Créer un site web avec odoo

**Remarque :** Ce guide nécessite une notion du langage python, et que Odoo soit déjà installé.

# Créer un module basique

Avec odoo, les taches son … par la création des modules.

Pour créer un module basique, il faut utiliser la commande suivant :

**$ ./odoo.py scaffold Academy mesmodules**

Cette commande crée automatiquement le module «Academy » dans le dossier « mesmodules »

# Module de démonstration

Maintenant nous avons un module prêt pour l’installation.

Premièrement, démarrer le serveur odoo en utilisant la commande suivante :

**$ ./odoo.py --addons-path= «addons,mesmodules »**

Puis allez à l’adresse <http://localhost:8069> (il faut d’abord créer une base de données en incluant les données de démonstration)

Ensuite, il faut installer notre module. Pour ce faire, il faut aller dans **Modules,** puis enlever le filtre « installed (installé en français)» dans la zone de recherche. Et enfin cliquer sur installer.

# Pour le navigateur

Le contrôleur interprète les requêtes du navigateur et renvoie les données en retour.

Configurons notre contrôleur qui se trouve dans le fichier  **controller/controllers.py**

**controller/controllers.py :**

**# -\*- coding: utf-8 -\*-**

**from openerp import http**

**class Academy(http.Controller):**

**@http.route('/academy/academy/', auth='public')**

**def index(self, \*\*kw):**

**return "Hello, world"**

Maintenant, redemarrons notre serveur et allons dans l’adresse <http://localhost:8069/academy/academy/>

# Les templates

Odoo accepte n’importe quel template python pour le « Vue ». Mais propose son propre template qui est « QWeb ».

En utilisant un template, notre contrôleur devient :

**controller/controllers.py :**

**class Academy(http.Controller):**

**@http.route('/academy/academy/', auth='public')**

**def index(self, \*\*kw):**

**return http.request.render('academy.index', {**

**'teachers': ["Diana Padilla", "Jody Caroll", "Lester Vaughn"],**

**})**

# Enregistrer les données avec odoo

Le **Model** s’occupe du mappage des tables de la base de données

## Définir le modèle de données

Dans notre cas nous allons définir un modèle **teacher** dans le fichier **models/models.py**

**models/models.py :**

**from openerp import models, fields, api**

**class Teachers(models.Model):**

**\_name = 'academy.teachers'**

**name = fields.Char()**

Puis ajouter ceci dans **\_\_openerp\_\_.py**

**\_\_openerp\_\_.py :**

**'data': [**

**'security/ir.model.access.csv',**

**'templates.xml',**

**],**

Et ceci dans **security/ir.model.access.csv :**

**access\_academy\_teachers,access\_academy\_teachers,model\_academy\_teachers,,1,0,0,0**

C’est juste un droit d’accès.

## Accéder aux donnés

La dernière étape est de modifier la modèle et le template pour utiliser nos données de démonstration.

Notre contrôleur devient :

**controller/controllers.py :**

**class Academy(http.Controller):**

**@http.route('/academy/academy/', auth='public')**

**def index(self, \*\*kw):**

**Teachers = http.request.env['academy.teachers']**

**return http.request.render('academy.index', {**

**'teachers': Teachers.search([])**

**})**

**views/templates.py :**

**<template id="index">**

**<title>Academy</title>**

**<t t-foreach="teachers" t-as="teacher">**

**<p><t t-esc="teacher.id"/> <t t-esc="teacher.name"/></p>**

**</t>**

**</template>**

Maintenant, redémarrer notre serveur (tout en mettant à jour la base de données) et actualiser la page.

# Support web

Odoo dédie un module pour la création des sites web.

Pour l’utiliser, ajouter **website** dans le champ **depends** de « academy »,

**\_\_openerp\_\_.py :**

**'version': '0.1',**

**# any module necessary for this one to work correctly**

**'depends': ['website'],**

**# always loaded**

**'data': [**

Puis « website=true » dans le contrôleur

**controller/controllers.py :**

**from openerp import http**

**class Academy(http.Controller):**

**@http.route('/academy/academy/', auth='public', website=True)**

**def index(self, \*\*kw):**

**Teachers = http.request.env['academy.teachers']**

**return http.request.render('academy.index', {**

**views/templates.py :**

**<template id="index">**

**<t t-call="website.layout">**

**<t t-set="title">Academy</t>**

**<div class="oe\_structure">**

**<div class="container">**

**<t t-foreach="teachers" t-as="teacher">**

**<p><t t-esc="teacher.id"/> <t t-esc="teacher.name"/></p>**

**</t>**

**</div>**

**</div>**

**</t>**

**</template>**

# URL and Routing