

TP Publication automatique de données sur internet



(I) Envoi d'informations au site Dweet.io

Les dispositifs Raspberry, Arduino, etc sont souvent reliés à des capteurs qui émettent des données (température, luminosité, ...) , et pour que ces données soient lues par un opérateur distant, on utilise souvent des sites internet, pour stocker ces données, et produire des représentations graphiques des valeurs relevées. Ces sites peuvent avoir un intérêt pour superviser tout type d'information, comme celles que nous émettrons par exemple dans le cadre des projets FAS 😊)

1. Rendez-vous sur le site [Dweet.io](https://dweet.io), la première page permet de comprendre comment fonctionne ce site. Faites de mêmes sur dweetpro.io . Comprenez-vous la différence ? **Faites votre choix pour le reste du TP** (conseil : dweetpro pour avoir un *lock*).
2. Depuis le terminal, utilisez l'outil `curl` pour dweeter en mode GET plusieurs mesures de position `x & y` pour un robot **RobotZ** (où Z est le numéro de votre binôme), comprises entre 0 et 300 en utilisant des requêtes http contenant une *query string* sur la base du modèle suivant
`curl https://dweet.io/dweet/for/my-thing-name?hello=world&foo=bar`
ou `curl https://dweetpro.io/dweet/for/my-thing-name?hello=world&foo=bar`
3. Normalement le contenu de la réponse reçue du site vous indique s'il a bien reçu les données ({{ « this »: « *succeeded* », ...} notez que la réponse reçue est au format **json**).
4. En utilisant la **Console** de dweetpro.io ou la page *Play* de dweet.io on peut gérer aussi manuellement les dweets d'un objet connecté (que ce soit un objet d'un compte pro ou pas). Explorez l'interface pour répondre à la question suivante.
5. Supposons que vous avez un programme qui écoute les mesures transmises pour votre robot, ce programme peut lire les données de positions en émettant des requêtes http lui aussi. Trouver sur la page dweet.io :
 1. quelle requête permet d'avoir juste la dernière position (x & y) postée pour votre robot
 2. toutes les positions postées pour votre robot.

Normalement vous récupérez aussi l'heure (GMT) à laquelle chaque dweet a été émis.

6. Il nous est plus facile de lire un graphique qu'une liste de chiffres. Pour cela le site nous fournit une interface rudimentaire : dans un navigateur, ouvrez l'adresse <https://dweetpro.io/follow/RobotZ> en remplaçant, comme précédemment, Z par votre numéro de robot. Emettez de nouvelles coordonnées dans le terminal (avec `curl`) et vérifiez dans le navigateur que les diagrammes se mettent à jour. Notez que ces diagrammes sont non paramétrables.
7. Les requêtes précédentes se sont faites en `https`, mais Dweet.io indique qu'elles peuvent être faits en `http`, vérifiez sur deux ou trois exemples. Vous noterez aussi, sans qu'on ait le temps de s'attarder ici, que ce site dispose d'une possibilité de nous envoyer une alerte par mail en cas de dépassement d'une seuil (par exemple sortie du terrain d'un robot, dépassement d'une température) et d'une possibilité de nous envoyer les nouvelles valeurs en continue (avec une connexion *keep-alive*).

Bon, c'est pas tout ça, mais si votre projet utilise ce site, ce sera en Python. Cherchez (et trouvez) un module python permettant de *dweeter* facilement depuis un programme (si plusieurs modules sont possibles, choisissez judicieusement). Installez-le (sur les machines TP, sinon sur Raspberry).

8. Faites un programme envoyant un couple de coordonnées x et y (**important : ne pas le mettre dans une boucle sans `time.sleep(5)`, sinon Polytech va se faire bannir**) . Notez que par défaut les fonctions de la bibliothèque envoient sur dweet.io et non dweetpro.io .
9. Vérifiez que Dweet.io reçoit bien les données depuis <https://dweet.io/follow/RobotZ> dans un navigateur.

10. **Challenge** : quel est le premier groupe qui va trouver comment modifier la bibliothèque pour que les données soient envoyées sur dweetpro.io ?

(II) Envoi d'informations au site Google Drive

Nous allons maintenant hacker le site de Google, rien que ça ! En fait, plus précisément il s'agit d'extraire des informations que Google aimerait maintenir plutôt secrètes pour qu'on ne détourne pas le fonctionnement de ses services ;)

Vous avez remarqué comme le suivi graphique des données envoyées sur Dweet.io était *cheap* ?

Pour passer à des représentations plus sympas, nous allons utiliser d'autres services, plus jolies, dont certains capables de s'interfacer avec un tableur de google drive (gs spreadsheet). Collecter les données de notre robot dans un tableur nous aidera aussi à faire des analyses (stats, analyse de données, etc).

Mais il y a un petit hic : pour publier des données dans un tableur du drive il faut en général montrer patte blanche et fournir une clef API qui identifie qu'on est bien le propriétaire du document. La procédure pour obtenir une telle clef et s'en servir est un peu longue, aussi nous allons décider de contourner l'étape d'authentification. Pour cela nous allons créer un formulaire qui est mis en lien avec un tableur de façon automatique :

1. Générez un formulaire contenant les données de position x et y pour votre robot : en vous identifiant par exemple comme raspberrylg@gmail.com (mdp donné par votre enseignant) : <https://docs.google.com/forms/u/0/> et choisissez un formulaire vide (blank)
2. Ajoutez deux champs de type *Question courte* dans le formulaire, l'un appelé x, et l'autre y, donnez un titre « RobotZ » (où Z est ...) à votre formulaire. Notez l'id de votre formulaire qui est caché dans l'adresse url que visite votre navigateur quand vous éditez le formulaire :
<https://docs.google.com/forms/d/1d...id..ici..JCs/edit>
3. Allez ensuite dans l'onglet *Réponses* pour faire le lien avec un tableur créé exprès qui collectera les réponses reçues par le formulaire : cliquez sur l'icône croix blanche / fond vert « créer ... » puis indiquez un nom de feuille de calcul correspondant à votre robot. Le navigateur ouvre automatiquement ou vous propose d'ouvrir la feuille de calcul correspondante dans un nouvel onglet : très bien.
4. Dans le formulaire, allez maintenant dans ENVOYER/SEND (haut à droite), puis choisissez l'icône en forme de lien (–), copiez l'adresse et ouvrez là dans un nouvel onglet. Puis remplissez le formulaire, soumettez et vérifiez que la feuille de calcul s'est remplie avec la réponse. Faites une 2ème tentative pour vous convaincre que tout est ok.
5. Bien, on a donc une feuille de calcul qui se remplit avec nos données, maintenant on veut automatiser le processus pour que ce soit un de vos programmes qui envoie les données automatiquement : placez-vous dans l'onglet où on peut remplir le formulaire, chaque champ à remplir a un nom caché qu'il nous faut découvrir. Dans Safari : menu Développement -> Démarrez la sélection d'éléments puis cliquez sur l'endroit où on met les données pour la question «x» : normalement l'inspecteur web s'ouvre en indiquant le bout de code html qui lui correspond. On cherche quelque chose qui ressemble à ceci: name=« **entry.360037641** ». On fait pareil pour le champ «y». Et vous obtenez par exemple name=« **entry.1187150394** ». Si vous êtes sous Firefox, allez dans le menu Outils -> développement web -> ...)
6. Maintenant on a tous les éléments pour simuler une requête de notre robot. Testons d'abord ça dans le terminal à l'aide de notre ami curl :

```
curl "https://docs.google.com/forms/d/1d...id..ici..JCs/formResponse?ifq&entry.360037641=10&entry.1187150394=50&submit=Submit"
```



"I hacked into the school computer and changed all my grades. Then the school hacked into my computer and deleted all my games!"

par exemple pour indiquer un point de coordonnées $x=10$ et $y=50$. Vérifiez que la feuille de calcul google s'est mise à jour et contient maintenant ces coordonnées. *Question subsidiaire : pourquoi on n'a pas besoin d'écrire `\&` dans la commande précédente ?*

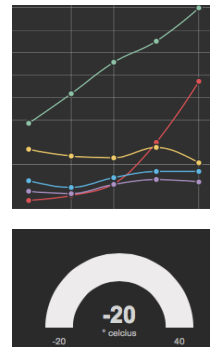
7. Ok, maintenant on va faire ça par programme, en utilisant le module python **requests** (qui est probablement installé suivant le module pour dweet que vous avez installé précédemment). Faites un programme qui publie deux coordonnées x et y pour votre robot avec `requests.post(...)` et vérifiez qu'un couple de données se sont bien ajoutées dans le tableur du compte google.
8. Mais alors... finalement, pour les interactions avec Dweet(pro) il vaut mieux utiliser uniquement ce module plutôt qu'une bibliothèque python ? Oui, car on peut contrôler nous même sur quel site on envoie (dweet ou dweetpro ou un autre site !) et quelle commande on fait (une fois qu'on connaît l'API, on peut fabriquer nous mêmes les adresses). Faites un essai rapide pour publier une donnée sur *Dweet* de cette façon.

(III) Monitoring des informations par des graphiques

Pour faire des graphiques nous pouvons déjà nous rendre dans le tableur qui se remplit automatiquement : menu Insert -> Chart et là c'est magique, la figure se complète au fur et à mesure que les données arrivent.

Certains sites proposent des affichages plus sympatiques et nous vous demandons d'explorer ici deux d'entre eux : Infogr.am et FreeBoard.io. Parvenez à représenter les coordonnées x et y de votre robot extraites de la feuille de calcul google drive :

- sous la forme de graphique en lignes sur les deux sites (on veut un graphique qui évolue automatiquement si possible)
- sur **Infogr.am** représentez les points parcourus par le robot sous la forme de Chart *Line*, puis sous la forme de Chart *Scatter plot*
- Sur **Freeboard** représentez x et y sous la forme de jauges



Pouvez-vous localiser un autre site qui permet de faire des graphiques sympatiques de vos données ? Discutez-en avec vos enseignants.

(IV) Rétroaction : alerte en cas de dépassement d'un seuil

Supposons que vous voulez alerter quelqu'un ou un programme par le fait qu'une mesure dépasse certaines coordonnées (x, y en dehors d'une zone délimitée par exemple). Comment pouvez-vous faire ça depuis les sites examinés dans ce TP / depuis d'autres sites.

Décrivez comment un programme python pourrait réagir à une telle alerte.

(V) Explorez d'autres solutions

De nombreux autres sites IoT (internet of things) existent sur la toile, comme PubNub ou BeeBotte qui ont l'air tous les deux de permettre à la fois de stocker des données et d'en faire des graphiques. Explorez l'un d'entre eux et rédigez un petit tutoriel avec texte+images à l'usage des autres étudiants (on le placera sur Moodle après relecture).