

Digital Lab - Librairie, capteurs, snippets & pour aller plus loin

lundi 14 avril 2014 10:20

1 - Librairies

- 1 Qu'est-ce qu'une librairie

Si processing permet de réaliser de nombreuses choses il nous est parfois nécessaire d'étendre son champs d'action. C'est là où les librairies entrent en action.

Une librairie est un ensemble de fonctions pré-écrite et de classes pouvant être utilisées dans processing et permettant ainsi d'accéder à différent inputs (micro, camera, video...) ou de réaliser des actions complexes (moteur physique, calculs de géométrie 3D, analyse de donnée...)

Par défaut processing intègre certaines librairies qui sont :

- DFX : permet l'export de fichier 3D
- Minim : permet l'analyse audio de fichier ou du micro et la synthèse sonore
- Net : permet l'écriture et la lecture de données via internet
- PDF : permet l'écriture de PDF
- Serial : permet l'écriture et la lecture de données via communication serial (Arduino notamment)
- Video : permet l'analyse et la lecture de fichier vidéo ou de capture vidéo

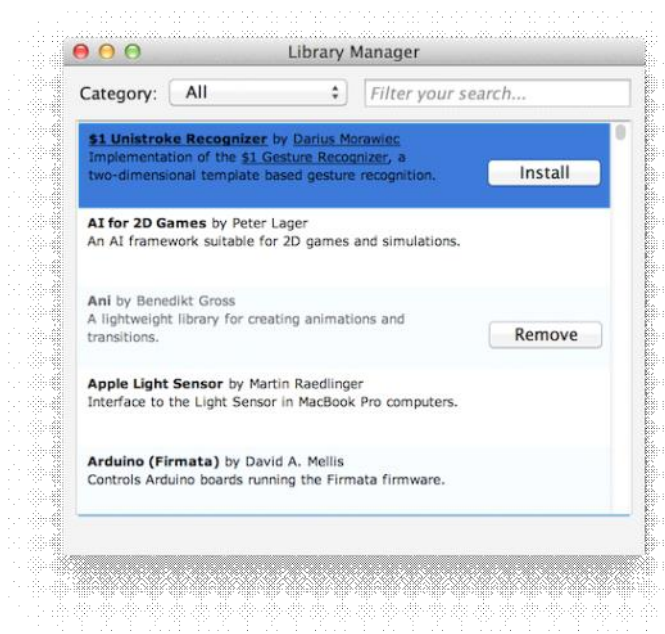
Il est également possible d'installer de nombreuses librairies réalisées par différents contributeurs. Le site de processing propose une sélection de librairies mais il est également possible d'en trouver sur différents site (*google is my friend*).

- 2 Installer une librairie

L'installation d'une librairie peut se faire de deux manière suivante.

La plus simple consiste à passer par l'outil fourni par processing disponible la manager de librairie disponible via le menu :

- **Sketch > import Library > Add Library**



Le manager de librairie permet d'installer diverses librairies parmi un sélection validée par les différents contributeurs de processing.

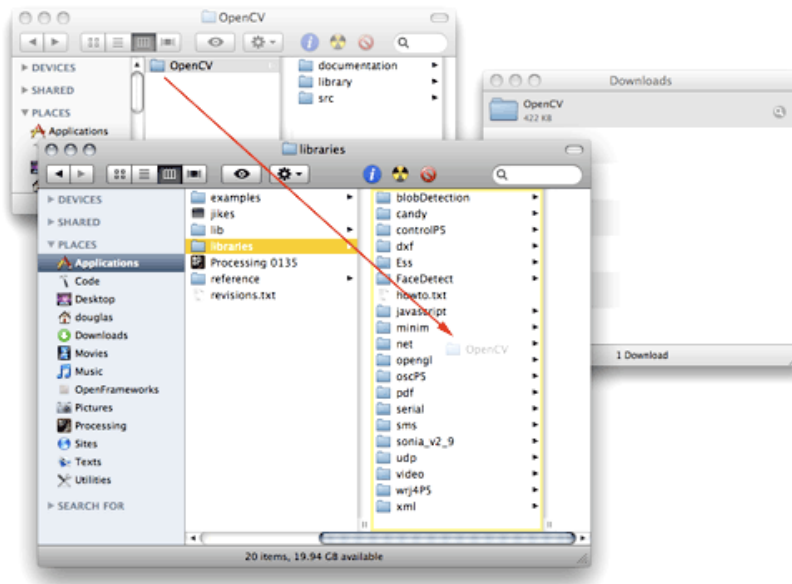
Vous pourrez retrouver cette sélection et leur description sur le site processing à l'adresse suivante:

- <http://processing.org/reference/libraries/>

Une seconde méthode permet d'installer une librairie qui ne se trouverait pas dans le manager de librairie. Pour ce faire il suffit de télécharger la librairie sur le site qui lui est propre. Les librairies téléchargées ont généralement la même architecture de dossier, à savoir

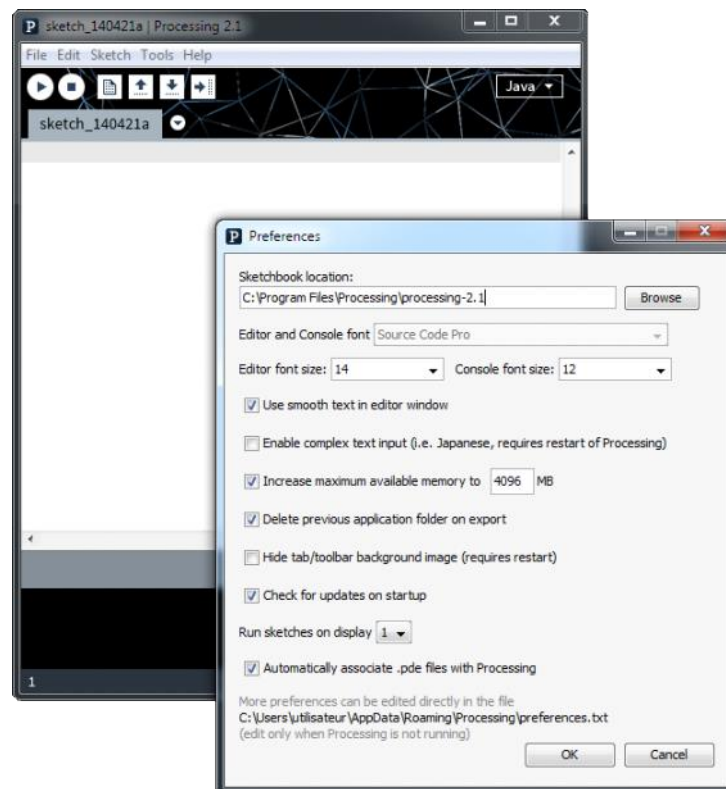
- nomDeLaLibrairie
 - Library
 - Exemples
 - Références
 - Src ou bin
- ReadMe.txt/rf

Pour installer la librairie téléchargée il suffit de copier/coller le dossier "nomDeLaLibrairie" dans le dossier library de processing.



Par défaut ce dernier se trouve dans :
 - Document/Processing

On peut également retrouver le lien vers le dossier librairie depuis le menu préférence



- 3 Utiliser une librairie

Pour utiliser une librairie une fois installée, il suffit de l'importer dans son sketch.
 Pour se faire nous pouvons utiliser le menu

- **Sketch > import library > librairie voulue**

Nous remarquons alors l'ajout d'éléments suivant dans notre code :

- **Import lienDeLaLibrairie.*;**

Cela permet de signifier à notre code d'importer la librairie (et donc ses méthodes) au lancement dans notre sketch.

Il peut arriver qu'une librairie puisse s'importer en plusieurs parties afin d'être moins lourdes. C'est le cas de la librairie ToxicLib qui se divise en plusieurs éléments, tous pouvant être utilisés de manière indépendante.

NB 1 : par défaut la ligne d'import d'une librairie sera **import librairie.*;** mais il n'est pas rare des import s'écrivant de la manière suivante : **import com.nom.nomDeLaLibrairie.*;**
 Cette écriture est une convention permettant de retrouver le lien de téléchargement de la librairie ainsi que son créateur. Par défaut nous remontons le lien de la manière suivante :

- Domaine (com/net/fr/en/io...)
 - nomDuSite
 - Nom de la librairie

NB 2 : la convention `.*` à la fin de l'import de la librairie signifie au sketch que nous devons importer l'ensemble des dossier présent dans la librairie. Ainsi nous avons donc :

- Domaine (com/net/fr/en/io...)
 - nomDuSite
 - Nom de la librairie
 - ◆ Tous les sous-dossiers

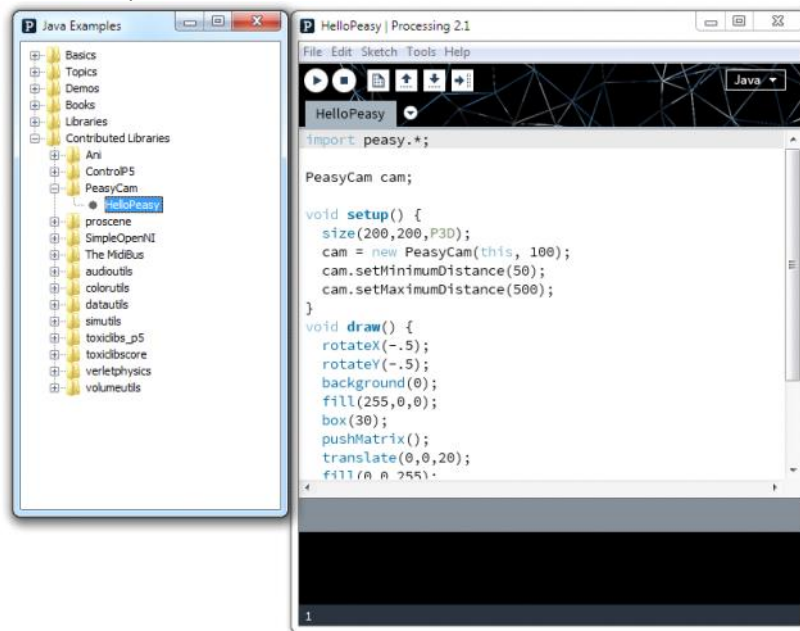
- 4 Apprendre à utiliser un librairie

Avant d'utiliser un librairie il est important d'apprendre à s'en servir.

Une librairie étant un ensemble de fonctions et de classes pré-écrites il nous est indispensable de les connaître pour les utiliser au mieux.

La méthode la plus simple pour se familiariser avec une librairie est de passer par ses exemples. Une fois la librairie installée ces exemple sont disponible dans le dossier exemple à l'emplacement **Contributed Libraries**

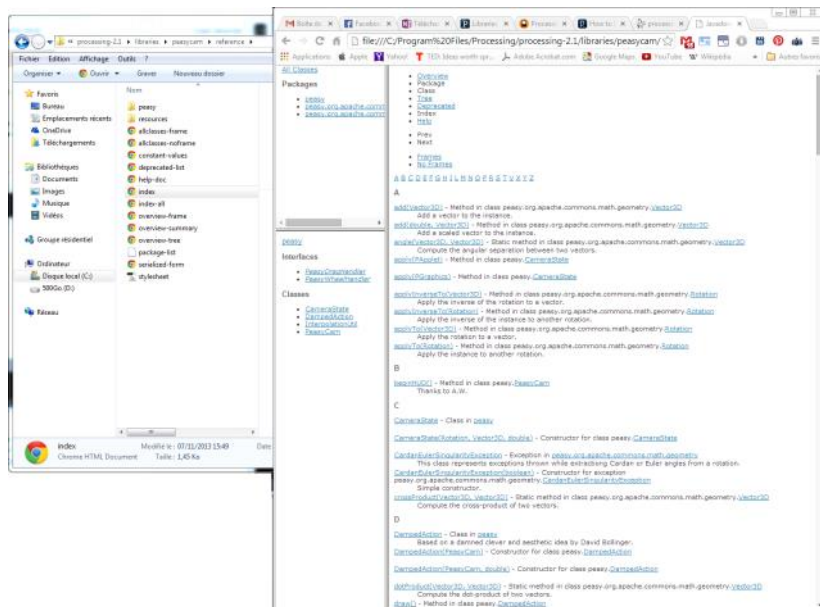
- File > Exemple



Les site de téléchargement des librairie possèdent habituellement une référence ainsi qu'un *quick start* permettant de nous familiariser avec la librairie. C'est le cas notamment de la librairie **peasyCam** de Jonathan Feinberg (<http://mrfeinberg.com/peasycam/>)

Enfin chaque librairie possède une documentation **.html** nommée **javadoc** et disponible dans le dossier où la librairie est installée (cf *part.02 - installer une librairie*).

Cette **javadoc** regroupe l'ensemble des méthode que possède la librairie et permet de savoir comment les utiliser. Celle-ci sont généralement auto-générée, il n'est donc pas rare de n'obtenir que peu d'information sur l'utilisation des méthodes. Celle-ci est cependant très utile lorsque vous maitrisez parfaitement la librairie et que vous souhaitez aller plus loin avec.



2 - Capteurs i/o

Si processing permet, de manière native, de réaliser des interaction clavier/souris, il est également possible, à l'aide de librairies, d'utiliser d'autre type de capteurs i/o.

I/O pour input et output permet de définir l'état d'un capteur. Ainsi un capteur input est un capteur envoyant une donnée entrante (du capteur vers l'ordinateur) et un capteur output est un capteur recevant une donnée (de l'ordinateur au capteur).

On donc donc définir un clavier et une souris comme un capteur input uniquement. Il en est de même pour une webcam ou un micro.



- 1 Les capteurs "de bases"

Depuis quelques années nos ordinateurs portable intègre automatiquement deux capteur : webcam et micro. Il nous est alors possible de les utiliser dans processing à l'aide de librairie intégrées nativement depuis la version 1.5 de processing.

Pour ce faire, il suffit de se rendre dans les exemples des librairie vidéo et minim du dossier librairie

- File > exemples

Il nous sera alors possible d'accéder au flux audio et vidéo afin de les afficher et/ou les analyser pour de la détection de présence, de mouvement, de couleur, de son...



- 2 Kinect

La **Kinect** est un capteur 3D permettant de détecter des corps et des squelette dans un espace 3D.
Si la kinect est le capteur le plus connu, il en existe d'autre dont **Carminé** (de primeSense) et **xTion** (de Asus).

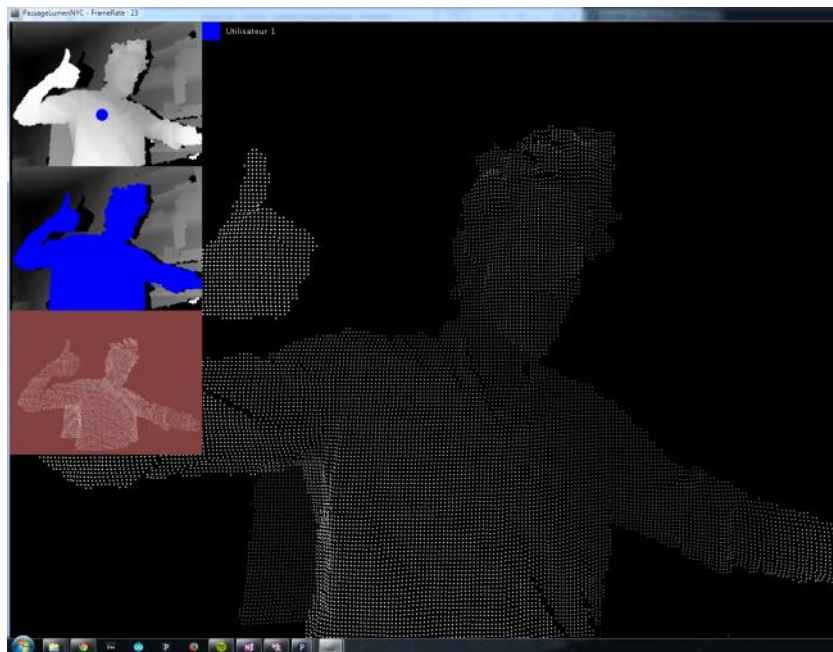
Ces capteurs se composent d'un micro, d'un projecteur infrarouge, d'une camera RGB ainsi que d'une caméra de profondeur.

NB : Une caméra de profondeur est une camera permettant d'obtenir le depthmap de la scène. Ce depthmap est une image noir et blanc où la valeur de blanc (de 0 à 100%) est définie par le distance entre l'objet/pixel dessiné est la capteur de la camera.

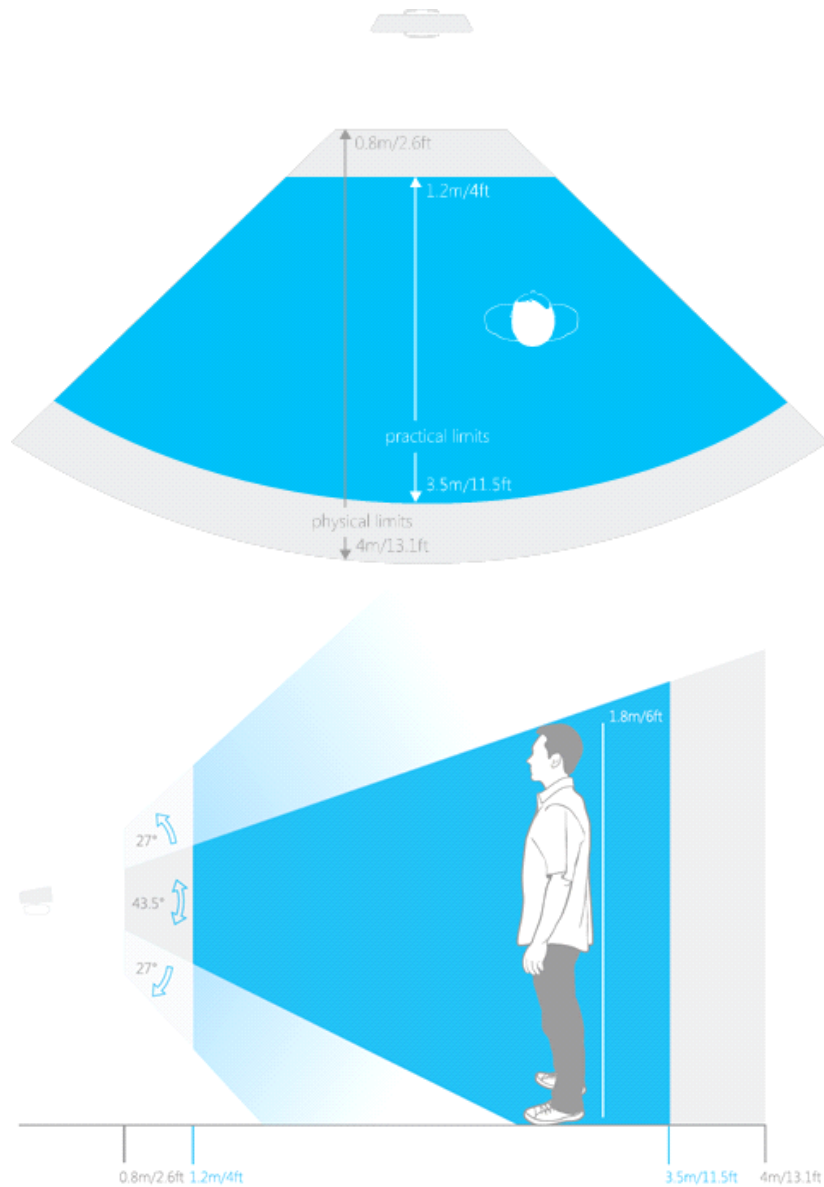


Il est possible d'utiliser ces capteur dans processing par le biais de la librairie SimpleOpenNI :
<https://code.google.com/p/simple-openni/>

Il nous sera alors possible d'accéder aux différentes camera du capteur mais aussi de détecter la position d'une main ou d'un squelette entier, chaque utilisateur, leur centre de gravité...



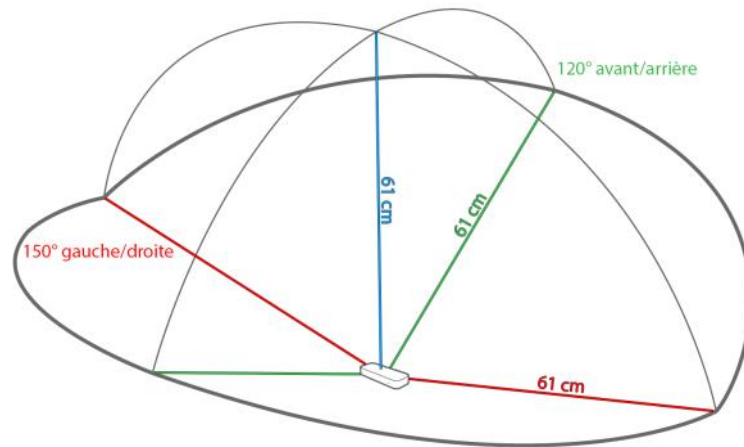
Lorsqu'on travaille avec des capteur 3d, il est important d'en connaître la portée maximale et minimale afin d'établir une zone d'interaction correct. Ainsi pour la kinect pour obtenons une zone active de 1.2m à 3.5m de profondeur.



- 3 Leap Motion

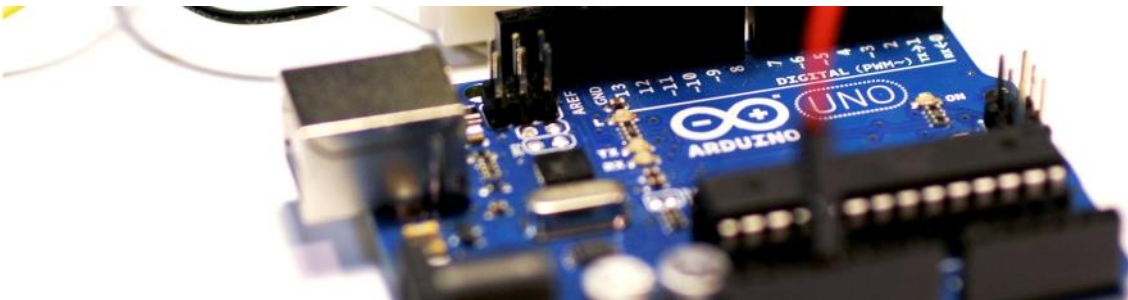
La **leapmotion** est un capteur 3D dédié à la détection de main. Ce dernier fonctionne, à peu de chose près, sur le même principe que la **kinect**, c'est-à-dire qu'il utilise des projecteur infrarouge ainsi que des caméras de profondeur afin de déterminer un depthmap de la scene.

Comme pour la **kinect** la **leapmotion** possède une zone active qu'il est important de connaitre. Cette dernière est constituée d'une pyramide de 61cm de côté.



Il est possible d'utiliser la leapMotion dans processing à l'aide des librairie suivante :

- **Leap Motion for processing** <https://github.com/voidplus/leap-motion-processing>
- **LeapMotionP5** <http://www.onformative.com/blog/leap-motion-library-for-processing/>



- 4 Arduino

Arduino est un programme de prototypage électronique englobant plusieurs éléments dont :

- Une interface électronique permettant de contrôler une grande variété de capteurs
- Un environnement de développement dont l'IDE (interface) est très proche de celui de processing
- Un langage de programmation (dérivé du C)

Arduino permet de réaliser des objet connecté autonome ou d'étendre les capacités de captation d'une installation.

Il est très facile de communiquer entre processing et arduino. Que ce soit via une communication Serial ou à l'aide d'un protocole de communication firmata

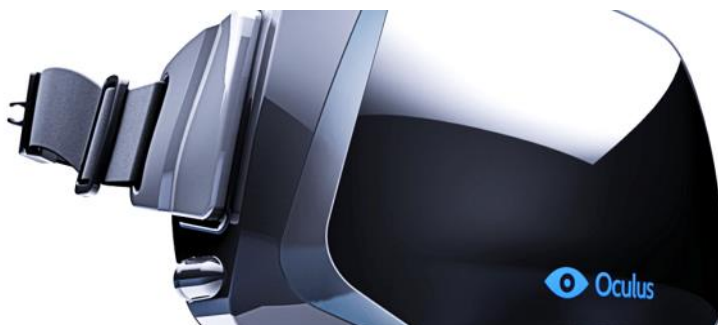
Processing intègre divers exemples de communication serial entre capteur et programme dans sa section librairies

- **File > exemples**

De nombreuses références sont disponible ici :

- <http://arduino.cc/>
- <https://learn.sparkfun.com/tutorials>
- <http://www.cooking-hacks.com/>

- 5 D'autres capteurs



Il existe de nombreux capteurs supplémentaires et de nombreux autres sont à venir. Ainsi on peut très utiliser les capteurs suivant dans divers projets

- Imprimantes 3D
- Wiimote
- Playstation move
- Brain Sensor
- Oculus Rift
- Midi Controller

- CL/duo

3 - Snippets

Vous trouverez à l'adresse suivante une liste non exhaustive de divers snippets (code d'aide) pouvant vous aider dans vos projets.

Ces derniers sont classé par type

- 3D
- 2D
- Algorithmes

<https://github.com/alexr4/eartsup-course/tree/master/3eme%20annee/snippets>

D'autres snippets seront à venir sur <https://github.com/alexr4/>