

Passerelle - Development #38237

factoriser simplify()

05 décembre 2019 17:36 - Nicolas Roche

Statut:	Fermé	Début:	05 décembre 2019
Priorité:	Normal	Echéance:	
Assigné à:	Nicolas Roche	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		

Description

Beaucoup de connecteur utilise un code similaire pour le paramètre (q) de recherche.

```
$ git grep "unicodedata.normalize('NFKD'" | cut -d: -f1 | sort | uniq
passerelle/apps/base_adresse/models.py
passerelle/apps/csvdatasource/models.py
passerelle/apps/gdc/views.py
passerelle/apps/solis/models.py
passerelle/apps/sp_fr/models.py
passerelle/contrib/agoraplus/normalize.py
passerelle/contrib/agoraplus/views.py
passerelle/utls/conversion.py
```

Révisions associées

Révision e4a063fa - 10 avril 2020 11:43 - Nicolas Roche

utls: factorize use of unicodedata module (#38237)

Historique

#1 - 09 décembre 2019 13:02 - Nicolas Roche

- Fichier 0001-utls-factorize-use-of-unicodedata-module-38237.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

Je me demande si je ne devrais pas utiliser six à la place d'unicode ?

```
<<<
if not isinstance(s, unicode):
    s = unicode(s, 'utf-8', 'ignore')
---
if not isinstance(s, six.text_type):
    s = six.text_type(s, 'utf-8', 'ignore')
>>>
```

#2 - 30 mars 2020 18:02 - Nicolas Roche

- Fichier 0001-utls-factorize-use-of-unicodedata-module-38237.patch ajouté
- Assigné à mis à Nicolas Roche

Rebasé.

La question ci-dessus ne se pose plus :

```
s = force_text(normalize(s))
```

#3 - 30 mars 2020 19:38 - Benjamin Dauvergne

J'ai un peu du mal avec les multiples passages ascii->unicode, tant qu'à factoriser on pourrait pas plutôt avoir des fonctions qui travaillent uniquement avec des unicode/str et convertir en sortie vers des bytes quand c'est nécessaire (parce que là ça fait mal à la tête).

#4 - 06 avril 2020 07:41 - Nicolas Roche

- Fichier 0001-utils-factorize-use-of-unicodedata-module-38237.patch ajouté

avoir des fonctions qui travaillent uniquement avec des unicode/str

oui, désolé pour le mal de tête.

#5 - 09 avril 2020 15:07 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée

Le gain ne me paraît pas terrible mais c'est surtout parce que les usages sont un peu trop divers (on a encore deux versions de simplify, tout ça parce que dans sp_fr on veut des '_' et pas des ') et les .decode('ascii', 'replace').encode('ascii') mériteraient des explications à chaque fois.

L'usage de la version locale de normalize() dans csvdatasource() me paraît louche en python3 (on devrait faire decode/encode comme ailleurs, on ne devrait pas générer des bytes dans les expressions), en fait en python2 pareil mais faudrait rendre implicite unicode_literals dans le eval() (voir <https://stackoverflow.com/questions/54224792/how-can-a-representation-of-a-literal-be-safely-evaluated-assuming-unicode-lite>).

#6 - 10 avril 2020 11:44 - Nicolas Roche

- Statut changé de Solution validée à Résolu (à déployer)

```
commit e4a063fac6ebe57b077c3846031c0bca956e1a82
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date: Mon Dec 9 12:27:55 2019 +0100
```

```
utils: factorize use of unicodedata module (#38237)
```

#7 - 15 avril 2020 14:17 - Frédéric Péters

- Statut changé de Résolu (à déployer) à Solution déployée

Fichiers

0001-utils-factorize-use-of-unicodedata-module-38237.patch	10 ko	09 décembre 2019	Nicolas Roche
0001-utils-factorize-use-of-unicodedata-module-38237.patch	8,58 ko	30 mars 2020	Nicolas Roche
0001-utils-factorize-use-of-unicodedata-module-38237.patch	8,48 ko	06 avril 2020	Nicolas Roche