

CREACIÓN DE UN MÓDULO EN ODOO9

Alberto Merino Chico

SISTEMAS DE GESTIÓN EMPRESARIAL IES Ribera del Tajo

En esta práctica final alcanzaremos a crear un módulo en nuestro ERP Odoo, esta vez en su novena versión.

Odoo interface showing the 'Policias' module. The breadcrumb is 'Policias / comisaria.policias,1'. The left sidebar shows 'Policias' and 'Patrullas'. The main area displays fields for 'Placa' (1423-47), 'Nombre' (SuperPoli), 'Edad' (25), 'Localidad' (Talavera de la Reina), and 'Patrulla'.

Odoo interface showing the 'Patrullas' module. The breadcrumb is 'Patrullas / comisaria.patruillas,1'. The left sidebar shows 'Patrulla', 'Nombre de la patrulla', 'Zona', and 'policia'. The main area displays fields for 'Patrulla' (12), 'Nombre de la patrulla' (PatrullaCanina), and 'Zona' (NORTE). Below these fields is a table with columns 'Placa', 'Nombre', 'Edad', 'Localidad', and 'Patrulla'. The table contains one row with values: 1452-786, PerroCop, 0, PerroVillie, and comisaria.patruillas,1.

Comenzaremos creando el documento `__init__.py`. En el cuál se definirá el modelo a implementar.

```
1 import gestioncomisaria
```

A continuación, en `__openerp__.py`, indicaremos las características del mismo.

```
1 {
2     'name': 'Gestion de Comisaria',
3     'description': 'Gestion de comisarias completo',
4     'author': 'Alberto Merino Chico',
5     'data': ['policias_view.xml', 'patrullas_view.xml'],
6     'application': True,
7 }
```

Ahora se definirán las clases que contendrán la información de las vistas.

```

1  -*- coding: utf-8 -*-
2  from openerp import models, fields, api
3
4  class Policias(models.Model):
5      _name = 'comisaria.policias'
6      placa = fields.Char('Placa', required = True, size = 8)
7      nombre = fields.Char('Nombre', required = True, size = 150)
8      edad = fields.Integer('Edad', required = True)
9      localidad = fields.Char('Localidad', required = True)
10     patrulla = fields.Many2one('comisaria.patrullas', 'Patrulla')
11
12     class Patrullas(models.Model):
13         _name = 'comisaria.patrullas'
14         idpatrulla = fields.Integer('Patrulla', required = True)
15         nombrepatrulla = fields.Char('Nombre de la patrulla', required = True)
16         zonapatrulla = fields.Char('Zona', required = True)
17         policiasenpatrulla = fields.One2many('comisaria.policias', 'patrulla', 'policia')
18

```

Y por último damos forma a los documentos de las vistas. Éstos serán iguales para ambas clases, obviamente cambiando sus campos y variables.

```

<openerp>
<data>
<act_window id = "action_policias" name="Policias" res_model="comisaria.policias" view_mode="tree,form"/>
<menuitem name="Policias" id = "menu_policias"/>
<menuitem name="Policias" id="menu_policias_policias" parent="menu_policias" action="action_policias"/>

<record model = "ir.ui.view" id="view_policias_form">
<field name="name">Policias</field>
<field name="model">comisaria.policias</field>
<field name="type">form</field>
<field name="arch" type="xml">
<form string="Policias">
<header></header>
<sheet>
<group name="group_top">
<group name="group_left">
<field name="placa"/>
<field name="nombre"/>
<field name="edad"/>
<field name="localidad"/>
<field name="patrulla"/>
</group>
</group>
</sheet>
</form>
</field>
</record>
<record model = "ir.ui.view" id="view_policias_tree">
<field name="name">Policias</field>
<field name="model">comisaria.policias</field>
<field name="type">tree</field>
<field name="arch" type="xml">
<tree colours="grey:active==True">
<field name="placa"/>
<field name="nombre"/>
<field name="edad"/>
<field name="localidad"/>
<field name="patrulla"/>
</tree>
</field>
</record>
</data>
</openerp>

```