

Campos Claudomir

RATSIMATAHOMANANA Ny Heja Finaritra

ROGERALY Naresh



Projet Python

RankingBuilder – League of Legends

Table des matières

Description du projet	3
Déroulement	4
Manuel d'utilisation	5
Difficultés rencontrées	6

Description du projet

Rakingbuilder est un projet utilisant la bibliothèque riotwatcher qui permet à l'application de communiquer avec l'API de league of legends dont le but d'avoir des informations sur des joueurs.

Ce projet propose deux grandes fonctionnalités :

- Ranking Factory – Menu pouvant superviser les différents classements présents dans la base de données : comme le fait de créer des classements, supprimer des classements, d'ajouter des joueurs, programmer les mises à jour, préparation de l'environnement pour exploiter les classements (page html, création de la base de données, planification des tâches)
- Main Ranking – Application qui permet de communiquer avec l'API et mettre à jour les différentes bases de données (classements)

Package	Version	Usage
LolWatcher	base	Communiquer avec LOL
Pandas	0.20.3	Package de calcul et d'analyse de données
Sys	Base	Package d'interface avec l'interpréteur python
Math	base	Package de fonctions mathématiques
Numpy	base	Package de calcul et manipulation de données
Psycopg2	2.8.3	Package de database adaptation
Time	base	Package pour les informations liés à la date, temps
Os	base	Package d'interface avec le système d'exploitation

Déroulement

Factory.py :

FICHIERS ENTREE	
Players.txt (opt)	Une liste déjà fournie si l'on veut ajouter des joueurs à un classement lors de sa création

PROGRAMME PYTHON	
USE	Factory.py

FICHIERS DE SORTIE	
OUT	NONE

k.py :

FICHIERS ENTREE	
NOM DU CLASSEMENT en arg[1]	Le nom du classement dans lequel on va effectuer des mises à jour

PROGRAMME PYTHON	
USE	Tâche CRON -> k.py [nomClassement]

FICHIERS DE SORTIE	
OUT	nomClassement/Log.txt

Manuel d'utilisation

Pour la création d'un classement et la génération de la page WEB veuillez suivre ses étapes :

- Lancer factory.py avec python 3

```
Ranking Generator, veuillez choisir une option
1 - Acceder a un classement
2 - Creer un classement
3 - Quitter
```

- Choisir « 2 » pour créer son classement puis entrez le nom du classement : pour le manuel ça sera « ESGI »
- Une fois le classement créé on pourra voir l'existence d'une base de données portant le nom du classement, la création d'un répertoire dans le ftp de kodaline.fun/esgi et d'un répertoire sur ./factory/esgi et ./factory/esgi/logs et ./factory/esgi/players
- Pour ajouter des joueurs sur le classement soit passer par l'interface d'administration sur kodaline.fun/esgi/adm générée soit en plaçant un players.txt dans ./factory/esgi/players et choisir l'option «Ajouter des joueurs à ce classement » et entrer le nom du fichier texte.

```
Veuillez choisir l'action a faire avec le classement : ESGI
1 - Ajouter des joueurs a ce classement
2 - Ajouter une planification pour la maj de ce classement
3 - Visualiser le classement
4 - Visualiser les dernieres logs de ce classement
5 - Delete database
6 - Revenir au menu principal
```

- Maintenant il faut planifier la tâche pour que le classement se mette à jour de manière quotidienne et donc choisir l'option « Ajouter une planification pour la maj de ce classement »
- Nous avons maintenant fini avec les tâches liées à la factory de classement

Maintenant que toutes ces tâches sont réalisées il suffit d'attendre le cron qui va lancer votre classement en passant en argument le nom de votre classement à k.py [nomClassement] une fois qu'il s'est terminé vous pourrez aller à kodaline.fun/esgi et voir votre classement.



Ajoutez votre pseudo

Ajouter

Rank	Pseudo	Win	Loose	Winrate	Division	Palier	League Points
1	Kvu	124	98	56	DIAMOND	I	22
2	O K	77	87	47	PLATINUM	I	13
3	Fungala kwaki	296	286	51	PLATINUM	II	0
4	Thomas cafard	68	87	44	GOLD	II	58
5	Actae	76	111	41	GOLD	IV	0

Difficultés rencontrés

Durant ce projet nous avons rencontrés de diverses difficultés :

- Demande de clef d'API provenant de LEAGUE OF LEGENDS : délais de deux semaines afin de l'obtenir fallait attendre que les dev de riot nous fournisse une clef d'API
- Temps d'exécution qui peut être long :
On a essayé de divers moyens comme le multi-threading mais on a vite été limité aux nombres de requêtes HTTP à la seconde.