

TP7 Python Django

Jinja2 & Template latex : Génération des documents

<https://docs.djangoproject.com/en/5.1/>

L3 - IFNTI Sokodé – 26 novembre 2024

Ce TP a pour objectif de générer les documents pdf à partir du moteur jinja2 et du template latex.

Jinja2 est un langage permettant de mettre en œuvre des templates. Il est très répandu dans l'écosystème python.

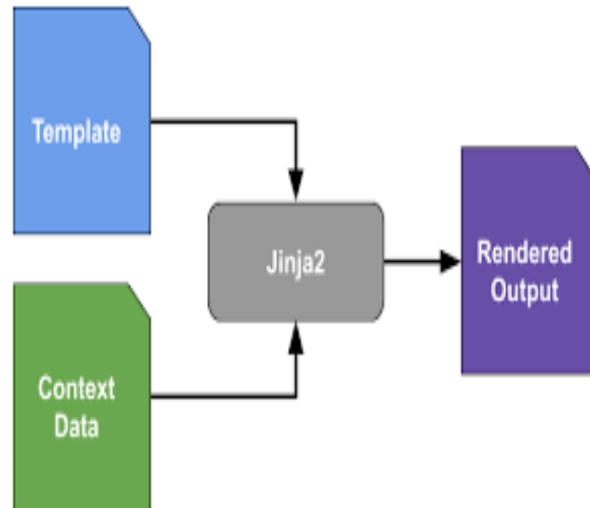
Les templates permettent de générer des documents texte avec du contenu généré dynamiquement. Les fichiers résultant peuvent être du HTML, du XML, du Json, du latex, ... ou quoi que ce soit qui soit du texte.

Plusieurs utilisations sont fréquentes :

- génération de page web
- génération de configuration (installation,...)
- génération de document à échanger (impression, import, export ...)

Plutôt que d'avoir dans votre code, des vastes portions réservées à des chaînes de caractères qui vont être copiées et modifiées, les langages de templates utilisent un fichier contenant le gabarit avec la logique d'insertion, laissant la partie python propre.

Jinja2 effectue la fusion entre le document de template et les données pour générer le document final.



En Jinja2, tout ce qui se trouve entre une double pair d'accolade est considéré par le moteur de rendu. Jinja2 va les interpréter comme des noms de variables à remplacer.

Il y a plusieurs façon de passer les données au template que ce soit du point de vue du format des données, que de l'invocation du moteur. Nous allons nous concentrer sur l'utilisation de python et l'utilisation d'un dictionnaire.

1 Mise en route

Installer les paquets suivants dans votre environnement via pip :

- pip install latex
- pip install jinja2
- sudo apt install texlive-lang-french

2 Création du module controleur et du template

1. Créer le dossier nommé Templating_ifnti à la racine du projet.
Créer en suite le fichier controleur.py dans ce dossier.
Ouvrez le fichier Templating_ifnti/controleur.py. Placez-y le code suivant :

```
def generate_pdf(context) :
    '''INSTANCIATION D'UN NOUVEL ENVIRONNEMENT
    AVEC DES OPTIONS DE BALISES PERSONNALISÉES'''
    j2_env = Environment(variable_start_string="\VAR{",
                        variable_end_string="}", block_start_string="\BLOCK{",
                        block_end_string="}", comment_start_string="\COMMENT{",
                        comment_end_string="}")

    '''DECLARATION DE FICHER'''
    #fichier à lire contenant le template avec les balises
    fichier_in = open("ifnti/liste_eleves.tex", 'r')

    #fichier en sortie accueillant les donnéesfournies
    fichier_out = open("out/template_out.tex", 'w')

    template = fichier_in.read() #lecture du template
    monContext = context
    monContext["image_path"] = dirname(abspath(__file__)) + "/out/images/"

    '''APPLICATION DE L'ENVIRONNEMENT EDITE SUR LE TEMPLATE'''
    j2_template = j2_env.from_string(template)
    # écriture dans le fichier en sortie
    fichier_out.write(j2_template.render(monContext))
    fichier_out.close()
    mon_pdf = build_pdf(open("out/template_out.tex", 'r'))
    mon_pdf.save_to("out/liste_eleves.pdf")
    '''FERMETURE DE CANAUX'''
    fichier_in.close()
```

Que fait ce code ?

2. Créer les dossiers nommés **ifnti** et **out** à la racine du projet.
 Créer en suite le fichier **liste_eleves.tex**(le template) dans le dossier **ifnti**.
 Ouvrez le fichier **ifnti/liste_eleves.tex**. Placez-y le code suivant :

```

\documentclass[10pt,a4paper]{article}
\usepackage[utf8]{inputenc}
\usepackage[français]{babel}
\usepackage[T1]{fontenc}
\usepackage{amsmath}
\usepackage{amsfonts}
\usepackage{amssymb}
\usepackage{makeidx}
\usepackage{graphicx}
\usepackage[left=2cm, top=3cm,bottom=2cm, right=2cm]{geometry}
\usepackage{multirow}
\usepackage{lmodern}
\usepackage{fmtcount}
\usepackage{tcolorbox}
\usepackage{fancybox}
\usepackage{color}
\usepackage[autolanguage,np]{numprint}
\usepackage{longtable}
\usepackage{phonenumbers}
\usepackage{telprint}
\graphicspath{ {\VAR{image_path}} }

\begin{document}

\begin{minipage}{12cm}
\textsf{REGION CENTRALE} \\\hspace{1cm}
\end{minipage}
\begin{minipage}{5cm}
REPUBLIQUE TOGOLAISE\\
Travail-Liberté-Patrie
\end{minipage}

\begin{minipage}{3cm}
\begin{center}
\includegraphics[scale=0.8]{logoIfnti.jpg} \\\hspace{1cm}

IFNTI-Sokodé TOGO\\300 BP 40
\end{center}
\end{minipage}

\begin{center}
\underline{\textbf{LISTE DES ETUDIANTS}}
\end{center}
\begin{longtable}{|l|l|l|l|l|} \hline
\textbf{Matricule} & \textbf{Nom} & \textbf{Prénom} & & \\
\textbf{Sexe} & \textbf{Date Naissance} & & & \\
\BLOCK{for elv in eleves}
\VAR{elv.matricule} & \VAR{elv.nom} & \VAR{elv.prenom} & & \\
\VAR{elv.sexe} & \VAR{elv.dateNais} & & & \\
\BLOCK{endfor}

```

```
\end{longtable}  
\end{document}
```

Créer le dossier nommé **images** dans le dossier **out**. Placer ensuite un logo de ifnti dans ce dossier. Le nom de l'image doit correspondre au nom indiqué dans le template.

3 Définition de vues django

Nos vues ici doivent produire les informations sous forme d'un dictionnaire(context) qui sera un argument de notre contrôleur generate_pdf()

1. Définir la vue **listEleves** permettant d'ouvrir un fichier .pdf quelconque . Copier votre code et insérer une capture d'écran de votre document
2. Modifier la vue **listEleves** pour que cette vue génère la liste de tous les élèves. Copier votre code et insérer une capture d'écran de votre document
3. Définir la vue **liste_niveauElv** permettant de générer la liste des élèves d'un niveau donné. Copier votre code et insérer une capture d'écran de votre document

4 Création de template, controleur et vue

1. Définir la vue **notesEleves** permettant de générer les notes des élèves d'une matière donnée. Copier votre code et insérer une capture d'écran de votre document
2. Définir la vue **notesSynthese** permettant de générer pour chaque élève sa moyenne dans chaque matière. Copier votre code et insérer une capture d'écran de votre document