

UT5 DESARROLLO DE MÓDULOS EN ODOO PARTE 5

AÑADIR MODELOS

Tabla de contenido

1. NUEVO MODELO. AUTOR	2
2. TAREA.....	4

1. NUEVO MODELO. AUTOR

Vamos a crear un nuevo modelo autor para más adelante relacionarlo con el modelo libros.

Lo mejor es que separemos los modelos en archivos distintos. Vamos a cambiar el nombre de models.py a libro.py y vamos a crear un nuevo modelo con el nombre autor.py

```
LIBRERIA_B
├── __pycache__
│   └── controllers
│       └── __pycache__
│           ├── __init__.py
│           └── controllers.py
└── models
    ├── __pycache__
    │   ├── __init__.py
    │   └── autor.py 1
    └── libro.py 1
```

```
1  from odoo import models, fields, api
2
3
4  class autor(models.Model):
5      _name = 'libreria_b.autor'
6      _description = 'Descripción del modulo autor'
7      # Define que el campo "nombre" será el que se muestre en vistas ManyZone/Many2many
8      _rec_name = 'nombre'
9
10     nombre = fields.Char(string="Autor", required=True)
11
```

Si creamos un nuevo modelo **SIEMPRE** tendremos que hacer lo siguiente ☺

1. Hacer el import del nuevo modelo

```
EXPLORER ...   __init__.py X  libro.py 1
OPEN EDITORS
LIBRERIA_B
__pycache__
controllers
__pycache__
__init__.py
controllers.py
demo
demo.xml
models
__pycache__
__init__.py
```

```
models > __init__.py
1  from . import libro
2  from . import autor |
```

2. Añadir la entrada en el fichero security/ir.model.access.csv

```
security > ir.model.access.csv
1  id,name,model_id:id,group_id:id,perm_read,perm_write,perm_create,perm_unlink
2  access_libreria_libro,libreria.libro,model_libreria_libro,base.group_user,1,1,1,1
3  access_libreria_autor,libreria.autor,model_libreria_autor,base.group_user,1,1,1,1
```

3. Crear la vista list

```
<record model="ir.ui.view" id="libreria.autor_list">
    <field name="name">Autores list</field>
    <field name="model">libreria.autor</field>
    <field name="arch" type="xml">
        <list>
            <field name="nombre" />
        </list>
    </field>
</record>
```

4. Crear la vista form (opcional).

Hasta ahora hemos creado siempre la vista de tipo `list` pero en este caso vamos a definir la vista de `form` para configurar el aspecto del formulario. Las vistas de tipo `form` siempre se crean automáticamente pero claro, no podemos personalizarlas.

```
<!-- Vista de formulario para autores -->
<record id="view_autor_form" model="ir.ui.view">
    <field name="name">librería.autor_form</field>
    <field name="model">libreria.autor</field>
    <field name="arch" type="xml">
        <form>
            <sheet>
                <group>
                    <field name="nombre"/>
                </group>
            </sheet>
        </form>
    </field>
</record>
```

5. Añadir el action para las vista list y form

```
<!-- actions opening views on models -->
<record model="ir.actions.act_window" id="libreria.autor_action_window">
    <field name="name">Listado de Autores</field>
    <field name="res_model">libreria.autor</field>
    <field name="view_mode">list,form</field>
</record>
```

6. Añadir la opción de menú

```
<menuitem name="Autores" id="libreria.menu_2_list"
parent="libreria.menu_1" action="libreria.autor_action_window"/>
```

2. TAREA

Crea el modelo autor de este ejemplo y un fichero de carga de datos para dar de alta 10 autores.