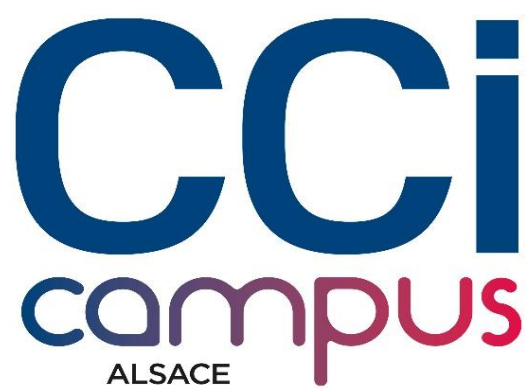


HENRY Alexis,

23/02/2022



Découverte de Sass

The word 'Sass' written in a stylized, cursive script font. The letters are a vibrant pink color. The 'S' is large and loops around, and the 'ss' at the end has a long, flowing tail.

SOMMAIRE

<https://alxishenry.github.io/Learn-to-Use-SASS/>

I. Introduction.

II. Machine de démonstration.

III. Exploration Sass .

IV. Ressources utilisées.

V. Liens utiles.

I- INTRODUCTION

Le SASS, Syntactically Awesome Stylesheets, est un langage compilé, interprété en CSS.

Il permet d'utiliser des variables, des règles imbriquées, des mixins, des fonctions, etc... le tout avec une syntaxe entièrement compatible CSS.

Sass aide à garder les feuilles de style volumineuses bien organisées et facilite le partage de la conception au sein et entre les projets.

Sass nous permet de convertir des feuilles de styles réalisés en Sass, Scss, en Css.

Sass, est un langage assez complexe, car il n'utilise pas du tout la syntaxe basique de css. C'est-à-dire qu'il n'utilise pas d'accolades, de doubles points, de points virgules... Le moindre espace manquant peut provoquer une erreur.

Quant au Scss, qui lui utilise une syntaxe beaucoup plus agréable, en proposant les mêmes fonctionnalités.

Définition proposée par Wikipédia : « Sass (Syntactically awesome stylesheets) est un langage de script préprocesseur qui est compilé ou interprété en CSS (Cascading Style Sheets). SassScript est le langage de script en lui-même. Sass est disponible en deux syntaxes. La syntaxe originale, appelée "syntaxe indentée"³ qui utilise l'indentation pour séparer les blocs de code et les sauts de ligne pour les séparer les directives. La nouvelle syntaxe, "SCSS", utilise les mêmes séparateurs de blocs que CSS. Les fichiers de la syntaxe indentée et SCSS utilisent respectivement les extensions .sass et .scss »

II- MACHINE DE DEMONSTRATION

utiliser obligatoirement multipass comme « virtualisateur »,

Installation de multipass depuis : <https://multipass.run/>.

Utilisation de multipass comme « virtualisateur ».

```
PS C:\Windows\system32> multipass list
No instances found.
PS C:\Windows\system32> multipass launch --name Sass_Instance
launch failed: Invalid arguments supplied
Invalid instance name supplied: Sass_Instance
PS C:\Windows\system32> multipass launch --name sassInstance
Creating sassInstance \
```

multipass list : permet de lister les instances créées.

multipass launch --name NAME : permet de créer une instance avec un nom

J'ai donc créé une instance nommée sassInstance.

```
Invalid instance name supplied: Sass_Instance
PS C:\Windows\system32> multipass launch --name sassInstance
Starting sassInstance -
```

Une fois l'instance créée, je liste les instances actuelles :

```
PS C:\Windows\system32> multipass list
Name                State      IPv4          Image
sassInstance        Running    172.20.25.45 Ubuntu 20.04 LTS
PS C:\Windows\system32>
```

On voit donc bien sur ce screen, que l'instance a été créée, est bien sous Ubuntu.

Ainsi qu'une IP lui a bien été attribuée : 172.20.25.45, ce qui signifie qu'on pourra y accéder depuis l'host, et ainsi modifier le nom de domaine lié à celle-ci.

La commande ci-dessous permet de se connecter en shell à l'instance :

```
PS C:\Windows\system32> multipass shell sassInstance
```

Une fois connecté, on a tout d'abord un récapitulatif de la configuration de l'instance :

```
System load: 0.27          Processes:                115
Usage of /: 28.1% of 4.67GB Users logged in:         0
Memory usage: 45%         IPv4 address for eth0: 172.20.25.45
Swap usage: 0%
```

```
PS C:\Windows\system32> multipass shell sassInstance
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-99-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Wed Feb 23 11:38:43 CET 2022

System load: 0.27          Processes:                115
Usage of /: 28.1% of 4.67GB Users logged in:         0
Memory usage: 45%         IPv4 address for eth0: 172.20.25.45
Swap usage: 0%

1 update can be applied immediately.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@sassInstance: $
```

La dernière ligne : 'ubuntu@sassInstance' nous permet de savoir qu'on est connecté.

Une fois connecté, la première chose à faire est de vérifier l'os installé :

```
ce:~$ cat /proc/version  
Linux 5.0-99-generic (buildd@lgw01-amd64-007) (gcc version 9.3.0 (Ubuntu 9.3.0-17ubuntu1~20.04))
```

Ensuite, j'ai mis à jour le système avec la commande ci-dessous :

```
ubuntu@sassInstance:~$ ls^C  
ubuntu@sassInstance:~$ sudo apt update && apt upgrade -y
```

installer Apache et Sass dans la machine virtuelle obtenue

Ce qu'on le souhaite est héberger une page web, donc pour ceci il faut créer un serveur web.

Cela nécessite l'installation d'Apache via la ligne de commande suivante :

```
ubuntu@sassInstance:~$  
ubuntu@sassInstance:~$  
ubuntu@sassInstance:~$ sudo apt install apache2_
```

La commande ci-dessous permet de vérifier le statut d'Apache :

```
ubuntu@sassInstance:~/var/www/html$ systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enable
   Active: active (running) since Wed 2022-02-23 12:34:07 CET; 7min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 651 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
  Main PID: 680 (apache2)
    Tasks: 55 (limit: 1054)
   Memory: 8.7M
   CGroup: /system.slice/apache2.service
           └─680 /usr/sbin/apache2 -k start
             └─681 /usr/sbin/apache2 -k start
               └─682 /usr/sbin/apache2 -k start

Feb 23 12:34:02 sassInstance systemd[1]: Starting The Apache HTTP Server...
Feb 23 12:34:04 sassInstance apachectl[668]: AH00558: apache2: Could not reliably deter
Feb 23 12:34:07 sassInstance systemd[1]: Started The Apache HTTP Server.
ubuntu@sassInstance:~/var/www/html$
```

Ensuite, l'énoncé nous demande d'installer Sass :

```
ubuntu@sassInstance:~$
ubuntu@sassInstance:~$ sudo apt-get install sass
```

```
ubuntu@sassInstance:~$ sudo apt-get install sass
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'ruby-sass' instead of 'sass'
The following additional packages will be installed:
  fonts-lato javascript-common libjs-jquery libruby2.7 rake ruby ruby-minitest ruby-net-telnet ruby-power-assert ruby-test-unit ruby-xmlrpc ruby-x
  unzip zip
Suggested packages:
  apache2 | lighttpd | httpd ri ruby-dev ruby-sass-listen bundler
The following NEW packages will be installed:
  fonts-lato javascript-common libjs-jquery libruby2.7 rake ruby ruby-minitest ruby-net-telnet ruby-power-assert ruby-sass ruby-test-unit ruby-x
  rubygems-integration unzip zip
0 upgraded, 16 newly installed, 0 to remove and 0 not upgraded.
Need to get 7409 kB of archives.
After this operation, 33.7 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

L'installation de Sass a été réalisée ci-dessus.

Lors de l'installation d'Apache, celui-ci reçoit une configuration de base qui est la suivante :

Contenu du dossier /var/www/html, qui contient le fichier affiché par Apache de base.

```
ubuntu@sassInstance:~$ ls -l /var/www/html
total 12
-rw-r--r-- 1 root root 10918 Feb 23 11:54 index.html
ubuntu@sassInstance:~$
```

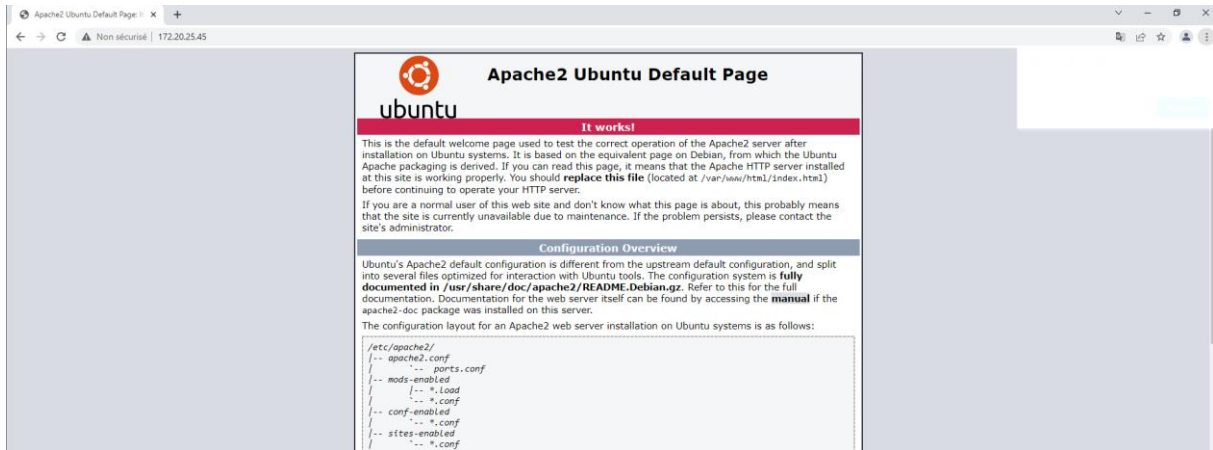
Contenu du dossier /etc/apache2/sites-available, qui contient la configuration des sites :

```
ubuntu@sassInstance:/etc/apache2/sites-available$ ls -al
total 20
drwxr-xr-x 2 root root 4096 Feb 23 11:54 .
drwxr-xr-x 8 root root 4096 Feb 23 11:54 ..
-rw-r--r-- 1 root root 1332 Oct  1  2020 000-default.conf
-rw-r--r-- 1 root root 6338 Oct  1  2020 default-ssl.conf
```

L'adresse IP liée à la machine est 172.20.25.45.

```
ubuntu@sassInstance:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 52:54:00:25:8d:74 brd ff:ff:ff:ff:ff:ff
    inet 172.20.25.45/20 brd 172.20.31.255 scope global dynamic eth0
        valid_lft 85196sec preferred_lft 85196sec
    inet6 fe80::5054:ff:fe25:8d74/64 scope link
        valid_lft forever preferred_lft forever
```

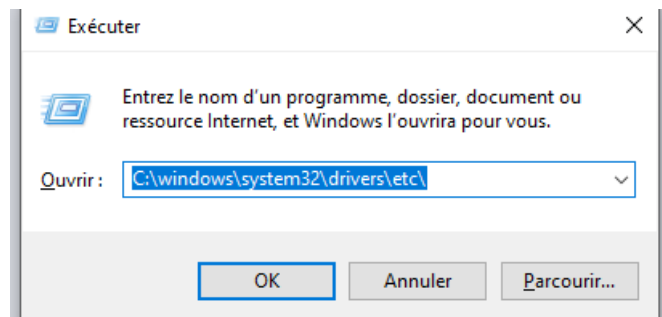

Page index.html présente dans le dossier racine d'Apache. Celui-ci est affiché de base lors de la connexion à l'adresse IP de l'instance.



machine hôte affiche en suivant l'URL :<http://www.nom-prenom.fr>

Pour faire en sorte que l'accès à la page web de la machine ne se fasse plus par l'adresse IP mais par un nom de domaine local à notre PC, il faut modifier le fichier host de Windows.

Voici le chemin pour se rendre au fichier host :



J'ai décidé de modifier le fichier host via Powershell :

```
PS C:\Windows\system32\drivers> cd etc
PS C:\Windows\system32\drivers\etc> ls

Répertoire : C:\Windows\system32\drivers\etc

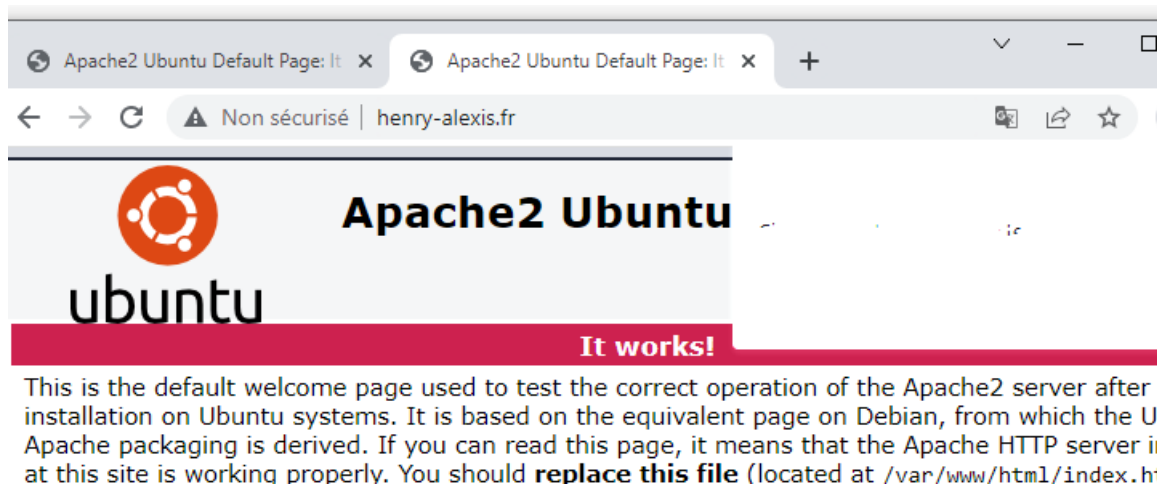
Mode                LastWriteTime         Length Name
----                -
-a----            22/02/2022   14:47           861 hosts
-a----            23/02/2022   11:35           551 hosts.ics
-a----            07/12/2019   10:12          3683 lmhosts.sam
-a----            07/12/2019   10:12           407 networks
-a----            07/12/2019   10:12          1358 protocol
-a----            07/12/2019   10:12         17635 services

PS C:\Windows\system32\drivers\etc> notepad hosts
PS C:\Windows\system32\drivers\etc> notepad hosts_
```

La commande [Notepad file] permet d'ouvrir le fichier 'hosts' dans le bloc-notes.

```
Last login: Wed Feb 23 11:38:44 2022 # 102.24.24.27 rmlno.acme.com # source server
ubuntu@sassInstance:~$ ip a # 38.25.63.10 x.acme.com # x client host
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 # localhost name resolution is handled within DNS itself.
   link/loopback 00:00:00:00:00:00 # 127.0.0.1 localhost
   inet 127.0.0.1/8 scope host lo # ::1 localhost
     valid_lft forever preferred_lft 172.31.157.170 www.henry-alexis.fr
   inet6 ::1/128 scope host
     valid_lft forever preferred_lft
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP>
   link/ether 52:54:00:25:8d:74 brd ff:ff:ff:ff:ff:ff
   inet 172.31.157.170/20 brd 172.31.157.255 scope global eth0
     valid_lft 86085sec preferred_lft 86400sec
   inet6 fe80::5054:ff:fe25:8d74/64 scope link
     valid_lft forever preferred_lft
```

Une fois le fichier host configuré, on peut donc accéder à la machine par le nom de domaine indiqué :



ajouter une page HTML au bon endroit dans la machine virtuelle, affichant le texte « Découverte de Sass par NOM Prénom.

J'ai décidé de garder la configuration de base d'Apache, et de mettre en place mon site à côté de celle-ci.

Pour cela, j'ai tout d'abord créé un dossier qui va contenir le contenu du site :

```
ubuntu@sassInstance:~/var/www$ cd
ubuntu@sassInstance:~$ ls -l /var/www/html
total 16
drwxr-xr-x 2 root root 4096 Feb 23 12:41 henry-alexis.fr
-rw-r--r-- 1 root root 10918 Feb 23 11:54 index.html
ubuntu@sassInstance:~$ p
```

J'ai ensuite copié le fichier de configuration du site de base d'Apache, pour l'adapter à mon site et le configurer de sorte qu'il redirige les requêtes vers ce domaine sur le bon dossier.

```
ubuntu@sassInstance:/etc/apache2/sites-available$ sudo cp 000-default.conf henry-alexis-fr.conf
ubuntu@sassInstance:/etc/apache2/sites-available$ ls -al
total 24
drwxr-xr-x 2 root root 4096 Feb 23 12:49 .
drwxr-xr-x 8 root root 4096 Feb 23 11:54 ..
-rw-r--r-- 1 root root 1332 Oct 1 2020 000-default.conf
-rw-r--r-- 1 root root 6338 Oct 1 2020 default-ssl.conf
-rw-r--r-- 1 root root 1332 Feb 23 12:49 henry-alexis-fr.conf
```

Ci-dessous, la configuration insérée dans le fichier henry-alexis-fr.conf.

Celui-ci va donc rediriger les requêtes sur henry-alexis.fr, www.henry-alexis.fr vers le document root /var/www/html/henry-alexis.fr.

```
ubuntu@sassInstance:/etc/apache2/sites-available$ ls -l
total 16
-rw-r--r-- 1 root root 1332 Oct 1 2020 000-default.conf
-rw-r--r-- 1 root root 6338 Oct 1 2020 default-ssl.conf
-rw-r--r-- 1 root root 1409 Feb 23 12:50 henry-alexis-fr.conf
ubuntu@sassInstance:/etc/apache2/sites-available$ cat henry-alexis-fr.conf
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    ServerAlias henry-alexis.fr
    ServerName www.henry-alexis.fr
    DocumentRoot /var/www/html/henry-alexis.fr

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
</VirtualHost>
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Pour que cela soit effectif, il faut activer le site avec la commande suivante :

```
ubuntu@sassInstance:/etc/apache2/sites-available$ sudo a2ensite henry-alexis-fr.conf
Enabling site henry-alexis-fr.
To activate the new configuration, you need to run:
  systemctl reload apache2
```

Il faut ensuite redémarrer Apache pour que les changements soit bien pris en compte :

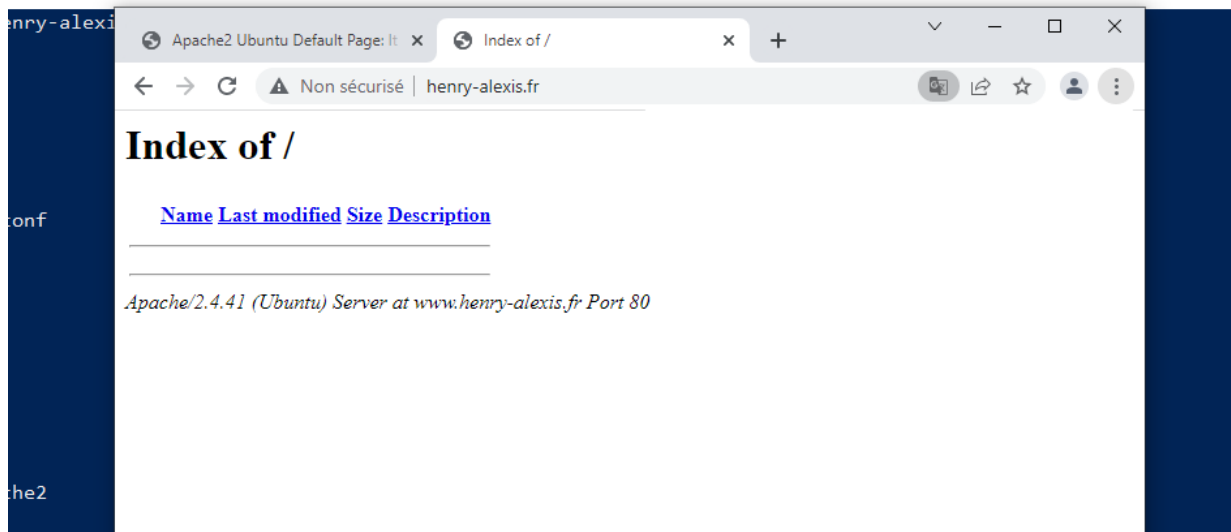
```
systemctl reload apache2
ubuntu@sassInstance:/etc/apache2/sites-available$ systemctl reload apache2
```

Une fois cela réalisé, si on regarde le contenu des dossiers /etc/apache2/sites-enabled et sites-available,

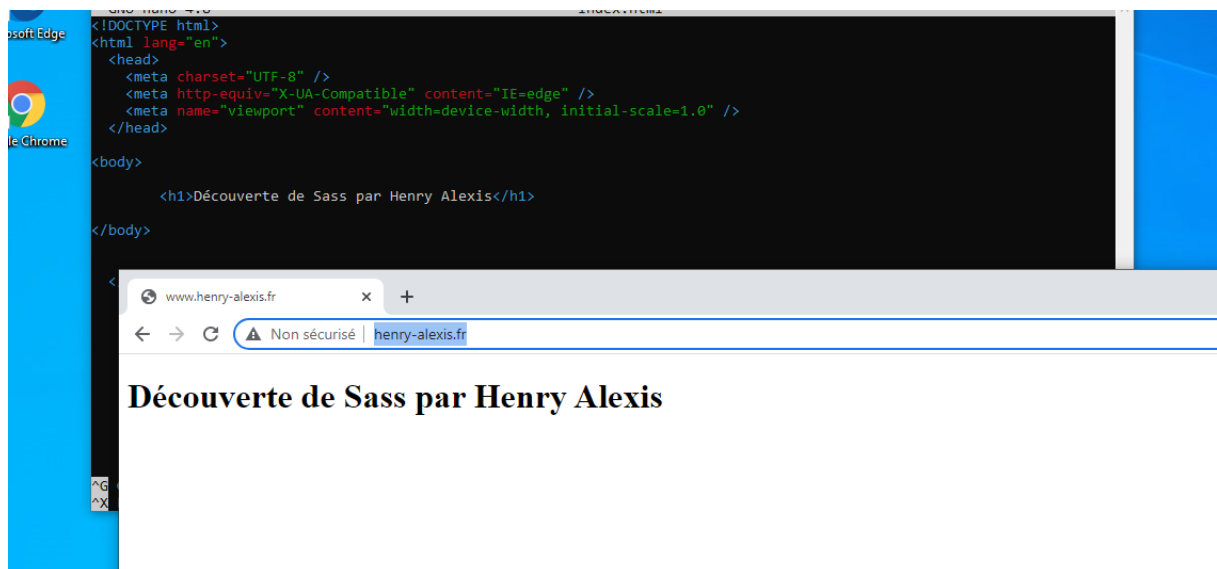
```
ubuntu@sassInstance:~$ ls -al /etc/apache2/sites-available/ && ls -al /etc/apache2/sites-enabled/
total 24
drwxr-xr-x 2 root root 4096 Feb 23 12:50 .
drwxr-xr-x 8 root root 4096 Feb 23 11:54 ..
-rw-r--r-- 1 root root 1332 Oct  1  2020 000-default.conf
-rw-r--r-- 1 root root 6338 Oct  1  2020 default-ssl.conf
-rw-r--r-- 1 root root 1409 Feb 23 12:50 henry-alexis-fr.conf
total 8
drwxr-xr-x 2 root root 4096 Feb 23 12:52 .
drwxr-xr-x 8 root root 4096 Feb 23 11:54 ..
lrwxrwxrwx 1 root root   35 Feb 23 11:54 000-default.conf -> ../sites-available/000-default.conf
lrwxrwxrwx 1 root root   39 Feb 23 12:52 henry-alexis-fr.conf -> ../sites-available/henry-alexis-fr.conf
```

On remarque que le dossier de configuration de henry-alexis-fr.conf a été placé dans sites-enabled. Ce qui signifie que le fichier de configuration est opérationnel.

Une fois qu'on se rend sur la page, on tombe sur un Index of /, cela est dû au fait que je n'ai pas encore créé de fichier html dans le dossier relié à ce domaine.



Je créer alors le fichier ci-dessous avec le texte demandé dans l'énoncé :



On peut voir que sur la page internet ci-dessus, il affiche le résultat attendu.

Version d'Apache (apache2 -v) :

```
ubuntu@sassInstance:~$ apache2 -v
Server version: Apache/2.4.41 (Ubuntu)
Server built:   2022-01-05T14:49:56
ubuntu@sassInstance:~$
```

Version de Sass (sass -v) :

```
ubuntu@sassInstance:~$ sass -v
Ruby Sass 3.7.4
ubuntu@sassInstance:~$
```

Contenu du dossier sites-available (ls -al /etc/apache2/sites-available) :

```
ubuntu@sassInstance:~$ ls -al /etc/apache2/sites-available/
total 24
drwxr-xr-x 2 root root 4096 Feb 23 12:50 .
drwxr-xr-x 8 root root 4096 Feb 23 11:54 ..
-rw-r--r-- 1 root root 1332 Oct  1  2020 000-default.conf
-rw-r--r-- 1 root root 6338 Oct  1  2020 default-ssl.conf
-rw-r--r-- 1 root root 1409 Feb 23 12:50 henry-alexis-fr.conf
```

Contenu du dossier sites-enabled (ls -al /etc/apache2/sites-enabled) :

```
ubuntu@sassInstance:~$ ls -al /etc/apache2/sites-enabled/
total 8
drwxr-xr-x 2 root root 4096 Feb 23 12:52 .
drwxr-xr-x 8 root root 4096 Feb 23 11:54 ..
lrwxrwxrwx 1 root root  35 Feb 23 11:54 000-default.conf -> ../sites-available/000-default.conf
lrwxrwxrwx 1 root root  39 Feb 23 12:52 henry-alexis-fr.conf -> ../sites-available/henry-alexis-fr.conf
```

Contenu du dossier html (ls -al /var/www/html) :

```
ubuntu@sassInstance:~$ ls -al /var/www/html
total 24
drwxr-xr-x 3 root root 4096 Feb 23 12:41 .
drwxr-xr-x 3 root root 4096 Feb 23 11:54 ..
drwxr-xr-x 4 root root 4096 Feb 23 13:45 henry-alexis.fr
-rw-r--r-- 1 root root 10918 Feb 23 11:54 index.html
```

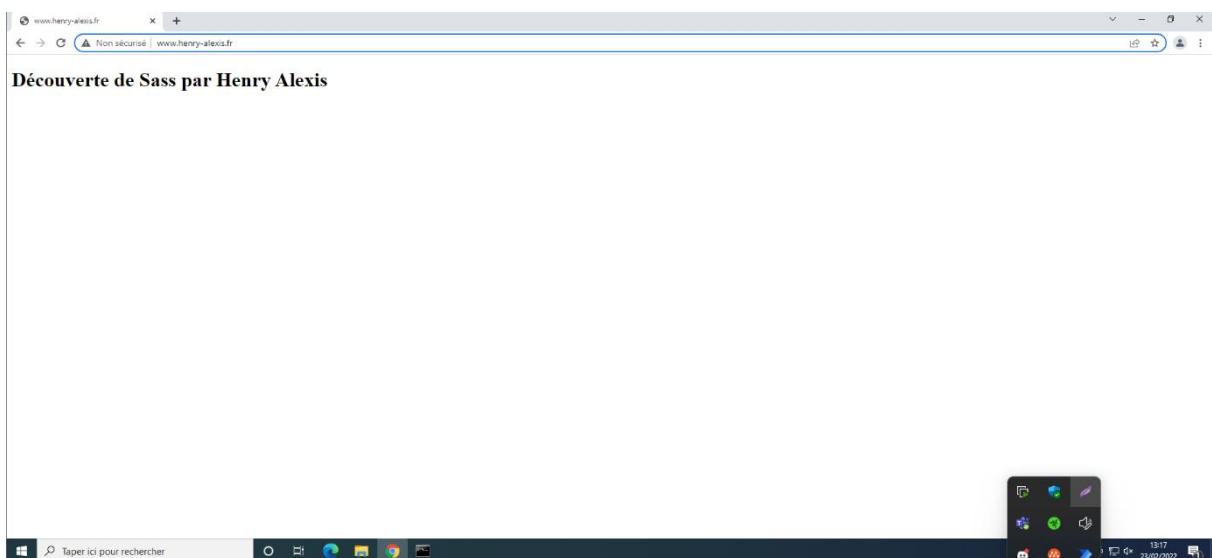
copie d'écran de la page du navigateur affichant la page créée (avec obligatoirement l'URL imposée).

Résultat de la page web

Nom de domaine modifié,

Fichiers de configurations modifiés, (renvoyant vers le chemin ci-dessous)

Code présent dans `/var/www/html/henry-alexis.fr/index.html`.



III- EXPLORATION SASS

préparer un fichier simple scss

Ci-dessous la création du fichier index.scss (compilé en css):

```
ubuntu@sassInstance:/var/www/html/henry-alexis.fr$ ls -al
total 12
drwxr-xr-x 2 root root 4096 Feb 23 13:14 .
drwxr-xr-x 3 root root 4096 Feb 23 12:41 ..
-rw-r--r-- 1 root root 295 Feb 23 13:09 index.html
ubuntu@sassInstance:/var/www/html/henry-alexis.fr$ touch index.scss
touch: cannot touch 'index.scss': Permission denied
ubuntu@sassInstance:/var/www/html/henry-alexis.fr$ sudo touch index.scss
ubuntu@sassInstance:/var/www/html/henry-alexis.fr$ ls -al
total 12
drwxr-xr-x 2 root root 4096 Feb 23 13:20 .
drwxr-xr-x 3 root root 4096 Feb 23 12:41 ..
-rw-r--r-- 1 root root 295 Feb 23 13:09 index.html
-rw-r--r-- 1 root root 0 Feb 23 13:20 index.scss
ubuntu@sassInstance:/var/www/html/henry-alexis.fr$
```

Ci-dessous la liaison du fichier css dans le l'index.html :

```
GNU nano 4.8
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="stylesheet" href="./index.css"/>
  </head>
  <body>
    <h1>Découverte de Sass par Henry Alexis</h1>
  </body>
</html>
```

Modifier la page HTML de découverte Sass

J'ai décidé de mettre une couleur sur la couleur du background de la page, pour cela j'ai créé une variable \$variableTest qui contient une couleur.

Cette variable est attribuée au background du body ci-dessous.

```
ubuntu@sassInstance:~$ nano index.html
$variableTest: #223c66;

body {
  background-color: $variableTest;
}
```

Compiler le fichier scss pour constater le résultat

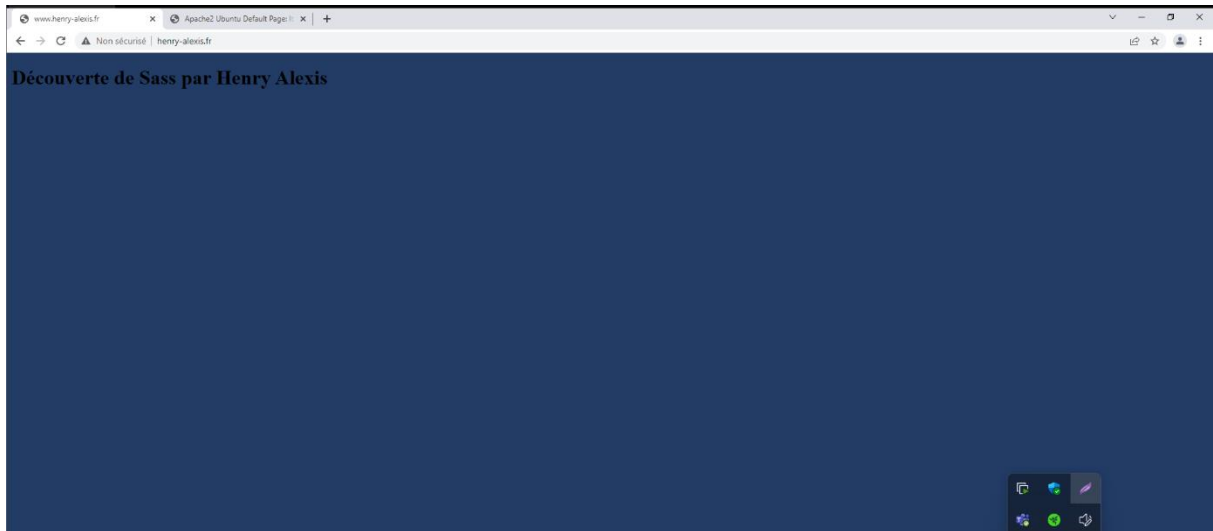
Ci-dessous la compilation du fichier scss en css :

```
ubuntu@sassInstance:~/var/www/html/henry-alexis.fr$ sudo nano index.scss
ubuntu@sassInstance:~/var/www/html/henry-alexis.fr$ sudo sass index.scss index.css
```

Cette commande génère les fichiers suivants :

```
ubuntu@sassInstance:~/var/www/html/henry-alexis.fr$ sudo sass index.scss index.css
ubuntu@sassInstance:~/var/www/html/henry-alexis.fr$ ls -al
total 28
drwxr-xr-x 3 root root 4096 Feb 23 13:23 .
drwxr-xr-x 3 root root 4096 Feb 23 12:41 ..
drwxr-xr-x 3 root root 4096 Feb 23 13:23 .sass-cache
-rw-r--r-- 1 root root 75 Feb 23 13:23 index.css
-rw-r--r-- 1 root root 125 Feb 23 13:23 index.css.map
-rw-r--r-- 1 root root 343 Feb 23 13:21 index.html
-rw-r--r-- 1 root root 68 Feb 23 13:22 index.scss
ubuntu@sassInstance:~/var/www/html/henry-alexis.fr$
```

Résultat de la page web avec Scss



La suite du document concerne l'installation et l'utilisation de Knacss.

Ci-dessous la récupération du repository Github de Knacss.

```
ubuntu@sassInstance: /var/www/html/henry-alexis.fr$ sudo wget https://codeload.github.com/alsacreations/KNACSS/zip/refs/heads/master
--2022-02-23 13:28:01-- https://codeload.github.com/alsacreations/KNACSS/zip/refs/heads/master
Resolving codeload.github.com (codeload.github.com)... 140.82.121.9
Connecting to codeload.github.com (codeload.github.com)[140.82.121.9]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/zip]
Saving to: 'master'

master [ <->
2022-02-23 13:28:02 (3.67 MB/s) - 'master' saved [48001]

ubuntu@sassInstance: /var/www/html/henry-alexis.fr$ ls -l
total 64
-rw-r--r-- 1 root root 77 Feb 23 13:26 index.css
-rw-r--r-- 1 root root 125 Feb 23 13:26 index.css.map
-rw-r--r-- 1 root root 343 Feb 23 13:21 index.html
-rw-r--r-- 1 root root 69 Feb 23 13:26 index.scss
-rw-r--r-- 1 root root 48001 Feb 23 13:28 master
ubuntu@sassInstance: /var/www/html/henry-alexis.fr$
```

Le dossier récupéré étant un fichier zip, il faut utiliser la commande suivante pour le rendre utilisable :

```
ubuntu@sassInstance: /var/www/html/henry-alexis.fr$
ubuntu@sassInstance: /var/www/html/henry-alexis.fr$
ubuntu@sassInstance: /var/www/html/henry-alexis.fr$
ubuntu@sassInstance: /var/www/html/henry-alexis.fr$
ubuntu@sassInstance: /var/www/html/henry-alexis.fr$ sudo unzip master_
```

Une fois cela fait, on peut supprimer le dossier zip.

```
ubuntu@sassInstance: /var/www/html/henry-alexis.fr$ ls -l
total 68
drwxr-xr-x 4 root root 4096 Jan 24 17:03 KNACSS-master
-rw-r--r-- 1 root root 77 Feb 23 13:26 index.css
-rw-r--r-- 1 root root 125 Feb 23 13:26 index.css.map
-rw-r--r-- 1 root root 343 Feb 23 13:21 index.html
-rw-r--r-- 1 root root 69 Feb 23 13:26 index.scss
-rw-r--r-- 1 root root 48001 Feb 23 13:28 master
ubuntu@sassInstance: /var/www/html/henry-alexis.fr$ sudo rm -rf master
```

Ensuite, on récupère seulement le dossier 'sass', car c'est le seul qui nous est utile. On le déplace donc dans la racine du site.

```
ubuntu@sassInstance:/var/www/html/henry-alexis.fr$
ubuntu@sassInstance:/var/www/html/henry-alexis.fr$ ls -l
total 68
drwxr-xr-x 4 root root 4096 Jan 24 17:03 KNACSS-master
-rw-r--r-- 1 root root 77 Feb 23 13:26 index.css
-rw-r--r-- 1 root root 125 Feb 23 13:26 index.css.map
-rw-r--r-- 1 root root 343 Feb 23 13:21 index.html
-rw-r--r-- 1 root root 69 Feb 23 13:26 index.scss
-rw-r--r-- 1 root root 48001 Feb 23 13:28 master
ubuntu@sassInstance:/var/www/html/henry-alexis.fr$ sudo rm -rf master
ubuntu@sassInstance:/var/www/html/henry-alexis.fr$ cd KNACSS-master/
ubuntu@sassInstance:/var/www/html/henry-alexis.fr/KNACSS-master$ ls `ls`
ls: cannot access '`ls`': No such file or directory
ubuntu@sassInstance:/var/www/html/henry-alexis.fr/KNACSS-master$ ls
LICENSE README.md changelog.md css gulpfile.js package.json sass
ubuntu@sassInstance:/var/www/html/henry-alexis.fr/KNACSS-master$ mv sass/
abstracts/ base/ components/ knacss.scss utils/
ubuntu@sassInstance:/var/www/html/henry-alexis.fr/KNACSS-master$ mv sass/
abstracts/ base/ components/ knacss.scss utils/
ubuntu@sassInstance:/var/www/html/henry-alexis.fr/KNACSS-master$ mv sass/
abstracts/ base/ components/ knacss.scss utils/
ubuntu@sassInstance:/var/www/html/henry-alexis.fr/KNACSS-master$ mv sass/ /var/www/html/henry-alexis.fr/_
```

Une fois cela fait, on peut supprimer le dossier KNACSS-master.

```
ubuntu@sassInstance:/var/www/html/henry-alexis.fr/KNACSS-master$ cd ..
ubuntu@sassInstance:/var/www/html/henry-alexis.fr$ ls -l
total 24
drwxr-xr-x 3 root root 4096 Feb 23 13:29 KNACSS-master
-rw-r--r-- 1 root root 77 Feb 23 13:26 index.css
-rw-r--r-- 1 root root 125 Feb 23 13:26 index.css.map
-rw-r--r-- 1 root root 343 Feb 23 13:21 index.html
-rw-r--r-- 1 root root 69 Feb 23 13:26 index.scss
drwxr-xr-x 6 root root 4096 Jan 24 17:03 sass
ubuntu@sassInstance:/var/www/html/henry-alexis.fr$ sudo rm -rf KNACSS-master/
```

Voici, ci-dessous le contenu du dossier 'sass' :

```
ubuntu@sassInstance:/var/www/html/henry-alexis.fr/sass$ ls -l
total 20
drwxr-xr-x 2 root root 4096 Jan 24 17:03 abstracts
drwxr-xr-x 2 root root 4096 Jan 24 17:03 base
drwxr-xr-x 2 root root 4096 Jan 24 17:03 components
-rw-r--r-- 1 root root 869 Jan 24 17:03 knacss.scss
drwxr-xr-x 2 root root 4096 Jan 24 17:03 utils
ubuntu@sassInstance:/var/www/html/henry-alexis.fr/sass$ ls -l
total 20
drwxr-xr-x 2 root root 4096 Jan 24 17:03 abstracts
drwxr-xr-x 2 root root 4096 Jan 24 17:03 base
drwxr-xr-x 2 root root 4096 Jan 24 17:03 components
-rw-r--r-- 1 root root 869 Jan 24 17:03 knacss.scss
drwxr-xr-x 2 root root 4096 Jan 24 17:03 utils
ubuntu@sassInstance:/var/www/html/henry-alexis.fr/sass$
ubuntu@sassInstance:/var/www/html/henry-alexis.fr/sass$
```

On ajoute le fichier 'knacss.css' au html, que nous allons créer après :

```
ubuntu@sassInstance:/var/www/html/henry-alexis.fr
GNU nano 4.8
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="stylesheet" href="knacss.css"/>
  </head>
  <body>
    <h1>Découverte de Sass par Henry Alexis</h1>
  </body>
</html>
```

Ci-dessous, la conversion du fichier knacss.scss en css :

```
use --trace for backtrace.
ubuntu@sassInstance:/var/www/html/henry-alexis.fr/sass$ sudo sass knacss.scss /var/www/html/henry-alexis.fr/knacss.css
```

On peut désormais, utiliser les classes que propose Knacss, pour affecter du style à nos éléments HTML. Ce qui est fait ci-dessous :

```
<meta charset="UTF-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<link rel="stylesheet" href="knacss.css"/>
</head>

<body>
  <h1 class="hidden flex">Découverte de Sass par Henry Alexis</h1>
</body>

</html>
```

Quand on regarde la documentation de Knacss, on peut observer qu'ils y ont mis les différentes classes possibles à ajouter, j'ai décidé d'ajouter une classe qui va rendre invisible le h1.

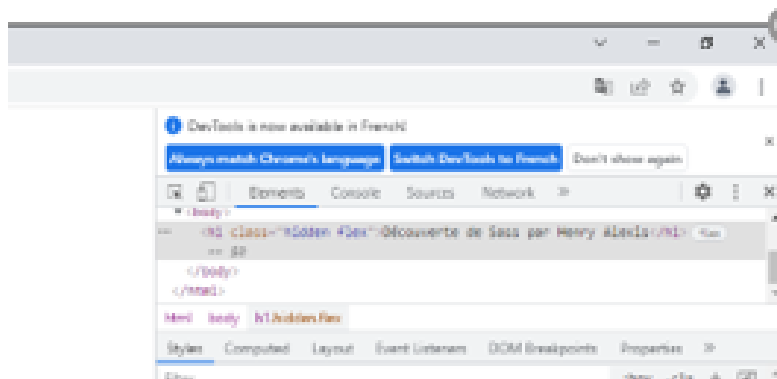
Ci-dessous, les différentes valeurs de display existantes :

valeurs de display

Classe	Propriété
<code>.hidden</code>	<code>display: none;</code>
<code>.block</code>	<code>display: block;</code>
<code>.inline</code>	<code>display: inline;</code>
<code>.inline-block</code>	<code>display: inline-block;</code>
<code>.flex</code>	<code>display: flex;</code>
<code>.grid</code>	<code>display: grid;</code>

Ci-dessous, le résultat obtenu avec les classes 'hidden flex' attribués :

On peut observer, en inspectant la page, que l'élément h1 est contenu par les deux classes hidden et flex, et on peut observer à droite qu'il est en display flex. Cela est dû au fait que la classe flex se situe après la classe hidden.



Résultat de la page web réalisée avec Knacss (flex)



Je vais désormais tester en enlevant la classe 'flex' du h1, ce qui fait que normalement la classe 'hidden' devrait être prise en compte, et de ce fait le h1 devrait se mettre en display none.


```
GNU nano 4.8
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="stylesheet" href="knacss.css"/>
  </head>

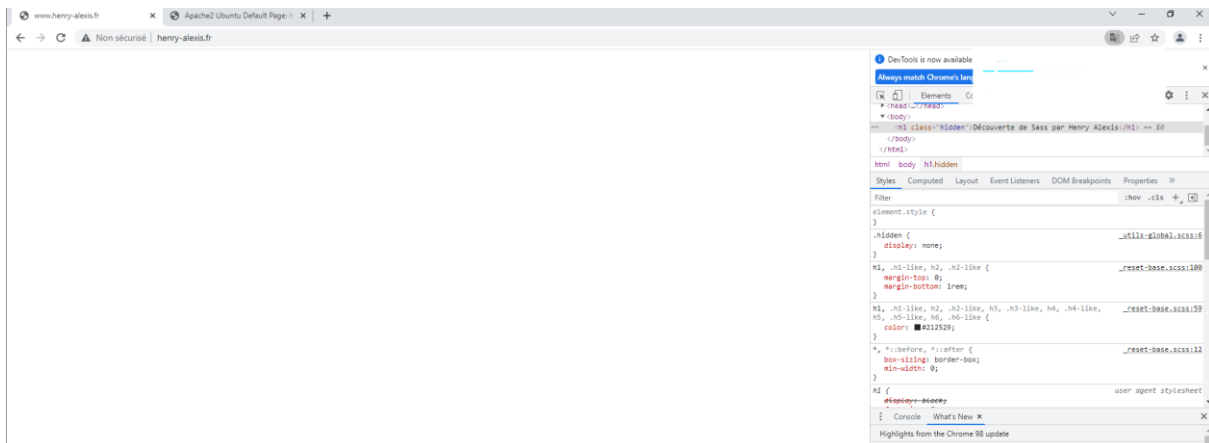
  <body>

    <h1 class="hidden">Découverte de Sass par Henry Alexis</h1>

  </body>

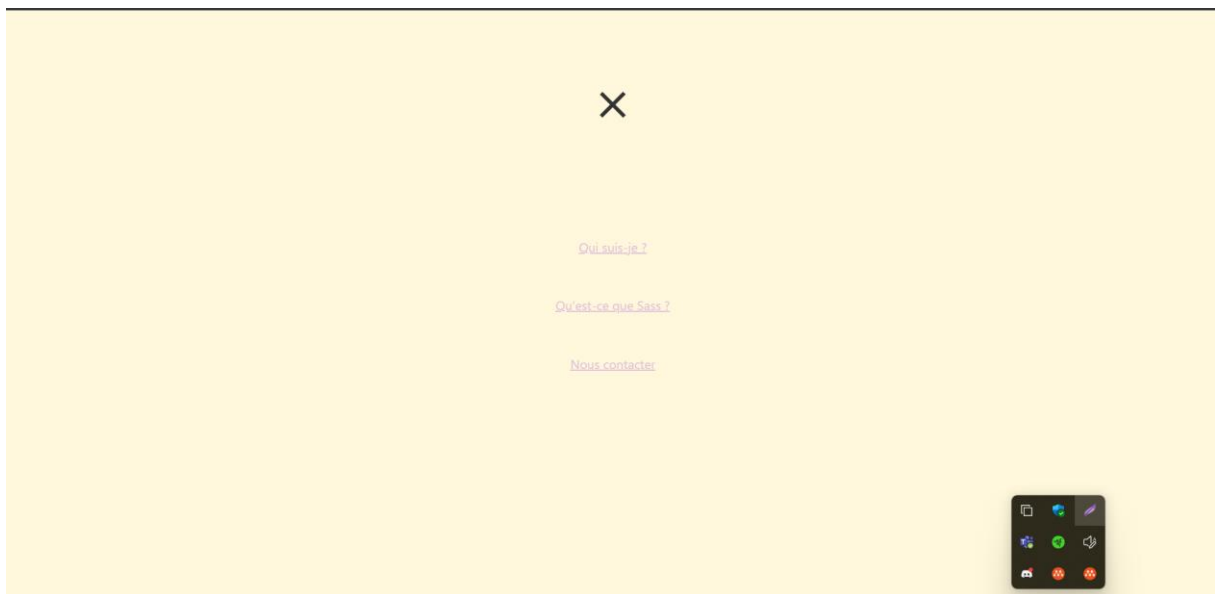
</html>
```

Résultat de la page web réalisée avec Knacss (hidden)



On peut voir ci-dessus, que c'est bien le cas, le h1 a disparu, et s'est bel et bien mit en display none.

Résultat final d'une page avec Scss, Knacss et Schnapss



IV- RESSOURCES UTILISEES

<https://multipass.run>

[https://fr.wikipedia.org/wiki/Sass_\(langage\)](https://fr.wikipedia.org/wiki/Sass_(langage))

<https://sass-lang.com/install>

<https://www.knacss.com/doc.html>

<https://schnaps.it>

<https://www.alsacreations.fr/>

<https://github.com/alsacreations/KNACSS>

<https://astuce-pc.com/comment-modifier-le-fichier-des-hotes-dans-windows-10/>

V- LIENS UTILES

Github repository :

<https://github.com/AlxisHenry/Learn-to-Use-SASS>

Github Website :

<https://alxishenry.github.io/Learn-to-Use-SASS/>