WordCamp Toulouse 2025

HTML To Gutenberg

Le développement de blocs Gutenberg accessible à tous.

WordCamp Toulouse 2025

Qui suis-je

Julien VERNEAUT

- Strasbourg
- Développeur freelance depuis 2018
- WordPress depuis 2016
- Travaille avec des agences webs pour des clients dans les secteurs du luxe, de l’éducation supérieure et de l’industrie
- WordPress, Drupal, Shopify, eco-système Node.js (Express/Fastify, React.js, etc.)

WordCamp Toulouse 2025

Pourquoi utiliser Gutenberg ?

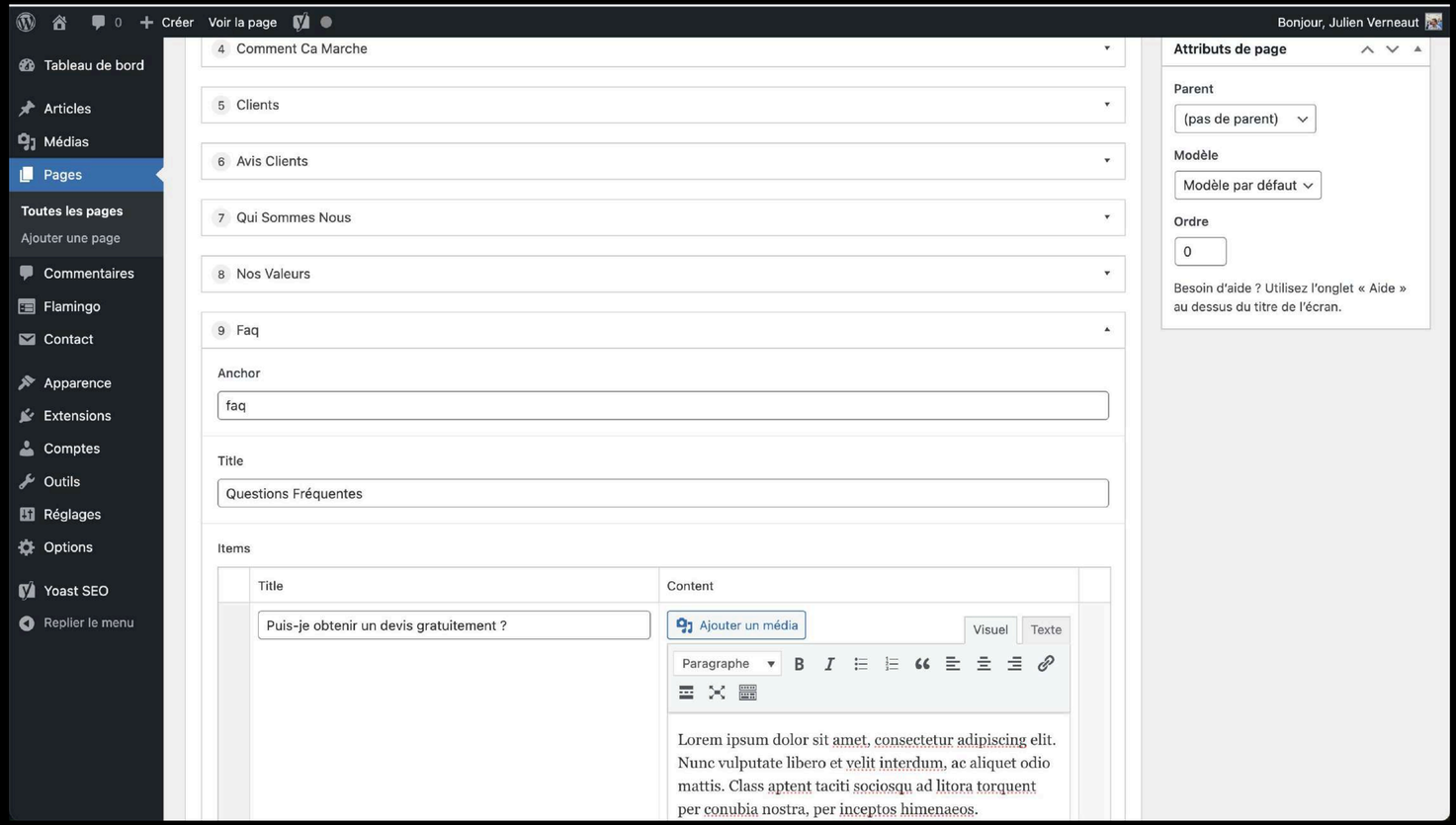
Avant Gutenberg

Les avantages

- Facilité de développement
- Éco-système mature (ACF, Bedrock, etc.)
- Séparation claire entre les données et leur représentation
- Évolutivité

Les inconvénients

- Liste de champs à rallonge
- **Pas de relation visuelle entre les données et leur représentation**



Après Gutenberg

Les avantages

- **Relation directe entre le contenu et sa représentation**
- WYSIWYG au sens pur

Les inconvénients

- Demande une réflexion plus importante au moment de la conception (blocks storming)
- Paradigme de développement différent
- Complexité opérationnelle (si développement de blocs customs)

WordCamp Toulouse 2025

Les développeurs adorent, n’est-ce pas ?

...n’est-ce pas ?

WordCamp Toulouse 2025

J’ai demandé à Reddit.

WordCamp Toulouse 2025

mattbeck ·

I have wat
or easy to

It's also qu

Tru

I tri

lear2000 · -3 a

Creating a custo
is a hot mess. I n
editor ftw.

10

r/Wordpress · il y a 4 a
Ceofreak

Why does Gutenberg suck so fucking hard?

Discussion

Let me elaborate... (This is a rant)

I've been running WordPress blogs since 2014 or so and always had been using the classic editor. I write very long tutorials on my website (ceos3c.com) with images and some of them are 5000+ words long. I never had any kind of issue using the classic editor, even with a 5000+ words article, editing was smooth.

I couple of months ago I decided to give Gutenberg a try because I like the idea of just dropping in blocks and keeping things simple. Also, just writing /h2 and stuff like this to create something speeds up my workflow tremendously.

I generally like how it's set up - it's pretty clean and intuitive. But here comes the problem: As soon as an article gets longer than ~1000 words, this fucking thing starts to get

Lire la suite

builder I know. It uses React,
r me, the pros surpass the
an know. Because I believe that

difficult and expensive to get
dency. I saw companies paying

y to change its structure.
r. Not a problem if that block
om block is on dozens of
one on my staging
t there, there are devs that
the reasons behind this
seems done on purpose.

little bit more time to learn
agine this will be solved in the
devs behind Gutenberg.
Because it's limiting the potential of this page builder, which I believe is a revolution for WordPress.

4 Répondre Récompenser Partager

WordCamp Toulouse 2025

...alors j'ai demandé à WordPress.

WordCamp Toulouse 2025

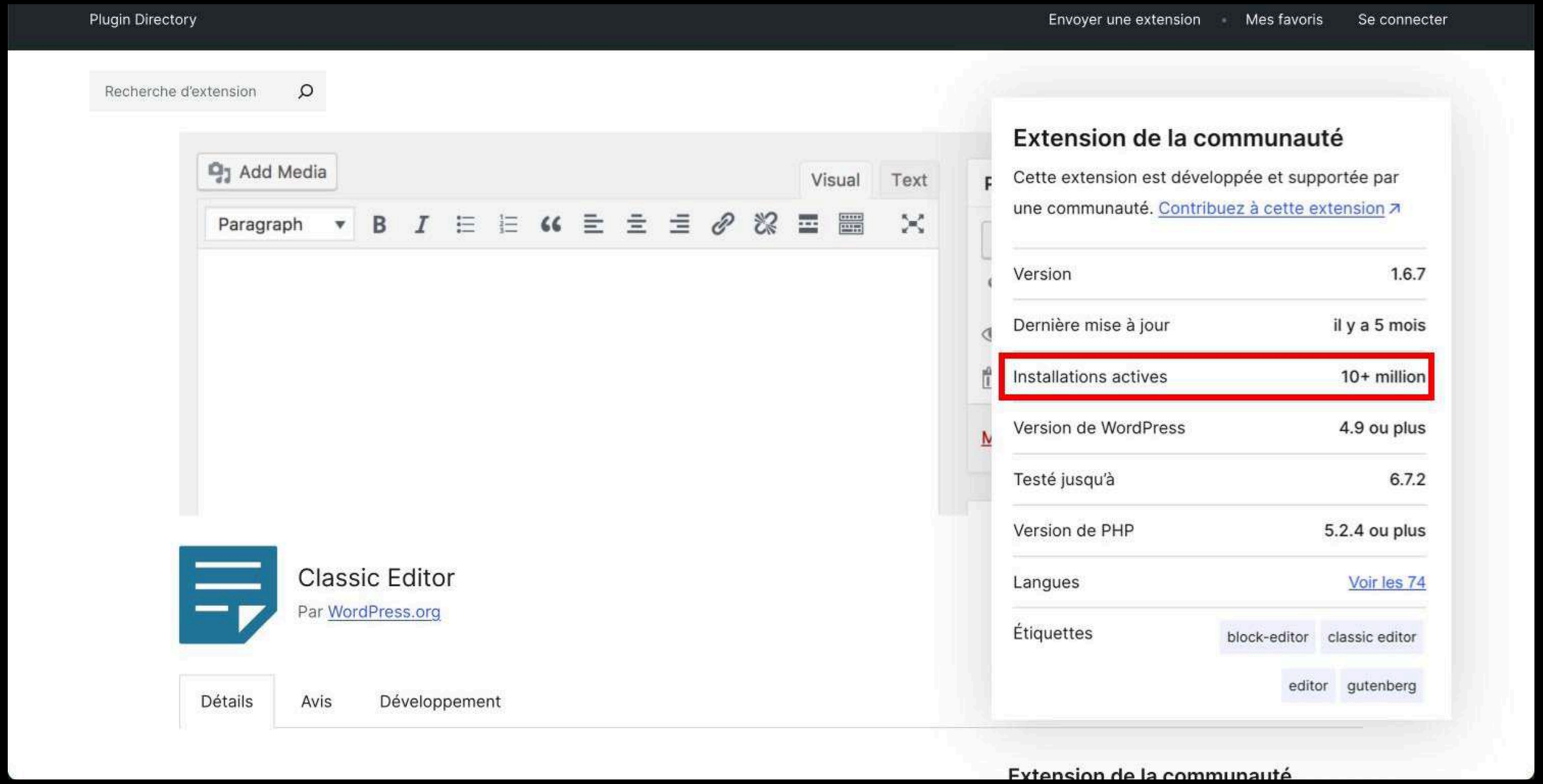
Forums

Welcome to Sup

Unreso

Review

Topic	Participants	Replies	Last Post
I dont know why wordpress has done this ★☆☆☆☆ Started by: kailashaddanki	1	0	2 weeks, 3 days ago kailashaddanki
If you want to blow up your site, instal Gutenberg ★☆☆☆☆ Started by: cristianmartinus	1	0	2 weeks, 4 days ago cristianmartinus
Turn it off! ★☆☆☆☆ Started by: Vika Oksana	1	0	3 weeks ago Vika Oksana
a joke! ★☆☆☆☆ Started by: Rabbi Islam rony	1	0	3 weeks, 1 day ago Rabbi Islam rony
Uma tragédia ★☆☆☆☆ Started by: dariocoutinho	1	0	4 weeks, 1 day ago dariocoutinho
Unintuitive ★☆☆☆☆ Started by: t2121	1	0	1 month ago t2121
even 1 star is too much ★☆☆☆☆ Started by: stardust666	1	0	2 months, 3 weeks ago stardust666
Should not be included by default ★☆☆☆☆	1	0	3 months ago nspirov


Plugin Directory
Envoyer une extension
Mes favoris
Se connecter
Recherche d'extension
Add Media
Visual
Text
Paragraph
Classic Editor
Par WordPress.org
Détails
Avis
Développement
Extension de la communauté
Cette extension est développée et supportée par une communauté. Contribuez à cette extension
Version 1.6.7
Dernière mise à jour il y a 5 mois
Installations actives 10+ million
Version de WordPress 4.9 ou plus
Testé jusqu'à 6.7.2
Version de PHP 5.2.4 ou plus
Langues Voir les 74
Étiquettes block-editor classic editor editor gutenberg
Extension de la communauté

WordCamp Toulouse 2025

Gutenberg, le mal aimé ?

WordCamp Toulouse 2025

~~Gutenberg, le mal aimé ?~~

Gutenberg, le mal compris

Un usage différent selon le type d'utilisateur

Pour les bloggeurs

Gutenberg offre une expérience d'édition native plus agréable que TinyMCE.

Il se rapproche de l'utilisation d'outils comme Notion, et est globalement bien reçu par cette communauté.

Utilisation des blocs natifs.

Pour les site builders

Il permet une souplesse dans la construction de sites plus poussée qu'auparavant avec l'utilisation de thèmes basés sur les blocs et l'utilisation du FSE.

Utilisation des blocs natifs, des blocs fournis par le thème et de blocs installés via des plugins.

Pour les agences/les développeurs

Il permet d'offrir à ses clients la meilleure expérience d'édition possible, mais il demande de repenser complètement sa méthodologie de conception de sites web.

Customisation des blocs natifs et développement de blocs personnalisés.

WordCamp Toulouse 2025

Les avantages de l’approche custom

Un site sans aucune extension tierces

Montées en version de WordPress simplifiées

Une expérience d’édition la plus intuitive possible

Un site performant sans surcouche additionnelle

WordCamp Toulouse 2025

Présentation live

À quoi ressemble un site construit avec des blocs Gutenberg personnalisés

WordCamp Toulouse 2025

Comment développer un bloc Gutenberg personnalisé

2 approches :

- Le développement de blocs statiques
- Le développement de blocs dynamiques

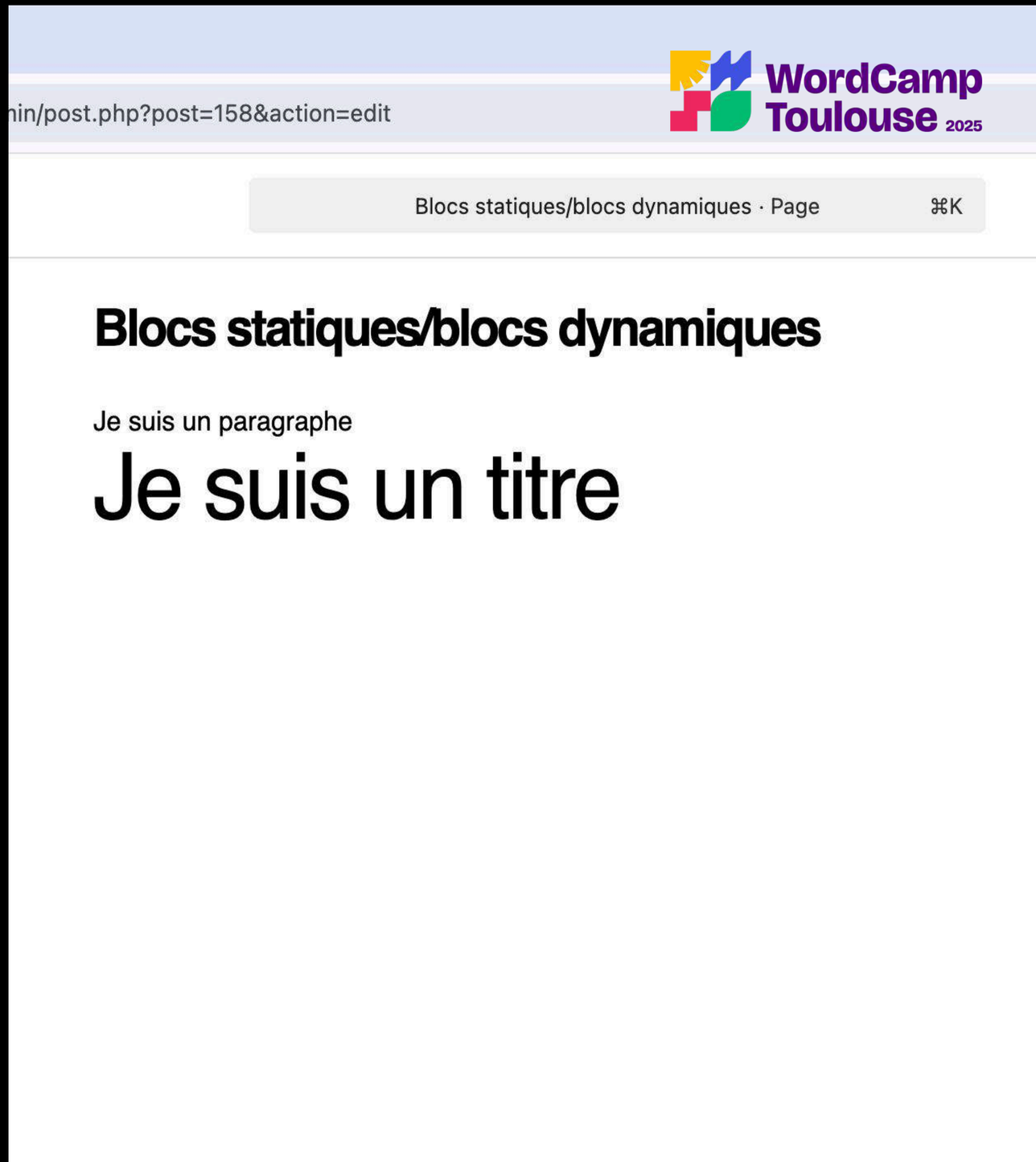
Les blocs statiques

Les blocs statiques stockent tout leur contenu et balisage directement dans le contenu du post (dans la base de données sous forme de HTML avec des commentaires de blocs). Lors du rendu du post en front-end, WordPress se contente d'afficher ce HTML sauvegardé.

Exemples de blocs statiques dans le core :

- core/paragraph
- core/heading
- core/button
- ...

Plus simples à développer, mais plus compliqués à faire évoluer.



Les blocs statiques

Les blocs statiques stockent tout leur contenu et balisage directement dans le contenu du post (dans la base de données sous forme de HTML avec des commentaires de blocs). Lors du rendu du post en front-end, WordPress se contente d'afficher ce HTML sauvegardé.

Exemples de blocs statiques dans le core :

- core/paragraph
- core/heading
- core/button
- ...

Plus simples à développer, mais plus compliqués à faire évoluer.

WordCamp Toulouse 2025

Blocs statiques/blocs dynam

```
<!-- wp:paragraph -->

<p>Je suis un paragraphe</p>

<!-- /wp:paragraph -->


<!-- wp:heading -->

<h2 class="wp-block-heading">Je suis un titre</h2>

<!-- /wp:heading -->
```

Les blocs dynamiques

Les blocs dynamiques ne stockent pas le contenu rendu du post. À la place, ils utilisent le rendu côté serveur (PHP) via render_callback lorsque le post est affiché. Le contenu est généré à la volée.

Exemples de blocs dynamiques dans le core :

- core/shortcode
- core/query
- core/latest-posts
- ...

Demande plus de développement, mais offre plus de souplesse et une mise à jour simplifiée.

WordCamp Toulouse 2025

Les bords de Garonne

Ici, l'eau écoute. Elle reflète le ciel, les rêves et les confidences.

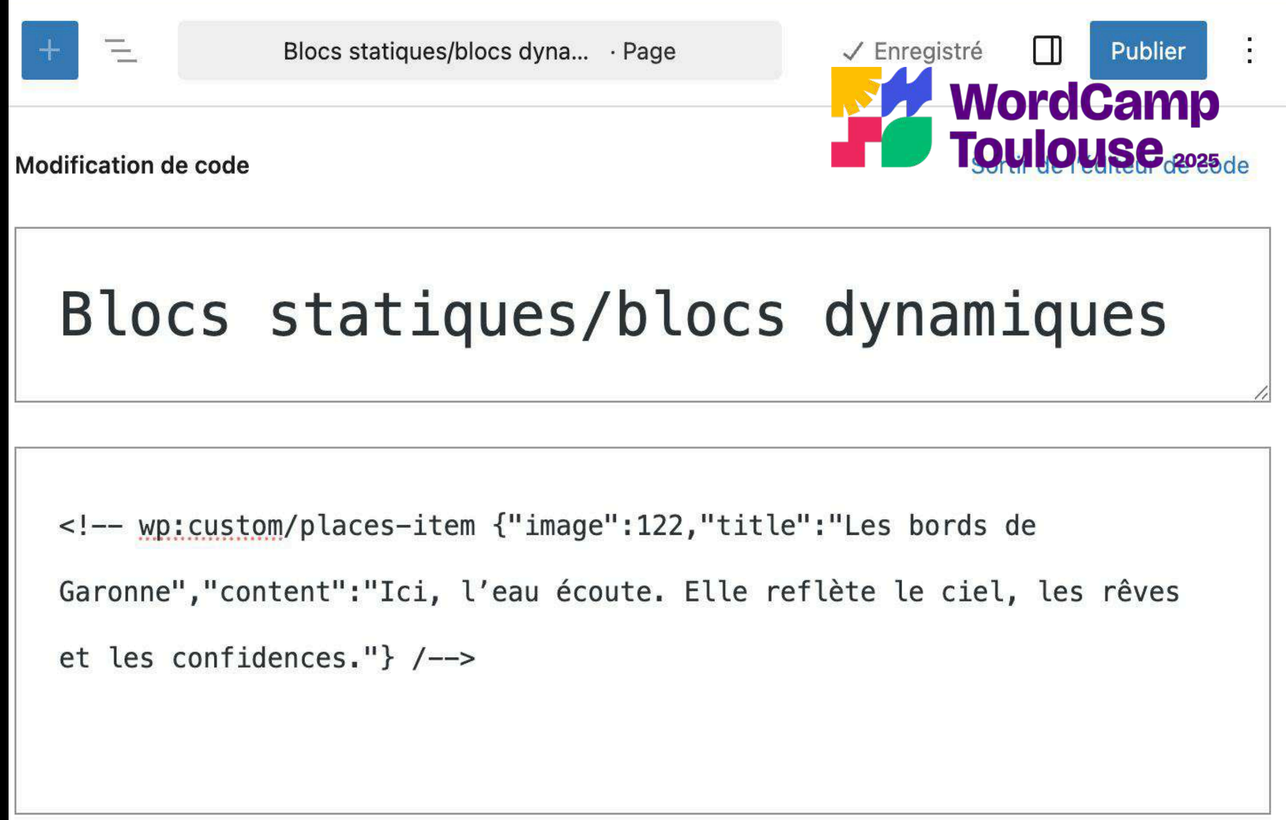
Les blocs dynamiques

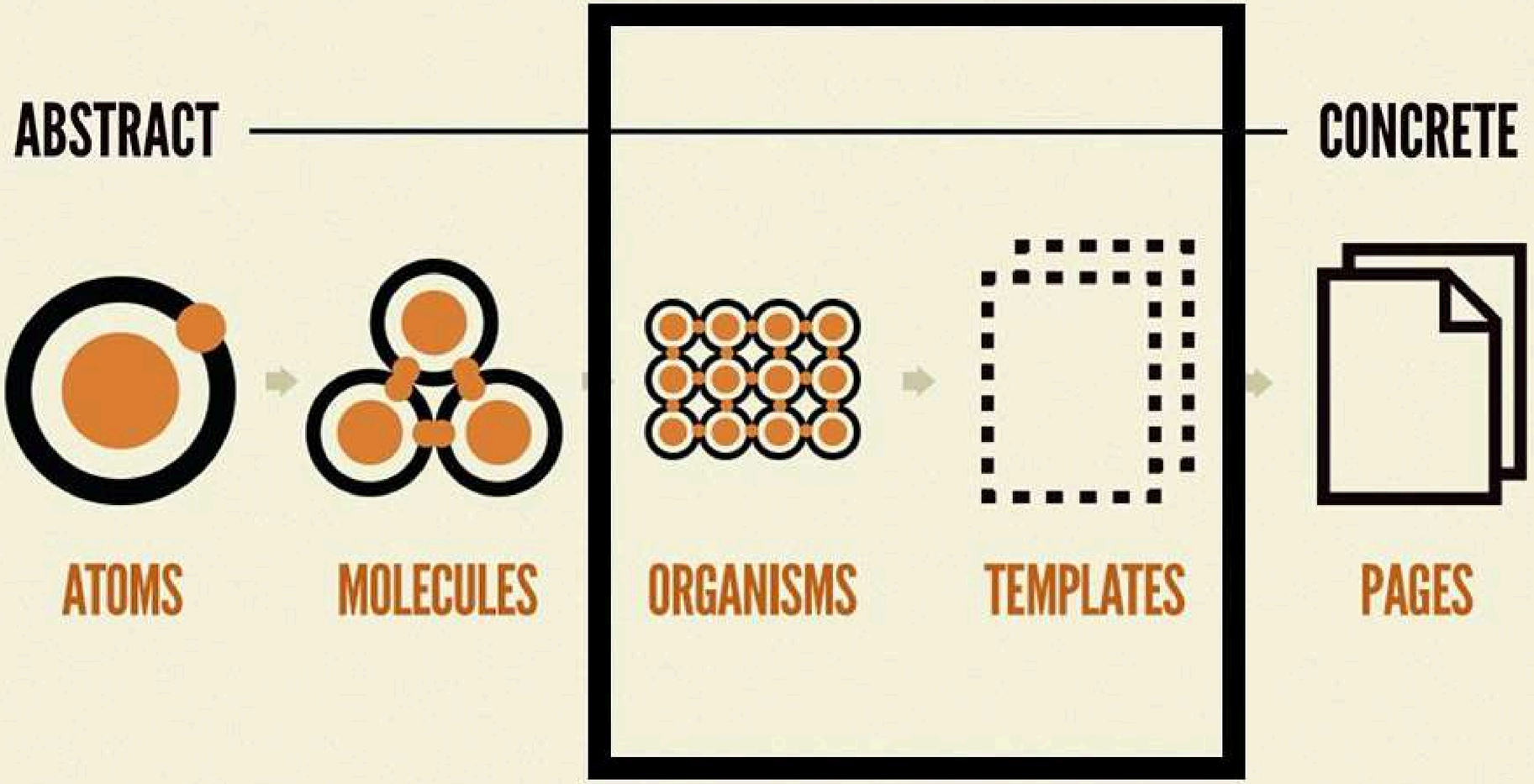
Les blocs dynamiques ne stockent pas le contenu rendu du post. À la place, ils utilisent le rendu côté serveur (PHP) via render_callback lorsque le post est affiché. Le contenu est généré à la volée.

Exemples de blocs dynamiques dans le core :

- core/shortcode
- core/query
- core/latest-posts
- ...

Demande plus de développement, mais offre plus de souplesse et une mise à jour simplifiée.



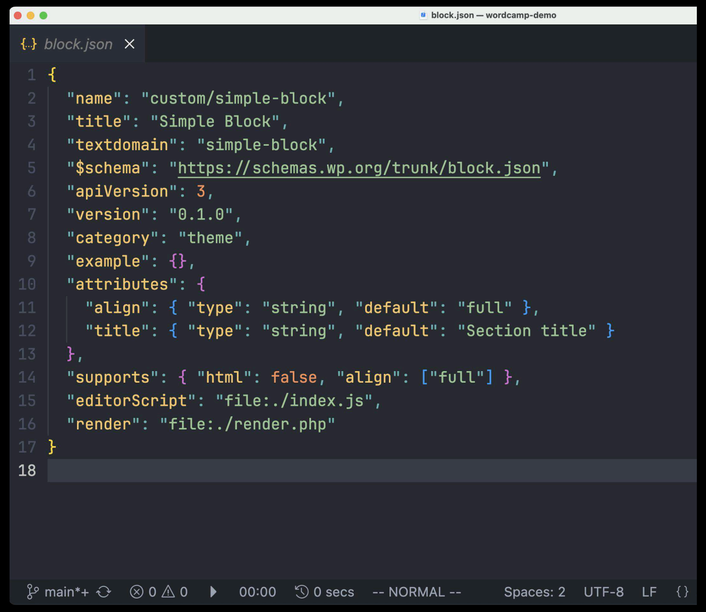


Blocs Dynamiques

WordCamp Toulouse 2025

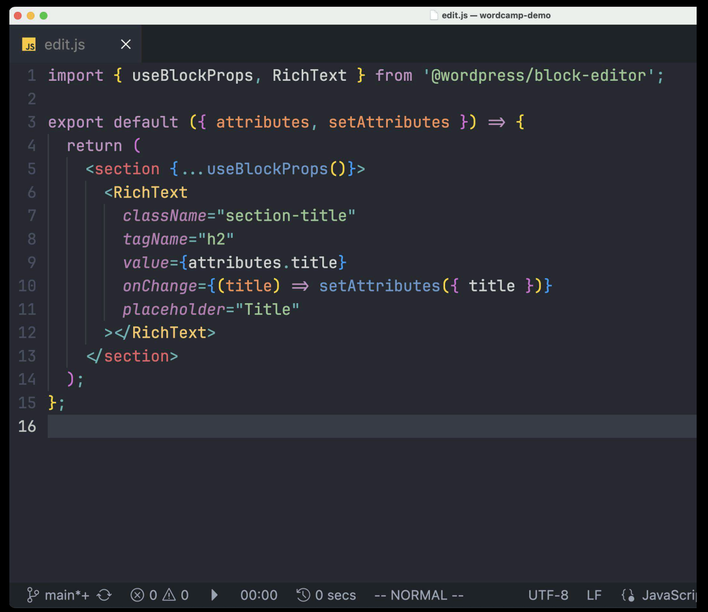
Mon conseil : privilégiez les blocs dynamiques pour la construction de blocs personnalisés.

Les composants d’un bloc dynamique



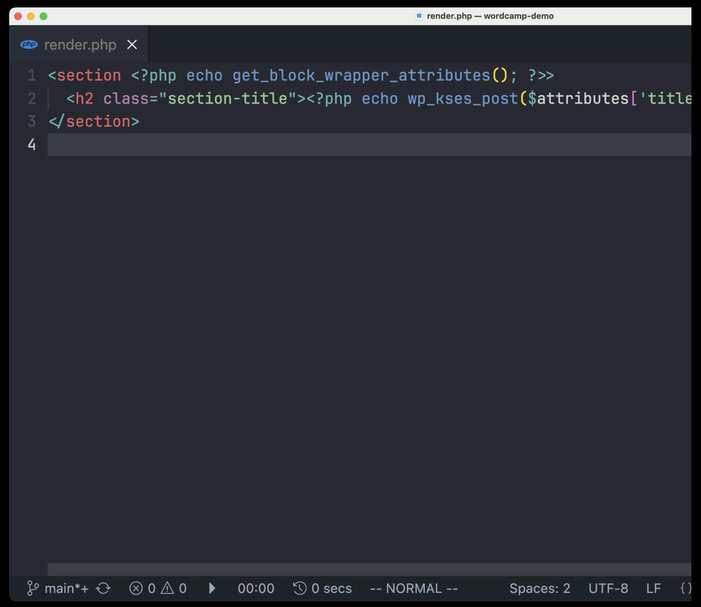
block.json

Défini les attributs du bloc : son nom machine, son titre, ses attributs éditables, etc.



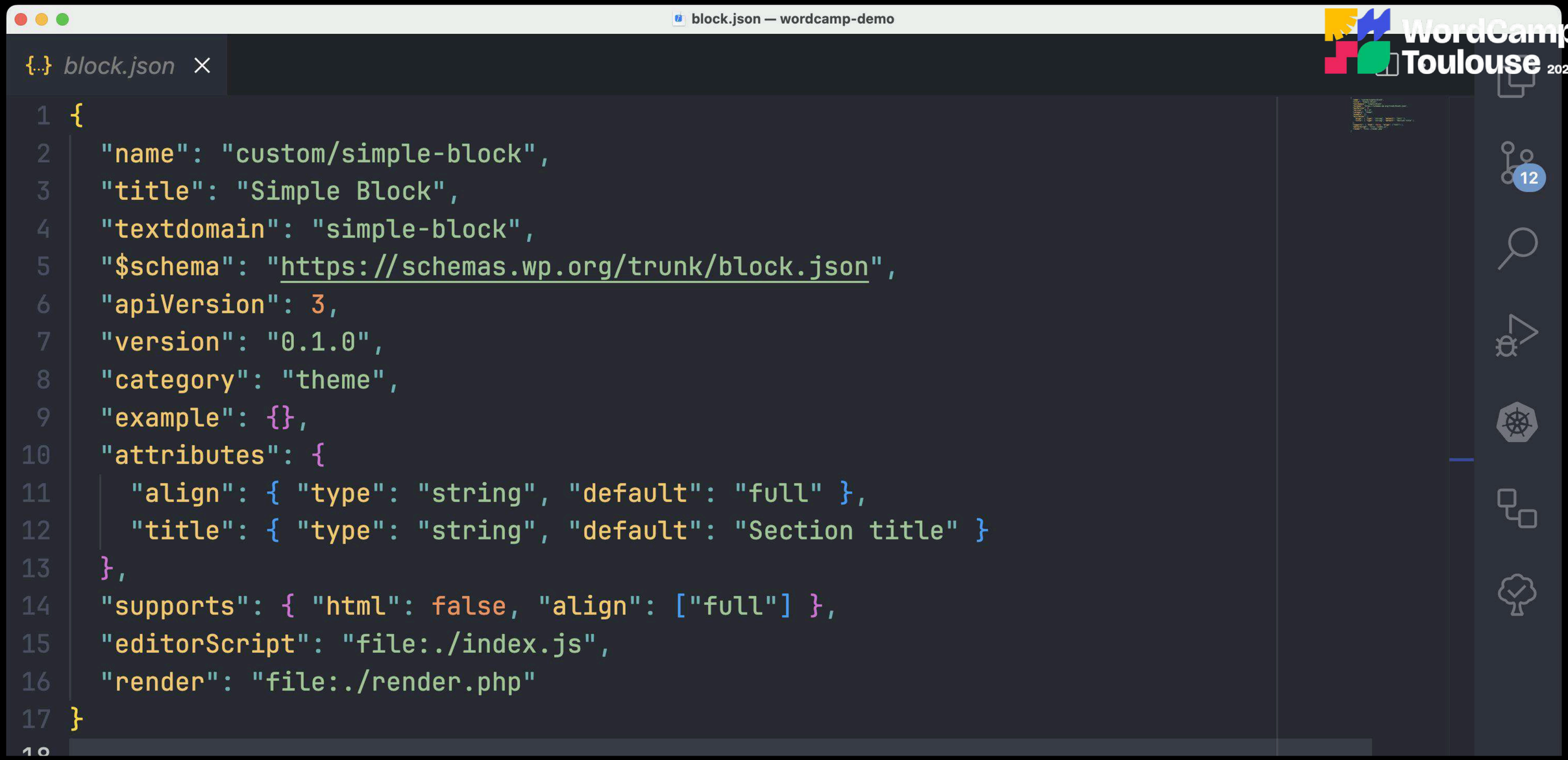
edit.js

Composant React.js responsable du rendu et de l’édition du bloc en back-office. Utilise les composants React.js fournis par WordPress.



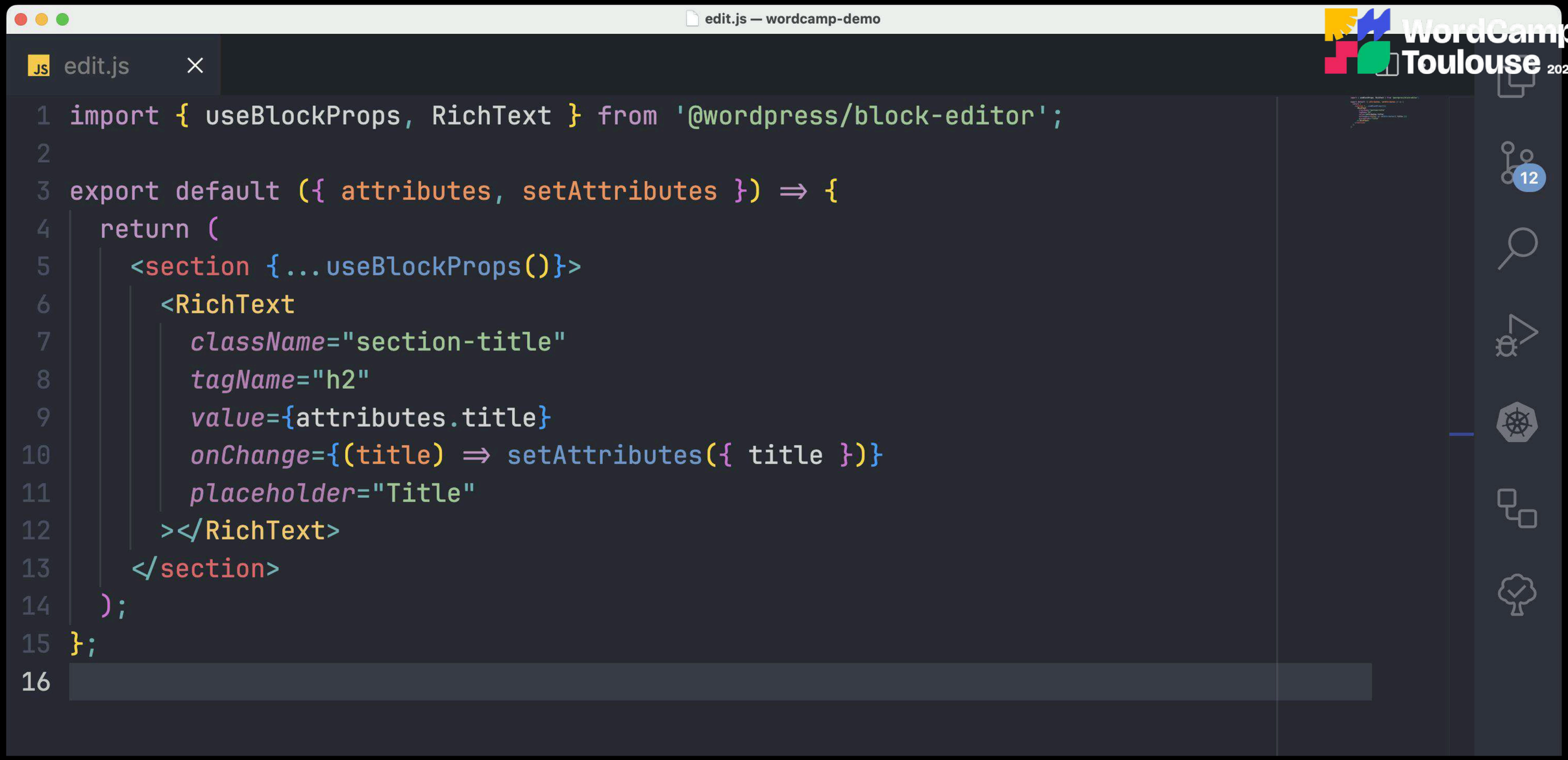
render.php

Fichier de rendu front-end du bloc. Il consomme les attributs définis dans le fichier block.json et édités en back-office via le composant edit.js.



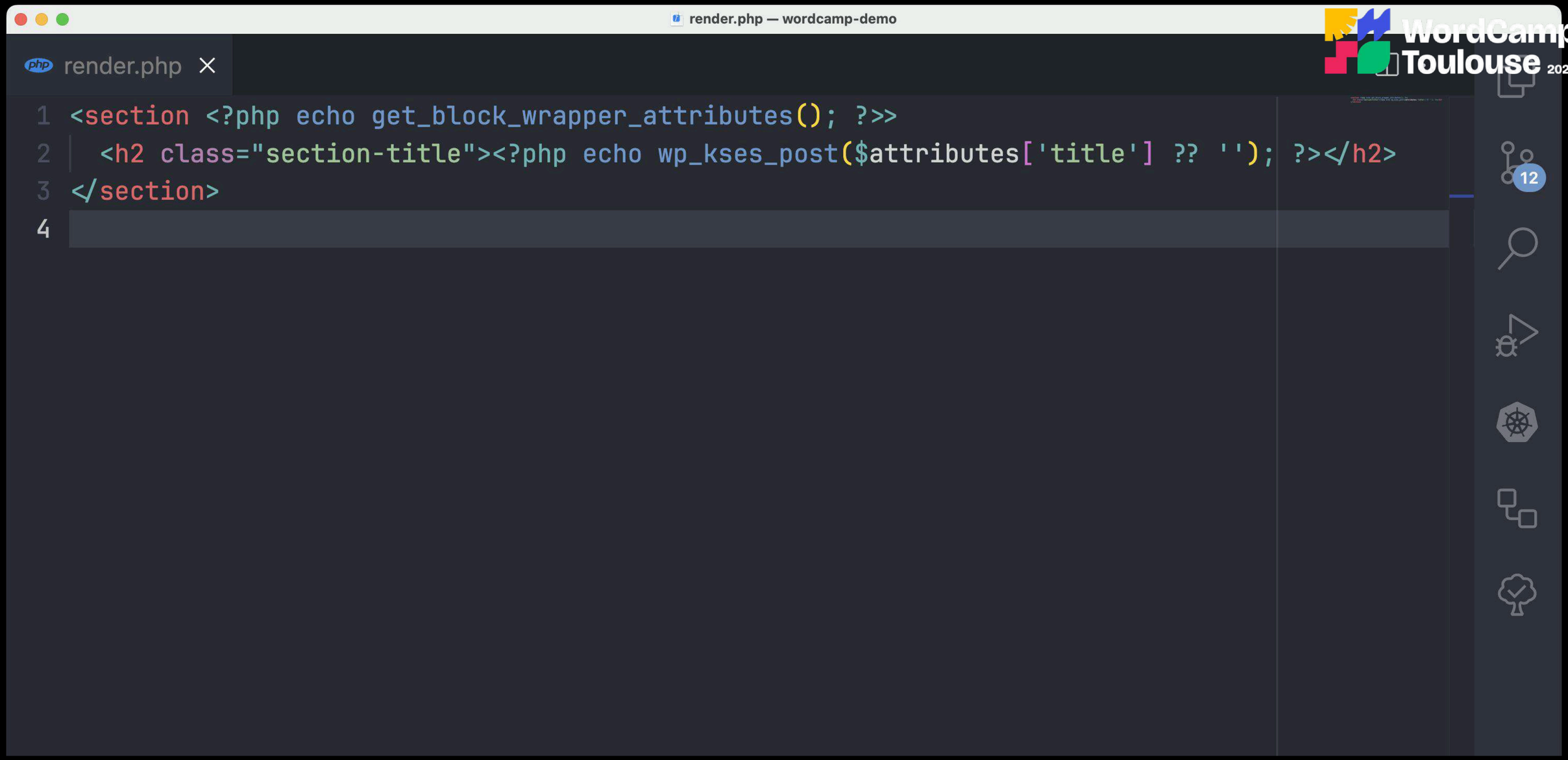
WordCamp Toulouse 2025

block.json



WordCamp Toulouse 2025

edit.js


render.php — wordcamp-demo
render.php
1 <section <?php echo get_block_wrapper_attributes(); ?>>
2 <h2 class="section-title"><?php echo wp_kses_post($attributes['title'] ?? ''); ?></h2>
3 </section>
4
12

render.php

WordCamp Toulouse 2025

Problème N°1 :
Une partie du code est dupliqué entre les fichiers

WordCamp Toulouse 2025

Problème N°2 :
Barrière à l'entrée technologique (React.js, JSX, etc.)

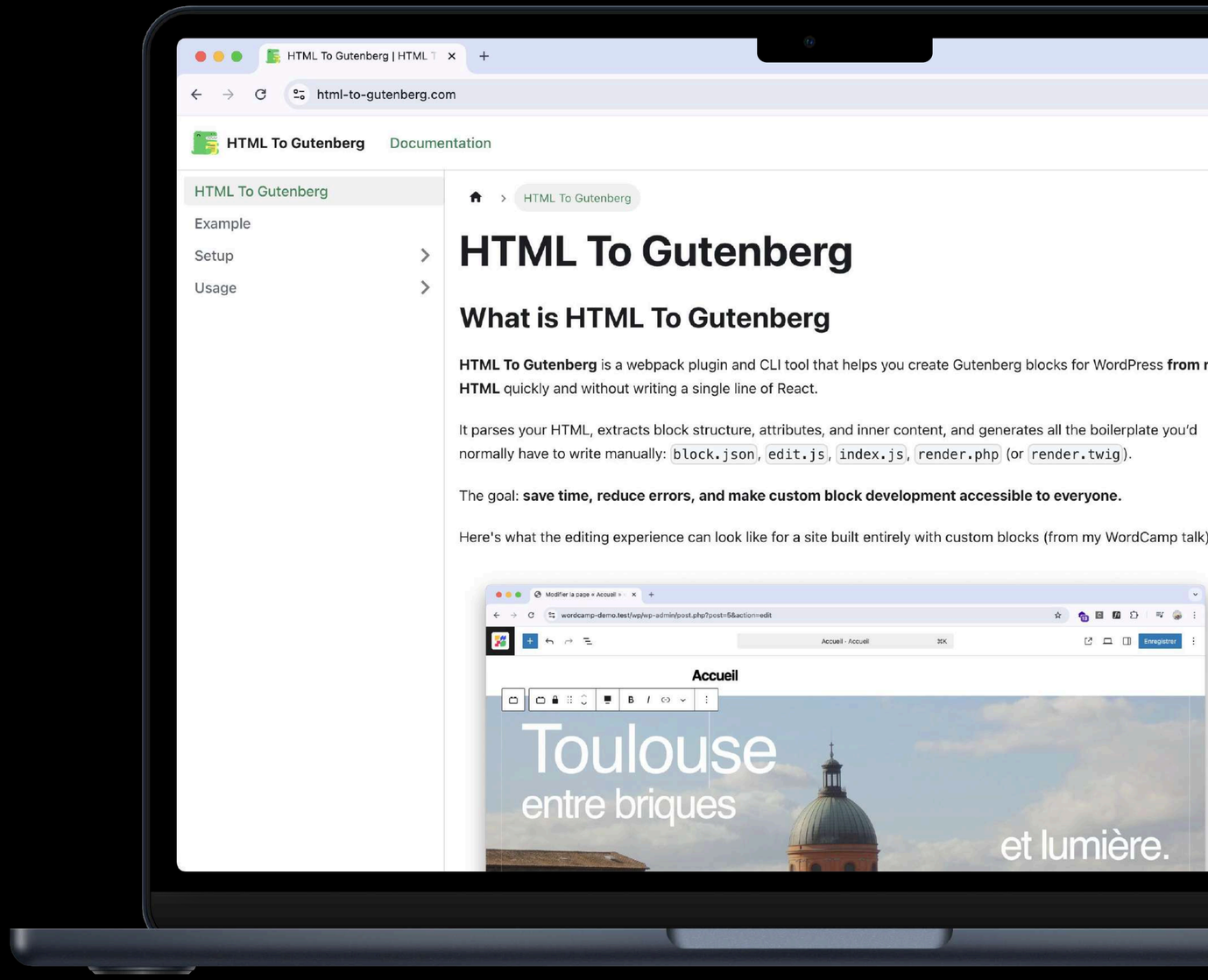
WordCamp Toulouse 2025

Ma solution ✅

HTML To Gutenberg

Un plugin webpack et un outil en ligne de commande pour générer un bloc Gutenberg dynamique à partir d'un simple fichier HTML.

html-to-gutenberg.com



WordCamp Toulouse 2025

simple-block.html M

```
...
1 <section>
2   <h2
3     class="section-title"
4     data-attribute="title">
5     Section title
6   </h2>
7 </section>
8
```

simple-block.html

edit.js — wordcamp-demo

edit.js

```
1  import { useBlockProps, RichText } from '@wordpress/block-editor';
2
3  export default ({ attributes, setAttributes }) => {
4    return (
5      <section {...useBlockProps()}>
6        <RichText
7          className="section-title"
8          tagName="h2"
9          value={attributes.title}
10         onChange={(title) => setAttributes({ title })}
11         placeholder="Title"
12       ></RichText>
13     </section>
14   );
15 };
16
```

main*+ ⊗ 0 ⚠ 0 ▶ 00:00 0 secs -- NORMAL -- UTF-8 LF JavaScript

simple-block/edit.js

block.json — wordcamp-demo

block.json

```
1  {
2    "name": "custom/simple-block",
3    "title": "Simple Block",
4    "textdomain": "simple-block",
5    "$schema": "https://schemas.wp.org/trunk/block.json",
6    "apiVersion": 3,
7    "version": "0.1.0",
8    "category": "theme",
9    "example": {},
10   "attributes": {
11     "align": { "type": "string", "default": "full" },
12     "title": { "type": "string", "default": "Section title" }
13   },
14   "supports": { "html": false, "align": ["full"] },
15   "editorScript": "file:./index.js",
16   "render": "file:./render.php"
17 }
18
```

main*+ ⊗ 0 ⚠ 0 ▶ 00:00 0 secs -- NORMAL -- Spaces: 2 UTF-8

simple-block/
block.json

render.php — wordcamp-demo

render.php

```
1 <section <?php echo get_block_wrapper_attributes(); ?>>
2   <h2 class="section-title"><?php echo wp_kses_post($attributes['title']) ?></h2>
3 </section>
4
```

simple-block/
render.php

WordCamp Toulouse 2025

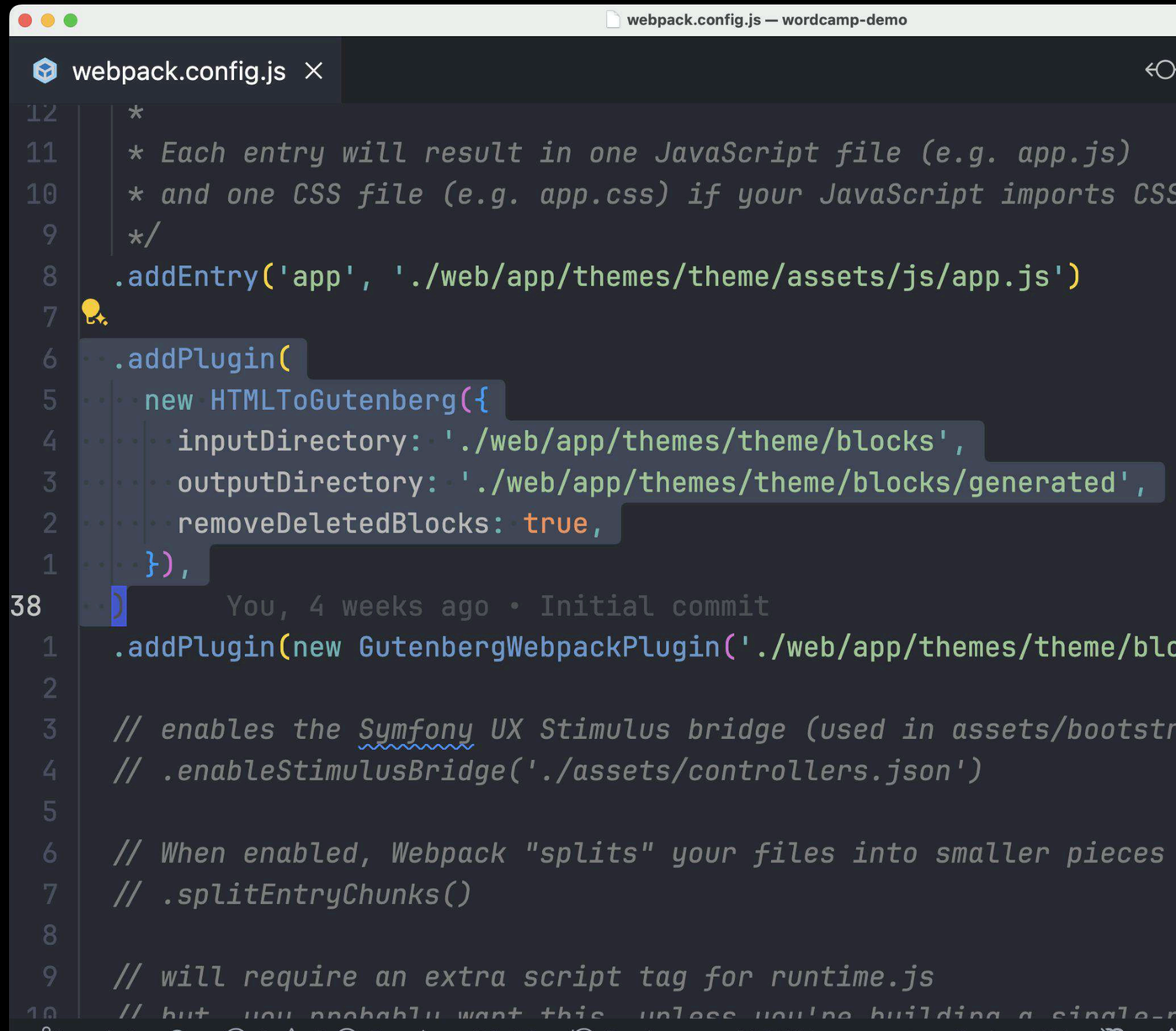
Documentation

Configuration

Ajout à une config webpack existante via
npm install @jverneaut/html-to-gutenberg

OU

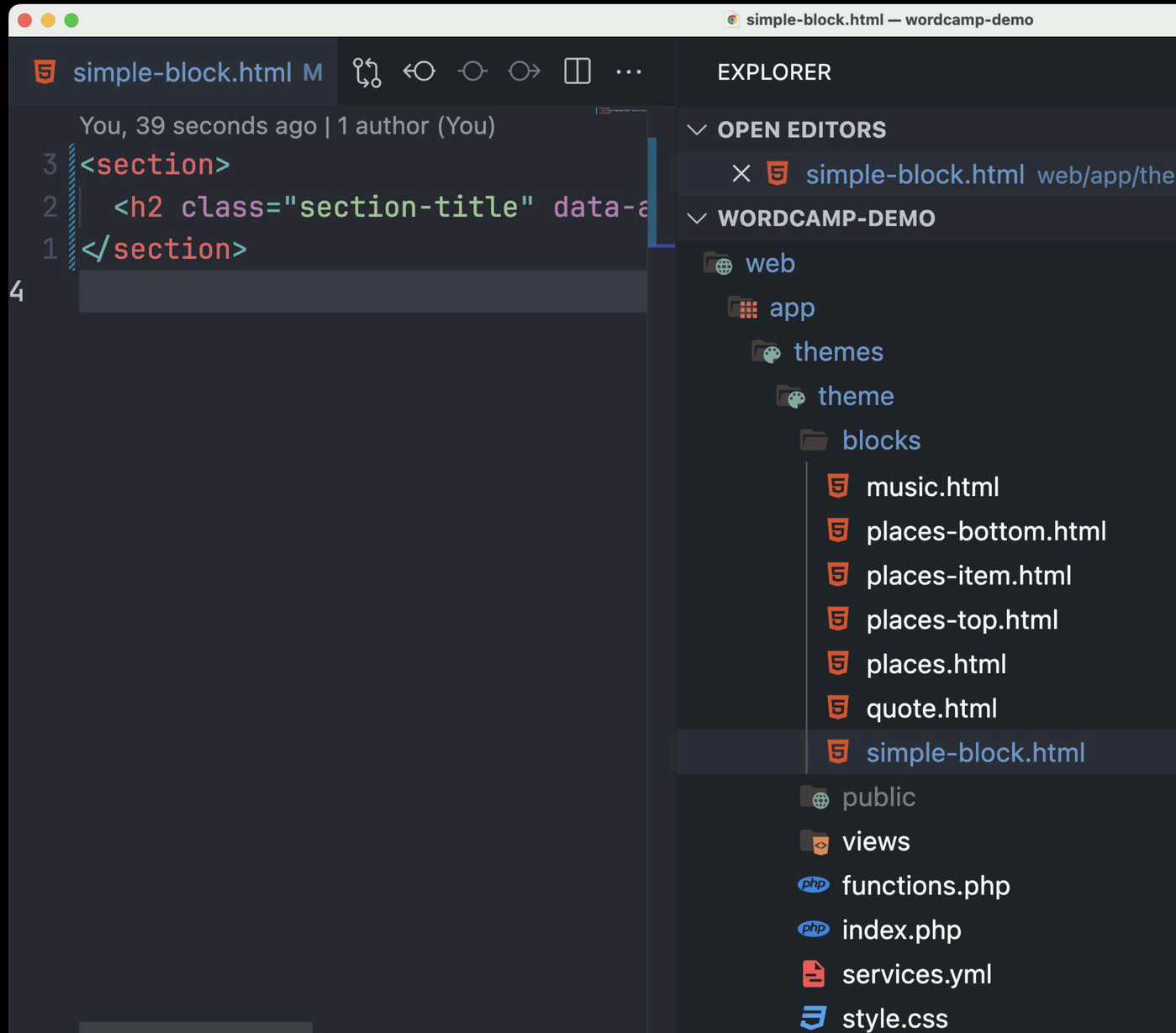
npx @jverneaut/html-to-gutenberg



WordCamp Toulouse 2025

Documentation

Création d'un bloc



Ajouter un fichier HTML dans le dossier inputDirectory défini via le plugin Webpack.

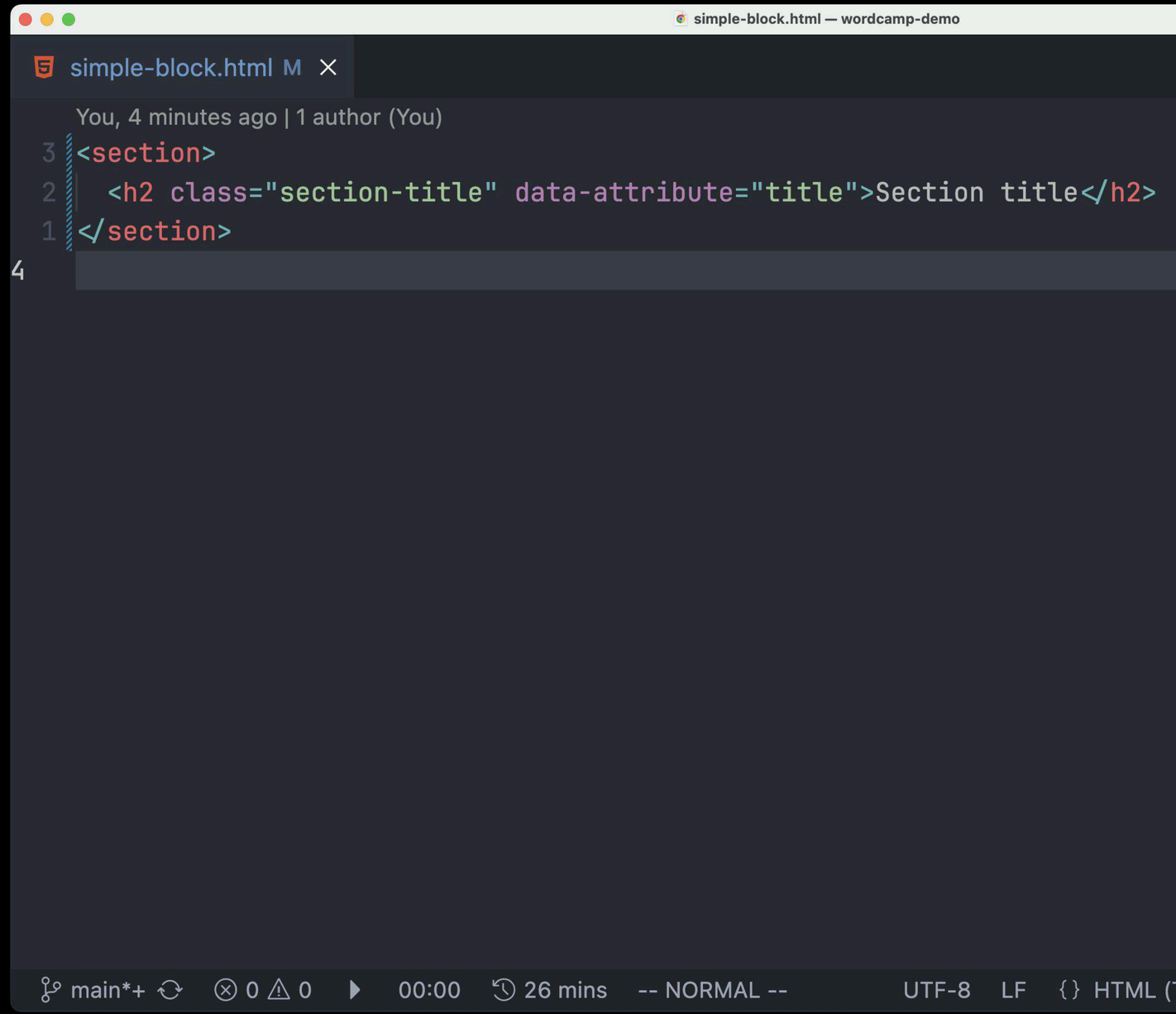
WordCamp Toulouse 2025

Documentation

Ajout d’un attribut texte

Ajouter un data-attribute=”[nom de l’attribut]” sur l’élément à éditer.

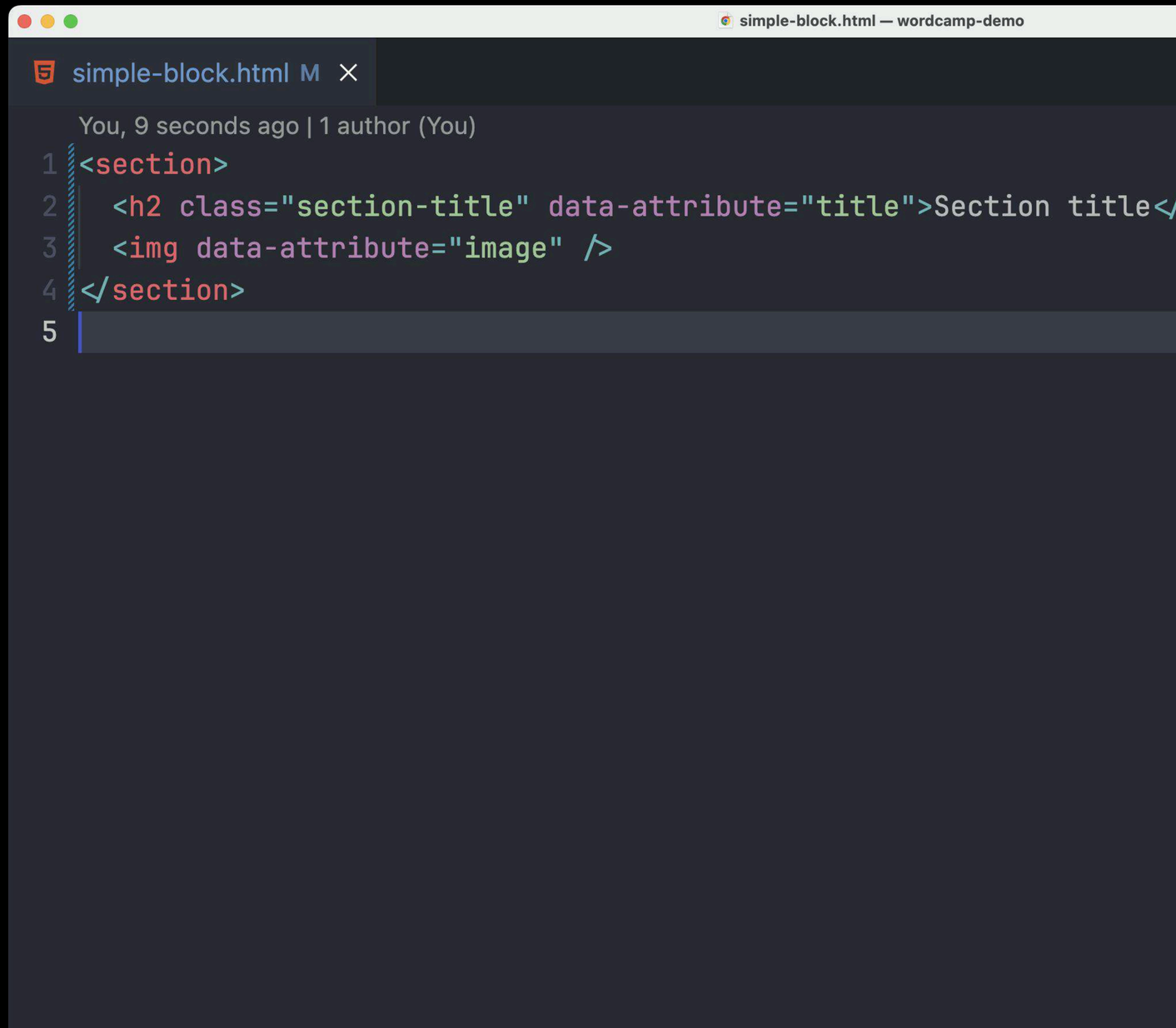
Il devient automatiquement éditable en back-office et rendu en front-end. Son contenu initial est également utilisé comme défaut.



WordCamp Toulouse 2025

Documentation

Ajout d’un attribut image

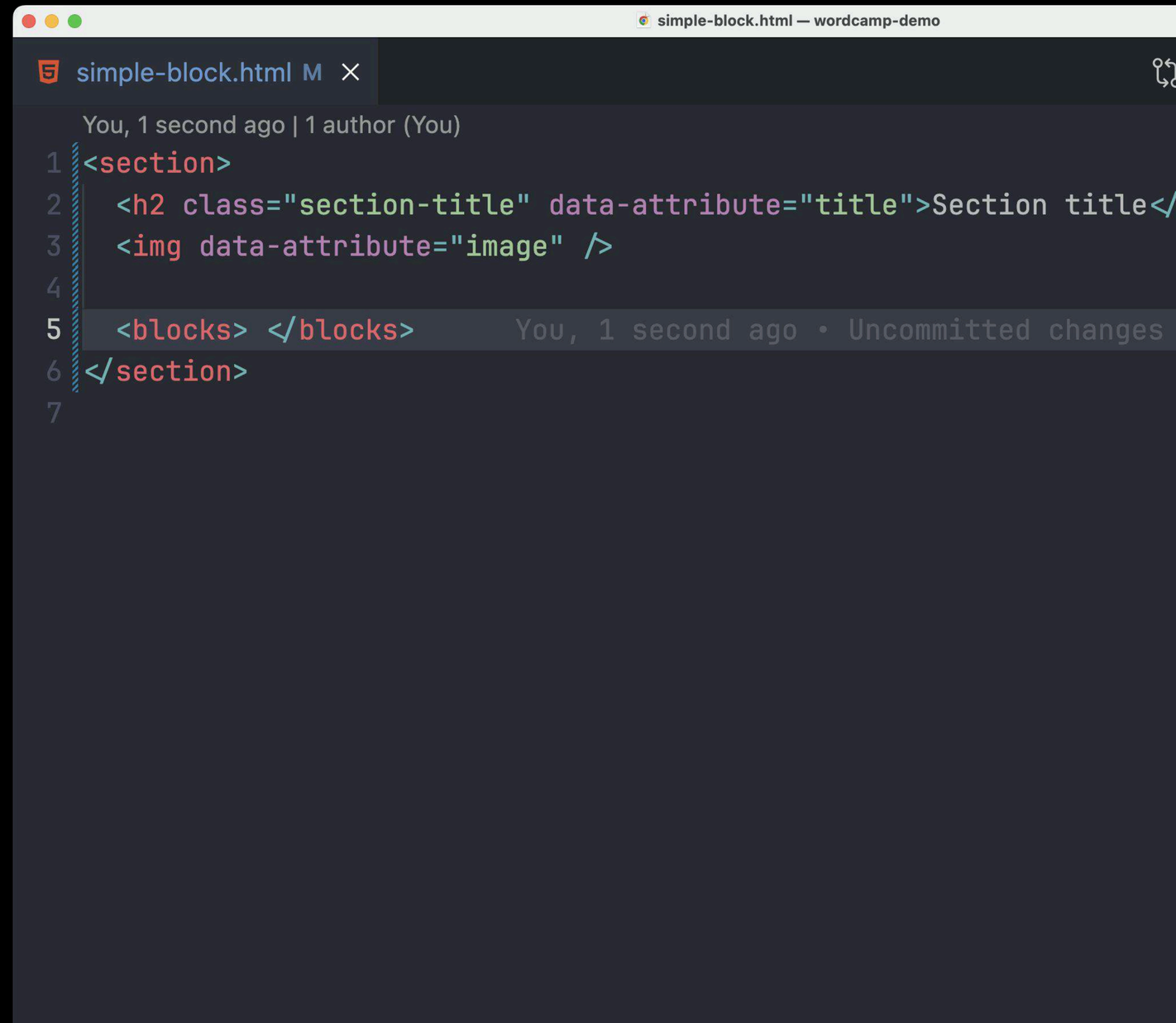


Ajouter un data-attribute=”[nom de l’attribut]” sur l’image à éditer.

WordCamp Toulouse 2025

Documentation

Utilisation des InnerBlocks

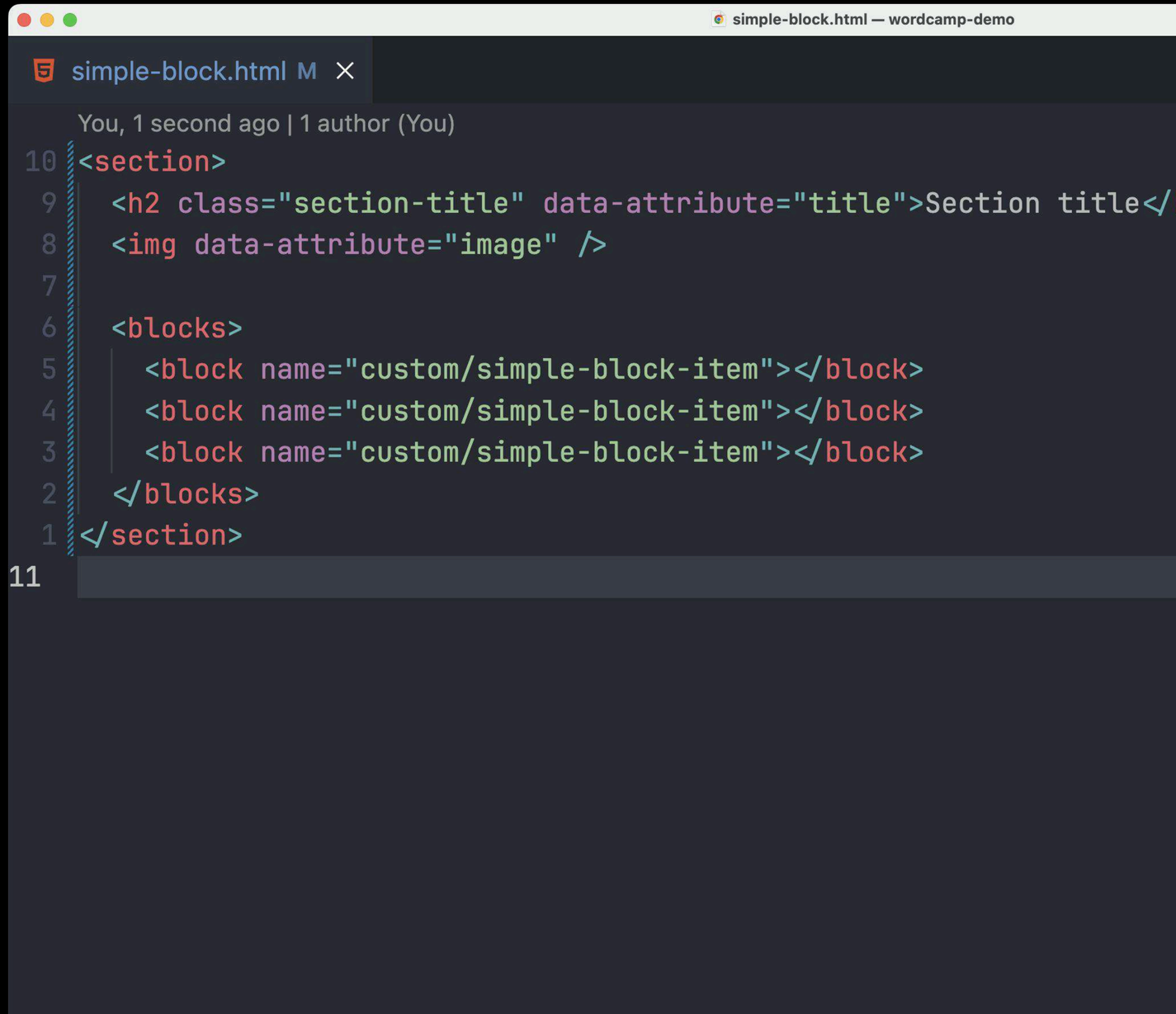


L'élément <blocks> permet de définir une zone unique pour les InnerBlocks.

WordCamp Toulouse 2025

Documentation

Utilisation avancée des InnerBlocks (1/3)

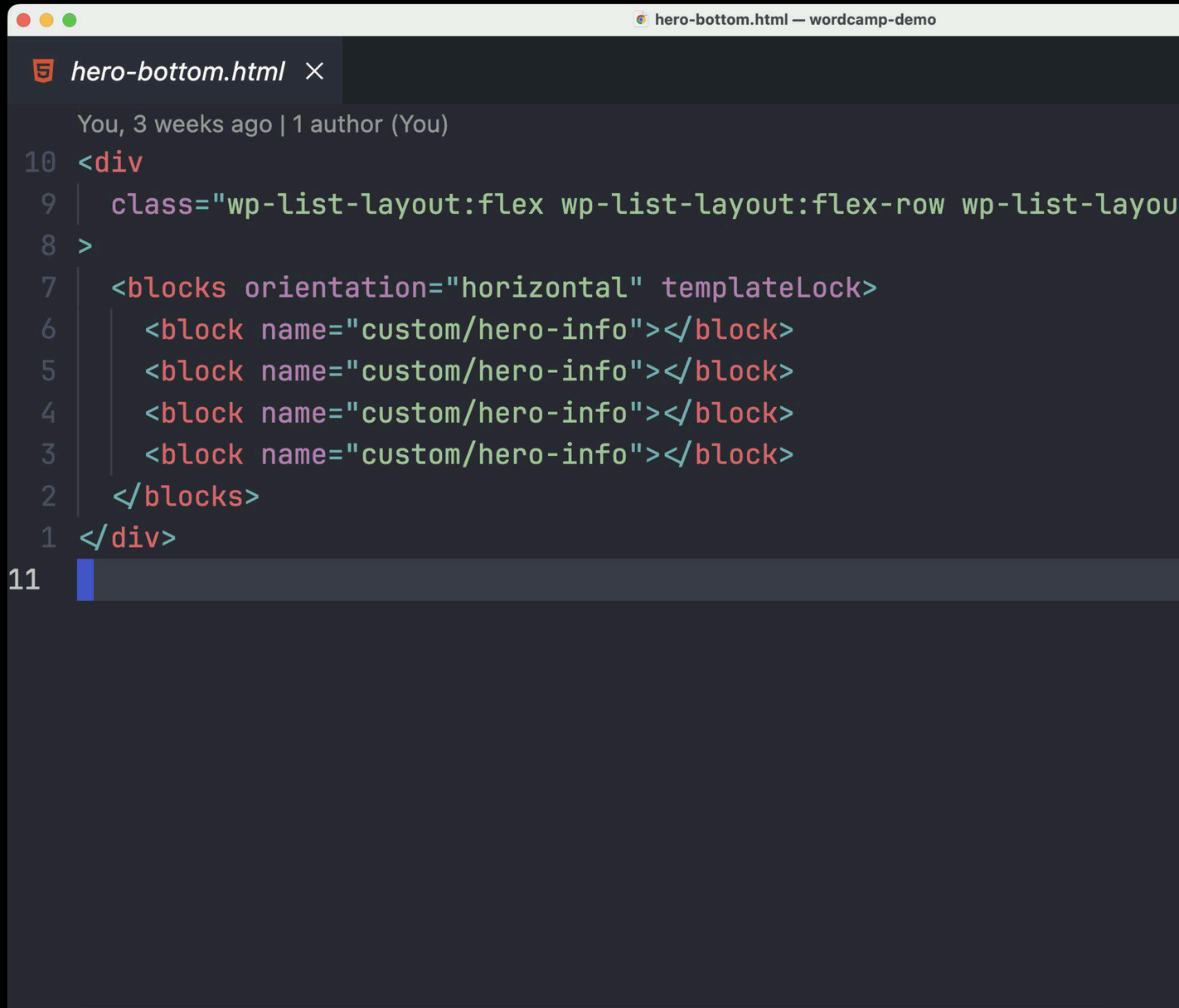


Utiliser l'élément <block> pour définir la template des blocs internes.

WordCamp Toulouse 2025

Documentation

Utilisation avancée des InnerBlocks (2/3)

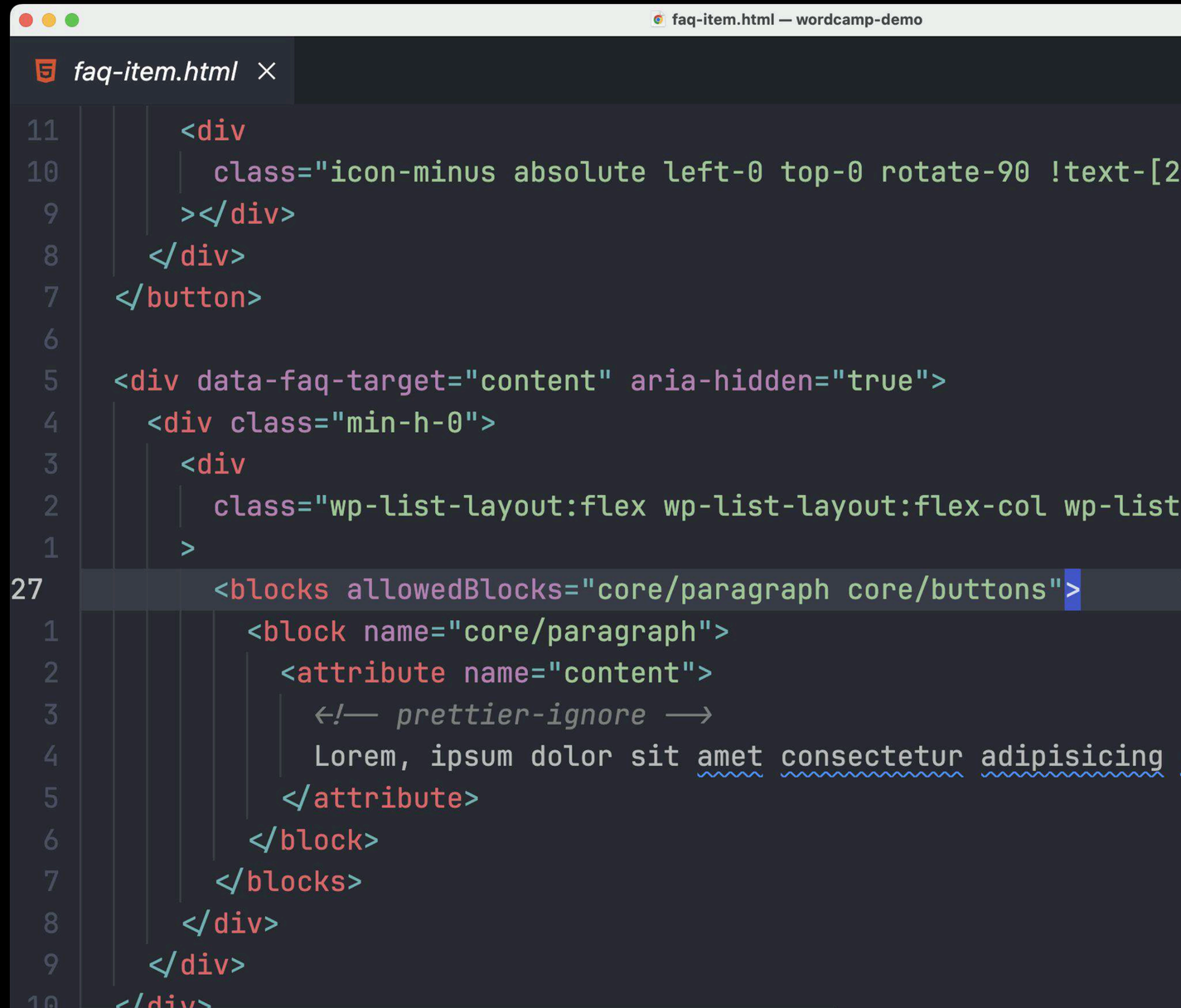


Possibilité de définir l'orientation et de verrouiller la template.

WordCamp Toulouse 2025

Documentation

Utilisation avancée des InnerBlocks (3/3)



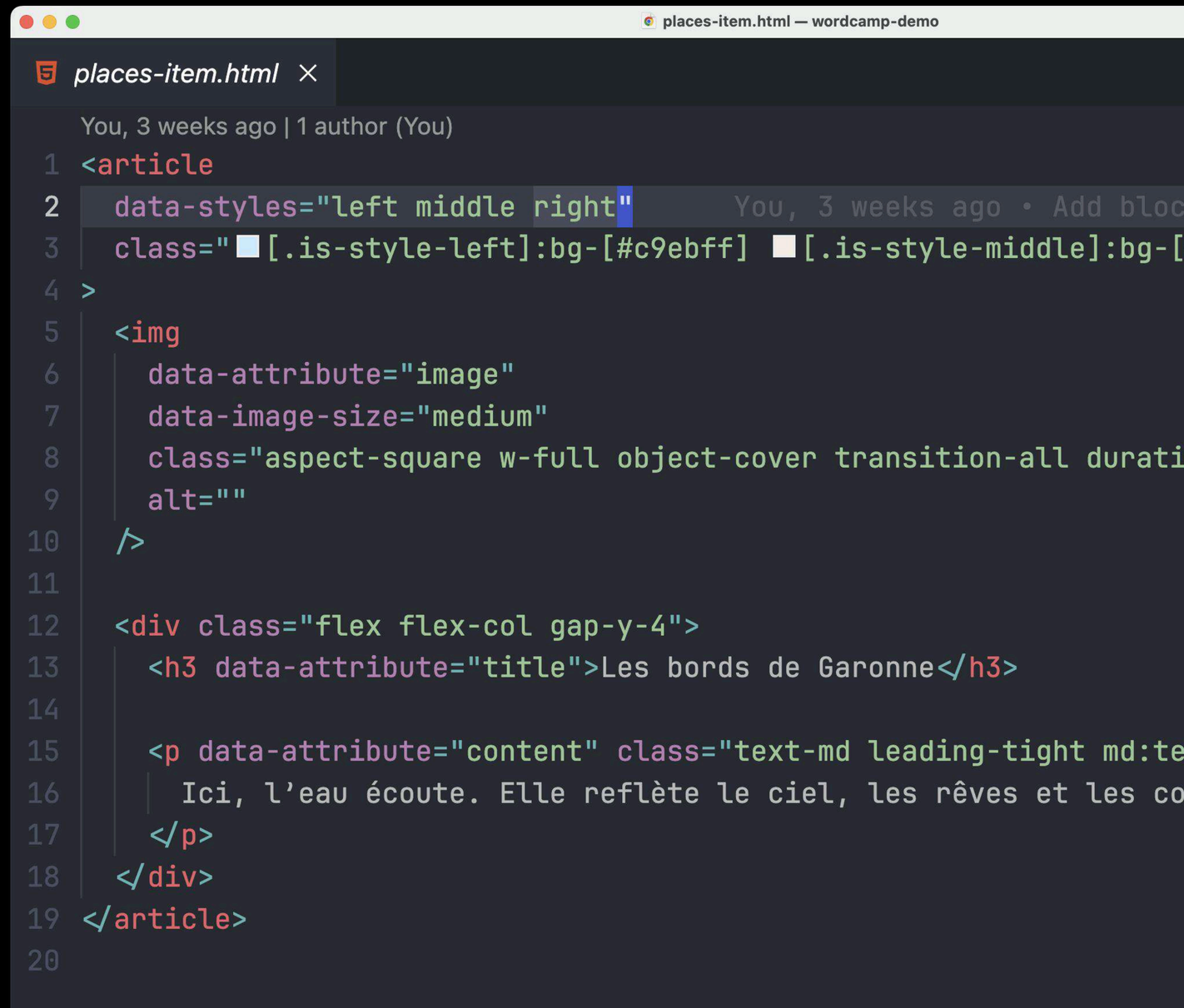
Possibilité de définir les blocs autorisés à être utilisés.

WordCamp Toulouse 2025

Documentation

Déclaration des variations de styles

Il est possible de définir des variations de styles en utilisant l'attribut data-styles sur l'élément de premier niveau du bloc.



WordCamp Toulouse 2025

Présentation live

Démonstration de HTML To Gutenberg

WordCamp Toulouse 2025

Les avantages de HTML To Gutenberg

Pas de duplication de code

Un seul fichier source est utilisé pour générer tous les fichiers nécessaires pour enregistrer un bloc.

Pas d’utilisation de React.js

Les développeurs ne connaissant pas React.js peuvent désormais créer des blocs Gutenberg natifs sans avoir besoin d’écrire une seule ligne de JavaScript.

Mise à jour simplifiée

En modifiant le fichier source, on modifie le rendu du bloc en back-office et en front-end.

Portabilité

Comme les blocs sont contenus dans un seul fichier, ils peuvent facilement être réutilisés sur différents projets.

Cela ouvre également la voie au développement d’une librairie de blocs réutilisables à partir de templates existantes, comme Tailwind Plus par exemple.

Questions/réponses