

# Projet Responsive Web Design SN2

## Objectifs :

1. Réaliser une page personnelle de présentation (type CV en ligne)
2. Héberger la page sur Github Pages

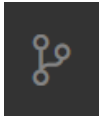
## I. Prérequis : Configuration de Git

Une fois VS Code installé, utiliser le raccourci clavier **Ctrl + Shift + P** afin d'ouvrir la palette de commandes. Une fois ouverte, taper **"settings json"** et sélectionner la ligne **Preferences : Open Settings (JSON)**.

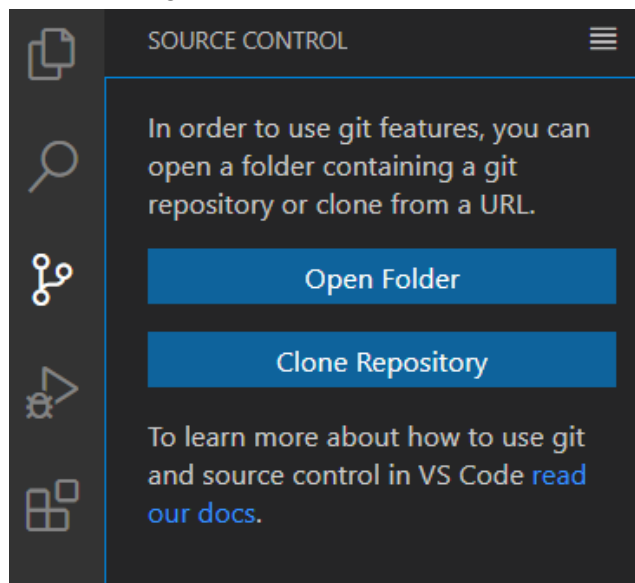
Dans le fichier qui s'ouvre, coller le texte suivant :

```
{
  "git.path": "C:\\laragon\\bin\\git\\bin\\git.exe",
  "terminal.integrated.shell.windows":
"C:\\Laragon\\bin\\cmd\\cmd.exe",
  "terminal.integrated.shellArgs.windows": "."
}
```

Sauvegarder (Ctrl + S) et fermer l'onglet, puis redémarrer VS Code. Pour vérifier que les préférences ont bien été enregistrées, cliquer sur l'icône **"Source Control"** dans la barre de gauche :



La barre de gauche doit se développer et afficher :



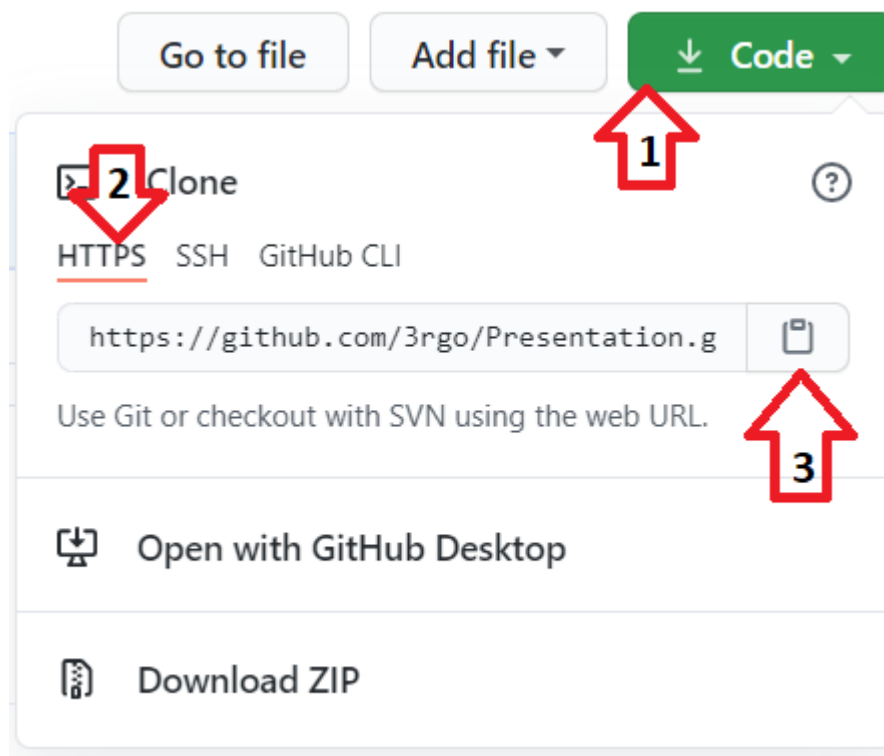
## Dépôt de code sur GitHub (<https://pages.github.com/>)

Tout d'abord, créer un compte sur GitHub (<https://github.com/>), qui vous permettra de partager du code en "open source" (source ouverte, librement accessible). Ensuite, une fois le compte créé, créer un projet :

- En haut à droite de l'écran, cliquer sur +, puis sur **New repository**
- Dans **repository name**, saisir votre pseudonyme Github, suivi de ".github.io" (par exemple : "grosnounoursdu31.github.io")  
**Attention : n'utilisez pas de caractères accentués, ni d'espaces (mettre un tiret ou tiret-bas)**
- (Facultatif) Saisir une description
- Cocher la case **Add a README**
- Valider la création en cliquant sur le bouton **Create repository**

Une fois le repository créé :

- Cliquer sur le bouton vert **Code**
- Puis sur l'onglet **HTTPS** (sauf si vous avez déjà configuré une clé SSH)
- Puis copier l'URL qui s'affiche



Ensuite dans VS Code, sur l'onglet **Source Code** (cf. *partie précédente*), cliquer sur le bouton **Clone Repository**, coller l'URL du dépôt copié sur GitHub, et presser **Entrée**.

VS Code demande ensuite l'emplacement où télécharger le code source. N'importe quel emplacement peut faire l'affaire mais pour gagner du temps, on choisira de le mettre directement dans le dossier "**www**" de Laragon (**C:\laragon\www**). Une fois le code téléchargé (quasiment instantané car il n'y a qu'un fichier vide à récupérer), une popup en bas à droite propose d'ouvrir le dépôt téléchargé : cliquer sur **Open** pour le faire. Sinon on

pourra toujours le faire grâce au menu **Fichier > Ouvrir un dossier** (**File > Open folder**).

## Familiarisation avec Git

Actions (commandes) de base :

- `git init` : création d'un dépôt
- `git clone [XXX]` : clonage d'un dépôt existant
- `git add [XXX]` : ajout d'un ou plusieurs fichiers à la version courante
- `git commit -m "[XXX]"` : enregistrement de la version courante avec un message
- `git push [origin] [master]` : envoi des nouvelles version sur la branche ([master]) d'un dépôt distant ([origin])
- `git checkout [XXX]` : déplacement entre versions/branches
- `git pull` : récupération des modifications distantes (travail collaboratif)

Tutoriel (simplifié) : <https://rogerdudler.github.io/git-guide/index.fr.html>

*N.B. : Vous pouvez arrêter la lecture lorsque vous atteignez la section "tags". La suite n'est pas nécessaire pour ce cours, mais pourra vous être utile plus tard*

Toutes ces commandes sont à lancer via un terminal (celui de VS Code ou de Windows/macOS), mais elles sont également présentes via les menus de VS Code.

Tutoriel Git + VSCode : <https://docs.microsoft.com/fr-fr/learn/modules/use-git-from-vs-code/>

Avant la première utilisation de Git, il faut le configurer avec votre nom et email, à l'aide des deux commandes suivantes :

- `git config --global user.name "Jean Lassalle"`
- `git config --global user.email "john.laroom@64forever.fr"`

## II. Création d'une page web responsive

Créer une page web respectant les consignes suivantes :

- La page doit contenir du HTML et du CSS (JavaScript facultatif)
- La structure de page doit utiliser les balises HTML sémantiques
- La disposition de la page doit utiliser une Grid ou le système Flexbox
- Le rendu doit être adapté à la consultation sur smartphone et tablette (responsive)

Le choix du contenu et de sa mise en forme est libre, mais un site personnel de type portfolio/CV est un bon outil à avoir pour qui compte travailler dans le monde du développement et/ou du digital.

N'hésitez pas à consulter des sites existants (cherchez "exemple site personnel" ou "personal page template" sur Google) pour trouver l'inspiration. Exemples de templates : <https://colorlib.com/wp/cat/personal/>

Vous pouvez bien entendu utiliser des librairies front (Bootstrap, Bulma, Tailwind, Materialize, ...)

### III. Déploiement du code

Poussez sur Github !

### IV. Rendu & Critères de notation

Afin de pouvoir être noté, chaque apprenant devra envoyer par email les 2 éléments suivants :

- L'adresse du dépôt de code sur GitHub (attention de bien avoir "git push" votre code)
- L'URL pour accéder à la page

#### **Critères de notation :**

- 14 pts : Page Web
  - 2 pts : Utilisation du HTML et du CSS
  - 2 pts : Utilisation des balises sémantiques
  - 4 pts : Utilisation d'une grid ou du flexbox
  - 8 pts : Affichage adapté aux mobiles et tablettes
- 4 pts : Hébergement Github Pages fonctionnel
- Bonus :
  - +1 point pour l'utilisation d'un framework front
  - +1 point pour l'utilisation (judicieuse) de JavaScript
  - +1 point pour la présence d'un GIF ou d'une photo marrante de vous

La beauté et l'esthétisme étant des critères subjectifs, ceux-ci ne rentrent pas en compte dans la notation. Seuls comptent le respect des critères et d'avoir fait l'effort de mettre un minimum de contenu sur la page.

L'hébergement Github Pages est gratuit à vie (sans aucune manipulation à faire). Donc n'hésitez pas à faire évoluer progressivement votre site. Par exemple :

- Analyse de visites avec Google Analytics
- Conception d'une meilleure interface
- Formulaire de contact (avec un outil comme [Formspree](https://formspree.io/) par exemple car Github Pages ne permet que d'utiliser du HTML/CSS/JS)
- ...