# Le site de nooSFere

Le URL de nooSFere est https://https://www.noosfere.org/.

Voici un site extrêmement riche consacré aux publications à propos de l’imaginaire.

On y trouve des références aux livres, aux auteurs, aux 4me de couverture, pour ne nommer que quelques-uns des sujets présentés dans l’encyclopédie : [Ce que contiennent les bases de données (noosfere.org)](file:///C:\Users\Papa\AppData\Roaming\Microsoft\Word\Ce%20que%20contiennent%20les%20bases%20de%20données%20(noosfere.org))..

nooSFere héberge des sites amateurs : [nooSFere - Sites d'adherents](https://www.noosfere.org/noosfere/heberges.asp)

nooSFere héberge des sites d’auteurs et d’illustrateurs : [nooSFere - Sites d'auteurs](https://www.noosfere.org/noosfere/sites_auteurs.asp)

Je ne veux pas reproduire ici ce que le site dit bien mieux que moi…

Voir :

[nooSFere : Qu'est-ce](https://www.noosfere.org/noosfere/assoc/qu_estce.asp)  en bref

[nooSFere - Presentation de l'association](https://www.noosfere.org/noosfere/assoc/statuts.asp) en détail

[Questions à la nooSFere](https://www.noosfere.org/icarus/articles/article.asp?numarticle=463)

[nooSFere - Plan du site](https://www.noosfere.org/actu/news.asp)

Le site nooSFere ne vend ni n’édite des livres. L’association propose a ses membres (cout de 30€/an voire 10€/an en tarif réduit… on peut donner plus 😊) des services vraiment superbes. Voir [Pourquoi adhérer](https://www.noosfere.org/noosfere/assoc/pourquoi.asp).

Je ne fais PAS de pub, mais j’aime vraiment bien ce site… Attention, ce site va changer dans le futur, c’est annoncé :

[nooSFere - Soutenir l'association](https://www.noosfere.org/noosfere/assoc/don.asp)

Mon idée est de me permettre de télécharger les infos relative a un livre dans calibre

# L’API (actuel) de ce site, ou plutôt de l’encyclopédie de ce site.

### Une recherche simple par [nooSFere - Recherche](https://www.noosfere.org/noosearch_simple.asp)…

Il suffit de remplir la case recherche et envoyer par »enter » ou par <CR>. Le site répond tout ce qui correspond aux ‘Mots’ écrits dans la case avec interprétation libre (essayez riCH pour voir).

Bien sur c’est magnifique, mais pour filtrer ce que tu veux, il faut un humain ou une IA… (non je ne peux pas programmer une IA)

### Une recherche avancée par [nooSFere - Recherche dans les bases de nooSFere](https://www.noosfere.org/livres/noosearch.asp).

On a aussi une case pour y écrire les ‘Mots’ rechercher, mais on peut sélectionner le sujet (auteur, livres, série, etc.). De plus on peut préciser si on veut tous les mots, n’importe quel mot ou les mots proches l’un de l’autre. On peut exiger la correspondance exacte des mots plutôt que des phrases et des mots approchants…

## Recherche simple par programme :

On envoie une requête, méthode « post » vers :<https://www.noosfere.org/noosearch_simple.asp> avec pour arguments : "Mots" : "<entrée dans la boite>"

Extrait de code python

lrpauteur = input("auteur : ")

# noosfere accepts \* to complete the name or the surname, trips on the dot in van vogt A.E.

lrpauteur=str(lrpauteur.replace(","," "))

lrpauteur=str(lrpauteur.replace(".","\* "))

base\_url="https://www.noosfere.org"

search\_urn=base\_url+"/noosearch\_simple.asp"

def postrequest(requete):

# Essayons un acces en mode post sur noosearch\_simple.asp

req=urllib.parse.urlencode(requete).encode('ascii')

sr=urllib.request.urlopen(search\_urn,req)

soup = BS(sr, "html.parser")

return soup

def ret\_autr\_indx(soup):

# Trouve la reference de l'auteur dans la soupe produite par noosfere

# retourne auteurindex, un dictionnaire avec key=AUTEUR, val=href

auteurindex={}

for child in soup.recursiveChildGenerator():

if child.name=="span" and "class" in child.attrs:

if 'resultat\_search' in child["class"]:

if "auteur.asp" in child.find\_next("a")["href"]:

auteurindex[child.text]=(child.find\_next("a")["href"])

if len(auteurindex) > 1:

print("Sorry, too many author found, please input one of the following :")

for key in auteurindex:

print(key.title())

elif len(auteurindex) < 1:

print("Sorry could not find the author")

return auteurindex

#auteur

requete = {"Mots": lrpauteur}

soup = postrequest(requete)

auteurindex = ret\_autr\_indx(soup) # quel est l'indexe de l'auteur? auteur is a dict

if len(auteurindex) != 1: sys.exit("Désolé")

for key,ref in auteurindex.items():

auteura,indexa = key,ref

## Recherche avancée par programme

On envoie une requête, méthode « post » vers :<https://www.noosfere.org/noosearch_simple.asp> avec pour arguments :

* "Mots" : "<entrée dans la boite>"