

# Sistemas de Gestión Empresarial

## Andreu Sanz Sanz

2  
D  
A  
M

2  
3  
I  
2  
4



# Tema VI.

## Fields bàsics i relacionals

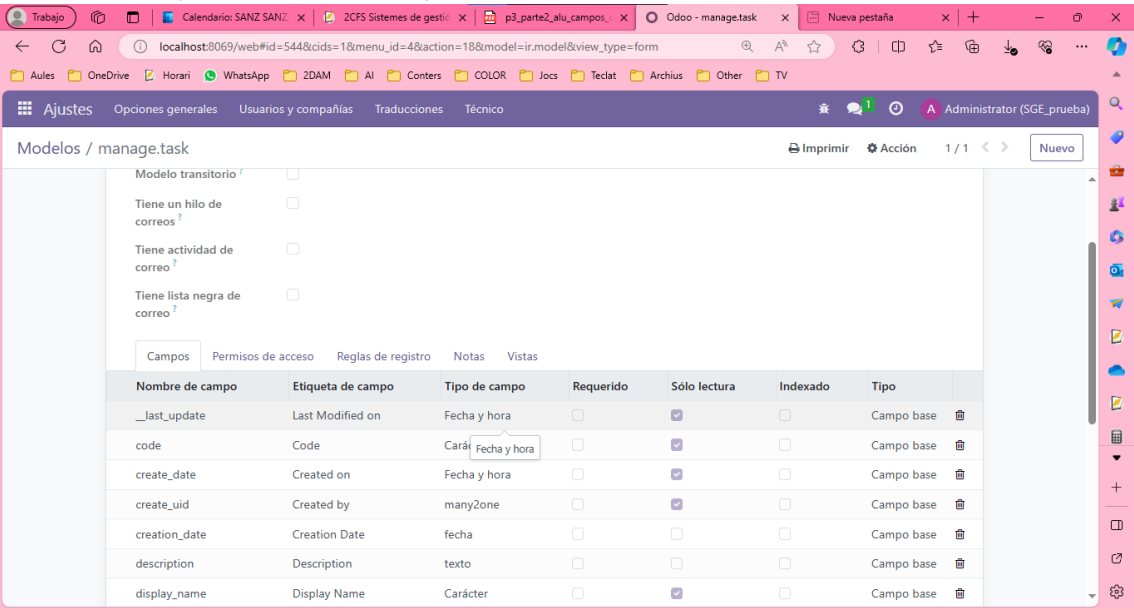
Sistemes de Gestió Empresarial | 2023-2024  
Andreu Sanz Sanz

### Tabla de contenidos

Objetivo 1 .....	2
1. Realiza captura del campo nuevo creado desde Odoo, en el apartado de Ajustes Técnico >> Modelos.....	2
2. ¿Cómo se crea una función en Python? ¿Y el bucle for? Realiza captura/s en Odoo de las pruebas realizadas para ver que la función realizada en Python es la correcta. ....	3
3. Explica con pseudocódigo las función <code>_get_code</code> creada. ....	3
Objetivo 2 .....	4
4. ¿Qué realiza la función <code>isinstance</code> ? ¿Y la función <code>timedelta</code> ? .....	4
5. Explica con pseudocódigo las función <code>_get_end_date</code> creada.....	4
6. ¿Para qué sirve <code>@api.depends</code> ? .....	4

# Objetivo 1

- 1. Realiza captura del campo nuevo creado desde Odoo, en el apartado de Ajustes Técnico >> Modelos.



Nombre ?

Description ?

Code ?

Start Date ?

Is Paused ? ☐

Technology ?

Creation Date ?

End Date ?

Sprint ?

Name

Agregar línea

<input type="checkbox"/>	Code	Nombre	Description
<input type="checkbox"/>	TSK_1	AD	Practica_06
<input type="checkbox"/>	TSK_2	Andreu	asdas

2. ¿Cómo se crea una función en Python? ¿Y el bucle for?  
Realiza captura/s en Odoon de las pruebas realizadas para ver que la función realizada en Python es la correcta.

```
class task(models.Model):    Missing class docstring
    _name = 'manage.task'    Bad indentation. Found 5 spaces, expected 4
    _description = 'manage.task'    Bad indentation. Found 5 spaces, expected 4
    name = fields.Char(string="Nombre",readonly=False, required=True, help="Introduzca el nombre de la tarea")
    description = fields.Text()    Bad indentation. Found 5 spaces, expected 4
    code = fields.Char(compute="_get_code")    Bad indentation. Found 5 spaces, expected 4
    creation_date = fields.Date()    Bad indentation. Found 5 spaces, expected 4
    start_date = fields.Datetime()    Bad indentation. Found 5 spaces, expected 4
    end_date = fields.Datetime()    Bad indentation. Found 5 spaces, expected 4
    is_paused = fields.Boolean()    Bad indentation. Found 5 spaces, expected 4
    sprint = fields.Many2one("manage.sprint")    Bad indentation. Found 5 spaces, expected 4
    technology = fields.Many2many("manage.technology")    Bad indentation. Found 5 spaces, expected 4
    Trailing whitespace
    def get_code(self):    # self es una colección    Bad indentation. Found 5 spaces, expected 4
        for task in self:    Bad indentation. Found 10 spaces, expected 8
            if len(task.sprint) == 0:    Bad indentation. Found 15 spaces, expected 12
                task.code = "TSK_" + str(task.id)    Bad indentation. Found 20 spaces, expected 12
            else:    Bad indentation. Found 15 spaces, expected 12
                task.code = str(task.sprint.name).upper() + "_" + str(task.id)    Bad indentation. Found 25 spaces, expected 12
```

Para crear una función en Python, utilizas la palabra clave **def**, seguida del nombre de la función, paréntesis que pueden contener parámetros (si los hay), y un bloque de código indentado.

El bucle **for** se crea utilizando la palabra clave **for**, seguida de una variable de iteración, la palabra clave **in**, y una secuencia iterable.

### 3. Explica con pseudocódigo la función `_get_code` creada.

Función `_get_code(self)`:

Para cada tarea en `self`:

Si la longitud de la tarea.sprint es igual a 0:

Establecer tarea.code como "TSK\_" seguido de la cadena convertida de tarea.id

SI no:

Establecer tarea.code como la cadena en mayúsculas de tarea.sprint.name seguido de "\_" y la cadena convertida de tarea.id

## Objetivo 2

### 4. ¿Qué realiza la función `isinstance`? ¿Y la función `timedelta`?

La función `isinstance` se utiliza para determinar si un objeto es una instancia de una clase específica o de cualquiera de las clases proporcionadas.

La función `timedelta` se utiliza para calcular el `end_date` según su fecha de inicio y duración.

### 5. Explica con pseudocódigo la función `_get_end_date` creada.

```
Función _get_end_date(self):  
    Para cada sprint en self:  
        Si sprint.start_date es una instancia de datetime y  
        sprint.duration es mayor que 0:  
            sprint.end_date = sprint.start_date + timedelta  
            días=sprint.duration  
        De lo contrario:  
            sprint.end_date = sprint.start_date
```

### 6. ¿Para qué sirve `@api.depends`?

Se utiliza para especificar las dependencias entre campos en modelos. Indica qué campos afectan al campo en el que se coloca y, por lo tanto, cuándo se debe actualizar el valor del campo dependiente. En este caso, la función se ejecutará cuando cambien las fechas de inicio (`start_date`) o la duración (`duration`) de un `sprint`.