

Gestions Avancée admin - Actions

Les actions administrateurs

Les actions administrateur, accessibles depuis le menu Configuration > Gestion Avancée administration > Action administrateur, servent à lancer des méthodes liées à des calculs.

Les actions principales sont :

- All covers compute : recalcule des couvertures de stock de chaque produit
- Compute purchase received qty : recalcule des quantités reçus dans les lignes d'achats en fonctions des mouvements entrant
- Compute sale delivered qty : recalcule des quantités livrées dans les lignes de ventes en fonctions des mouvements sortant
- All disbursement compute : recalcule des taxes dans les paiements

Une image contenant texteDescription générée automatiquement Les actions fenêtres

Les actions-fenêtres sont nécessaires à l'affichages des éléments dans l'ERP : afficher une liste en cliquant sur un menu, afficher un formulaire, ouvrir un assistant de saisie... L'action-fenêtre permet de définir la ou les vues à afficher sur un objet.

Exemple : Pour les partenaires, on affiche une vue liste, une vue formulaire. Pour les factures, il est défini une vue liste, une vue formulaire, une vue calendrier et une vue graphe.

Les actions-fenêtre permettent également de restreindre les enregistrements en utilisant des domaines et des contextes.

Exemple : Client est un menu sur lequel l'action-fenêtre du même nom n'affiche que les partenaires dont le champ Client est vrai.

- **Objet source** : dans le cas d'un assistant, l'objet source est l'objet sur lequel l'assistant va agir.
Exemple : l'action-fenêtre « Modifier le mot de passe » est déclenchée par un bouton du même nom présent dans les formulaires d'utilisateur. L'objet source est donc `res.users`.
- **Utilisation de l'action** : ce champ est très peu utilisé, il est conseillé de le laisser vide.
- **Type d'action** : la valeur est toujours `ir.actions.act_window`. Cette liste ne référence en effet que les actions de type actions de fenêtre.
- **Fenêtre cible** :
 - Fenêtre courante : un bouton qui modifie le formulaire sur lequel vous êtes
 - Edition en ligne :
 - Nouvelle fenêtre : ouverture d'une nouvelle fenêtre
 - Window Stack :

L'onglet Paramètres Généraux

Cadre Vues

- **Type de vue** : formulaire ou en arborescence. Les vues arborescence sont les vues sous formes d'arbres.
Exemples de vue en arborescence : structure des emplacements (stocks), plan de taxes (comptabilité).
- **Mode de vue** : définit les différentes vues utilisées avec l'objet. Elles sont séparées par une virgule.
Exemples : `tree, form, calendar, graph`. Ces quatre actions se trouvent dans les actions-fenêtres des factures. Les vues des factures clients contiennent donc 4 icônes :
- **Ref. de vue** : nom de la [vue](#) affichée par défaut, parmi les vues existantes (à ne pas confondre avec le mode de vue). Mais dans la pratique, la liste en bas du formulaire est plus utilisée car elle permet de définir la vue pour chaque mode de vue.
- **Réf de vue de recherche** : choix de la [vue de recherche](#) associée à l'action.

Cadre Filtres

- **Valeur de domaine** : code python qui définit le filtre appliqué.
Dans l'illustration ci-dessus, l'action-fenêtre ne liste que les factures de type « `out_invoice` », autrement dit, les factures clients.
- **Valeur du contexte** : à la création d'un enregistrement depuis le menu lié à l'action-fenêtre, certains paramètres sont passés en contexte.

Dans l'illustration ci-dessus, à la création d'une facture, son type sera automatiquement « `out_invoice` » et elle sera affectée au journal de type « `sale` » (journal des ventes).

- **Limite** : nombre maximal d'enregistrements visibles dans la liste. Par défaut, il est fixé à 80.

- Rafraîchissement auto : champ technique réservé à l'éditeur d'Open-Prod.
- Recherche auto : à cocher pour exécuter les recherches automatiquement.
- Filtre : champ technique réservé à l'éditeur d'Open-Prod.
- Restreindre aux listes : champ technique réservé à l'éditeur d'Open-Prod.

Cadre Aide

Ce cadre permet d'ajouter un texte qui s'affichera dans la vue liste (tree) lorsqu'un aucun élément n'existe où que les filtres ne renvoient aucun résultat.

On retrouve des textes d'aide dans de nombreux sous-menus du menu Interface et du menu Comptabilité.

Cadre Vues

La liste des vues en fin de formulaire permet de définir la vue à appliquer à l'action-fenêtre pour chaque mode de vue.

Les actions serveurs

Une action-serveur désigne l'exécution d'une action dans le système. Cette action est activée à partir d'un déclencheur. Les déclencheurs sont par exemple des [actions automatisées](#), des signaux (dans les [transitions](#)) dans les flux de travail ou encore des actions définies dans le menu Action au-dessus de la vue liste ou formulaire.

Paramétrage

Pour créer une action-serveur, il faut compléter un certain nombre de paramétrages :

- Le nom de l'action tel qu'il apparaîtra dans les menus techniques
- Le modèle de base, c'est-à-dire l'objet sur lequel l'action s'exécutera
- La séquence spécifie l'ordre d'exécution des actions serveur dans l'action automatisée (du plus petit au plus grand). Si la séquence est égale à -99, l'action ne sera pas exécutée.
- L'action à effectuer parmi les celles proposées :
 - Execute python code : exécution d'une fonction Python à définir
 - Trigger a workflow signal : envoie un signal à un flux de travail et exécute un bouton d'un workflow dynamique
 - Run a client action : exécution d'une action client, à sélectionner
 - Create or Copy a new record : création ou duplication d'un enregistrement
- Write on a record : modification d'un enregistrement existant
- Execute several actions : exécution d'une liste d'actions-serveur
- Printing : impression d'un document
- Condition : l'exécution de l'action-serveur est soumise à une condition Python à définir.

Nom de l'action

De confirmée à Livrée

Ajouter dans le menu 'Plus'

Modèle de base

Stock picking

Condition

True

Action à effectuer

Trigger a Workflow Signal

Séquence

5

Signal de flux de travail

Aide

Modèle cible

☐ Utiliser le modèle de base de l'action
 ☒ Utilisez un champ relationnel sur le modèle de base

Champ Relation

Vente

Signal à déclencher

confirmed_to_delivered

Les onglets de paramétrage s'affichent en fonction de l'action à effectuer :

Onglet « Code Python »

- Champ libre dans lequel écrire la fonction python.
- Le "self" devient "model"

Exemple d'utilisation : envoyer un mail en automatique à la création d'un enregistrement.

Onglet « Signal de flux de travail »

- Utiliser le modèle de base de l'action : l'action agira sur le workflow du modèle de base (défini dans l'entête).
- Utiliser un champ relationnel sur le modèle de base : l'action agira sur le workflow du champ Relation. Il faut ensuite définir le signal (ou bouton) à déclencher.

Exemple d'utilisation : traiter une livraison fait évoluer le workflow d'une vente.

Onglet « Client »

- Sélectionner une action parmi l'objet des actions.

Exemple : ouvrir une fenêtre d'assistant à la validation d'une fiche produit.

Onglet « Créer / Ecrire / Copier »

- La règle de création détermine comment est créé le nouvel enregistrement. On distingue quatre options :

- Créer un nouvel enregistrement dans le modèle Base ;
- Créer un nouvel enregistrement dans un autre modèle ;
- Copier l'enregistrement courant ;
- Choisir et copier un enregistrement dans la base de données.
- Correspondance des valeurs : Valeur des champs à renseigner lors de la création / modification

Exemple d'utilisation : créer un événement pour un utilisateur lors de la conversion d'un devis en vente.

Onglet « Créer / Ecrire / Copier » (Write on a record)

- La politique de mise à jour détermine la méthode de sélection de l'enregistrement à mettre à jour :
- Mettre à jour l'enregistrement courant désigne automatiquement l'enregistrement à l'origine de l'action ;
- Mettre à jour un enregistrement lié à l'enregistrement courant en utilisant Python permet de définir un enregistrement lié à l'enregistrement d'origine, à l'aide d'une fonction Python ;
- Choisir et mettre à jour un enregistrement dans la base de données permet de sélectionner directement n'importe quel enregistrement de la base.
- Le champ Correspondances des valeurs permet de sélectionner les champs de l'enregistrement à mettre à jour, et d'en définir les nouvelles valeurs.

Exemple d'utilisation : modifier l'événement généré.
Il n'est pas possible d'utiliser de if ... else ...

Onglet « Exécuter plusieurs actions »

- Définir une liste d'actions serveurs à exécuter.

Exemple 1 : action serveur sur les ordres de fabrication, pour obtenir la liste des mouvements consommés et non consommés des matières premières lié a l'OF en cliquant sur un bouton

```
# action serveur de type python sur mrp.manufacturingorder
```

```
# Affichage des stock.move des MP des OFs sélectionnés
```

```
all_rm_rcs = env['stock.move']
```

```
for po_rc in objects:
```

```
for wo_rc in po_rc.workorder_ids:

    all_rm_rcs += wo_rc.rm_draft_ids + wo_rc.rm_done_ids

#view_id = env.ref('stock.view_tree_move').id

action = {

    'name': 'Matières premières',

    'view_type': 'form',

    'view_mode': 'tree,form',

    'res_model': 'stock.move',

    'type': 'ir.actions.act_window',

    'domain': [('id', 'in', all_rm_rcs.ids)],

    'target': 'current',

}
```

Exemple 2 : Action serveur avec syntaxe de date

Syntaxe récurrente :

- Renseigner des champs de type datetime via une expression python :
datetime.datetime.now().strftime('%Y-%m-%d %H:%M:%S')

Une image contenant texteDescription générée automatiquement

Revision #1

Created 3 November 2023 09:32:40 by Theotim COLIN DE VERDIERE

Updated 3 November 2023 13:29:26 by Theotim COLIN DE VERDIERE