



# QUI EST DANS L'ISS ?

## ➤ La station spatiale ISS

La **Station Spatiale Internationale**, ou ISS (d'après l'anglais International Space Station), est une station spatiale placée en orbite terrestre basse, occupée en permanence par un équipage international qui se consacre à la recherche scientifique dans l'environnement spatial. Ce programme, lancé et piloté par la NASA, est développé conjointement avec l'agence spatiale fédérale russe, avec la participation des agences spatiales européenne, japonaise et canadienne. La NASA partage de nombreuses données sur Internet concernant la position de la station, ou encore le nombre de personnes présentes.

Dans ce projet tu vas utiliser un web service pour trouver et afficher qui se trouve actuellement dans l'ISS.



## ➤ Le Web Service

Un web service a une adresse (url) comme toute page web. Au lieu de renvoyer du code HTML dans le cas d'une page web, il retourne des données.

Ouvre la page suivante dans ton navigateur : <http://api.open-notify.org/astros.json> (CTRL + clic).

L'activité que vous allez réaliser va vous permettre de répondre, entre autres, à ces trois questions :

Quel est le format des données enregistrés dans cette page ?

Combien de personnes sont actuellement dans l'ISS ?

Comment s'appellent-ils ?

Dans quels vaisseaux sont-ils ?



## ➤ Traitement des données par Python

Ouvrez l'IDE Python **Pyzo** via le dossier du bureau « Autres raccourcis/SNT/Pyzo\_SNT ». Seules les lignes sur fond « orange » doivent être saisies dans la zone de programmation (zone à gauche de la fenêtre de **Pyzo**).

Tu peux aussi t'aider du cours disponible sur le site sur la page dédiée au langage PYTHON.

Enregistre ton programme dans le dossier « **Documents/SNT/THÈME\_4** » avec le nom de fichier suivant : « **ISS-TonNomDeFamille** ». N'oublie pas de partager ton programme en fin de séance !

### 1. Écriture du code

**Note : ne fais pas de copier / coller, il faut saisir le code qui est sur fond orange**

```
import json
import urllib.request
```

- À quoi servent ces deux lignes : afin d'utiliser des bibliothèques spécifiques à cet exercice, il faut ajouter (importer) des modules supplémentaires au programme en ajoutant ces toutes premières lignes.

```
adresse = 'http://api.open-notify.org/astros.json'
```

- À quoi sert cette ligne : cette ligne de code permet de stocker l'adresse web du web service dans la variable « **adresse** ».

```
donnees = urllib.request.urlopen(adresse)
```

- À quoi sert cette ligne : cette ligne de code permet d'appeler le web service et de stocker le résultat dans la variable « **donnees** ».

```
dico = json.loads(donnees.read())
print(dico)
```

- À quoi servent ces deux lignes : la première ligne permet de charger la réponse « **json** » dans une structure de données Python. C'est une table de données (dictionnaire en langage Python) ; la seconde ligne affiche dans le « **shell** » le contenu de la variable dictionnaire « **dico** ».

Exécute le programme : menu **Exécuter** puis **Démarrer le script** !

Normalement tu devrais retrouver exactement les mêmes informations que sur le site en ligne mais, sous une forme bien différente.

Si un ou plusieurs messages d'erreur (en rouge) apparaissent lors de l'exécution, vérifie le code que tu as saisi ; il doit être identique, à la lettre près, aux lignes de code indiquées ci-dessus !

Et si tu as copié/collé le texte de ce document, les erreurs peuvent être issues de cette technique !!!



## a. Gestion du dictionnaire

Un dictionnaire en python est une **base de données** qui utilise des **descripteurs (clés)** alphanumériques associées à des **valeurs (valeurs)**. Chaque **objet (enregistrement)** est constitué de **clés**, une **valeur** étant affectée à chaque **clé**. Référence au cours : [lien](#) (CTRL + clic).

Exemple :

```
{'craft': 'ISS', 'name': 'Sergey Prokopyev'}
un dictionnaire python est encadré par des crochets
chaque enregistrement est encadré par des accolades
```

Pour la suite, il te faut identifier certaines informations, par exemple, comment s'appelle le dictionnaire de ton programme, quelles sont les clés et quels sont leurs noms.

Le dictionnaire principal « *dico* » utilise les **clés** suivantes : « *people* », « *number* » et « *message* ». « *people* » est un dictionnaire contenu dans le dictionnaire principal. Il comporte les **clés** suivantes : « *craft* » et « *name* ».

		<i>dico</i>	
		<i>number</i>	<i>message</i>
<i>people</i>			
<i>craft</i>	<i>name</i>		
Vaisseau A	Astro 1	Nb astronautes	
Vaisseau B	Astro 2		Success
...	...		

Dictionnaire principal
Dictionnaire contenu dans le dictionnaire « *dico* »

## b. Nombre d'astronautes dans l'espace

- Maintenant, tu vas chercher comment afficher les informations de façon plus lisible, par exemple, afficher le nombre d'astronautes présents dans l'espace :

```
print("Nombre d'astronautes dans l'espace : ", dico['??'])
```

Quelle **clé** parmi les **clés** présentées ci-dessus dois-tu inscrire à la place des points d'interrogation ?

Pour t'aider, tu peux consulter la page suivante : [Python.Doctor](#) (CTRL + clic)

Teste ton programme, celui devrait afficher le nombre d'astronautes présents dans l'espace.

## c. Nom des astronautes dans l'espace

La valeur associée à la clé « *people* » est aussi une base de données (*dictionnaire*) !

```
liste_astronautes = dico['people']
print(liste_astronautes)
```

- À quoi servent ces deux lignes : la première ligne permet de stocker en mémoire le contenu du dictionnaire « *people* » dans la variable « *liste\_astronautes* » afin de pouvoir l'utiliser ensuite ; la seconde affiche dans le « *shell* » le contenu de la variable « *liste\_astronautes* » et donc le dictionnaire « *people* ».

Maintenant, tu vas chercher comment afficher une ligne pour chaque astronaute.

Pour cela, il faut utiliser une boucle **for objet in \_\_\_** :

À chaque passage dans la boucle « **for** », *objet* correspondra aux données d'un astronaute différent.

```
for objet in !!! :
    print(objet ['??'])    #Il faut respecter le décalage à droite (tabulation)
```

Remplace les points d'exclamation par la bonne **variable** et les points d'interrogation par la bonne **clé** afin d'afficher les noms des astronautes présents dans l'espace.

Tu utilises les données en direct donc le résultat dépendra du nombre de personnes actuellement dans l'espace.



### d. Afficher le nom du vaisseau

Comme pour le nom des astronautes, le web service fournit le nom du vaisseau dans lequel se trouve l'astronaute (par exemple l'ISS).

Peux-tu modifier ton script pour qu'il affiche aussi le nom du vaisseau dans lequel se trouvent les astronautes ?

Exemple :

```
L'astronaute Mark Vande Hei est dans le vaisseau ISS
L'astronaute Pyotr Dubrov est dans le vaisseau ISS
L'astronaute Anton Shkaplerov est dans le vaisseau ISS
L'astronaute Zhai Zhigang est dans le vaisseau Shenzhou 13
L'astronaute Wang Yaping est dans le vaisseau Shenzhou 13
L'astronaute Ye Guangfu est dans le vaisseau Shenzhou 13
L'astronaute Raja Chari est dans le vaisseau ISS
L'astronaute Tom Marshburn est dans le vaisseau ISS
L'astronaute Kayla Barron est dans le vaisseau ISS
L'astronaute Matthias Maurer est dans le vaisseau ISS
```

### e. Afficher les astronautes de l'ISS

Comme indiqué dans le titre, le but de ton programme devait être d'afficher les spationautes présents dans l'ISS et non pas tous ceux présents dans l'espace.

Peux-tu modifier ton script afin qu'il n'affiche que les astronautes présents dans l'ISS ?

N'hésites pas à te référer au code saisi lors de la réalisation de ton premier programme (voir la page PYTHON, activité : Démarrer avec Python).

Sauvegarde ton programme dans ton dossier THEME\_4 !  
Nous l'utiliserons dans une activité future pour déterminer la position de l'ISS.



En fin de séance ou si tu as terminé cette activité, n'oublies pas de partager ton programme 👍