


TP0 : Découverte de Visual Studio Code

Ayoub KARINE (ayoub.karine@isen-ouest.yncrea.fr)

Installation

1. Télécharger Visual Studio (VS) Code (et non pas Visual Studio) à partir du site : <https://code.visualstudio.com/>
2. Procéder à l'installation de VS code en gardant les paramètres par défaut
3. Créer sur vos machines un nouveau dossier appelé **testvscode**
4. Ouvrir ce dossier dans VS code en utilisant l'une des deux techniques:
 - a. File>Open Folder
 - b. Glisser le dossier directement dans VS code
5. Créer un nouveau fichier "testvscode.html" dans le dossier tp2 en utilisant l'une des deux techniques
 - a. File>New File
 - b. En cliquant sur le bouton  du menu :

6. Créer une architecture de base d'une page HTML validé sans erreur dans le site : <https://validator.w3.org/>. Remarquer que sur VS code :
 - a. Les balises fermées sont insérées automatiquement
 - b. L'indentation est automatique aussi

Snippets

7. Les **snippets** servent à faire référence à des bouts de code en tapant juste un "préfixe". Ils sont très utilisés en programmation vu que le temps d'un programmeur dans un projet est très précieux.

Plusieurs snippets sont déjà programmés sur VS code. Tester les snippets suivants :

- a. ul
 - b. ul>li
 - c. ul>li*5
 - d. div>h1+p+h2
8. Pour éviter de taper, à chaque fois, l'architecture de base d'une page HTML, nous allons créer un **snippet** :
- a. clic droit sur le code HTML puis choisir : command palette
 - b. Taper dans la recherche "snippet" puis choisir : Preferences
Configure User Snippet
 - c. cliquer sur html.json :
 - i. Dans prefix, taper par exemple html
 - ii. Dans body, insérer l'architecture de base HTML déjà validé de la question 6.
 - iii. Dans description, taper par exemple html architecture
 - iv. Tester le snippet dans la page HTML. Sur VS code, il y'a moyen de partager l'écran en deux colonnes : View > Edit Layout > Split Right. Appliquer cela pour avoir le fichier .html dans une colonne et le fichier .json dans la deuxième colonne

Raccourcis

9. Les raccourcis (shortcuts) sont aussi indispensables pour un programmeur (et ce, peu importe le langage utilisé)

a. Quelques raccourcis souvent utilisés sur VS code :

- i. `Ctrl` + `s` pour sauvegarder
- ii. `Ctrl` + `:` pour mettre un commentaire
- iii. `Ctrl` + `c` pour copier
- iv. `Ctrl` + `x` pour couper
- v. `Ctrl` + `v` pour coller
- vi. `Ctrl` + `n` pour créer un nouveau fichier
- vii. `Tab` pour l'indentation
- viii. `Shift` + `Tab` pour l'indentation inverse

b. Pour voir tous les raccourcis de VS code :

- i. `Ctrl` + `Shift` + `P` (ou clic droit command palette) pour ouvrir "Command Palette"
- ii. Taper Shortcuts et clic sur le premier résultat apparu

c. Modification et suppression d'un raccourcis

- i. `Ctrl` + `Shift` + `P` (ou clic droit command palette) pour ouvrir "Command Palette"
- ii. Taper Shortcuts et clic sur le premier résultat apparu
- iii. clic droit sur le nom de la commande et puis choisir "Change Keybinding". Éventuellement, une commande peut être supprimé via "Remove Keybinding"

Extensions

10. Plusieurs extensions très utiles sont disponibles aussi dans VS code. Je recommande d'installer quelques unes qui seront utiles pour la suite des séances de travaux pratiques :

a. Clic sur l'icône des extensions



b. Chercher, installer puis découvrir les extensions (chaque extension est fourni avec une documentation utile pour son utilisation) :

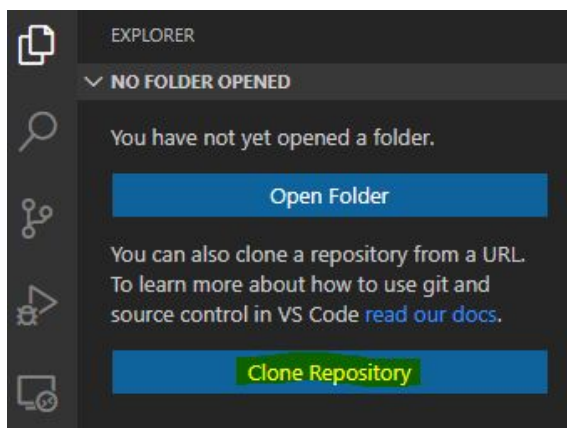
- i. **Open in browser** : pour ouvrir, depuis VS code, un fichier dans le navigateur
- ii. **Live Server** : modification en temps réel dans le navigateur. Attention, il faut enregistrer à chaque modification le fichier .html pour que le navigateur recharge la page web.
- iii. **Live Server Preview** : modification en temps réel dans un affichage interne à VS code. Attention, il faut enregistrer à chaque modification le fichier .html pour que le navigateur recharge la page web.
- iv. **Material Icon Theme** : pour une description artistique de l'extension de chaque fichier du projet

v. **Htmltagwrap** : pour insérer une sélection dans une balise

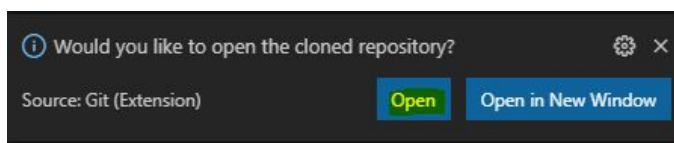
Github

On ne peut pas créer des "repository" directement à partir de VS code, on est obligé de créer au premier lieu la "repository" sur le site de Github

11. Créer un dossier vide dans votre ordinateur nommé "testgit"
12. Créer un compte Github : <https://github.com/>
13. Créer, sur <https://github.com/>, un "repository" nommé "testgit"
14. Ouvrir vscode puis cliquer sur Clone Repository



15. Dans le "repository URL", écrire le lien du "repository" puis cliquer sur entrer. Par exemple : <https://github.com/ayoubkarine/testgit>
16. Chercher le dossier "testgit" sur votre ordinateur puis cliquer sur "open" dans la fenêtre affichée

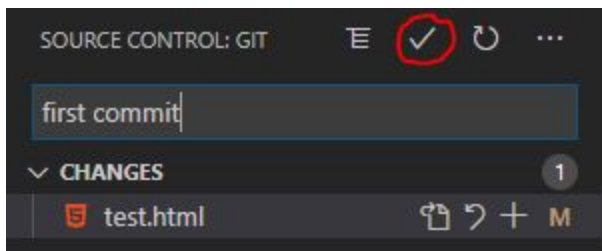


17. Créer un fichier html qui affiche "testgit"

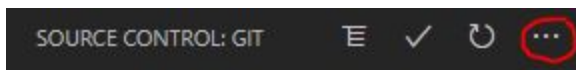
18. Cliquer sur  du menu :



19. Pour faire un commit (changement dans le repo local) cliquer sur :



20. Pour faire un push (changement dans le repo distant) cliquer sur :



Puis sur Push

21. Vérifier dans le site de Github si le fichier a été bien ajouté dans le "repository"

Débugueur Chrome sur VS code

22. Dans File → Preferences → Extensions, taper "Debugger for chrome" et procéder à son installation

23. Créer un nouveau dossier et puis l'ouvrir sur VS code. Dans ce dossier, il est demandé de créer un fichier HTML qui appelle un fichier JS dont le contenu est :

```
let a = "test"
console.log(a)
```

24. Dans Run → Open Configurations, remplacer le contenu du fichier launch.json par :

```

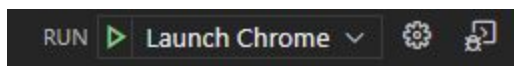
{
  "version": "0.2.0",
  "configurations": [
    {
      "type": "chrome",
      "request": "launch",
      "name": "Launch Chrome against localhost",
      "file": "${workspaceFolder}"
    }
  ]
}

```

25. Cliquer sur  dans le menu :



26. Ajouter un breakpoint dans la ligne 2 du fichier JS puis clique sur 



et choisir le fichier HTML

27. Tester les différentes fonctionnalités de débogage :

