

Authentic 2 - Development #32815

python3 : déprécier le builtin @file@

06 mai 2019 15:18 - Paul Marillonnet

Statut:	Fermé	Début:	06 mai 2019
Priorité:	Normal	Echéance:	
Assigné à:	Benjamin Dauvergne	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
N'existe plus en python3			

Révisions associées

Révision a516e02d - 19 juin 2019 09:38 - Paul Marillonnet

python3: define a FileType in tests/test_commands.py (#32815)

```
as file is not a builtin anymore
```

Révision 23145898 - 19 juin 2019 09:38 - Paul Marillonnet

python3: deprecate file builtin (#32815)

Historique

#1 - 06 mai 2019 15:30 - Paul Marillonnet

- Fichier 0002-python3-deprecate-file-builtin.patch ajouté
- Fichier 0001-python3-define-a-compatible-FileType.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

#2 - 06 mai 2019 15:43 - Thomas Noël

Met l'affaire "FileType" dans les tests directement, inutile de "pourrir" authentic.compat avec ça.

#3 - 06 mai 2019 16:15 - Paul Marillonnet

- Fichier 0002-python3-deprecate-file-builtin.patch ajouté
- Fichier 0001-python3-define-a-FileType-in-tests-test_commands.py.patch ajouté

Ok oui c'est vrai.

#4 - 06 mai 2019 16:37 - Paul Marillonnet

- Fichier 0002-python3-deprecate-file-builtin.patch ajouté
- Fichier 0001-python3-define-a-FileType-in-tests-test_commands.py.patch ajouté

On va y arriver.

#5 - 06 mai 2019 16:52 - Thomas Noël

- Statut changé de Solution proposée à Solution validée

Sounds good ; à pousser en un seul patch

Quand même, je ne pige pas trop le remplacement des simple « x=file.read » par des tordus « with open as f: x=f.read »

#6 - 06 mai 2019 16:59 - Paul Marillonnet

Thomas Noël a écrit :

Quand même, je ne pige pas trop le remplacement des simple « `x=file.read` » par des tordus « `with open as f: x=f.read` »

C'est très simple, les chances de voir le patch acké par Benj sont multipliées par 3 si on parvient à placer des gestionnaires de contexte `with` (ça permet aussi d'éviter une fermeture explicite du fichier, cette fermeture étant effectuée à la sortie du contexte.)

#7 - 06 mai 2019 17:11 - Thomas Noël

Paul Marillonnet a écrit :

(ça permet aussi d'éviter une fermeture explicite du fichier, cette fermeture étant effectuée à la sortie du contexte.)

Sauf erreur de ma part, `open().read()` ferme le fichier aussitôt après le `read`. Non ? Ah, je découvre que c'est de la magie CPython.

Ok pour le `with`, donc.

#8 - 06 mai 2019 17:15 - Paul Marillonnet

Ok oui, je pensais à un équivalent sans gestionnaire de contexte :

```
f = open(...)
f.read()
```

qui laisse des descripteurs de fichier ouverts, d'où, je crois, le conseil de Benj d'utiliser des gestionnaires de contexte.

#9 - 07 mai 2019 10:11 - Benjamin Dauvergne

Thomas Noël a écrit :

Sauf erreur de ma part, `open().read()` ferme le fichier aussitôt après le `read`. Non ? Ah, je découvre que c'est de la magie CPython.

La réponse c'est peut-être¹; mais avec `with` c'est certain et c'est comme ça qu'il faut faire et ça permet de prendre de bonnes habitudes de gestion des ressources en général, et en fait je trouve ça joli à force. Logiquement le compteur de référence doit tomber à zéro en sortie de scope et le descripteur de fichier être fermé, mais si il y a une exception dans `read()` ça peut ne pas le faire pour diverses raisons.

¹<https://stackoverflow.com/questions/4599980/python-close-file-descriptor-question>

#10 - 07 mai 2019 10:16 - Benjamin Dauvergne

ducktype powa ! remplacé `isinstance(x, file)` par `hasattr(x, 'read')`.

#11 - 19 juin 2019 09:54 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

#12 - 19 juin 2019 09:54 - Benjamin Dauvergne

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

```
commit 23145898bde4f26d51c8dfedba59f8a980dabf79
Author: Paul Marillonnet <pmarillonnet@entrouvert.com>
Date: Wed Apr 10 18:56:46 2019 +0200
```

```
python3: deprecate file builtin (#32815)
```

```
commit a516e02de09eff340f37b3ded2e59d7a289d66f2
Author: Paul Marillonnet <pmarillonnet@entrouvert.com>
Date: Wed Apr 10 19:20:46 2019 +0200
```

```
python3: define a FileType in tests/test_commands.py (#32815)
```

```
as file is not a builtin anymore
```

#13 - 21 juin 2019 09:15 - Frédéric Péters

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

Fichiers

0001-python3-define-a-compat.FileType.patch	1,24 ko	06 mai 2019	Paul Marillonnet
0002-python3-deprecate-file-builtin.patch	5,2 ko	06 mai 2019	Paul Marillonnet

0002-python3-deprecate-file-builtin.patch	4,91 ko	06 mai 2019	Paul Marillonnet
0001-python3-define-a-FileType-in-tests-test_commands.py.patch	1,38 ko	06 mai 2019	Paul Marillonnet
0002-python3-deprecate-file-builtin.patch	4,91 ko	06 mai 2019	Paul Marillonnet
0001-python3-define-a-FileType-in-tests-test_commands.py.patch	1,07 ko	06 mai 2019	Paul Marillonnet