

# Cadre photo Neopixel pour pixel art


Il s'agit d'un petit cadre photo, tout à fait ordinaire vu de l'extérieur, mais abritant en son sein une matrice de LED capable d'afficher de petits dessins en Pixel art.

Le cadre est recouvert d'une feuille de papier, ce qui lui donne un rendu organique et "naturel" à la manière du e-paper. Très simple à utiliser et esthétique, cet objet est également adapté à un usage en atelier, où les participants pourront dans un premier temps apprendre à faire de petite oeuvre en pixel art sur GIMP puis les afficher chez eux ou sur leur bureau.

 Difficulté Facile

 Durée 2 heure(s)

 Catégories Art, Décoration, Électronique, Maison, Jeux & Loisirs

 Coût 10 EUR (€)

## Sommaire

Introduction

Étape 1 - Raccorder la matrice de LEDs à l'arduino

Étape 2 - Intégrer la matrice au cadre photo

Étape 3 - Dessiner un pixel art de 8x8 pixels

Étape 4 - ALTERNATIVE : Redimensionner une image en 8x8 pixels

Étape 5 - Convertir une image en code Arduino

Étape 6 - Installer les bibliothèques et Téléverser le code

Notes et références

Commentaires

## Introduction

INFOS : Pour ce tutoriel, chaque étape est illustrée par un GIF animé.

Matériel requis :

- Arduino nano (<5€)
- Matrice de LEDs adressables, type NeoPixel (ws2812b), 8x8 (5€) ou 16x16 (15€)
- Cadre Leroy Merlin (1€)

Outils :

- Pistolet à colle
- Perceuse
- Cutter

## Matériaux

## Outils

 <https://drive.google.com/drive/folders/1qSXMHZJhxVvjEIZN5BhCXURWi191kWFJ?usp=sharing>

 Cadre\_photo\_Neopixel\_pour\_pixel\_art\_test\_de\_la\_matrice\_arduino.ino

# Étape 1 - Raccorder la matrice de LEDs à l'arduino

La première chose à faire est de réaliser le raccordement entre la matrice et l'arduino nano (ou uno/leonardo) pour qu'il puisse communiquer.

Il existe plusieurs modèles de matrice de LED adressage (ws2812b).

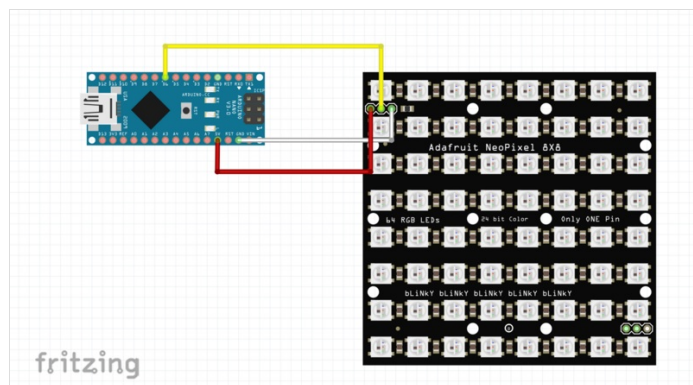
De manière générale elles sont structurées de la même manière :

- Un câble qui achemine les informations (Data In and Data Out)
- Un câble qui achemine le courant (VCC ou 5V)
- Un câble qui ferme le circuit en ramenant à la terre (Ground ou GND)

- Le câble VCC est à raccorder au 5V de l'arduino
- Le câble GND est à raccorder au Ground de l'arduino

Le câble DI doit être raccordé à la PIN 6 (pour fonctionner avec les codes, proposez dans ce tutoriel).

- Le câble DO sert à raccorder une matrice avec une autre matrice. Dans se cas le DO de la matrice 1 doit être connecté avec le DI de la matrice 2.



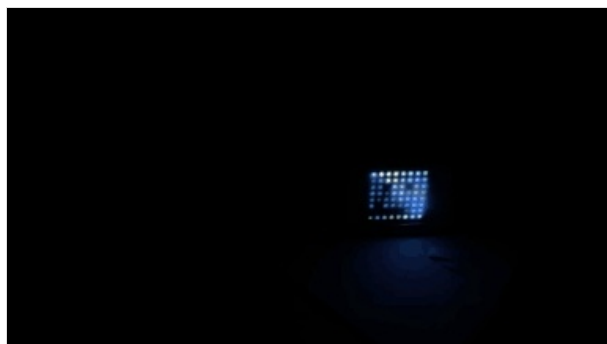
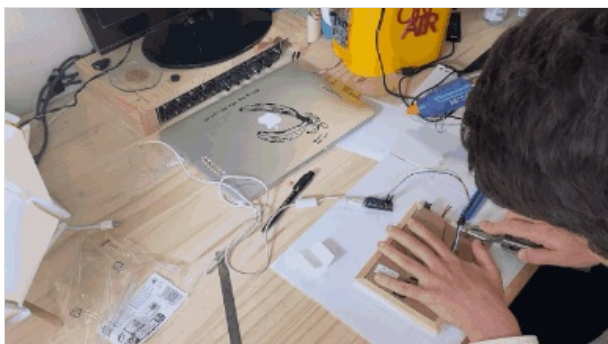
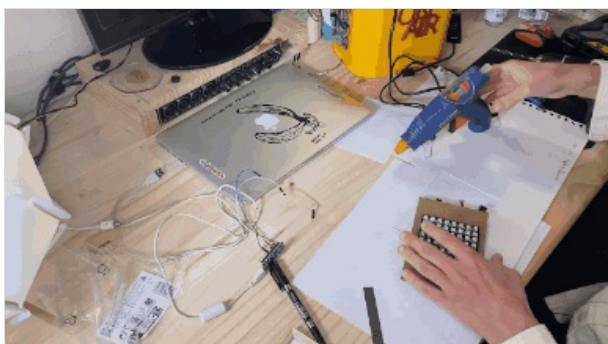
## Étape 2 - Intégrer la matrice au cadre photo

L'idée est d'afficher dans son salon ou dans son bureau, ou dans les couloirs de l'école, l'entreprise ou ailleurs ses oeuvres en pixel art.

Cette étape sert à enjoliver l'objet. Dans mon exemple j'intègre la matrice dans un cadre 10x15 acheté chez Leroy Merlin.

Pour ajouter un effet E-Paper, j'ai ajouté une feuille de papier devant la vitre, côté regarder.

1. Ouvrir le cadre et utiliser le dos du cadre comme patron pour découper un rectangle de papier à la bonne taille (un peu épais, du papier à dessin 80g c'est pas mal)
2. **Facultatif** : Téléverser le code "test\_de\_la\_matrice\_arduino.ino" dans l'arduino nano (instruction à l'étape 6) pour avoir un aperçu du résultat.
3. À l'aide d'un marqueur, marquer le dos du cadre pour savoir où percer pour laisser passer les câbles et où coller la matrice
4. Percer avec une perceuse, nettoyer avec un cutter
5. Pisto-coller la matrice. Le but est de coller la matrice de LED tout en la surélevant un petit peu pour laisser de la place aux câbles.
6. Disposer le rectangle de papier devant la vitre (côté regarder) puis assembler les deux parties du cadre.
7. Fermer le tout en repliant les petits ergots.





# Étape 3 - Dessiner un pixel art de 8x8 pixels

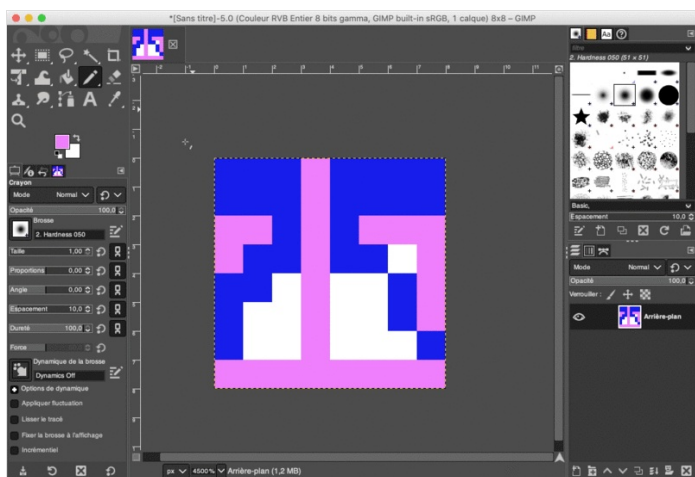
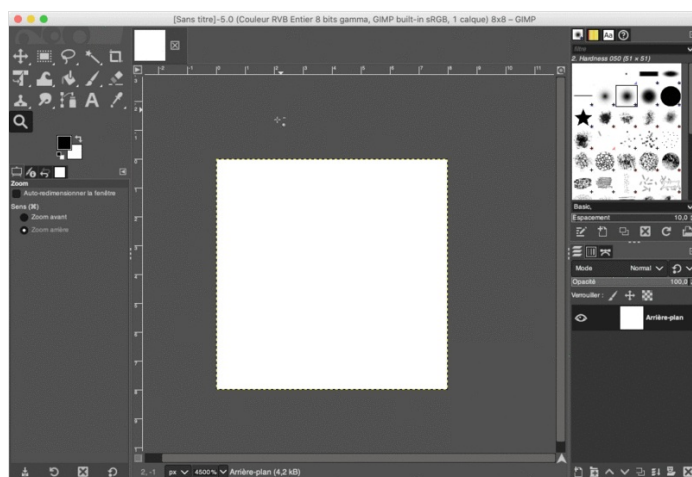
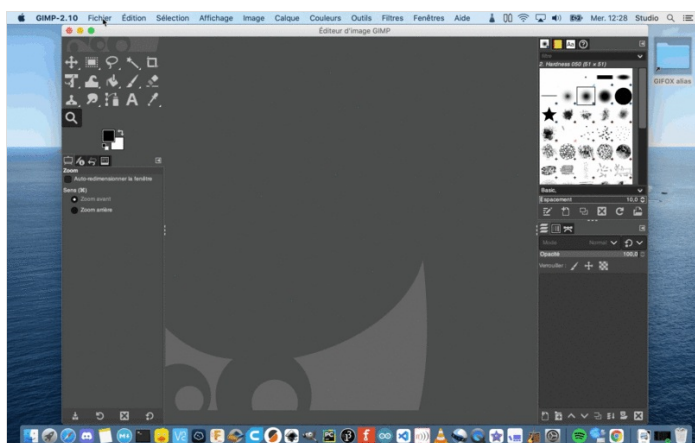
En atelier ou pour les fans de retro gaming : dessiner ses propres oeuvres.

1. Télécharger et installer GIMP à cette adresse
2. Créer une nouvelle image (ctrl + N) de 8 pixels par 8 pixels
3. Zoomer pour que le canva occupe tout l'écran
4. Sélectionner l'outil crayon (N)
5. Modifier la taille du crayon à 1 pixel et la dureté à 100%
6. Changer les couleurs et dessiner pixel par pixel.
7. Exporter : Fichier►Exporter sous►Choisir PNG►Renommer1.png

 S'il s'agit d'une série d'images ou de dessins, nommer les fichiers comme suis : "nom1.png, nom2.png, nom3.png, etc."


 S'il n'y a qu'une seule image il est nécessaire de la nommer "nom1.png"....


 Pour un meilleur rendu, restreindre le nombre de couleurs utilisées et utiliser des couleurs qui contrastent bien.

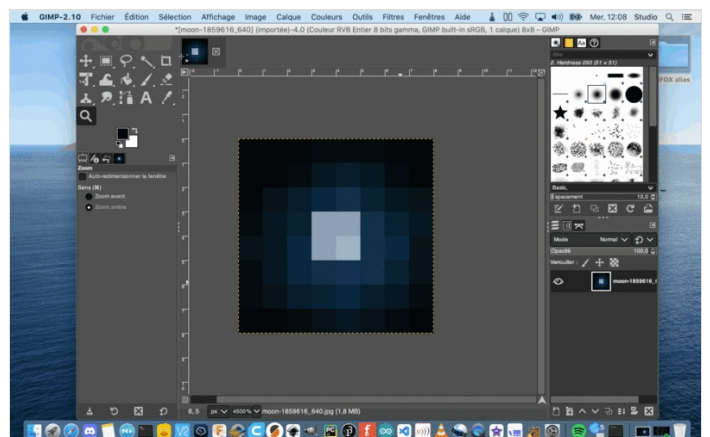
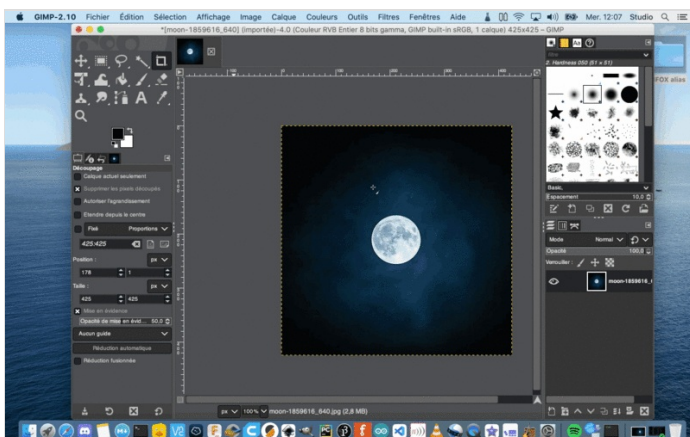
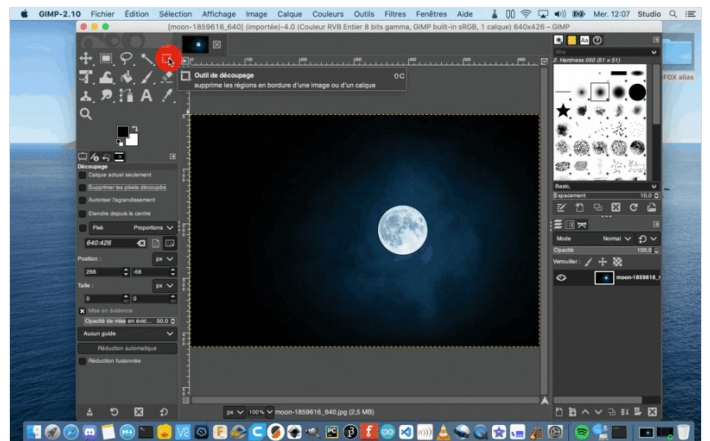
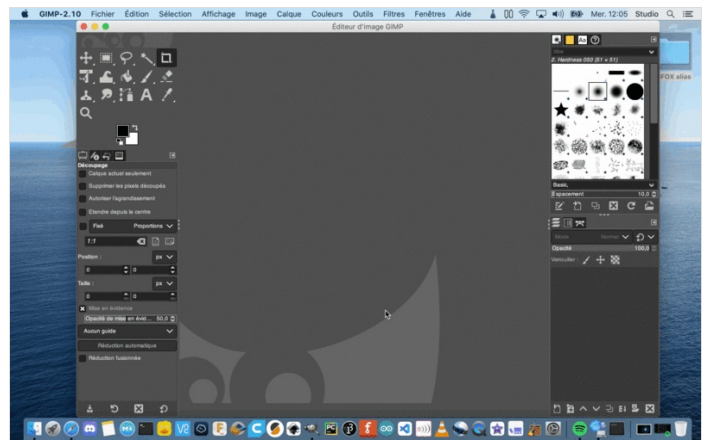
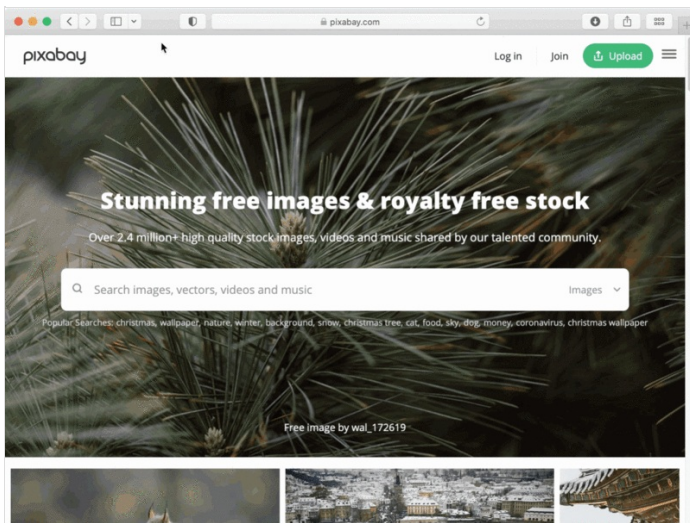


# Étape 4 - ALTERNATIVE : Redimensionner une image en 8x8 pixels

1. Télécharger et installer GIMP à cette adresse
2. Pour cette exemple j'ai téléchargé une image depuis la banque d'image libre de droit pixabay.com
3. Ouvrir l'image en question dans GIMP (ctrl + o)
4. Redimensionner l'image dans un format carré (Shift + C)
  1.  Cocher l'option "supprimer les pixels découpés"
  2.  Cocher l'option "proportion fixe", avec le ratio 1:1
5. Changer la taille de l'image : Image>Echelle et taille de l'image
6. Changer la taille de l'image en 8px par 8px
7. Exporter. Fichier>Exporter sous>Choisir PNG>Renommer1.png

 S'il s'agit d'une série d'images ou de dessins, nommer les fichiers comme suis : "nom1.png, nom2.png, nom3.png, etc."

 S'il n'y a qu'une seule image il est nécessaire de la nommer "nom1.png"....



# Étape 5 - Convertir une image en code Arduino

Pour faciliter et automatiser le passage d'un fichier .png à son affichage sur la matrice de LED un code python à été conçu.

## METTRE EN PLACE L'ENVIRONNEMENT PYTHON

1. Installer Pycharm version communauté (gratuit)
2. Créer son projet . Choisir python 3.
3. Attendre que l'indexation soit terminé
4. Supprimer le fichier "main.py" créer par défaut
5. Glisser déposer les fichier "main.py" et "create\_arduino\_code.py" (fournis dans ce tutoriel) dans le projet acuel
6. Ouvrir le terminal et lancer les deux commandes suivantes

```
pip3 install pillow  
pip3 install pyperclip
```

## UTILISER LE CODE POUR TRANSFORMER UNE IMAGE EN CODE ARDUINO

Il s'agit surtout de changer ces paramètres

```
#####  
##### PARAMETERS #####  
#####  
# Don't forget to edit the parameters  
folder_path = '/Users/Studio/Desktop/PIXEL/'  
name_of_the_drawings = 'EXEMPLE'  
number_of_drawings = 1 # 7 is the maximum for 8x8 matrice and arduino nano  
matrix_width = 8 # this works for an 8x8 matrice  
time_between_scene = 2000 # in milliseconds  
brightness = 50  
#####
```

*folder\_path* = chemin absolu menant jusqu'au dossier contenant les images

*name\_of\_the\_drawings* = le nom de la série de dessins ou d'images sans les chiffres.

**i** Si la série est 'MOON1, MOON2, MOON3, etc...' *name\_of\_the\_drawings* = 'MOON' S'il n'y a qu'une seule image, elle doit s'appeler 'MOON1' et *name\_of\_the\_drawings* = 'MOON'

*number\_of\_drawings* = le nombre d'images de la série

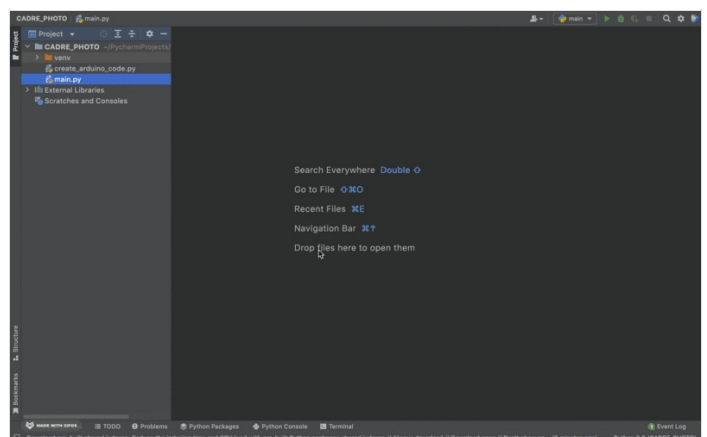
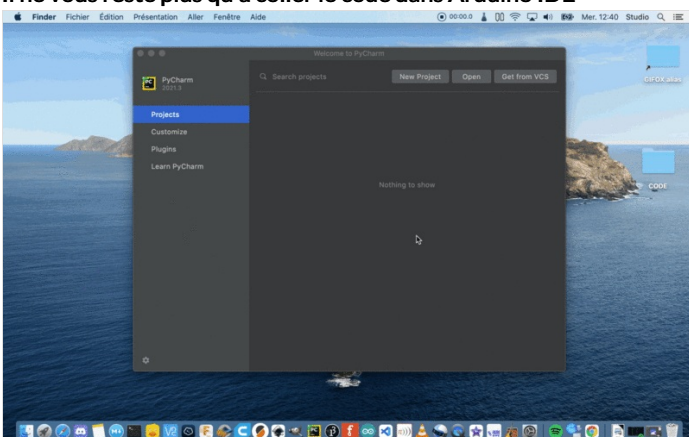
*matrix\_width* = largeur de la matrice de led (généralement 8 ou 16)

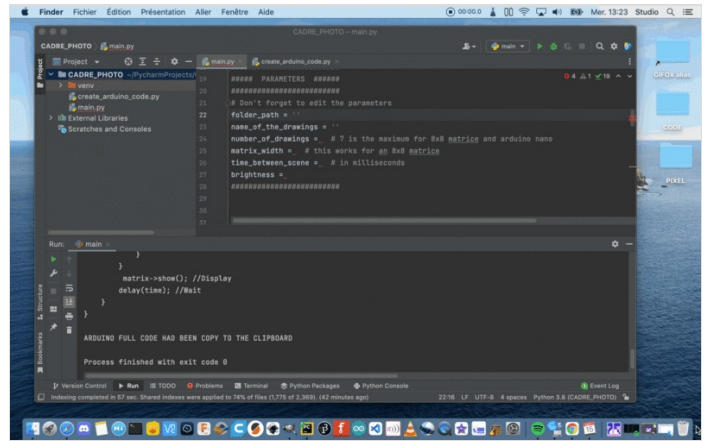
*time\_between\_scene* = le temps d'attente entre une image et la prochaine

*brightness* = luminausité des LEDs

Une fois ces informations modifié faite tourner le scripte. Le code Arduino s'inscrit dans la console python et est automatiquement copier dans le presse papier à la fin (grâce à la bibliothèque pyperclip).

## Il ne vous reste plus qu'a coller le code dans Arduino IDE





## Étape 6 - Installer les bibliothèques et Téléverser le code

C'est l'étape ultime : envoyé le code sur l'arduino.

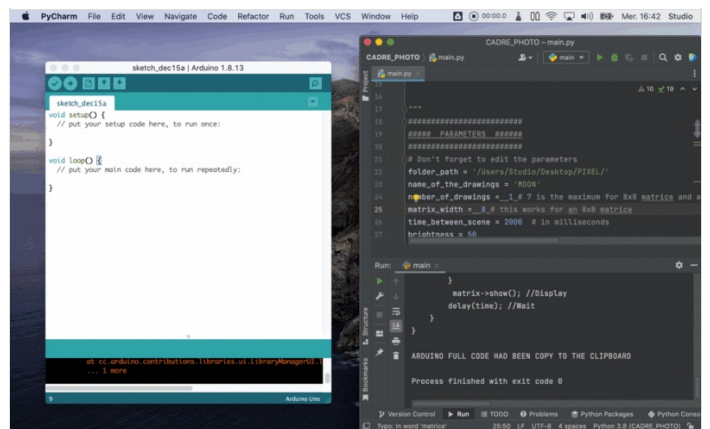
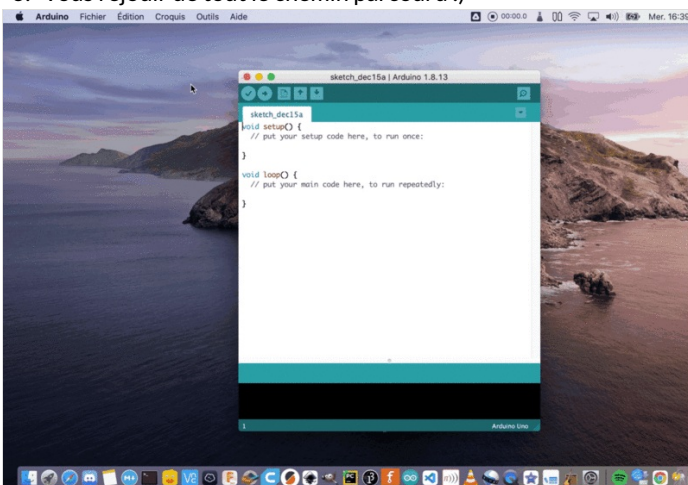
### INSTALLER LES LIBRAIRIES

**i** À ne faire qu'une fois, lors de la première utilisation

1. Télécharger et installer Arduino IDE
2. Croquis ► Inclure une librairie ► Gérer les bibliothèques
3. Rechercher et installer "FastLED NeoMatrix". Lorsque demandé, choisir "Install all", ce qui installera toutes les bibliothèques nécessaires au bon fonctionnement du code.

### TELEVERSER LE CODE (Upload en anglais)

1. En fonction de la carte utilisé renseigner les champs suivant dans le menu outil.
  1. Outils ► Types de cartes ► (Dans mon cas Arduino Nano)
  2. Outils ► Procéssueur ► (Dans mon cas ATmega328P old boot loader)
  3. Outils ► Port ► (Dépend d'un ordinateur à l'autre)
2. Si un "Sketch" n'est pas ouvert par défaut créer en un nouveau : Fichier ► Nouveau (ctrl +N)
3. Supprimer tout et remplacer par le code obtenu grâce au code python
4. Cliquer sur le bouton "Téléverser" ou cliquer sur Croquis ► Téléverser (ctrl +U, "U" pour Upload en anglais)
5. Attendre
6. Vous réjouir de tout le chemin parcouru :)



## Notes et références

J'espère que ce tutoriel vous à plu. Pour info, je suis codeur amateur et serait ravi d'intégrer vos améliorations. Amusez-vous bien :)

**Information sur la bibliothèque FastLED :**

<https://www.arduino.cc/reference/en/libraries/fastled-neomatrix/>

[https://github.com/marcmerlin/FastLED\\_NeoMatrix](https://github.com/marcmerlin/FastLED_NeoMatrix)