day_20180422': {'temperature': '13℃~26℃', 'weather': '晴', 'weather_id': {'fa': '00', 'fb': '00'}, 'wind': '东南风微风', 'week': '星期日', 'date': '20180422'}}}

Web API

```python
from flask import Flask
from flask import request
import json, urllib.request
from urllib.parse import urlencode

app = Flask(__name__)

@app.route('/', methods=['GET', 'POST'])
def home():
    return '''<h1>Home</h1>
            <p><a href="/CityWeather">Weather Forecast</a></p>'''

@app.route('/CityWeather', methods=['GET'])
def login_form():
    return '''<form action="/CityWeather" method="POST">
              <p>City Name <input name="CityName"></p>
              <p><button type="submit">Weather</button></p>
              </form>'''

@app.route('/CityWeather', methods=['POST'])
def login():
    # 配置您申请的APPKey
    appkey = " Your APPKEY "

    CityName = request.form["CityName"]

    # 1.城市天气查询
    return CityName+"未来天气："+request1(CityName, appkey, "GET")
```

<-Login

<-Main()

```python
# 城市天气查询
def request1(CityName, appkey, m="GET"):
    url = "http://v.juhe.cn/weather/index"
    params = {
        "cityname": CityName,  # 需要查询的城市
        "key": appkey,  # 应用APPKEY(应用详细页查询)
        "dtype": "",  # 返回数据的格式,xml或json，默认json
        "format": "",  # //未来6天预报(future)两种返回格式，1或2，默认1
    }
    params = urlencode(params)
    if m == "GET":
        f = urllib.request.urlopen("%s?%s" % (url, params))
    else:
        f = urllib.request.urlopen(url, params)

    content = f.read()
    res = json.loads(content)
    if res:
        error_code = res["error_code"]
        if error_code == 0:
            # 成功请求
            return str(res["result"])

        else:
            return str(res["error_code"])+":"+ str(res["reason"])

    else:
        return "request api error"

if __name__ == '__main__':
    app.run()
```

<-almost the same

Web API

Tips:

1. Register, Login, and get your **API KEY** from each website before you use API.

2. Different websites have different API Keys.

3. Copy and Paste, **Standing on the Shoulders of Giants**

Web API

*What are the **advantages** and **disadvantages** of using API?*

- *Advantages:*
  - *much easier to get useful information*
  - *more efficient to develop website platforms*
  - *more helps from professionals*
- *Disadvantages:*
  - *maybe not free*
  - *maybe limitations (cannot use it sometime in the future)*
  - *maybe some mistakes or exceptions*

human-readable text for data transmission

JSON

Analyzing JSON

1. import json
2. json.loads
3. json.dumps
4. get data following the tree
5. output results

JSON结构视图

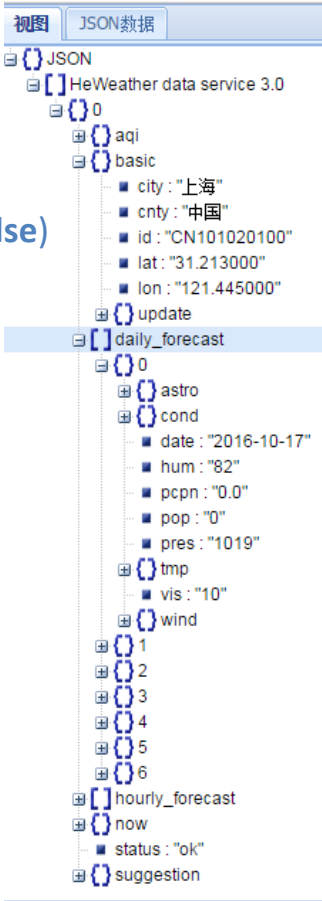
http://www.bejson.com/jsonviewernew/

```
city = json.dumps(response_json['HeWeather data service 3.0'][0]['basic'].get("city"),ensure_ascii=False)
```

有中文

```
date = json.dumps(response_json['HeWeather data service 3.0'][0]['daily_forecast'][0].get("date"))
```

无中文



JSON

- The result of weather forecast:

{'sk': {'temp': '13', 'wind_direction': '东南风', 'wind_strength': '2级', 'humidity': '57%', 'time': '21:10'}, 'today': {'temperature': '10℃~18℃', 'weather': '阴转多云', 'weather_id': {'fa': '02', 'fb': '01'}, 'wind': '东风微风', 'week': '星期一', 'city': '杭州', 'date_y': '2018年04月16日', 'dressing_index': '较舒适', 'dressing_advice': '建议着薄外套、开衫牛仔衫裤等服装。年老体弱者应适当添加衣物，宜着夹克衫、薄毛衣等。', 'uv_index': '最弱', 'comfort_index': '', 'wash_index': '较适宜', 'travel_index': '较适宜', 'exercise_index': '较适宜', 'drying_index': ''}, 'future': {'day_20180416': {'temperature': '10℃~18℃', 'weather': '阴转多云', 'weather_id': {'fa': '02', 'fb': '01'}, 'wind': '东风微风', 'week': '星期一', 'date': '20180416'}, 'day_20180417': {'temperature': '10℃~21℃', 'weather': '多云', 'weather_id': {'fa': '01', 'fb': '01'}, 'wind': '东风微风', 'week': '星期二', 'date': '20180417'}, 'day_20180418': {'temperature': '13℃~26℃', 'weather': '晴', 'weather_id': {'fa': '00', 'fb': '00'}, 'wind': '东南风微风', 'week': '星期三', 'date': '20180418'}, 'day_20180419': {'temperature': '14℃~28℃', 'weather': '晴转多云', 'weather_id': {'fa': '00', 'fb': '01'}, 'wind': '东南风微风', 'week': '星期四', 'date': '20180419'}, 'day_20180420': {'temperature': '18℃~29℃', 'weather': '多云转阴', 'weather_id': {'fa': '01', 'fb': '02'}, 'wind': '东南风3-5级', 'week': '星期五', 'date': '20180420'}, 'day_20180421': {'temperature': '14℃~28℃', 'weather': '晴转多云', 'weather_id': {'fa': '00', 'fb': '01'}, 'wind': '东南风微风', 'week': '星期六', 'date': '20180421'}, 'day_20180422': {'temperature': '13℃~26℃', 'weather': '晴', 'weather_id': {'fa': '00', 'fb': '00'}, 'wind': '东南风微风', 'week': '星期日', 'date': '20180422'}}}

JSON

视图 JSON数据

```
JSON
  sk
    temp : "13"
    wind_direction : "东南风"
    wind_strength : "2级"
    humidity : "57%"
    time : "21:10"
  today
    temperature : "10℃~18℃"
    weather : "阴转多云"
    weather_id
    wind : "东风微风"
    week : "星期一"
    city : "杭州"
    date_y : "2018年04月16日"
    dressing_index : "较舒适"
    dressing_advice : "建议着薄外套、开衫牛仔衫裤等服装。年老体弱者应适当添加衣物，宜着夹克衫、薄毛衣等。"
    uv_index : "最弱"
    comfort_index : ""
    wash_index : "较适宜"
    travel_index : "较适宜"
    exercise_index : "较适宜"
    drying_index : ""
  future
    day_20180416
    day_20180417
    day_20180418
    day_20180419
    day_20180420
    day_20180421
    day_20180422
```

Name	Value
future	...
sk	...
today	...

JSON

```python
def login():
    # 配置您申请的APPKey
    appkey = "bb198cded47261b9db35abc76f5f53d5"

    CityName = request.form["CityName"]

    # 1.城市天气查询
    response = request1(CityName, appkey, "GET").replace("'", '"')
    response_json = json.loads(response)
    CurrentTemp = json.dumps(response_json['sk'].get("temp"))
    CurrentWindDirection = json.dumps(response_json['sk'].get("wind_direction"), ensure_ascii=False)
    CurrentWindStrength = json.dumps(response_json['sk'].get("wind_strength"), ensure_ascii=False)
    CurrentHumidity = json.dumps(response_json['sk'].get("humidity"))
    CurrentTime = json.dumps(response_json['sk'].get("time"))
    return "城市："+CityName+"<p>当前天气：<br><li>气温："+CurrentTemp[1:-1]+"<li>风向："
           +CurrentWindDirection[1:-1]+"<li>风力："+CurrentWindStrength[1:-1]+"<li>湿度："+CurrentHumidity[1:-1]+"<li>时间："+CurrentTime[1:-1]
```

Question： Can you return the weather of today?

JSON

LIVE
DOPPLER
7 HD
7 DAY FORECAST
MIKE'S OUTLOOK
EXAMPLE 2:
Weather Forecast 2
MON
TUE
WED
THU
COOLER BREEZY
A LITTLE MILDER
SPRING WARMTH
MILD TO WARM
PARTLY CLOUDY
CHANCE LIGHT RAIN
SUN & CLOUDS
INLAND
72
74
76
78
68
66
68
BAY
68
70
72
74
66
64
66
COAST
58
60
62
64
60
58
60
POWERED BY AccuWeather

Example

127.0.0.1:5000/CityWeather

城市：**北京**　2018年04月16日　星期一　11℃~23℃　晴转多云　南风微风

今日提醒：

- 穿衣：较舒适　建议：建议着薄外套、开衫牛仔衫裤等服装。年老体弱者应适当添加衣物，宜着夹克衫、薄毛衣等。
- 紫外线：中等　外出：较适宜　洗衣：较适宜　锻炼：较适宜

当前天气：

- 气温：17
- 风向：西南风
- 风力：3级
- 湿度：47%
- 时间：23:38

上海外国语大学
SHANGHAI INTERNATIONAL STUDIES UNIVERSITY

some examples on social media

Web API Applications

OAuth

An open protocol to allow secure API authorization in a simple and standard method from web, mobile and desktop applications.

```
+--------+                               +---------------+
|        |--(A)- Authorization Request ->|   Resource    |
|        |                               |     Owner     |
|        |<-(B)-- Authorization Grant ---|               |
|        |                               +---------------+
|        |
|        |                               +---------------+
|        |--(C)-- Authorization Grant -->| Authorization |
| Client |                               |     Server    |
|        |<-(D)----- Access Token -------|               |
|        |                               +---------------+
|        |
|        |                               +---------------+
|        |--(E)----- Access Token ------>|    Resource   |
|        |                               |     Server    |
|        |<-(F)--- Protected Resource ---|               |
+--------+                               +---------------+
```

API Key needed

Access Token needed

Ref. http://www.ruanyifeng.com/blog/2014/05/oauth_2_0.html

Web API Applications

EXAMPLE 3:
Friendship of TVs on Twitter

Problem Description

This research aims to investigate the significance and activeness of TV media companies in the world. Twitter friendships between TV broadcasting companies were employed for this research. Totally, about 104 TV channels from different countries are selected for this research. In-Degree and Out-Degree are used as features in this study.

Features:

In-Degree: Followed by Others, which shows the significance to the world

Out-Degree: Following Others, which shows the activeness in the world

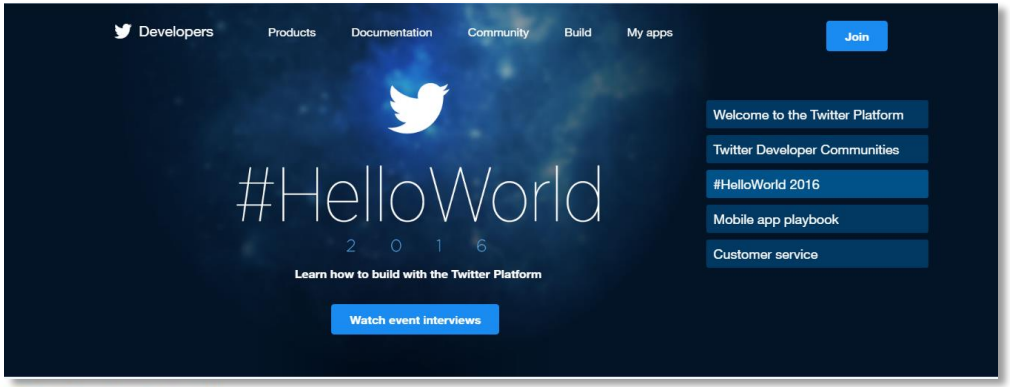
twitter and Its API

twitter

Do NOT tell me you do NOT know twitter, even you are a Chinese.

If you really do not know it, visit https://twitter.com right now!

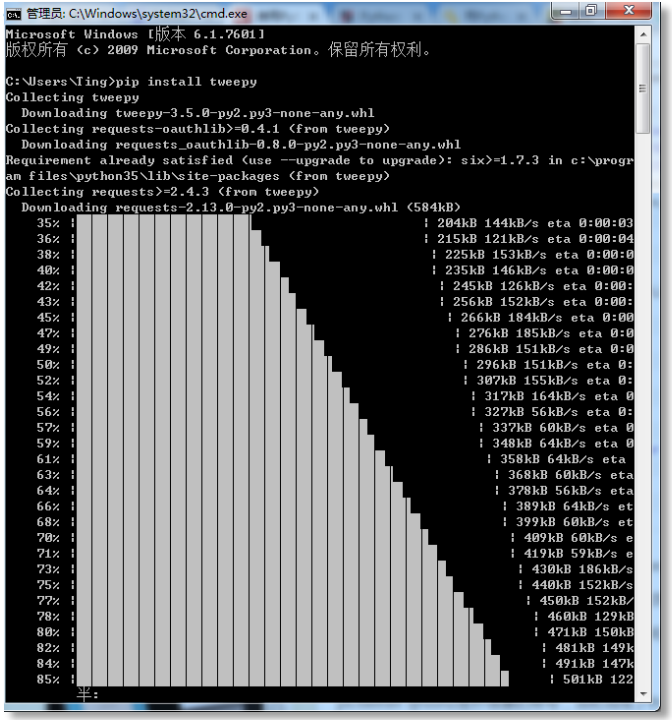
twitter API: https://dev.twitter.com/



Friendship of TVs on Twitter

Tweepy

An easy-to-use Python library for accessing the Twitter API



Installation

pip install tweepy

Reference to Tweepy

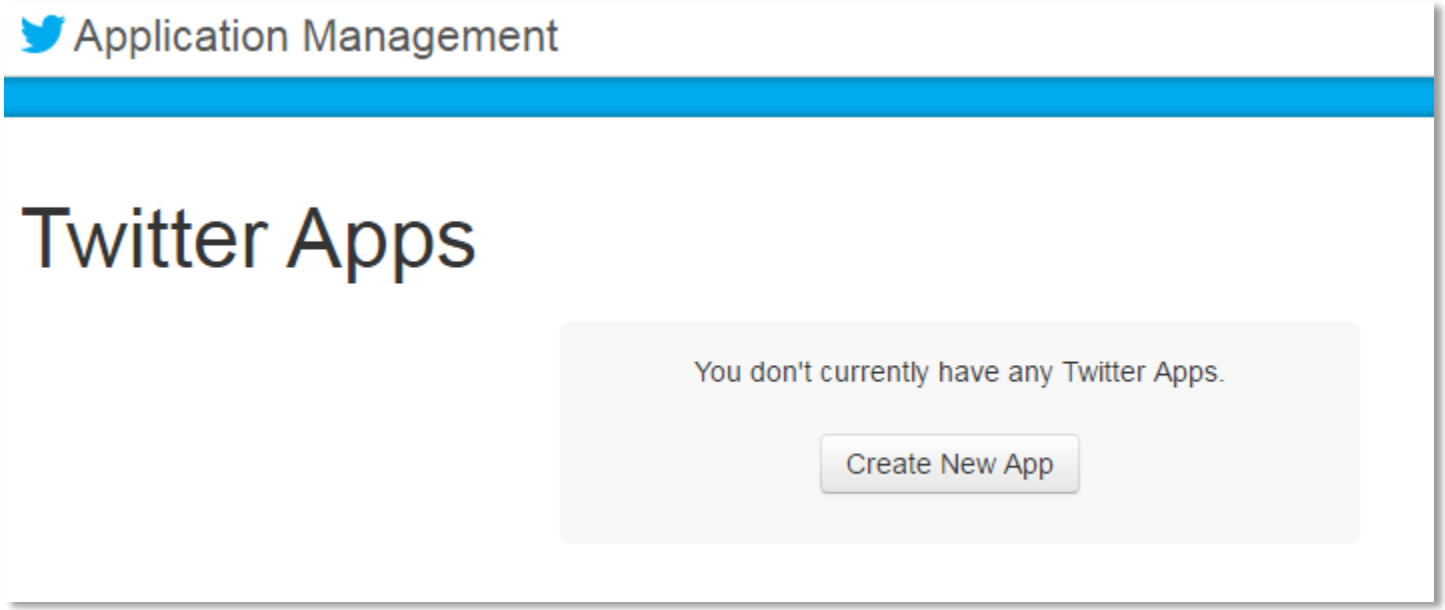
https://github.com/tweepy/tweepy

http://www.tweepy.org/

http://pythonhosted.org/tweepy/

Register Your Twitter APPs

Step 1 : visit **https://apps.twitter.com/,** and click on the button "**Create New APP**".



Friendship of TVs on Twitter

Step2 : Input the detailed information of your new app.

Create an application

Application Details

Name *

Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.

Description *

Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.

Website *

Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens.
(If you don't have a URL yet, just put a placeholder here but remember to change it later.)

Friendship of TVs on Twitter

Step3: Get your API keys and Access Tokens

[Notes]

You may get all these information when you click on "**Keys and Access Tokens**".

Details | Settings | Keys and Access Tokens | Permissions

https://twitter.com/tingandwang

Organization

Information about the organization or company associated with your a

Organization	None
Organization website	None

Application Settings

Your application's Consumer Key and Secret are used to authenticate

Access level	Read and write (modify app permissions)
Consumer Key (API Key)	qcQYzdOBWYMArsRHp3vTbogC tokens)
Callback URL	None
Callback URL Locked	No
Sign in with Twitter	Yes
App-only authentication	https://api.twitter.com/oauth2/token
Request token URL	https://api.twitter.com/oauth/request_token
Authorize URL	https://api.twitter.com/oauth/authorize

Friendship of TVs on Twitter

Steps for the System

1. Connect to twitter by API Keys and Access Tokens
2. Find the relationship using Tweepy functions

`API.show_friendship`(*source_id/source_screen_name, target_id/target_screen_name*)

Returns detailed information about the relationship between two users.

Parameters:

- **source_id** – The user_id of the subject user.
- **source_screen_name** – The screen_name of the subject user.
- **target_id** – The user_id of the target user.
- **target_screen_name** – The screen_name of the target user.

Return type: `Friendship` object

3. Save the results

Friendship of TVs on Twitter

Results

In-Degree for the significance of TV media

In-Degree Ranking

	Country	TV Media	Twitter ID	In-Degree
1	USA	CNN	CNN	26
2	Russia	Russia Today	RT_com	14
3	UK	BBC	BBCNews	13
4	Qatar	Al Jazeera television	AJENews	6
5	Spain	Radio Televisión Española	rtve	5
6	Netherland	Nederlandse Omroep Stichting	NOS	2
7	South Africa	South African Broadcasting Corporation, SABC	SABCNewsOnline	2
8	Lebanon	Al-Manar TV	almanarnews	2

Friendship of TVs on Twitter

Out-Degree for the activeness of TV media

Out-Degree Ranking

	Country	TV Media	Twitter ID	Out-Degree
1	Ireland	Raidió Teilifís Éireann, Ireland's National Public Service Multi-Media Organisation	rte	17
2	Ecuador	Ecuavisa	ecuavisa	6
3	Kenya	Kenya Broadcasting Corporation	KBCChannel1	5
4	Namibia	Namibian Broadcasting Corporation	NBCNamibia	4
5	Russia	Russia Today	RT_com	4
6	Bolivia	Bolivia TV	Canal_BoliviaTV	3
7	Singapore	Television Corporation of Singapore	ChannelNewsAsia	3
8	Fiji	Fiji Television Limited	FijiOneNews	3
9	Guyana	National Communications Network, Guyana	NCNGuyana	3
10	Tonga	Tonga Broadcasting Commission	TBCa3z	3
11	Afghanistan	TOLONews	TOLONews	3
12	Macedonia	Macedonian Radio-Television	mrt1web	3

Friendship of TVs on Twitter

Conclusions

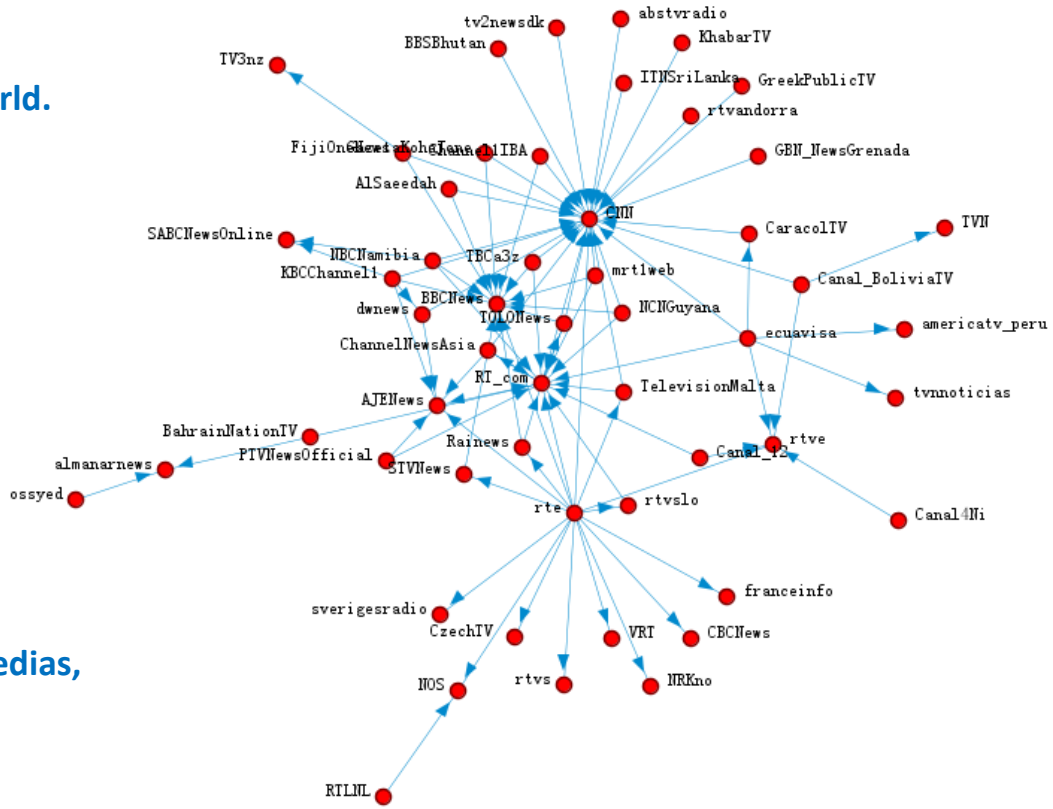
1. CNN, BBCNews, and Russia Today have the greatest significance to the world.

2. Raidió Teilifís Éireann from Ireland is the most active TV media.

3. Media from developed countries have greater influence.

4. Media from developing countries are more active.

5. CCTV, China neither follow any other medias, nor be followed by any other medias.



Reference

Reference

- XML
  - http://www.w3school.com.cn/x.asp
- JSON
  - http://www.w3school.com.cn/json

Reference

Baidu (http://apistore.baidu.com/)
Weibo (http://open.weibo.com/wiki/微博API)
WeChat (http://dev.wechat.com/)
QQ Tencent (http://wiki.open.qq.com/wiki/API文档)
360 (http://open.360.cn/)
Taobao (http://open.taobao.com/docs/api_list.htm?spm=a219a.7386653.1.22.3XIaLs)
JD (http://jos.jd.com/)
Suning (http://open.suning.com/)
Douban (https://developers.douban.com/)
Juhe (https://www.juhe.cn/)
Google (https://developers.google.com/)
Facebook (http://developers.facebook.com/)
Twitter (https://dev.twitter.com/)
Linkedin (https://developer.linkedin.com/)

Homework

Homework

1. Weather Forecast

- Print the weather for the whole week, including today and the next six days.

This homework should be finished before May 8.
The source code should also be sent to my email box at the same time.
Onsite homework checking will be carried out on May 8.

Homework

2. Get Data for your group work by APIs

- Search APIs from the Internet, and download useful data for your group work by them.
- Description about how to employ APIs in your system should be written in your system design documents.

The End of Lecture 6

Thank You

http://www.wangting.ac.cn