CS 5970 I Activity Police Report Extraction

Due date 2/24/17 (14 days)

In this activity, we are going to stretch the super powers you have learned thus far. Use your knowledge of Python and the Linux command line tools to extract information from a scraped file and add it to an SQLite database.

The Norman, Oklahoma police department regularly reports of incidents arrests and other activity. This data is hosted on their website. This data is distributed to the public in the forma of PDF files.

The website has three types of `arrests`, `incidents`, and `case summaries`. Your assignment in this project is to collect **just the incidents**. To do so, you need to write code to (1) download the data; (2) extract the id, number, datetime, location and incident ori; (3) create a SQLite database to store the data; (4) insert the data into the database; (5) return the status of the database.

Below we describe the assignment structure and each required function. To complete the project, you may use any Python3 library available from pypi.

Project Descriptions

Your code structure should be in a directory with the following format:

```
normanpd/
    normanpd/
        normanpd.py
        __init__.py
    README
    setup.py
    requirements.txt
    main.py
```

Create a `README` file that will act a a write-up for your project. You should include drections on how to install and use the code. You should describe all functions and your approach to develop the database. You should describe any known bugs and cite any sources or people you used for help. **Besure to include any assumptions you make for your solution.**

The `setup.py` and `requirement.txt` files should be used to describe your code package the code and describe all external packages, respectively. To test the code, we will install your package, essentially using the `pip -e install .` command. Ensure your code is able to be installed and executed on the GPEL machines.

The `main.py` will be used to execute youre code. Create a function called `main()` that imports the your project and sequentially calls each of the function described. Below is a template main function. Feel free to combine or optimize functions as long as you code preserves the behavior below.

Python

```python
#!/usr/bin/env python
# -*- coding: utf-8 -*-
import normanpd
from normanpd import normanpd

def main():
    # Download data
    normanpd.fetchincidents()

    # Extract Data
    incidents = normanpd.extractincidents()

    # Create Dataase
    normanpd.createdb()

    # Insert Data
    normanpd.populatedb(incidents)

    # Print Status
    normanpd.status(db)

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

The your code folder should have a package called `normanpd/`. In side the folder should be an empty `__init__.py` file and python file called `normanpd.py`. The latter is where the majority of your code will reside. Below are the five main functions this code should perform.

Download Data

The function `fetchincidents()` takes no parameters, it uses the python urllib.request library to grab all the insident pdfs for the norman police report webpage.

You can use the code below to grab the daily activity web page.

```python
request.urlopen('http://normanpd.normanok.gov/content/daily-activity')
    .read()
    .decode('utf-8')
```

You may then use regular expressions to discover the url strings of all the pdf documents. Construct those URLs to download the PDFs to a defined location. The locations of files can be stored as a global object, a config file, any other method of your choosing.

Extract Data

The function `extractincidents()` takes no parameters and it reads data from the pdf files and extracts the incidents. The each incident includes a `date/time`, `incident number`, `location`, `nature`, and `incideent ori`. This data is hidden inside of a PDF file.

To extract the data from the pdf files, use the PyPdf2.PdfFileReader class. It will allow you to extract pages and pdf file and search for the rows. Extract each row and add it to a list.

This function can return a list of rows.

Create Database

The `createdb()` function creates an SQLite database file named `normanpd.db` and inserts a table with the schema below.

```sql
CREATE TABLE incidents (
    id INTEGER,
    number TEXT,
    date_time TEXT,
    location TEXT,
    nature TEXT,
    ORI TEXT
);
```

Note, the `id` column can be a unique counter for all incidents in the table. The other columns correspond directly to the columns in the incident pdfs.

Insert Data

The function `populatedb(incidents)` function takes the rows created in the `extractincidents()`

function and adds it to the `normanpd.db` database. The signature of this function can be changed as needed.

Status Print

The `status()` function prints to standard out, first, the total number of rows in the database and then prints five random rows from the database.

Submission Instructions

Wrap the database file, README file, and scripts and any python or linux scripts you used in the assignment in a compressed .tar.gz file with the name normanpd_4x4. For example `normanpd_bond0007.tar.gz`.

Grading Criteria

- 30 % README file is thoroughly written
- 20 % Code is installed and executes with no errors
- 50 % Each function runs successfully

Plagiarism will be treated as a violation of academic integrity policy.