Exercise 2 - Literature and the French Theater

First name: Brian

Last name: Schweigler

Matriculation number: 16-102-071

Theoretical Q1: Cite four differences between XML and HTML standards

- XML allows plain text files to be used to share data.
- XML aims to describe information, HTML to display information
- XML tags are not predefined, as it is designed to be self-descriptive.
- XML is case-sensitive; HTML is case-insensitive.

Theoretical Q2: Are both XML and HTML fully declarative languages?

Both languages are declarative Languages but only HTML is a fully declarative one.

Q1: How many unique author names can you find?

General imports and set-up and solving the question:

In [1]:

```
%load_ext autoreload
%autoreload 2
%matplotlib inline
import lxml.etree
import os

authors = []
authors_set = set()
for fn in os.scandir('theatre-classique'):
    tree = lxml.etree.parse(fn.path)
    for author in tree.iterfind('//author'):
        if author.text is not None and author.text != "None" and "anonym" not in str(author.tex
            #print(str(author.text).lower()) # Was used for debug purposes
            authors.append(author.text.lower())
            authors_set.add(author.text.lower())
print( f'There are {len(authors_set)} distinct author names' )
```

```
There are 208 distinct author names
```

Q2: Are you sure that all these unique names refer to distinct authors?

There can be commas and "et" between authors; so we only have the distinct author groups in Q1.

In [2]:

```
author_dict = {}

for fn in os.scandir('theatre-classique'):
    tree = lxml.etree.parse(fn.path)
    for author in tree.iterfind('//author'):
        if author.text is not None and author.text != "None" and "anonym" not in str(author.tex
            current_author = str(author.text).split( " et ")
            for a in current_author:
                current_value = author_dict.get(a.split(",")[0].lower(),[])
                author_dict[a.split(",")[0].lower()] = current_value + [title.text for title ir
length_dict = { a : len(set(titles)) for a , titles in author_dict.items() }
print(f'We now have {len(length_dict.keys()) } distinct author names')
```

```
We now have 195 distinct author names
```

Q3 Can you reduce the variability around the author names?

There isn't a lot of variability after having everything in a set and lowercase, but we could remove any weird characters or numbers that we notice.

In [3]:

```
import re
regex_num = re.compile('[0-9]')
regex_bracket1 = re.compile('\)')
regex_bracket2 = re.compile('\(')
regex_tab = re.compile('\\t')
clean_dict = {}
for key in length_dict:
    new_key = regex_num.sub('', key)
    new_key1 = regex_bracket1.sub('', new_key)
    new_key2 = regex_bracket2.sub('', new_key1)
    new_key3 = regex_tab.sub('', new_key2)
    clean_dict[new_key3] = length_dict[key];
print(clean_dict)
```

```
{'aigueberre': 4, 'carmontelle': 22, 'chabanon': 3, 'chamfort': 2, 'champmeslé': 1, 'chazet':
3, 'dubois': 1, 'gassicourt': 1, 'chevreau': 1, 'colardeau': 1, 'colleville': 1, 'colle': 1, 'a
nseaume': 1, 'archambault': 2, 'coupigny': 2, 'crï¿½billon': 1, 'crébillon': 9, 'artaud': 1, 'c
ubières-palmézeaux': 8, 'baptiste': 1, 'du moutier': 1, 'cyrano': 2, 'dalibray': 1, 'dancourt':
49, 'barante': 3, 'dufesny': 2, 'barbier': 1, 'barré': 1, 'desfontaines': 4, 'beaumarchais': 1,
'deshoulières': 2, 'desmarets de saint-sorlin': 1, 'desportes': 1, 'diderot': 3, 'donneau de vi
sé': 3, 'beaunoir': 3, 'dorat': 2, 'gazon-doruxigné': 1, 'du bosc de montandre': 1, 'duche de v
ancy': 1, 'regnard': 27, 'dufresny': 16, 'du fresny': 4, 'bensérade': 2, 'dugazon': 1, 'dumania
nt': 1, 'feriol de pont-de-veyle': 1, 'durant': 9, 'du ryer': 14, 'bergasse': 1, "fabre d'eglan
tine": 1, 'favart': 1, 'beys': 1, 'florian': 5, 'folard': 1, 'fontenelle': 1, 'fuzelier': 5, 'l
egrand': 4, 'lesage': 2, 'biancolelli pierre-françois -': 2, 'genest': 1, 'gilbert': 1, 'gillet
de la tessonerie': 2, 'gouges': 2, 'goullinet': 1, 'biancolelli': 2, 'gresset': 1, 'gudin de la
brenellerie': 1, 'guerin de bouscal': 2, 'gueullette': 14, 'guibert': 1, 'harny de guerville':
2, 'hauteroche': 1, "henault d'armorezan": 1, 'hoffmann': 2, 'jaquelin': 1, 'philidor': 1, 'la
calprenede': 9, 'bideau de montigny': 1, 'la chapelle': 1, 'la croix': 1, "l'affichard": 1, 'la
fontaine': 7, 'bievre': 1, 'lafont': 1, 'la forge': 1, 'la fosse': 2, 'alain': 1, 'blanc': 1,
'la grange chancel': 1, 'la harpe': 5, 'la motte': 4, 'bohaire-dutheil': 2, 'lantier': 1, 'lebe
au de schosme': 1, 'pompignan': 1, 'legouvé': 4, 'le mierre': 3, 'dorneval': 1, 'le tellier':
1, 'le vayer de boutigny': 1, 'linant': 1, 'longepierre': 1, 'magnon': 1, 'boisrobert': 2, 'mai
ret': 1, 'maréchal': 1, 'marivaux': 33, 'boissy': 7, 'marmontel': 1, 'mathieu': 1, 'mercier':
3, 'merle': 1, 'desessarts': 1, 'molière': 30, 'molii¿½re': 1, 'plaute': 1, "soulas d'allainva
l": 2, 'moline': 2, 'françois-augustin paradis de moncrif': 1, 'nadal': 6, 'nericault-destouche
s': 1, 'nivelle de la chaussée': 5, 'ouville': 2, 'boucher': 1, 'ouzicourt': 2, 'pain': 2, 'rio
u': 1, 'palissot de montenoy': 5, 'bouilly': 1, 'patrat': 2, 'pellegrin': 1, 'pellet desbarreau
x': 1, 'piron': 5, 'guilbert-pixerécourt': 1, 'boursault': 11, 'plancher de valcour': 1, 'poins
inet de sivry': 3, 'poinsinet': 1, 'poisson': 2, 'pompigny': 1, 'prade': 1, 'pradon': 4, 'quina
ult': 15, 'euripide': 5, 'riccoboni père': 1, 'riccoboni': 1, 'romagnesi': 1, 'rochon  de chaba
nnes': 1, 'rosimond': 1, 'rotrou': 12, 'rousseau': 8, 'saint-aignan': 2, 'saint-evremond': 1,
'saint-priest': 1, 'sallebray': 1, 'saurin': 3, 'scarron': 10, 'scudery': 10, 'sedaine': 3, 'se
gur': 1, 'somaize': 2, 'taconet': 3, 'tanevot': 1, "tristan l'hermite": 5, 'urfé': 1, 'boyer':
12, 'vadé': 2, 'viau': 1, 'villiers': 2, 'voisenon': 1, 'voltaire': 38, 'racine': 12, 'quinte-c
urce': 1, 'justin': 1, 'plutarque': 1, 'arrien': 1, 'diodore de sicile': 1, 'virgile': 1, 'andr
ieux': 1, 'm. le chevalier de nantouillet': 1, 'tacite': 1, 'lucrèce': 1, 'pausanias': 1, "appi
en d'alexandrie": 1, 'aristophane': 1, 'sophocle': 1, 'garnier': 1, 'brécourt': 1, 'brueys': 1,
'palaprat': 1, "cailhava d'estendoux": 1, 'campistron': 8, 'carbon de flins': 2}
```

Q4 Can you count the number of plays per author?

This was already done in prior code and can simply be outputted:

In [4]:

```
print("Number of plays per author is given by: ", length_dict)
```

```
Number of plays per author is given by:  {'aigueberre': 4, 'carmontelle': 22, 'chabanon': 3, 'c
hamfort': 2, 'champmeslé': 1, 'chazet': 3, 'dubois': 1, 'gassicourt': 1, 'chevreau': 1, 'colard
eau': 1, 'colleville': 1, 'colle': 1, 'anseaume': 1, 'archambault': 2, 'coupigny': 2, 'crï¿½bil
lon': 1, 'crébillon': 9, 'artaud': 1, 'cubières-palmézeaux': 8, 'baptiste': 1, 'du moutier': 1,
'cyrano': 2, 'dalibray': 1, 'dancourt': 49, 'barante': 3, 'dufesny': 2, 'barbier': 1, 'barré':
1, 'desfontaines': 4, 'beaumarchais': 1, 'deshoulières': 2, 'desmarets de saint-sorlin': 1, 'de
sportes': 1, 'diderot': 3, 'donneau de visé': 3, 'beaunoir': 3, 'dorat': 2, 'gazon-doruxigné':
1, 'du bosc de montandre': 1, 'duche de vancy': 1, 'regnard': 27, 'dufresny': 16, 'du fresny':
4, 'bensérade': 2, 'dugazon': 1, 'dumaniant': 1, 'feriol de pont-de-veyle': 1, 'durant': 9, 'du
ryer': 14, 'bergasse': 1, "fabre d'eglantine": 1, 'favart': 1, 'beys': 1, 'florian': 5, 'folar
d': 1, 'fontenelle': 1, 'fuzelier': 5, 'legrand': 4, '\tlesage': 1, 'biancolelli pierre-françoi
s (1680-1734)': 2, 'genest': 1, 'gilbert': 1, 'gillet de la tessonerie': 2, 'gouges': 2, 'goull
inet': 1, 'biancolelli': 2, 'gresset': 1, 'gudin de la brenellerie': 1, 'guerin de bouscal': 2,
'gueullette': 14, 'guibert': 1, 'harny de guerville': 2, 'hauteroche': 1, "henault d'armoreza
n": 1, 'hoffmann': 2, 'jaquelin': 1, 'philidor': 1, 'la calprenede': 9, 'bideau de montigny':
1, 'la chapelle': 1, 'la croix': 1, "l'affichard": 1, 'la fontaine': 7, 'bievre': 1, 'lafont':
1, 'la forge': 1, 'la fosse': 2, 'alain': 1, 'blanc': 1, 'la grange chancel': 1, 'la harpe': 5,
'la motte': 4, 'bohaire-dutheil': 2, 'lantier': 1, 'lebeau de schosme': 1, 'pompignan': 1, 'leg
ouvé': 4, 'le mierre': 3, 'lesage': 2, 'dorneval': 1, 'le tellier': 1, 'le vayer de boutigny':
1, 'linant': 1, 'longepierre': 1, 'magnon': 1, 'boisrobert': 2, 'mairet': 1, 'maréchal': 1, 'ma
rivaux': 33, 'boissy': 7, 'marmontel': 1, 'mathieu': 1, 'mercier': 3, 'merle': 1, 'desessarts':
1, 'molière': 30, 'moliï¿½re': 1, 'plaute': 1, "soulas d'allainval": 2, 'moline': 2, 'françois-
augustin paradis de moncrif': 1, 'nadal': 6, 'nericault-destouches': 1, 'nivelle de la chaussé
e': 5, 'ouville': 2, 'boucher': 1, 'ouzicourt': 2, 'pain': 2, 'riou': 1, 'palissot de monteno
y': 5, 'bouilly': 1, 'patrat': 2, 'pellegrin': 1, 'pellet desbarreaux': 1, 'piron': 5, 'guilber
t-pixerécourt': 1, 'boursault': 11, 'plancher de valcour': 1, 'poinsinet de sivry': 3, 'poinsin
et': 1, 'poisson': 2, 'pompigny': 1, 'prade': 1, 'pradon': 4, 'quinault': 15, 'euripide': 5, 'r
iccoboni père': 1, 'riccoboni': 1, 'romagnesi': 1, 'rochon  de chabannes': 1, 'rosimond': 1, 'r
otrou': 12, 'rousseau': 8, 'saint-aignan': 2, 'saint-evremond': 1, 'saint-priest': 1, 'sallebra
y': 1, 'saurin': 3, 'scarron': 10, 'scudery': 10, 'sedaine': 3, 'segur': 1, 'somaize': 2, 'taco
net': 3, 'tanevot': 1, "tristan l'hermite": 5, 'urfé': 1, 'boyer': 12, 'vadé': 2, 'viau': 1, 'v
illiers': 2, 'voisenon': 1, 'voltaire': 38, 'racine': 12, 'quinte-curce': 1, 'justin': 1, 'plut
arque': 1, 'arrien': 1, 'diodore de sicile': 1, 'virgile': 1, 'andrieux': 1, 'm. le chevalier d
e nantouillet': 1, 'tacite': 1, 'lucrèce': 1, 'pausanias': 1, "appien d'alexandrie": 1, 'aristo
phane': 1, 'sophocle': 1, 'garnier': 1, 'brécourt': 1, 'brueys': 1, 'palaprat': 1, "cailhava
d'estendoux": 1, 'campistron': 8, 'carbon de flins': 2}
```