

E3FI

Python

Traitement et mise en forme de l'occupation des salles d'ESIEE Paris

Margerit Arthur – Vigier Thibault
21/11/2015

Table des matières

Introduction.....	2
But	2
Organisation du projet	2
Récupération des Données en ligne.....	3
En Ligne avec pycurl	3
Avec les fichiers	3
Parsing du calendrier.....	4
Avec iCalendar	4
Traitement des données	4
Pandas	4
Filtre.....	4
Sélection simple.....	4
Sélection Expression régulière	5
Affichage des données	5
Avec Matplotlib	5
Interface graphique.....	6
Avec Tkinter.....	6
Conclusion	6
Sources	7
Information.....	7
Documentation.....	7
Bibliothèque	7

Les sources du projet sont disponibles sur GITHUB: <https://github.com/tiregram/python-data>

Introduction

But

Le principe du projet est de mettre en place un outil permettant de visualisé les salles les plus utilisées et visitées à ESIEE Paris. Cette peut utiliser le service en ligne <https://planif.esiee.fr/> directement via internet ou utiliser un fichier.

Il est également possible de choisir les dates de début et de fin à observer.

Nous pouvons sélectionner et supprimer des éléments avec des expressions régulières.

Organisation du projet

Nous nous somme reparti le projet et avons travaillé ensemble à la fusion et à la mise en place du code.

data/

Contient les .ics et les données téléchargées par l'application. Il y a aussi le cookie qui permet l'authentification sur le site planif ESIEE.

src/

Contient toute les sources python nécessaire à l'installation du projet.

src/main.py

Importe tous les autre .py il permet de mettre en place l'interface graphique et les autres class

src/frame.py

Ce fichier contient la class qui s'occupe de l'interface graphique

src/DataGet.py

Ce fichier contient la class qui récupère et parse les données du calendrier en ligne ou du fichier.

src/DataCal.py

Ce fichier contient la class qui effectue le calcul du nombre d'heure total, et applique les filtres.

CONFIGURE

Ce script bash permet l'installation des lib python nécessaires au lancement des projets.

Récupération des données en ligne

Avec pycurl

PycURL est une interface de libcurl qui permet la récupération de fichier via une adresse URL. Gratuite, simple d'utilisation, polyvalente, elle supporte un nombre conséquent de protocoles de transfert. Nous avons donc décidé de l'utiliser dans sa version 7.19.5.3 disponible à l'adresse <http://pycurl.sourceforge.net/>

PycURL est utilisé dans la fonction getICS() de la classe DateDataFrom. Afin de télécharger le planning sur une durée définie, le site <https://planif.esiee.fr> nécessite préalablement une authentification. Pour pallier à ce problème, nous nous connectons une première fois au site en spécifiant des cookies ainsi que des logins, qui vont permettre de stocker un certains nombres d'informations de connexion à l'adresse: data/cookie.txt

Par la suite, une ultime requête comportant les dates que nous souhaitons est envoyée, celle-ci permet de récupérer le calendrier ics de toutes les salles de l'ESIEE sur la période définie. Ce calendrier est enregistré à l'adresse data/output.ics

La mise en forme des dates pour l'URL est définie dans la fonction getDateToUrl(). A partir de la date de début et la date de fin, elle retourne les informations de la date sous format URL.

Avec les fichiers

Par défaut, lorsque la demande de calendrier est générée par un clic sur le bouton "Generate" de l'interface graphique, deux champs de l'objet DateDataForm sont initialisés. Le premier "online" contient le statut de la checkbox correspondante à l'appel du fichier local; le deuxième "icsFileName" contient le chemin du fichier en local (null si vide).

Lorsque le programme arrive dans la méthode parseICS(), si la checkbox du fichier en local n'est pas cochée, la fonction getICS() est appelée afin de télécharger le calendrier et de modifier la variable "icsFileName" pour qu'elle pointe vers le fichier téléchargé.

De ce fait, le programme favorisera toujours l'appel du fichier en local si la demande en est expressément faite.

Parsing du calendrier

Avec iCalendar

Ics est une [extension de fichier](#) informatique utilisée pour désigner un fichier contenant des agendas et de la programmation d'informations compatibles avec [iCalendar](#) et bien d'autre.

La bibliothèque que nous avons choisie est la suivante [icalendar](#) en version 3.9.1 disponible à cette adresse <https://pypi.python.org/pypi/icalendar>.

Après avoir parsé les événements nous retournons une liste de dictionnaires contenant les informations qui nous intéressent (date de début, date de fin, localisation).

Traitement des données

Pandas

Pandas est une bibliothèque python qui fournit des outils pour l'analyse de Data.

Description officielle de pandas:

pandas is a [Python](#) package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labeled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, **real world** data analysis in Python. Additionally, it has the broader goal of becoming **the most powerful and flexible open source data analysis / manipulation tool available in any language**. It is already well on its way toward this goal.”

Cet outil est donc parfaitement approprié à notre projet et pourra être réutilisé en entreprise.

Filtre

Il existe deux types de filtre, que nous avons mis en place.

Sélection simple

La sélection simple permet de choisir uniquement une liste de salles séparées par une virgule.

Exemple:

210,110,260,160

Ne sélectionnera que les amphithéâtres.

Cette solution est facile à mettre en place avec l'option `query` des DataFrames.

```
self.query('piece in @roomList')
```

Et avec un “not” pour la sélection inverse.

```
self.query('piece not in @roomList')
```

Sélection expression régulière

La sélection par expression régulière est basée sur le système suivant:

Exemple:

`1[0-9]{3}`

Sélectionne toutes les sales de l'épi 1.

```
self['test'] = self.pieces.str.match(exprre)//on rajoute une colonne test qui a comme valeur le résultat du match de l'expression régulière.
```

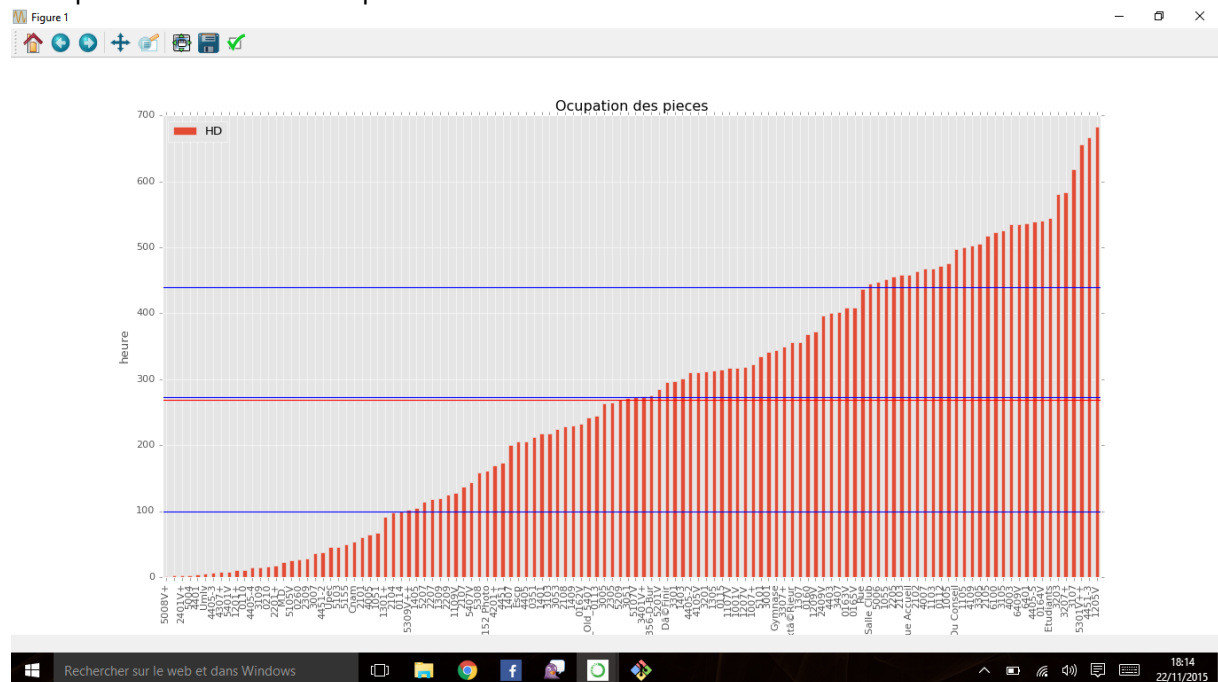
```
return DataCal(self.query('test == True').drop('test',1))
```

```
// On retourne le nouveau élément, en drop la colonne test qui a permis de faire la query.
```

Affichage des données

Avec Matplotlib

Afin d'afficher les résultats graphiquement nous avons utilisé Matplotlib car il est déjà implémenté dans pandas avec la fonction plot.



Interface graphique

Avec Tkinter

Tkinter est un module de TK, une librairie open-source permettant la création d'interface Homme Machine (IHM) en python. Sa portabilité envers les différentes plateformes, ainsi que sa simplicité d'utilisation ont été des critères de choix pour ce projet.

L'IHM de l'application se découpe en trois parties (panel + label) :

- Planif Dates, composé de deux champs permettant la saisie, un bouton "Generate", ainsi qu'un label informant sur la validité des dates saisies.
- File Options, composé d'une checkbox indiquant si oui ou non le programme doit tenir doit utiliser un fichier en local, un "file browser" permettant de sélectionner le fichier en local (accessible que si la checkbox est activée), ainsi qu'un label indiquant le chemin d'accès du fichier sélectionné (null par défaut)
- Advanced Panel, composé successivement de quatre ensemble checkbox + champs de saisie. Lorsqu'une checkbox est activée, le programme prend en compte les champs associé pour sélectionner/supprimer une salle ou un groupe de salle, du calendrier.

Dans l'application, la classe Frame hérite de tkinter.TK

Conclusion

Ce projet nous a permis de renforcer notre maîtrise du Python dans un projet concret. Il nous a aussi permis de découvrir de nombreuses bibliothèques de code tel que pandas, pycurl, Tkinter et icalendar. Nous avons mis en place des aspects de la programmation objet tels que l'héritage en python.

Sans oublier l'utilisation d'un gestionnaire de version git.

Sources

Information

<https://fr.wikipedia.org/wiki/.ics>

<https://planif.esiee.fr>

<https://icampus.esiee.fr/course/view.php?id=170>

Documentation

<https://docs.python.org/3/>

<http://www.programmez.com/magazine/programmez-189-pdf>

Bibliothèque

<http://pandas.pydata.org/>

<http://pycurl.sourceforge.net/>

<https://pypi.python.org/pypi/icalendar>

<https://docs.python.org/2/library/tkinter.html>