Odoo

COMANDAMENTS & SNIPPETS

[**1) @api.onchange('field')**](#_7s60fba6q8kx) **2**

[**2) @api.constrains('field')**](#_iyx0ixsmu1zt) **3**

[**3) @api.depends('field')**](#_y8im8hbpket0) **4**

[**4) @api.one**](#_fbydge27p3cy) **5**

# 1) @api.onchange('field')

El **mètode** anotat s’executa quan hi ha un **canvi** en el **camp** passat com a paràmetre.

Snippet:

@api.onchange('fuel\_litres')

def \_check\_under\_fuel(self):

for record in self:

if record.fuel\_litres < 50:

record.under\_fuel = True

else:

record.under\_fuel = False

Si el camp 'fuel\_litres' es modifica, s’actualitza el valor de un altre camp 'under\_fuel'.

El bucle significa que per cada instancia de la classe, es comprova i el valor de 'fuel\_litres' i s’actualitza el de 'under\_fuel'. Açò és per a que es carreguen correctament les dades quan s’obri la vista amb varios objectes de la classe.

# 2) @api.constrains('field')

El **mètode** anotat s’executa quan es guarda un registre i **valida** el **camp** passat com a paràmetre.

Snippet:

@api.constrains('cv')

def \_validate\_cv(self):

if self.cv <= 0:

raise exceptions.ValidationError('El valor de CV no puede ser menor o igual a 0')

Si el camp 'cv' no compleix les restriccions, es llança una excepció.

# 3) @api.depends('field')

El **mètode** anotat s’executa quan hi ha un **canvi** en el **camp** passat com a paràmetre. Es sol utilitzar per a camps computats. Mentre que **@api.onchange** per a no computats.

Snippet:

@api.depends('fuel\_litres')

def \_check\_under\_fuel(self):

for record in self:

if record.fuel\_litres < 50:

record.under\_fuel = True

else:

record.under\_fuel = False

Si el camp 'fuel\_litres' es modifica, s’actualitza el valor de un altre camp 'under\_fuel'.

El bucle significa que per cada instancia de la classe, es comprova i el valor de 'fuel\_litres' i s’actualitza el de 'under\_fuel'.

Açò és per a que es carreguen correctament les dades quan s’obri la vista amb varios objectes de la classe.

Si volem bloquejar el camp per a que l’usuari no el puga modificar, s’afegeix compute = 'function' al camp.

**NOTA** -> El atribut compute no és exclusiu per a l’anotació @api.depends. Es pot utilitzar en qualsevol camp que volem que siga calculat pel sistema.

Snippet:

under\_fuel = fields.Boolean(string = 'Necessita repostar', compute = '\_check\_under\_fuel')

# 4) @api.one

El **mètode** anotat s’executa per a **tots** els registres de la **classe**.

Mètode sense @api.one:

Snippet:

@api.depends('fuel\_litres')

def \_check\_under\_fuel(self):

for record in self:

if record.fuel\_litres < 50:

record.under\_fuel = True

else:

record.under\_fuel = False

Mètode amb @api.one:

Snippet:

@api.one

@api.depends('fuel\_litres')

def \_check\_under\_fuel(self):

if record.fuel\_litres < 50:

record.under\_fuel = True

else:

record.under\_fuel = False