

onnet

2021
VIE

Odoo Models and Fields

www.on.net.vn
www.arrowhitech.com

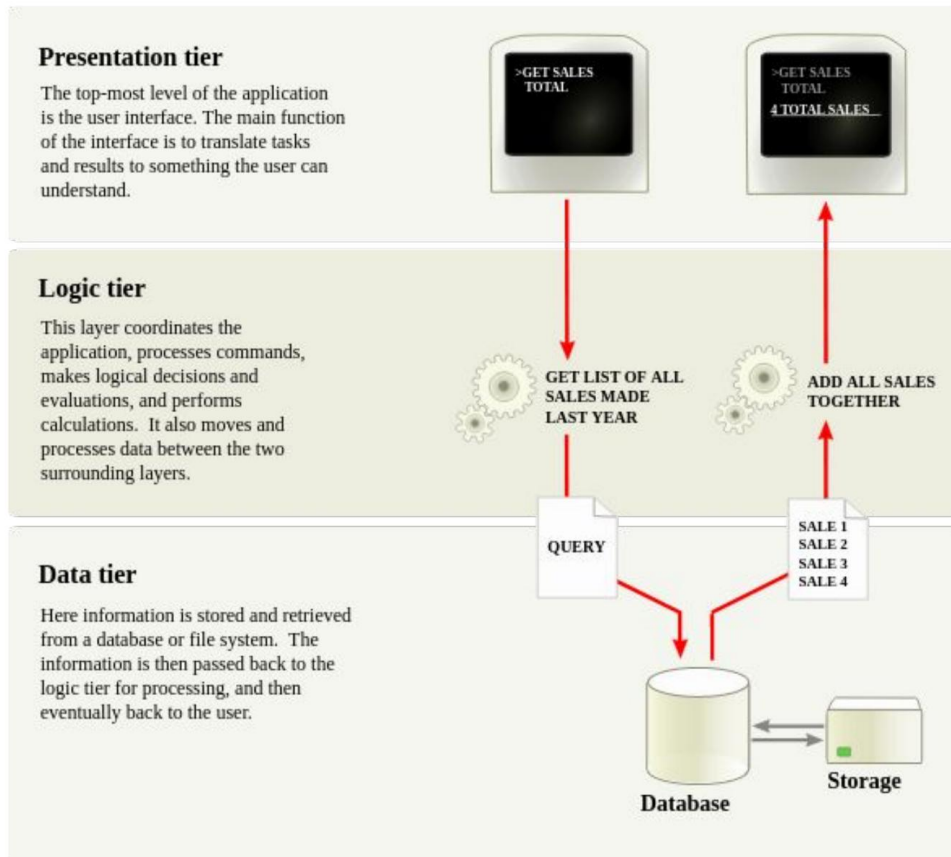


Table of contents

- **Architecture Overview**
- **Odoo module (addon)**
- **Models**
- **Fields**



Tổng quan về kiến trúc



Mô-đun Odoo (addon): Định nghĩa



- Mọi thứ trong Odoo đều bắt đầu và kết thúc bằng các mô-đun.
- Cả phần mở rộng máy chủ và máy khách đều được đóng gói dưới dạng mô-đun được tải tùy chọn trong cơ sở dữ liệu. Mô-đun là một tập hợp các chức năng và dữ liệu nhằm đến một mục đích duy nhất.
- Mô-đun Odoo có thể thêm logic nghiệp vụ hoàn toàn mới vào hệ thống Odoo hoặc thay đổi và mở rộng logic nghiệp vụ hiện có. Một mô-đun có thể được tạo để thêm các quy tắc kế toán của quốc gia bạn vào hỗ trợ kế toán chung của Odoo, trong khi một mô-đun khác có thể hỗ trợ thêm cho việc hình dung theo thời gian thực về đội xe buýt.
- Các nhà phát triển nhóm các tính năng kinh doanh của họ trong các mô-đun Odoo. Người dùng chính phải đối mặt các mô-đun được gắn cờ và hiển thị là Ứng dụng, nhưng phần lớn các mô-đun không phải là Ứng dụng. Các mô-đun cũng có thể được gọi là phần bổ trợ và các thư mục nơi máy chủ Odoo tìm thấy chúng tạo thành đường dẫn bổ sung.



Mô-đun Odoo (addon): Thành phần của một mô-đun



- Đối tượng nghiệp vụ: Một đối tượng nghiệp vụ (ví dụ: hóa đơn) được khai báo là một lớp Python. Các trường được định nghĩa trong các lớp này được tự động ánh xạ tới các cột cơ sở dữ liệu nhờ lớp ORM.
- Chế độ xem đối tượng: Xác định hiển thị giao diện người dùng
- Tập dữ liệu: tệp XML hoặc CSV khai báo dữ liệu mô hình: chế độ xem hoặc báo cáo, dữ liệu cấu hình (tham số hóa mô-đun, quy tắc bảo mật), dữ liệu trình diễn và hơn thế nữa
- Bộ điều khiển web: Xử lý các yêu cầu từ trình duyệt web

Cấu trúc mô-đun Odoo

- Mỗi mô-đun là một thư mục trong một thư mục mô-đun.
Các thư mục mô-đun được chỉ định bằng cách sử dụng tùy chọn `--addons-path`.
- Một mô-đun Odoo được khai báo bởi tệp kê khai của nó.
- Khi một mô-đun Odoo bao gồm kinh doanh các đối tượng (tức là tệp Python), chúng được tổ chức dưới dạng một gói Python với tệp `__init__.py`.
- Tệp này chứa các hướng dẫn nhập cho các tệp Python khác nhau trong mô-đun.

mô-đun

```

├─ mô hình
| └─ * .py
|   └─ __init__.py
├─ dữ liệu
|   └─ * .xml
├─ lượt xem
|   └─ * .xml
├─ __init__.py
└─ __manifest__.py
  
```

Mô hình: Mô hình là gì?



Mô hình xác định cấu trúc logic của cơ sở dữ liệu và về cơ bản xác định cách dữ liệu có thể được lưu trữ, tổ chức và thao tác.

Các mô hình có thể được định cấu hình bằng cách đặt các thuộc tính trong định nghĩa của chúng. Thuộc tính quan trọng nhất là `_name`, thuộc tính này được yêu cầu và xác định tên cho mô hình trong hệ thống Odoo .

Đây là định nghĩa tối thiểu về một mô hình:

từ các mô hình nhập khẩu của odoo

```
lớp TestModel (mô hình.Model):
```

```
    _name = "test.model"
```



Các loại mô hình

Các mô hình Odoo được tạo ra bằng cách kế thừa một trong những điều sau:

- Mô hình cho các mô hình cơ sở dữ liệu được duy trì thông thường
- TransientModel cho dữ liệu tạm thời, được lưu trữ trong cơ sở dữ liệu nhưng tự động được hút chân không thường xuyên như vậy
- AbstractModel cho các siêu lớp trừu tượng có nghĩa là được chia sẻ bởi nhiều kế thừa người mẫu

Lĩnh vực



Các trường được sử dụng để xác định những gì mô hình có thể lưu trữ và chúng được lưu trữ ở đâu.

Các trường được định nghĩa là thuộc tính trong lớp mô hình:

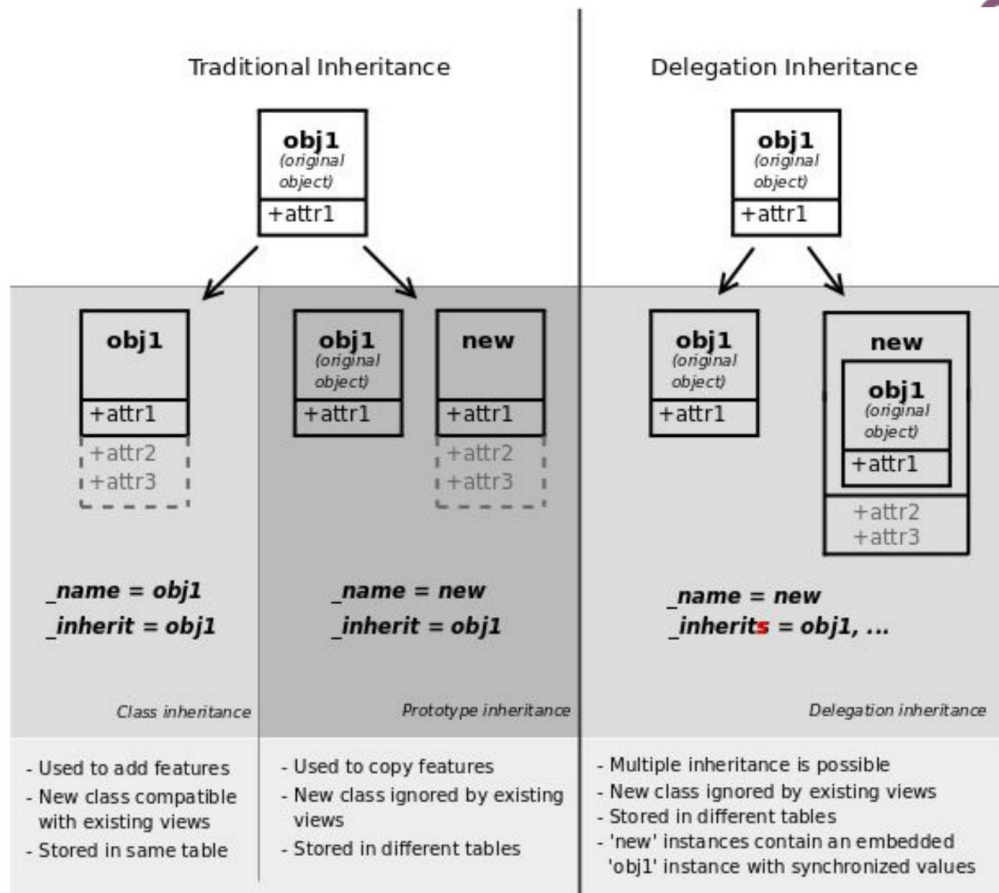
từ các trường nhập khẩu odoo, mô hình

```
class TestModel (models.Model): _name =  
    "test.model" _description = "Mô hình  
    thử nghiệm"  
  
    name = fields.Char ()
```

Kế thừa mô hình

Odoos cung cấp ba cơ chế khác nhau để mở rộng mô hình theo cách mô-đun:

- Kế thừa Cổ điển (`_name`, `_inherit`)
- Phần mở rộng (`_inherit`)
- Ủy nhiệm (`_inherits`)



Các trường cơ bản



Boolean: Một boolean (true, false).

Integer: Một số nguyên.

Float: Một số dấu phẩy động.

Char: Một chuỗi có độ dài giới hạn. Tham số kích thước yêu cầu xác định kích thước của nó.

Bản văn: Một trường văn bản không có giới hạn về độ dài.

Date: Một buổi hẹn hò.

Datetime: Cho phép lưu trữ ngày và giờ trong ngày trong cùng một trường.

Binary: Một chuỗi nhị phân

Lựa chọn: Một trường cho phép người dùng thực hiện lựa chọn giữa các giá trị được xác định trước khác nhau

Trường quan hệ



Mỗi cá thể mô hình là một “tập bản ghi”, tức là một tập hợp các bản ghi có thứ tự của mô hình.

- Many2one: Giá trị của trường như vậy là một tập bản ghi có kích thước 0 (không có bản ghi) hoặc 1 (một bản ghi duy nhất).
- One2many -
- Many2many - Lệnh

Trường tự động



Odoo tạo một vài trường trong tất cả các mô hình. Các trường này được quản lý bởi hệ thống và không thể được ghi vào, nhưng chúng có thể được đọc nếu hữu ích hoặc cần thiết:

- `id (Id)`: Định danh duy nhất cho bản ghi của mô hình.
- `create_date (Datetime)`: Ngày tạo bản ghi.
- `create_uid (Many2one)`: Người dùng đã tạo bản ghi.
- `write_date (Datetime)`: Ngày sửa đổi cuối cùng của bản ghi.
- `write_uid (Many2one)`: Người dùng sửa đổi bản ghi lần cuối.



Người giới thiệu

- Kế thừa và Mở rộng
- Mô hình
- Các lĩnh vực



Thực hành



Tạo một mô-đun mới. Trong mô-đun của mình, bạn nên:

- Tạo ít nhất một mô hình mới
- Kế thừa ít nhất một mô hình và thêm một trường mới
- Với một số trường

Bạn có thể làm theo ý tưởng của mô-đun học viện (bao gồm khóa học, sách, buổi học, sinh viên, ..) hoặc bạn có thể thử với ý tưởng của riêng mình.

Công việc của bạn đã hoàn thành khi bạn có thể cài đặt thành công mô-đun của mình

Một số lý tưởng cho các mô-đun mới:

- học viện: khóa học, sách, buổi học, người tham dự, người dẫn chương trình, ...
- bệnh viện: bệnh nhân, phòng, giường, ..
- thư viện: sách, thành viên, ...



Q&A