**ODOO 视图层详解**

2017年11月14日

主要有四个视图：

菜单视图：把 数据模型——菜单——视图（tree、form） 连接起来

表单视图：创建、编辑数据模型所用视图。

列表视图：展示数据模型（显示数据）时使用。

搜索视图：制定odoo右上角对于当前数据模型的可搜索字段以及可用过滤器。

1）菜单

在定义了数据模型之后，我们要操作该模型。就需要把模型——菜单——视图 连接起来。这可以通过act\_window+menuitem来实现。

<act\_window>元素定义了一个客户端窗口动作，该动作将以列表和表单视图的顺序打开todo.task模型。 <menuitem>定义了一个顶级菜单项，调用前面定义的action\_todo\_task动作。

两个元素都包含了id属性。 此id属性也称为XML ID，它用于唯一标识模块内的每个数据元素，并且可以由其他元素使用来引用它。

<!-- 视图动作 -->

<act\_window id="action\_qingjia\_qingjiadan" //该标签的id

name="请假单" //视图名

res\_model="qingjia.qingjiadan" //视图要操作哪个数据模型：模块.数据模型

view\_mode="tree,form" /> //依次用什么视图来打开这个数据模型

<!-- 顶级菜单 -->

<menuitem name="请假" id="menu\_qingjia"/> //顶级菜单：出现在导航栏上

<!-- 二级菜单 --> //二级菜单：点击顶级菜单跳转到请假模块后，出现在左侧边栏。

<menuitem name="请假单" id="menu\_qingjia\_qingjiadan" parent="menu\_qingjia" action="action\_qingjia\_qingjiadan"/> //指定菜单的响应动作：依次用什么视图来操作数据模型

最后，把视图文件所在路径注册到manifest的data中：

'data': [

'views/views.xml', //注意：路径要全。

],

2）表单视图

所有的视图都存储在数据库中。我们在XML文件中声明一个描述视图的<record>元素，该模块在安装模块时将被加载到数据库中。

<record id="view\_form\_模块名" model="ir.ui.view">

<field name="name">表单名</field>

<field name="model">数据模型（模块.模型）</field>

<field name="arch" type="xml"> //重点：视图类型定义

<form>

<group> //表单中一列

<field name="name"/> //字段为数据模型中的字段内容

<field name="is\_done"/>

<field name="active"/>

</group>

</form>

</field>

</record>

业务凭证窗体视图（仿纸页风格）

此表单包含两个元素：<header>包含操作按钮，<sheet>包含数据字段。

Form视图可以添加按钮以完成特定动作。这些按钮可以打开一个新的包含Form表单的窗口或运行定义在模块中的Python函数。

它们可以定义在Form视图内的任意位置，但是对于文档形式的窗体，建议把它们放在<header>标签中。

<form>

<header> //操作按钮所在区域

<button name="按钮响应事件（定义在数据模型中）" type="动作的类型（执行的操作）" string="按钮显示文本" class="按钮样式" />

</header>

<sheet> //数据字段操作区域

<group name="group\_top" string="请假单">

<field name="name"/> //数据模型中的字段们

<field name="days"/>

<field name="startdate"/>

<field name="reason"/>

</group>

</sheet>

</form>

group标签（相当于div）

<group>标签允许组织Form表单里的内容。在<group>中放置<group>，可以创建一个两列的列表。Group标签使用时，建议定义它的name属性，这样可以更方便的让其它模块扩展它或者用于标识该列内容的显示位置。

使用group方便地安排内容的分布显示：

<sheet>

<group name="group\_top">

<group name="group\_left">//左边列

<field name="name"/>

</group>

<group name="group\_right">//右边列

<field name="is\_done"/>

<field name="active" readonly="1"/>

</group>

</group>

</sheet>

所以一个完整的form视图如下：

<form>

<header>//操作按钮区域

<button name="数据模型中定义的按钮事件" type="操作类型" string="按钮显示文本" class="按钮样式" />

</header>

<sheet>//数据字段区域

<group name="group\_top">//使用group进行布局

<group name="group\_left">

<field name="name"/>

</group>

<group name="group\_right">

<field name="is\_done"/>

<field name="active" readonly="1" />

</group>

</group>

</sheet>

</form>

Tab 分页效果：notebook标签

<form>

......

<notebook>

<page string="页名">

<field name="显示内容" nolabel="页号"/>

</page>

</notebook>

</form>

3）列表视图

查看模型时，将使用<tree>视图。 树视图能够显示按层级结构组织的行，但大多数时候，它们用于显示简单列表。

<record id="view\_tree\_数据模型名" model="ir.ui.view">

<field name="name">列表名</field>

<field name="model">模块.数据模型</field>

<field name="arch" type="xml"> //指明视图类型

<tree colors="可以指明下面字段值为何值时，对应行使用什么背景颜色（这是通过bootstrap来实现的）">

<field name="字段.."/>

<field name="字段.."/>

</tree>

</field>

</record>

4）搜索视图

在列表的右上角，Odoo显示一个搜索框。 它搜索的字段和可用的过滤器是由<search>视图定义的。

<record id="view\_filter\_数据模型名" model="ir.ui.view">

<field name="name">过滤器名</field>

<field name="model">模块.数据模型</field>

<field name="arch" type="xml"> //视图类型

<search> //搜索视图定义

<field name="可搜索字段"/> //可搜索字段定义

<filter string="过滤条件名" domain="[('字段','操作符',值)]"/> //过滤条件定义

</search>

</field>

</record>