

w.c.s. - Development #56614

|filter_value vs plusieurs champs avec le même identifiant

02 septembre 2021 22:27 - Frédéric Péters

Statut:	Fermé	Début:	02 septembre 2021
Priorité:	Normal	Echéance:	
Assigné à:	Frédéric Péters	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
Ça fait			
<pre>def apply_filter_value(self, value, exclude=False): assert self.pending_attr field = self.get_field(self.pending_attr)</pre>			
qui fait			
<pre>def get_field(self, key): for field in self._formdef.get_all_fields(): if getattr(field, 'varname', None) == key: return field</pre>			
et la recherche se trouve ainsi limitée au premier champ avec l'identifiant fourni.			
Il faudrait itérer sur tous les champs et faire un critère			
<pre>Or(f1 == value, f2 == value, f3 == value)</pre>			

Révisions associées

Révision 8fca1ed0 - 20 octobre 2021 15:09 - Frédéric Péters

misc: extend |filter_by to work with fields with non-unique varname (#56614)

Historique

#2 - 02 septembre 2021 22:27 - Frédéric Péters

- Description mis à jour

#3 - 02 septembre 2021 22:28 - Frédéric Péters

- Description mis à jour

#4 - 19 octobre 2021 16:46 - Frédéric Péters

- Fichier 0001-misc-extend-filter_by-to-work-with-fields-with-non-u.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Assigné à mis à Frédéric Péters

- Patch proposed changé de Non à Oui

Un peu plus chaotique que je n'imaginai, surtout parce qu'à tester la prise en compte correcte des colonnes NULLes est importante, ce qui a amené à remplacer un "Not(Equal(value))" en "NotEqual(value) or Null", ce qui change le comportement actuel en présence de NULL, vers quelque chose de correct.

En pratique dans les tests, c'est

```
- queryset = lazy_formdata.objects.filter_by('datefield').apply_exclude_value(...)  
- assert queryset.count == 1  
+ assert queryset.count == 6 # 1 + 5 null
```

précédemment les valeurs null étaient juste zappées, pas considérées différentes du critère.

~~

Bref il y a ça et puis la prise en compte de l'objet de ce ticket, un `get_fields` qui retourne tous les champs avec le varname demandé, puis la construction d'une liste de critères, `Or(field1 value, field2 value, field3 == value)` pour le cas basique.

Formellement pour le cas `!exclude` le choix est d'appliquer ça sur l'ensemble des champs, i.e. `field1 != value AND field2 != value AND field2 != value` (et non OR).

#5 - 20 octobre 2021 11:30 - Thomas Noël

- *Statut changé de Solution proposée à Solution validée*

Frédéric Péters a écrit :

Un peu plus chaotique que je n'imaginai, surtout parce qu'à tester la prise en compte correcte des colonnes NULLes est importante, ce qui a amené à remplacer un "Not(Equal(value))" en "NotEqual(value) or Null", ce qui change le comportement actuel en présence de NULL, vers quelque chose de correct.

Impecc.

Formellement pour le cas `!exclude` le choix est d'appliquer ça sur l'ensemble des champs, i.e. `field1 != value AND field2 != value AND field2 != value` (et non OR).

Choix tout à fait logique selon moi.

Rien à redire du tout.

Peut-être qu'il faudra faire ruisseler vers la doc que dans le cas de champ avec le même identifiant, ils doivent être du même type, au risque de planter sur ça (et ailleurs, peut être... mais de fait je ne sais pas si on a quelque part dans la doc une explication du fonctionnement de w.c.s. quand plusieurs champs ont le même identifiant...)

#6 - 20 octobre 2021 15:11 - Frédéric Péters

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

```
commit 8fcaled0b540fd64c694d559cab7e10259400b5a
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Tue Oct 19 15:18:42 2021 +0200
```

```
misc: extend |filter_by to work with fields with non-unique varname (#56614)
```

#7 - 20 octobre 2021 19:17 - Frédéric Péters

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

Fichiers

0001-misc-extend-filter_by-to-work-with-fields-with-non-u.patch	7,77 ko	19 octobre 2021	Frédéric Péters
---	---------	-----------------	-----------------