

EBOOK

MICROSOFT VISUAL STUDIO CODE

# FLEXBOX CSS 3

*En Pratique avec  
Visual Studio Code  
(version 1.19)*



Patrice REY

*Formateur En Applications Informatiques*



# **FLEXBOX CSS3 (2eme edition)**

[Presentation](#)

[Sommaire](#)

[fiche\\_00](#)

[fiche\\_01](#)

[fiche\\_02](#)

[fiche\\_03](#)

[fiche\\_04](#)

[fiche\\_05](#)

[fiche\\_06](#)

[fiche\\_07](#)

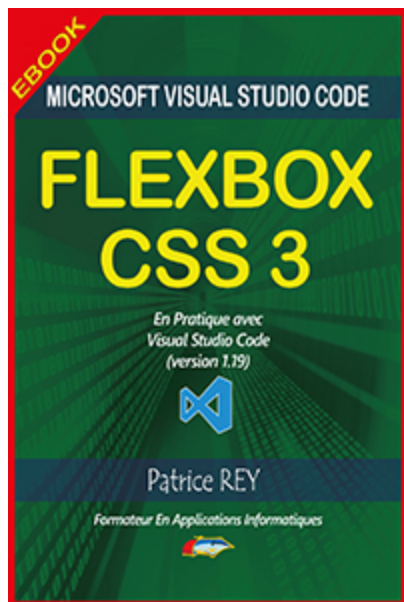
[fiche\\_08](#)

[fiche\\_09](#)

[fiche\\_10](#)

[Page de copyright](#)

# Presentation



## **FLEXBOX CSS 3 *par Patrice REY***

La spécification *Flexible Box Layout Module*, plus connue sous le nom de **Flexbox**, est une spécification CSS 3 du W3C qui définit un nouveau modèle de boîte et de positionnement jusqu'alors inédit.

Ce livre s'adresse à des personnes connaissant de façon générale l'usage et l'utilisation des CSS, qui souhaitent apprendre en pratique ce nouveau modèle de positionnement qu'est Flexbox.

Le code de programmation de tous les exemples de cet ouvrage est disponible en téléchargement gratuit en se rendant sur la fiche du livre sur le site [www.reypatrice.fr](http://www.reypatrice.fr).

# Sommaire

## Sommaire du livre

### Fiche 0 - [Avant-propos](#)

### Fiche 1 - [Mise en place de l'environnement de travail](#)

- 1 - La spécification *Flexible Box Layout Module*
- 2 - Créer un projet avec Visual Studio Code
- 3 - Installer des extensions
- 4 - Ajouter des dossiers et des fichiers

### Fiche 2 - [Les bases du design](#)

- 1 - Conteneur flexible et élément flexible
- 2 - Propriétés CSS agissant sur le conteneur flexible
  - 2.1 - La propriété *display*
  - 2.2 - La propriété *flex-direction*
  - 2.3 - La propriété *flex-wrap*
  - 2.4 - La propriété *flex-flow*
  - 2.5 - La propriété *justify-content*
  - 2.6 - La propriété *align-items*
  - 2.7 - La propriété *align-content*
- 3 - Propriété CSS agissant sur l'élément flexible enfant
  - 3.1 - La propriété *order*
  - 3.2 - La propriété *align-self*
  - 3.3 - La propriété *flex-grow*
  - 3.4 - La propriété *flex-shrink*

3.5 - La propriété *flex-basis*

3.6 - La propriété *flex*

### **Fiche 3 - Utilisation des Media Queries CSS 3**

1 - Le sélecteur *@media*

2 - Mise en page classique Flexbox pour écran d'ordinateur

3 - Mise en page avec les *media queries*

### **Fiche 4 - Quelques astuces indispensables**

1 - Occuper l'espace restant

2 - Centrer verticalement

3 - Pousser un élément flexible

4 - Aligner un titre avec un retour à la ligne

5 - Alignement horizontal et vertical

6 - Le rétrécissement d'un élément flexible

7 - Ensemble de colonnes flexibles

### **Fiche 5 - TP: un gabarit simple typique**

1 - Mise en place des éléments HTML de base

2 - Mise en place de l'ordre de placement

3 - Fixer les propriétés *flex-grow*, *flex-shrink* et *flex-basis*

4 - Mise en place des liens avec le survol

### **Fiche 6 - TP: une grille flexible à colonne et à gouttière**

1 - Mise en place des éléments HTML de base

2 - Fixer les propriétés *flex-grow*, *flex-shrink* et *flex-basis*

3 - Fixer la largeur d'une cellule par un calcul

### **Fiche 7 - TP: un formulaire fluide**

- 1 - L'objectif à atteindre
- 2 - Mise en place des éléments HTML de base
- 3 - Application des styles CSS
- 4 - Une couche de *responsive webdesign*

## **Fiche 8 - TP: une galerie d'images**

- 1 - Mise en place des éléments HTML de base
- 2 - Répartition horizontale
- 3 - Répartition verticale

## **Fiche 9 - Techniques de positionnement d'éléments**

- 1 - Répartir des éléments flexibles sur une ligne unique
- 2 - Répartir les éléments sur une ligne avec une gouttière
- 3 - Répartition en forçant le retour à la ligne
- 4 - Répartition automatique

## **Fiche 10 - TP: grille flexible et préprocesseur LESS**

- 1 - Construction d'une grille flexible
  - 1.1 - Une grille flexible avec 4 colonnes
  - 1.2 - Une grille flexible avec colonnes et gouttières
  - 1.3 - Des colonnes qui s'étendent
- 2 - Automatiser avec un préprocesseur LESS

# fiche\_00

## Avant-propos

La spécification *Flexible Box Layout Module*, plus connue sous le nom de Flexbox, est une spécification CSS 3 du W3C qui définit un nouveau modèle de boîte et de positionnement jusqu'alors inédit. A lui seul, ce mode de positionnement rend élémentaires tous les problèmes classiques rencontrés avec CSS depuis sa naissance: les alignements s'effectuent d'une façon simple, la possibilité de réaliser un centrage vertical, les éléments s'emboîtent avec une fluidité naturelle, des hauteurs identiques entre éléments frères, etc. Flexbox intègre une gestion naturelle de la fluidité des éléments, rendant caduc l'usage de grilles d'affichage complexes. Cette spécification Flexbox est en train de révolutionner de manière profonde et pérenne notre façon de concevoir des designs et des composants en CSS.

Flexbox a été pensé et optimisé pour faire table rase de toutes les techniques historiques de positionnement ainsi que les contournements bancales des propriétés CSS. Le but affiché de Flexbox est de remettre de l'ordre en revenant aux bases d'un positionnement propre et adapté aux besoins actuels. Outre la simplification des schémas de positionnement, Flexbox apporte des solutions parfaites à une problématique bien ancrée à notre époque qu'est le *Responsive Webdesign*. En effet, les projets d'adaptation aux tablettes et aux smartphones nécessitent la réalisation de designs d'éléments flexibles, la réorganisation des blocs ainsi que la faculté à basculer très aisément d'un mode d'affichage horizontal vers un mode d'affichage vertical.



## Le contenu du livre

Ce livre s'adresse à des personnes connaissant de façon générale l'usage et l'utilisation des CSS, qui souhaitent apprendre en pratique ce nouveau modèle de positionnement qu'est Flexbox. Cet ouvrage est conçu avant tout pour répondre à des cas réels de la vie d'intégrateur web de tous les jours. Il est essentiellement constitué des notions à connaître, des exemples concrets décortiqués et des travaux pratiques.

Dans la fiche 1, nous allons mettre en place un environnement de travail avec l'utilisation du logiciel gratuit *Visual Studio Code 1.19* de façon à optimiser la compréhension et la réalisation des boîtes flexibles.

Dans la fiche 2, nous aborderons les bases du modèle de boîte flexible avec l'ensemble des propriétés et de leurs effets. Nous verrons les propriétés qui agissent sur le conteneur parent (*display*, *flex-direction*, *flex-wrap*, *flex-flow*, *justify-content*, *align-items* et *align-content*) et les propriétés qui agissent sur les éléments flexibles enfants (*flex-grow*, *flex-shrink*, *flex-basis* et *flex*).

Dans la fiche 3, nous aborderons le concept des *media queries* et leur utilisation dans un contexte de conteneur flexible de type Flexbox.

Dans la fiche 4, nous verrons les astuces indispensables qui permettent d'utiliser Flexbox de manière simple et efficace (occuper l'espace restant, centrer verticalement, pousser un élément flexible, aligner un titre avec un retour à la ligne, l'alignement horizontal et vertical, le rétrécissement d'un élément flexible et la construction d'un ensemble de colonnes flexibles).

Dans la fiche 5, nous réaliserons un travail pratique pour réaliser un gabarit simple typique avec le modèle de positionnement Flexbox.



Dans la fiche 6, nous réaliserons un travail pratique pour la mise en place d'une grille flexible à colonnes et à gouttières.

Dans la fiche 7, nous réaliserons un travail pratique pour la mise en place d'un formulaire fluide destiné à collecter des données avant de les envoyer par un script PHP.

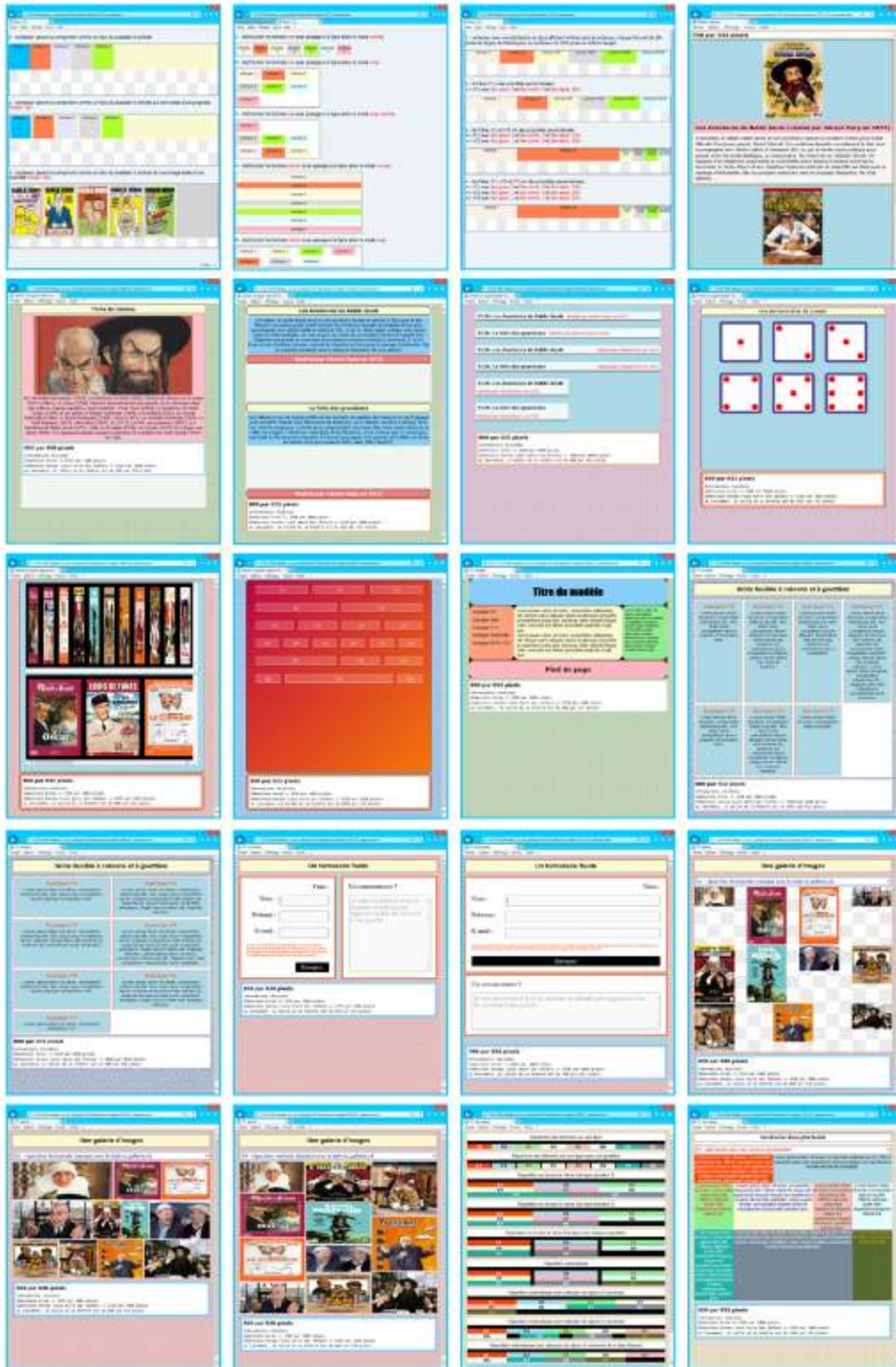
Dans la fiche 8, nous réaliserons un travail pratique pour composer une galerie d'images. De nombreux réglages permettront de mieux appréhender la façon de faire pour la mise en place de la galerie d'images.

Dans la fiche 9, nous aborderons les techniques courantes de positionnement d'éléments flexibles enfants au sein d'un conteneur flexible parent.

Dans la fiche 10, nous réaliserons un travail pratique pour la construction de grille flexible. Nous verrons comment automatiser les tâches pour la création de grille flexible en utilisant un préprocesseur de type **LESS** (mise en place d'une architecture côté client et utilisation avec la syntaxe LESS dans des fichiers au format *.less*).

La figure A1 visualise une partie des miniatures des pages web qui vont être réalisées au cours de ce livre avec Visual Studio Code pour mettre en pratique le modèle original de positionnement qu'est Flexbox. Vous vous apercevrez qu'avec l'utilisation de l'éditeur de code qu'est Visual Studio Code, il est très facile de réaliser un code de programmation dans différents formats (HTML, CSS, JavaScript, LESS, Json, etc.).

Figure A1



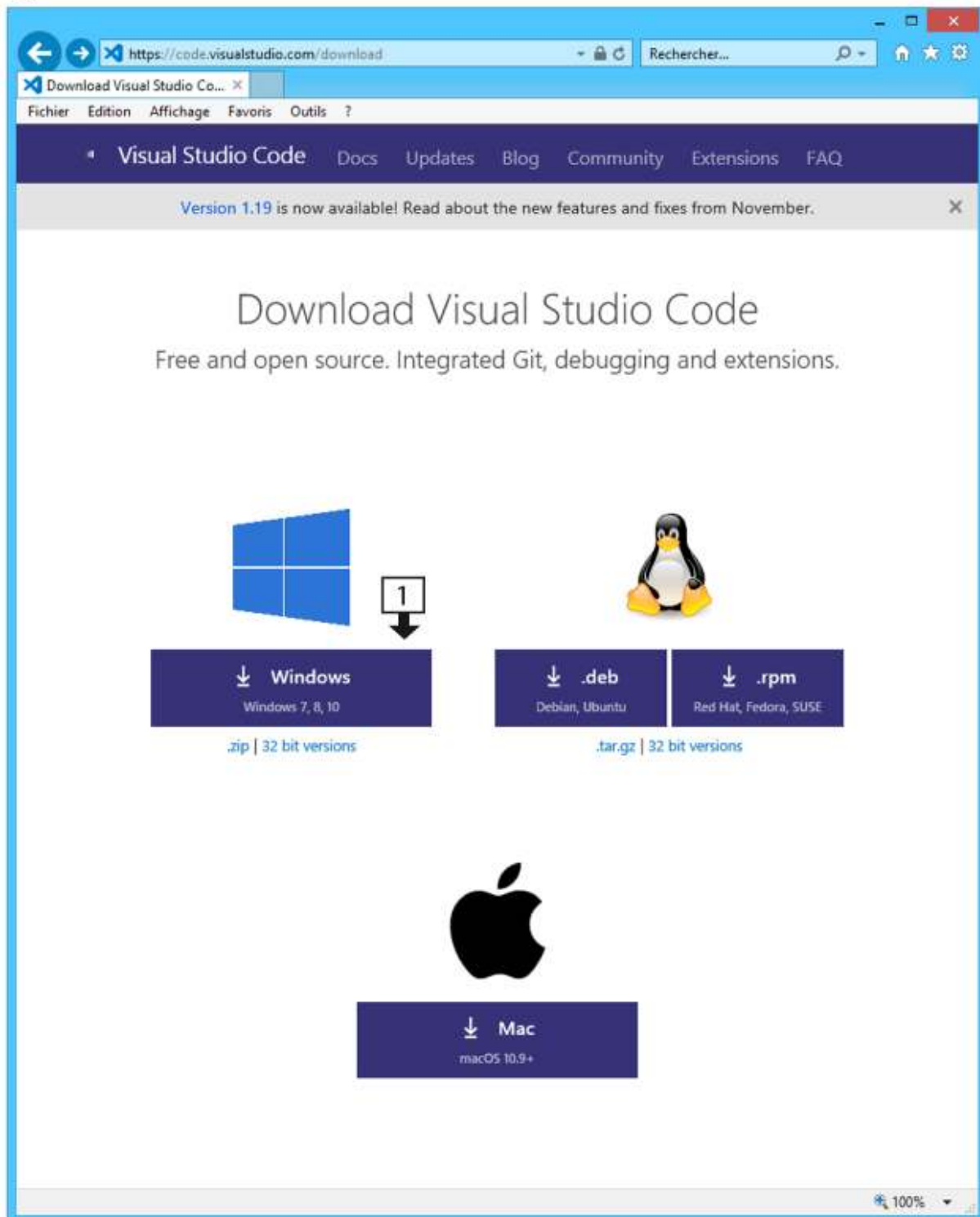
## Le logiciel Visual Studio Code pour le développement

Pour comprendre, mettre en pratique et réaliser plus facilement des programmes pour l'utilisation des boîtes flexibles, nous allons utiliser le logiciel gratuit *Visual Studio Code* dans sa version 1.19. Visual Studio Code est un éditeur de texte générique multiplateforme exceptionnel en français que vous pouvez télécharger gratuitement à l'adresse web <http://code.visualstudio.com/download> en cliquant sur le lien de la figure A2 au repère 1 si vous êtes sur Windows par exemple. C'est un logiciel *open source*, fournit par Microsoft, qui peut être utilisé gratuitement sans limite de temps.

Visual Studio Code s'exécute de manière native sur le système d'exploitation de votre choix (Mac OS X, Linux et Windows). Grâce à la colorisation de la syntaxe, la mise en surbrillance et la concordance des parenthèses, c'est un éditeur de premier choix pour toutes vos tâches de programmation. Il prend en charge JavaScript, C#, C++, PHP, Java, HTML, R, CSS, SQL, Markdown, TypeScript, LESS, SASS, JSON, XML et Python ainsi que de nombreux autres formats de fichiers courants. Il possède une modification avancée, rapide et adaptée pour clavier, avec plusieurs curseurs. L'assistant visuel complète automatiquement les descriptions des API lorsque vous les saisissez pour une plus grande rapidité et une plus grande précision. Vous diagnostiquez les problèmes de votre application avec les outils intégrés de débogage pour Node.js, TypeScript et JavaScript. Vous définissez des points d'arrêt dans votre code, des arrêts en cas d'exception et des variables espions, vous examinez votre code ou vous naviguez en remontant la pile des appels, ou vous rattachez aux processus d'exécution locaux. Vous adoptez les flux de travail modernes grâce à la puissance et

à la flexibilité de Git. Visual Studio Code prend en charge le processus de développement des applications ASP.NET 5 et Node.js de bout en bout. La version actuelle de Visual Studio Code est la version 1.19 disponible en français, et c'est celle utilisée tout au long de ce livre. Une documentation en ligne est disponible à l'adresse web *<http://code.visualstudio.com/Docs>*.

Figure A2



## Les liens de téléchargement

Tout le code source de programmation de cet ouvrage pourra être téléchargé gratuitement à l'adresse web suivante en allant sur la fiche du livre [\*http://www.reypatrice.fr\*](http://www.reypatrice.fr). Vous trouverez dans l'arborescence des fichiers du projet, un dossier pour chacun des chapitres. Chaque dossier contient tous les éléments nécessaires de programmation qui sont expliqués dans le livre. De cette façon, vous serez capable de suivre et de programmer assez rapidement et assez facilement. C'est une démarche volontaire, dans un but pédagogique, pour progresser rapidement. Bonne lecture à tous.

# fiche\_01

## Fiche 1 - Mise en place de l'environnement de travail

La spécification *Flexible Box Layout Module*, plus connue sous le nom de Flexbox, est une spécification CSS 3 du W3C qui définit un nouveau modèle de boîte et de positionnement jusqu'alors inédit. Nous allons commencer par voir, dans un premier temps, où se trouve la définition de cette spécification. Et dans un second temps, nous allons mettre en place un environnement de travail avec *Visual Studio Code* pour appréhender par la suite la programmation et l'utilisation de Flexbox.

### 1 - La spécification *Flexible Box Layout Module*

Les feuilles de style en cascade, généralement appelées CSS ( *Cascading Style Sheets*), forment un langage informatique qui décrit la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le **World Wide Web Consortium** (dont le sigle est **W3C**). Introduit au milieu des années 1990, CSS devient couramment utilisé dans la conception des sites web et bien pris en charge par les navigateurs web dans les années 2000. Le CSS 3 est le nom employé pour caractériser l'ensemble des nouveautés depuis le CSS 2.1. Il s'agit par exemple d'un ensemble de nouveaux effets à appliquer sur nos éléments HTML. Mais le CSS 3 c'est également un ensemble de nouveaux sélecteurs, de nouvelles manières de spécifier les couleurs, une détection des caractéristiques de l'appareil de l'utilisateur, des calculs dans les feuilles de style, des SVG en arrière-plan,

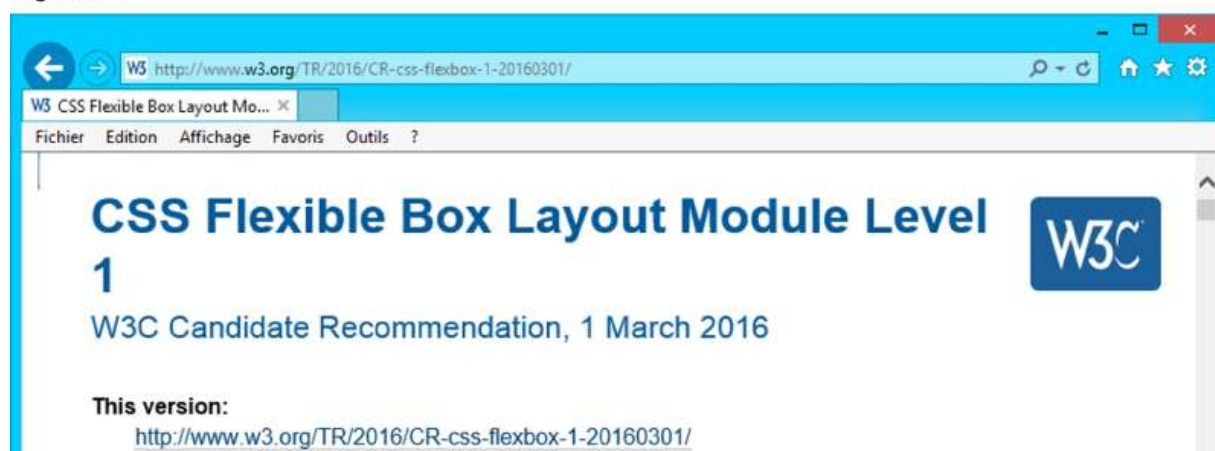


etc. Le CSS 3 est en fait un généreux enrichissement des feuilles de style qui, en plus d'être profitable à l'utilisateur, l'est au développeur également. La plupart des développeurs web ont découvert le CSS 3 en cherchant à appliquer certains effets sur leurs sites.

La spécification *Flexible Box Layout Module*, plus connue sous le nom de **Flexbox**, est une spécification CSS 3 du W3C. La consultation de la page web

<http://www.w3.org/TR/2016/CR-css-flexbox-1-20160301/> sur le site du W3C (figure 1.1) nous indique que la spécification de Flexbox est une spécification dont son stade officiel est aujourd'hui CSS Level 1. Cela signifie que tout est CSS 3 depuis la finalisation de CSS 2.1. Une nouvelle fonctionnalité, comme c'est le cas pour Flexbox, sera associée au *Level 1* tandis que les anciennes spécifications peuvent se situer au *Level 3* comme c'est le cas pour les polices et les modèles de boîtes.

Figure 1-1



Les technologies web sont un médium très plaisant à travailler car en constante évolution. Ces dernières années cette évolution s'accélère, nous offrant des possibilités de plus en plus grandes pour la création. Le seul inconvénient qu'apporte cette situation vient d'une particularité du web: il n'est pas centralisé. Les créateurs de navigateurs doivent donc implémenter ces nouvelles fonctionnalités eux-mêmes, ce qu'ils font à leur rythme et à leur manière. Dans ces

conditions, comment savoir quelles fonctionnalités sont assez répandues pour être utilisées? La réponse se trouve sur le site de **Can I Use** à l'adresse web <http://caniuse.com>. Le site *Can I Use* est un site vous fournissant tout ce dont vous pouvez avoir besoin pour vous en sortir dans cette jungle de fonctionnalités. Un large champ de recherches vous permettra de mettre la main sur la fonctionnalité en doute et un graphique clair vous indique son support en fonction des versions des navigateurs.

Quand vous vous trouvez sur cette page, inscrivez le terme "flexbox" en haut de la page comme le montre la figure 1.2 au repère 1. La fiche concernant Flexbox s'affiche au-dessous en visualisant un graphique de compatibilité des navigateurs. D'après cette fiche (figure 1.2 au repère 2), on apprend que 95.07% des navigateurs en France et 94.10% des navigateurs dans le monde appliquent les propriétés de Flexbox. En survolant les cases comme celle pour Internet Explorer 10, une info-bulle s'affiche (figure 1.2 au repère 3) indiquant qu'il faut préciser le préfixe vendeur "-ms-" à la propriété CSS spécifiée.

Figure 1-2



Les navigateurs utilisent des préfixes vendeurs (dits aussi préfixes propriétaires) qui sont théoriquement des

expérimentations des nouvelles propriétés. Ils sont ainsi libres de ne pas suivre les recommandations du W3C, aussi bien dans la manière d'afficher l'effet que dans la manière de l'écrire en CSS. Heureusement, ces navigateurs ont une forte volonté de respect des standards en suivant très exactement les spécifications du W3C. Par conséquent l'écriture des nouvelles propriétés est la même, mais en incluant simplement le préfixe spécifique à chaque navigateur comme *"-moz-"* pour Firefox, *"-webkit-"* pour Chrome/Safari/iPhone/Android, *"-o-"* pour Opera, *"-ms-"* pour Internet Explorer, etc. Par exemple la propriété CSS *border-radius* pour une bordure arrondie de 10 pixels s'écrit sous la forme:

```
-moz-border-radius: 10px;  
-webkit-border-radius: 10px;  
border-radius: 10px;
```

Cette bordure arrondie s'affichera correctement arrondie sur Internet Explorer 9, Firefox, Chrome, Safari, Opera, le navigateur iPhone, le navigateur Android et le navigateur Blackberry. Par contre elle ne s'affichera pas arrondie sur Internet Explorer 8 et versions inférieures puisqu'elle n'est pas implémentée.

## 2 - Créer un projet avec Visual Studio Code

Pour comprendre, mettre en pratique et réaliser plus facilement des programmes pour l'utilisation des boîtes flexibles, nous allons utiliser le logiciel gratuit **Visual Studio Code** dans sa version *1.19*. Visual Studio Code est un éditeur de texte générique multiplateforme exceptionnel en français que vous pouvez télécharger gratuitement à l'adresse web *<http://code.visualstudio.com/download>* en cliquant sur le lien correspondant à votre plateforme. C'est un logiciel *open source*, fournit par Microsoft, qui peut être utilisé gratuitement sans limite de temps.

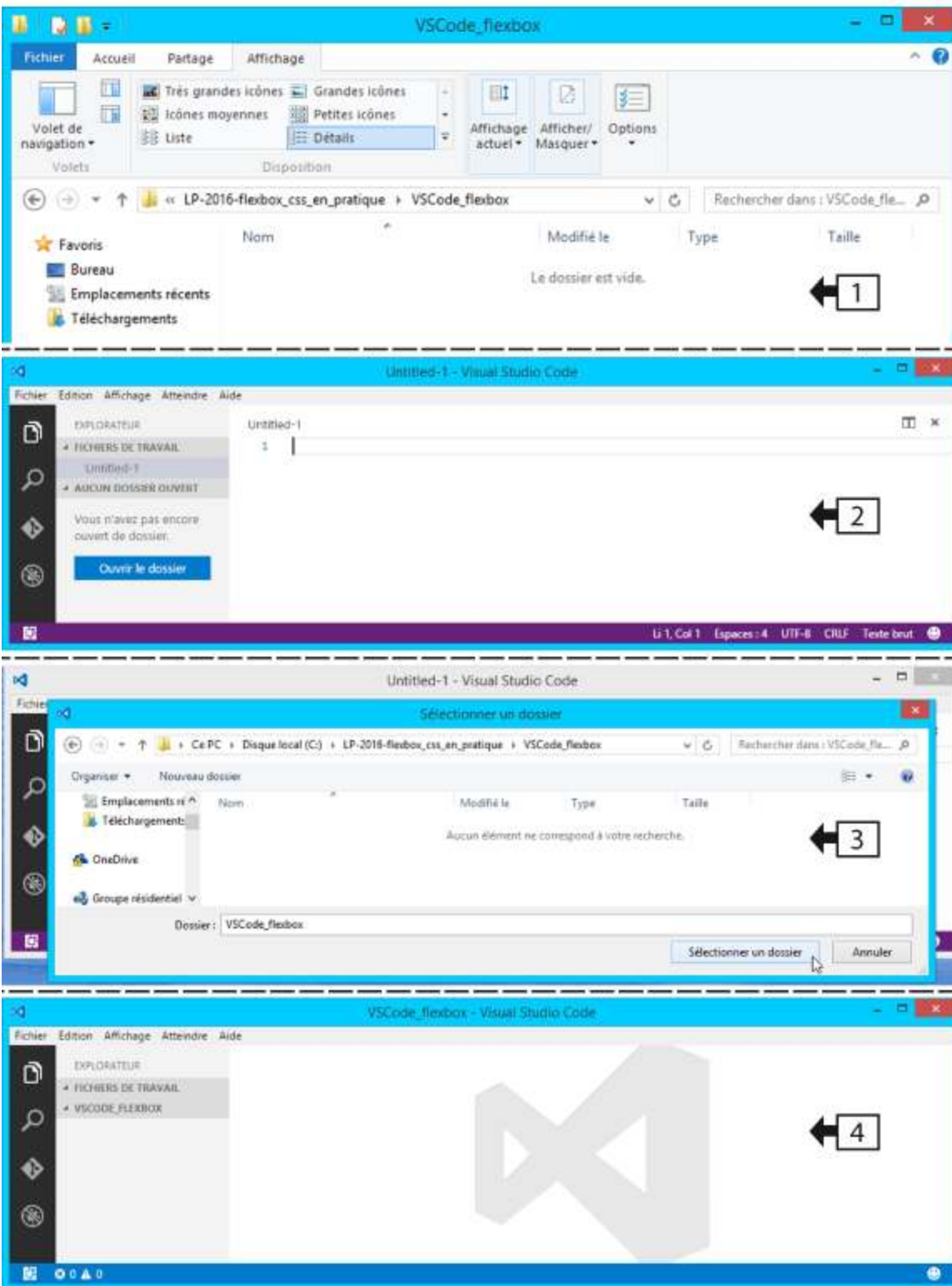
Visual Studio Code s'exécute de manière native sur le système d'exploitation de votre choix (Mac OS X, Linux et Windows). Grâce à la colorisation de la syntaxe, la mise en surbrillance et la concordance des parenthèses, c'est un éditeur de premier choix pour toutes vos tâches de programmation. Il prend en charge JavaScript, C#, C++, PHP, Java, HTML, R, CSS, SQL, Markdown, TypeScript, LESS, SASS, JSON, XML et Python ainsi que de nombreux autres formats de fichiers courants. Il possède une modification avancée, rapide et adaptée pour clavier, avec plusieurs curseurs. L'assistant visuel complète automatiquement les descriptions des API lorsque vous les saisissez pour une plus grande rapidité et une plus grande précision. Diagnostiquez les problèmes de votre application avec les outils intégrés de débogage pour Node, TypeScript et JavaScript. Définissez des points d'arrêt dans votre code, des arrêts en cas d'exception et des variables espions, examinez votre code ou naviguez en remontant la pile des appels, ou rattachez-vous aux processus d'exécution locaux. Adoptez les flux de travail modernes grâce à la puissance et à la flexibilité de Git. Visual Studio Code prend en charge le processus de développement des applications ASP.NET 5 et Node.js de bout en bout. La version actuelle de Visual Studio Code est la version 1.19 en français, et c'est celle utilisée tout au long de ce livre. Une documentation en ligne est disponible à l'adresse web

<http://code.visualstudio.com/Docs>.

Sur le disque, on ajoute un dossier nommé *VSCode\_flexbox* à un emplacement choisi (figure 1.3 au repère 1). On démarre le logiciel Visual Studio Code en double-cliquant sur son raccourci placé sur le bureau (figure 1.3 au repère 2). Normalement, l'explorateur est affiché sur le côté gauche de la fenêtre. Si ce n'est pas le cas, on le rend visible par la séquence des touches *Ctrl + Shift + E* ou bien par le menu *Affichage -> Explorateur*. Dans l'explorateur se trouve un bouton sur fond bleu nommé *Ouvrir un dossier*

(figure 1.3 au repère 2). On clique dessus et on sélectionne le dossier *VSCode\_flexbox* dans la boîte de dialogue Windows pour la sélection d'un répertoire (figure 1.3 au repère 3). En refermant la boîte de dialogue, le dossier *VSCode\_flexbox* est affiché dans l'explorateur du logiciel (figure 1.3 au repère 4) avec son contenu par une arborescence de dossiers et de fichiers.

Figure 1-3

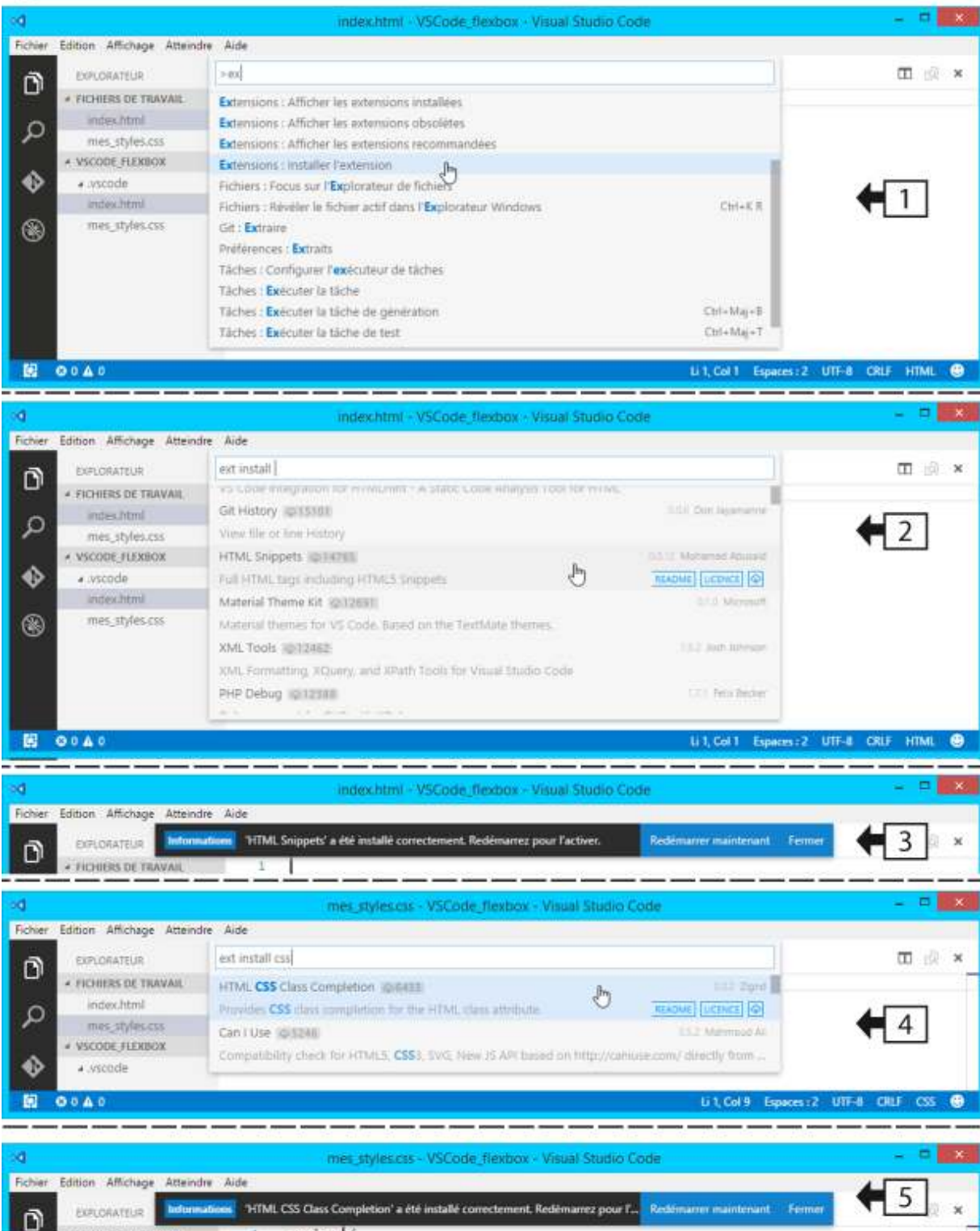


### 3 - Installer des extensions



Le logiciel Visual Studio Code est un éditeur de texte très élaboré et doté d'un ensemble d'extensions que l'on peut ajouter pour être encore plus performant dans l'écriture et la rédaction des codes de programmation. Nous allons ajouter deux extensions pour permettre une meilleure prise en charge des balises HTML et CSS par l'assistant visuel (*intellisense*) du logiciel. En tapant la touche *F1*, la séquence des touches *Ctrl + Shift + P* ou bien en choisissant le menu *Affichage -> Palette de commandes*, on affiche une palette des commandes qui permet d'accéder directement à une commande choisie dans la liste déroulante. Comme le montre la figure 1.4 au repère 1, dans le champ texte prévu à cet effet, on tape la séquence "*ext*" et on sélectionne dans la liste déroulante la rubrique "*Extensions: installer l'extension*". On appuie sur la touche *Entrée*. La liste déroulante propose alors un ensemble d'extensions disponibles au téléchargement. Comme le montre la figure 1.4 au repère 2, on sélectionne dans la liste déroulante en premier l'extension nommée *HTML Snippets* et on tape sur la touche *Entrée*. Le logiciel télécharge en arrière-plan l'extension puis l'installe dans la foulée. Quand c'est fait, une barre d'informations s'affiche en haut de la fenêtre (figure 1.4 au repère 3). On clique sur le bouton *Redémarrer maintenant*. Le logiciel se ferme et redémarre avec l'extension installée (au passage, il restaure l'explorateur tel qu'il était auparavant). On refait la même manipulation pour installer l'extension *HTML CSS Class Completion* comme le montre la figure 1.4 aux repères 4 et 5. A ce stade, nous avons un ensemble d'extensions qui vont nous faciliter la vie dans l'écriture du code de programmation en HTML et en CSS.

Figure 1-4



## 4 - Ajouter des dossiers et des fichiers

A la racine du dossier *VSCode\_flexbox*, nous allons ajouter deux fichiers nommés *index.html* de type HTML et *mes\_styles.css* de type CSS. Comme le montre la figure 1.5 aux repères 1 et 2, on clique sur l'icône *nouveau fichier* ou bien on choisit le menu *Fichier -> Nouveau fichier*. Un champ texte s'affiche dans le dossier et on entre la séquence *index.html* pour le fichier au format HTML et *mes\_styles.css* pour le fichier au format CSS. On obtient ainsi les deux fichiers au format choisi comme le montre la figure 1.5 aux repères 4 et 5. Le logiciel détecte automatiquement le type de format en fonction de l'extension du fichier. Sur la figure 1.5 au repère 6, l'extension du fichier correspondant s'affiche au bas de la fenêtre côté droit. Le paramétrage par défaut du logiciel permettra d'afficher le contenu de ces deux fichiers en utilisant la coloration syntaxique par défaut.