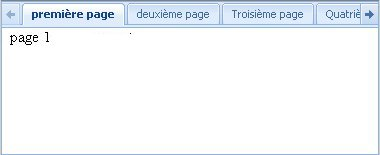
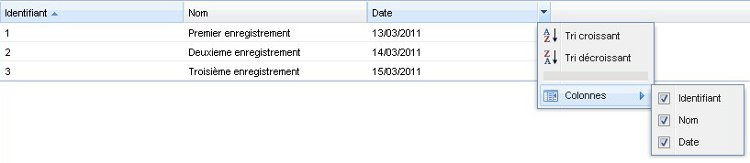
***Lexique :***

1. Le **ViewPort** permet d'afficher votre application en plein écran. Il se substitue à la balise body et se redimensionne automatiquement avec la fenêtre. Ce composant est idéal pour encapsuler toute votre application. Le Viewport s'affiche automatiquement au démarrage du programme.
2. Le **tabPanel** permet de créer des onglets à l'intérieur d'un panel.



1. Le **GridPanel** est la représentation d'un tableau en **ExtJs**.



1. Le **store** permet donc, via le proxy défini dans le modèle, de récupérer des données, de les stocker, de les filtrer (tris), etc. Le Store dans ExtJs est un conteneur d’enregistrement (Record). C’est cet objet que l’on manipulera pour charger les données depuis le serveur mais aussi pour accéder aux différents enregistrements.
2. **Ext.define** est la méthode pour créer ses propres classes dérivées d'une autre.
3. **Component :** Chaque composant est une sous-classe d'Ext.Component.

Composants de navigation :

**Ext.Toolbars** sont les plus utilisés généralement en tant qu'articles accouplés comme dans un Ext.Container. Ils peuvent être dessus ou bas accouplés utilisant la configuration accouplée. Ils te permettent d'insérer les articles (normalement Ext.Button) et d'ajouter également un titre.

**Ext.Button :** Il y a différents styles de bouton que vous pouvez créer à l'aide de l'icône, d'iconCls, d'iconAlign, d'ui, et de configurations des textes.

**Ext.TitleBar :** La principale différence entre un Ext.TitleBar et un Ext.Toolbar est que la configuration de titre est toujours centrée horizontalement dans un Ext.TitleBar entre les items qui sont à gauche ou à droite.

**SegmentedButton** est un récipient pour un groupe d'Ext.Buttons. Généralement un SegmentedButton serait un enfant d'un Ext.Toolbar et serait employé pour commuter entre différentes vues.

1. **Ext.menu.Item xtype: menuitem** Une classe pour toutes les commandes de menu qui exige la fonctionnalité liée au menu telle que le clic sélectionnant les sous-menus, icônes, etc.
2. Composants liés par store

**DataView** rend facile de créer un bon nombre de composants dynamiquement, habituellement basés d'un Ext.data.Store.

**List** est un DataView dénommé qui permet le groupement, l'indexation, les icônes, et une révélation.

1. Le **FormPanel** présente un ensemble de champs et fournit des manières de charger et sauvegarder des données.