

EXERCISE 9

DATA MANAGEMENT

Group 11

Alexandre Ducommun
Rabeb Ben Ramdhane
Abdallah Shukor

Professor

Iulian Ciorascu

02 December 2019

Question 1

Now the script `getregions.py` retrieve the historical data for France and Switzerland in a dictionary. During the web scraping task, we get data from the table tag of each region. The hidden tag `<tr>` for historical data are available when we parse the html tree with BeautifulSoup. The function of `getregions` return a dictionary of regions where the key is the city and the value is a list of attribute (country, region, city,

Question 2

For this task, we adapt the `download.py` script to create a csv file (`regions-metadata.csv`) if it doesn't already exist with the attribute country, region, city, date, file path and link. When a region is downloaded, a new line is added to this file (if it doesn't exist). In the script `checkforupdate.py`, the main function `verify()` get all the data using `getregions.py` and the data from `regions-metadata.csv`. If there is no match between the two set of data, the function `send_email` is called with a set of new region as argument. The list of regions passed in argument is added to the body of the message. Then the mail is sent from a gmail account created for this exercise.

```
def verify():
    regions = getregions.get_regions('http://insideairbnb.com/get-the-data.html')
    new = False
    regions_news = []
    with open('data/regions-metadata.csv', 'r', encoding='UTF8', newline='') as f:
        csvreader = csv.reader(f, delimiter=',')
        for row in csvreader:
            for r in regions:
                region = regions[r]
                for date, link in region[3].items():
                    filepath = 'data/' + region[0] + '/' + region[1] + '/' + region[2] + '/' + date + '/listings.csv'
                    if not filepath == row[3]:
                        new = True
                        regions_news.append(region[2])
    if new:
        send_email(regions_news)
```

The list of regions passed in argument is added to the body of the message. Then the mail is sent from a gmail account created for this exercise.

```
def send_email(regions):
    message = MIMEMultipart()
    message["from"] = "Data Management System - Exercise 9"
    message["to"] = "alexandre.ducommun@bluewin.ch"
    message["subject"] = "New data available"

    body = "new data available for :\n"
    for region in set(regions):
        r = "\n " + region
        body += r

    message.attach(MIMEText(body))

    with smtplib.SMTP(host="smtp.gmail.com", port=587) as smtp:
        smtp.ehlo()
        smtp.starttls()
        smtp.login("system.testdm@gmail.com", "UNINE12345678")
        smtp.send_message(message)
        print("Sent...")
```