

Pug (anciennement Jade) : *un moteur de rendu ‘concis’*

Pug est un moteur de templates implémenté en JavaScript qui permet de générer dynamiquement du HTML.

Sa syntaxe est inspirée de [Hamli](#) (HTML abstraction markup language). Elle est donc minimaliste et basée sur les indentations.

Avant, **Pug** s'appelait **Jade**. Seulement Jade est devenu une marque déposée, les responsables du dépôt sur Github ont choisi de le renommer en Pug.

Qu'est-ce que Pug?

Pour comprendre Pug, vous devez vous rappeler que le navigateur lit le code HTML et le code CSS, puis affiche des images et du texte formatés au client en fonction de ce que le code HTML et le code CSS lui indiquent.

Pug est un ‘intermédiaire’

Pug est un moteur de template pour Node.js.

Un moteur de template nous permet d'injecter des données puis de produire du HTML.

En bref: au moment de l'exécution, Pug -et d'autres moteurs de modèles- remplacent les variables de notre fichier par les valeurs réelles, puis envoient la chaîne HTML résultante au client.

Pug est un moteur de gabarit élégant, principalement utilisé pour la gabarit côté serveur dans NodeJS. En termes simples, Pug vous offre un nouveau moyen puissant d'écrire des balises, avec de nombreux avantages par rapport au HTML simple.

Utiliser Pug est très simple.

npm install pug --save

Configurez votre moteur de visualisation: `app.set('view engine', 'pug')`

Créer un fichier .pug

Alors, à quoi ressemble un fichier .pug?

C'est du html simplifié/indenté

Avec le fichier .pug suivant :

```
div
```

```
  p Hello!
```

```
  p World!
```

on obtient le HTML suivant :

```
<<div>
  <p>Hello!</p>
```

```
<p>World!</p>
</div>
```

Pug compile cela avec précision en traitant le premier mot de chaque ligne comme une balise, tandis que les mots suivants de cette ligne sont traités comme du texte à l'intérieur de la balise.

Un petit exemple ? Regardons ce que fait ce petit source .pug

(/index)

```
div
  p Salut
  p Les gars!
```

Puis celui-ci...

(/pug1)

```
div
  h1 Ocean's Eleven
  ul
    li Comedie
    li Policier
  p.
    Danny Ocean et onze complices
    braquent trois casinos simultanément.
```

Et enfin celui-là

(/pug2)

```
html
  head
    title= title
  body
    h1= message
```

et maintenant celui-ci

(/pug3)

```
div(class="movie-card", id="oceans-11")
  h1(class="movie-title") Ocean's 11
  img(src="https://s3-us-west-2.amazonaws.com/s.cdpn.io/123941/oceans11.jpg",
class="movie-poster")
  ul(class="genre-list")
    li Comedie
    li Policier
```

et le source node.js associé

```
var pug = require('pug');
var express = require('express');
var app = express();
app.set('view engine', 'pug'); //on délègue le moteur de rendu à pug
// par défaut, les fichiers .pug sont placés dans le répertoire views

/// sinon il faut fixer le répertoire des vues : app.set('views', './views')

app.get('/', function (req, res) {
  res.render('index');
})

app.get('/pug1', function (req, res) {
  res.render('pug1');
})

app.get('/pug2', function (req, res) {
  res.render('pug2', { title: 'Salut', message: 'Les gars de la licence Pro' });
});
app.listen(8080);
```

Un peu de JS dans le code PUG ? Facile...
(/pug4)

```
- var x = 5;
div
  ul
    - for (var i=1; i<=x; i++) {
      li= i + ". Hello"
    - }
```

Pug est implémenté avec JavaScript, il est donc très facile d'utiliser JavaScript dans Pug. **En commençant une ligne avec un trait d'union, nous indiquons au compilateur Pug que nous souhaitons commencer à utiliser JavaScript.**
(/pug5)

```
- var droids = ["R2D2", "C3PO", "BB8"];
div
  h1 Quelques Droids issus de Star Wars
  for name in droids
    div.card
      h2= name
```

(/pug6)

On peut aussi mélanger du javascript dans du texte, avec PUG rien de plus simple

```
- var profileName = "Danny Ocean";  
div  
  p Hi there, #{profileName}. How are you doing?
```

Vous trouverez la doc de base de Pug ici :

<https://pugjs.org/api/getting-started.html>

Une notion importante, les Mixins

Les mixins sont comme des fonctions. Ils prennent des paramètres en entrée et donnent un balisage en sortie. Les mixins sont définis à l'aide du mot clé mixin.

Une fois le mixin défini, vous pouvez appeler le mixin avec la syntaxe +
(/pug7)

```
mixin vignette(imageName, caption)  
  div.vignette  
    img(src=imageName style='width:200px; height:200px')  
    h4.image-caption= caption  
  
+vignette("https://s3-eu-west-1.amazonaws.com/darwinjocelyn/chat2.jpg", "Un joli  
chat.")  
+vignette("https://s3-eu-west-1.amazonaws.com/darwinjocelyn/chat1.jpg", "Encore un  
beau chat")
```

Travail à faire

Utilisez pug afin de présenter de façon élégante et concise chacun des CaptChat enregistrés dans votre base de données. Vous présentez : Le nom de l'artiste, le thème, le nom du jeu d'images, l'URL d'usage.