

# Club Robotique - Station de mesure (T°, H)

- [Description du projet](#)
- [Modélisation et impression en 3D de la boîte de la station](#)
- [Phase 2 - Impression 3D](#)
- [Fonctionnement et programmation des composants](#)
- [Phase 4 - Mesure et Analyse des Données de Température avec le Capteur DHT22](#)

# Description du projet

## Introduction

Pour ce cycle, nous allons réaliser une station de mesure de température de l'air.

## Description des compétences visées :

### **Compétences techniques et mécaniques :**

- Conception mécanique : savoir imaginer et concevoir la boîte qui va contenir les composants
- Assemblage de composants : apprendre à manipuler et assembler les différentes parties (boîte, vis, composants électroniques, etc.)
- Choix des matériaux : identifier les matériaux adaptés au projet

### **Compétences en électronique :**

- Découverte et câblage de capteurs pour mesurer une température.
- Savoir réaliser un schéma de câblage et connecter les composants sur une carte (ex. Arduino).

### **Compétences en programmation :**

- Programmation d'un microcontrôleur Arduino nano : écrire un code qui gère les capteurs, les délais.
- Optimisation du fonctionnement : ajuster les temps de prise de données, etc.

### **Compétences transversales :**

- Gestion de projet : planifier les étapes de réalisation, répartir les tâches, respecter les délais.
- Résolution de problèmes : savoir réagir face aux imprévus techniques ou fonctionnels.
- Travail en équipe : collaborer efficacement si le projet est collectif.

## Description du déroulé des séances :

~6 séances :

- ~Introduction du projet (brainstorming)
- ~2 séances sur TinkerCAD et sur l'impression 3D (création d'une boîte pour fixer les composants)
- ~1 séance Capteur : température
- ~2 séance Assemblage et programmation

## **Matériel :**

- 1 x Arduino Nano ou compatible ([seeeduino](#), funduino,...) : ~10€
- 1 x câble USB C : ~1€
- 1 x [Sensor Shield](#) pour Arduino Nano : ~3€
- 1 x Capteur Température (DHT 22): ~2€
- 1 x porte pile 4xaa : ~2€
- Une imprimante 3D pour imprimer les pièces du projet (fichier disponible à la prochaine page)(~4€ de filament) :

# Modélisation et impression en 3D de la boîte de la station

## Création d'un boîte pour la station

La station d'analyse de l'air ne possède pas de boîte pour contenir ni fixer les composants électroniques, c'est pour ça que je vais vous décrire comment la réalisée.

## Objectif (en deux séances)

L'objectif est de réussir à modéliser la boîte de la station qui va contenir tout nos composants électronique et à apprendre à utiliser une imprimante 3D pour imprimer cette dernière.

## Ressources disponibles

Vous pouvez retrouver le fichiers STL de la boîte dans les fichiers en pièce jointe.

Avant de commencer la modification, je vous invite à consulter si vous n'êtes pas à l'aise avec tinkercad :

- Cette [page](#) expliquant comment utiliser Tinkercad pour modifier un fichier 3D
- Cette [page](#) expliquant comment créer une classe sur Tinkercad (utile pour le travail en groupe ou en classe).

## 1. Modélisation 3D

Une fois la classe créée sur Tinkercad , avec le fichier de référence correctement chargé, et que tous les élèves sont connectés, nous pouvons passer à la phase de modification.

---

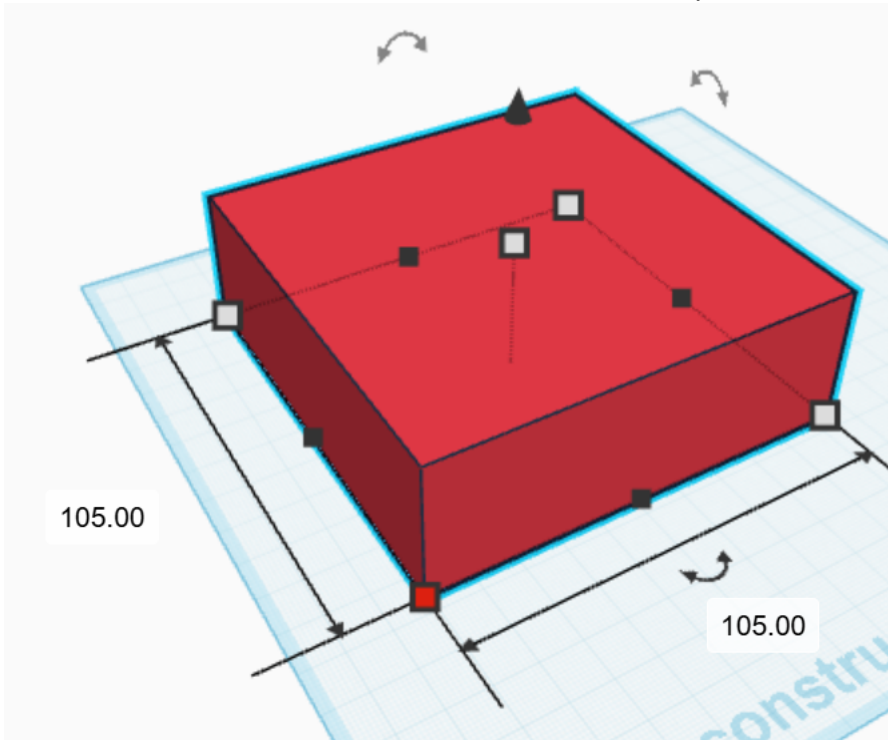
Comment créer une classe sur tinkercad :

<https://innovation.iha.unistra.fr/books/2-fablab-formation-machines-logiciels/page/creation-dune-classe-tinkercad>

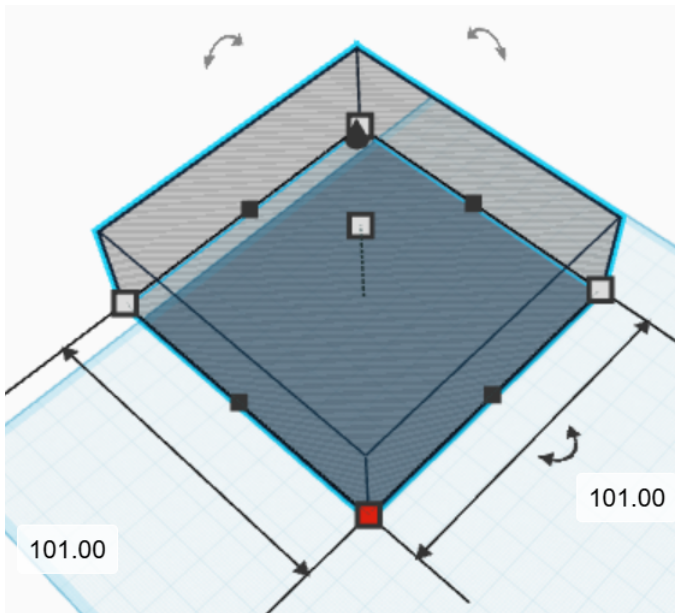
Sur l'interface principale de Tinkercad :


### 1. Création du Boîtier Principal:

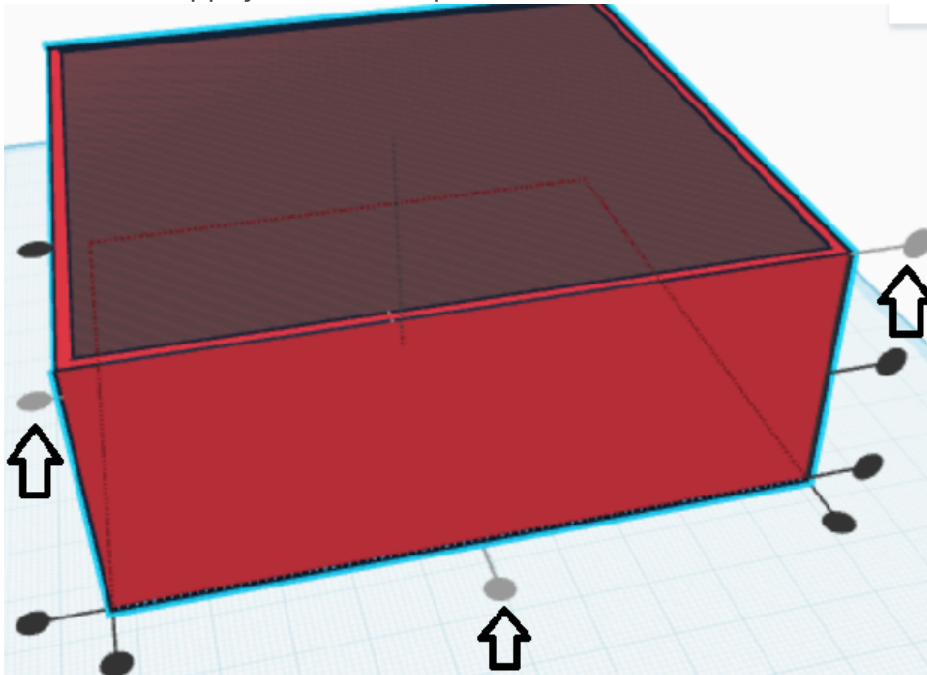
- Ouvrir TinkerCad et démarrer un nouveau projet.
- Placer un cube de base de 105 x 105 x 4 mm depuis la bibliothèque d'objets.




- Ajouter un deuxième cube de 101 x 101 x 38 mm (Ne pas oublier à le convertir en perçage!).



- Venir centrer le 2ème cube au premier en sélectionnant les 2 cubes puis appuyez sur ce bouton  bouton
- Et ensuite appuyer sur ces 3 points noir.



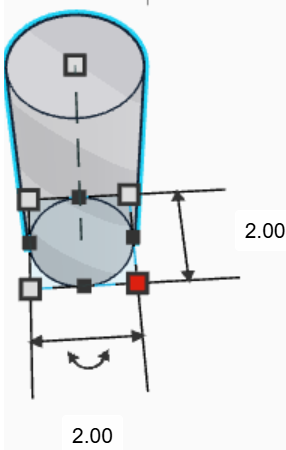
- Venir percer le deuxième cube au premier en appuyant sur ce bouton  (Bien sélectionner les 2 cubes en même temps).

### Création des trou pour les vis :

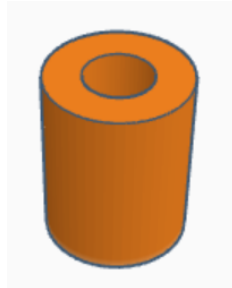
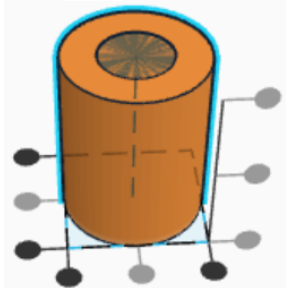
- Créer un cylindre de 4 x 4 x 5 mm.



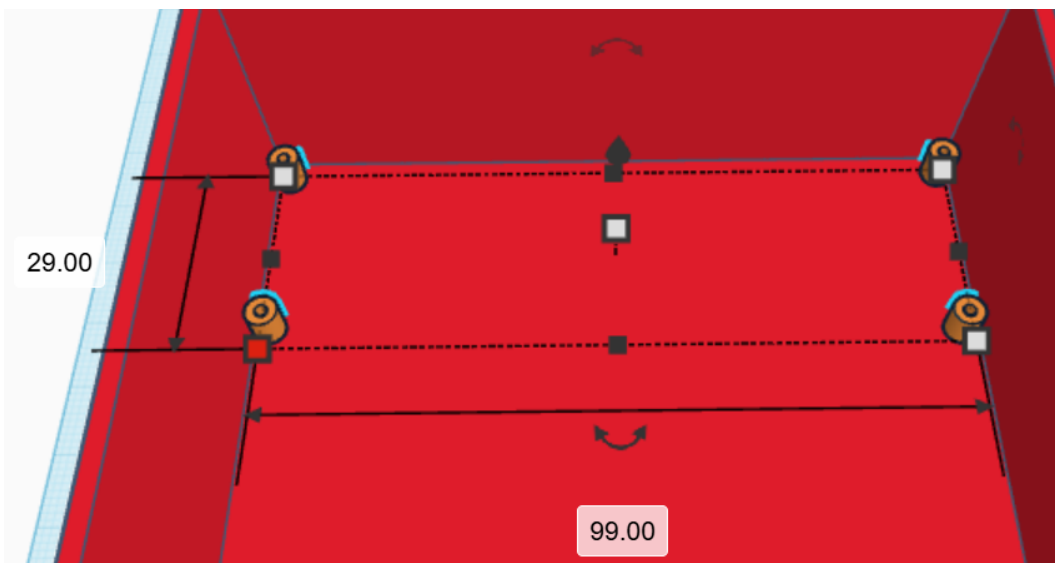
- Créer un cylindre de perçage de 2 x 2 x 5 mm.



- Créer.

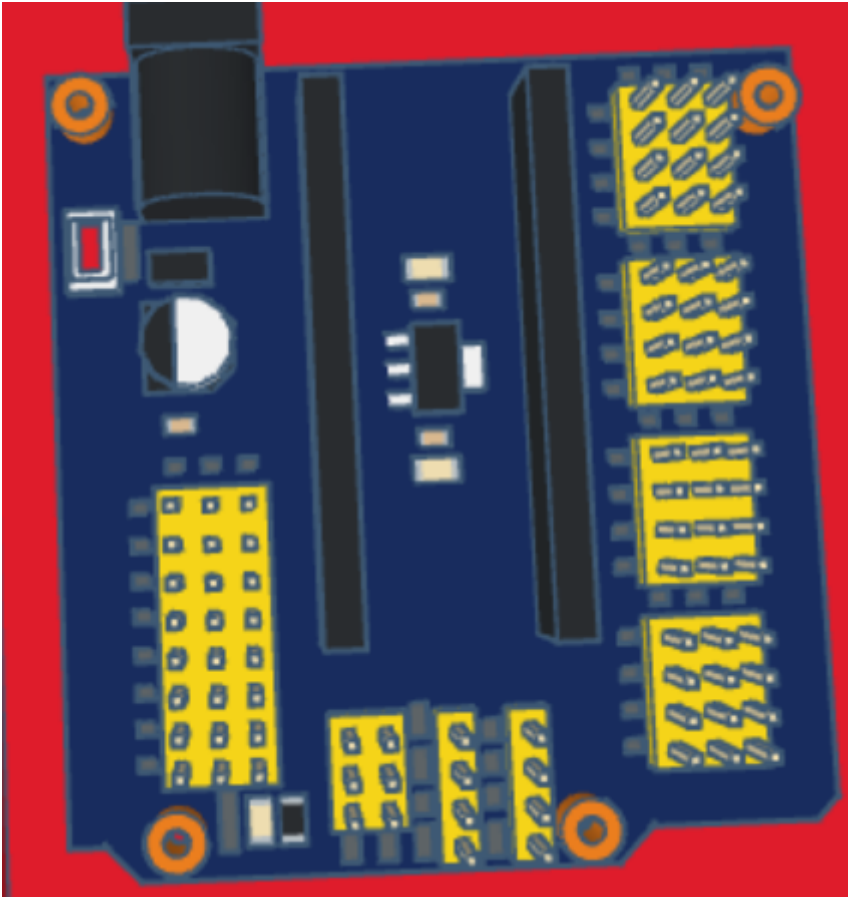


- Placer 4 cylindre en rectangle avec une longueur entre 2 cylindre de 99 mm et une largeur de 29 mm.
- Ces cylindres vont permettre de placer le shield pour la pile.



- Pour le shield dans le champs de recherche à droite des formes simple taper "**shield**".

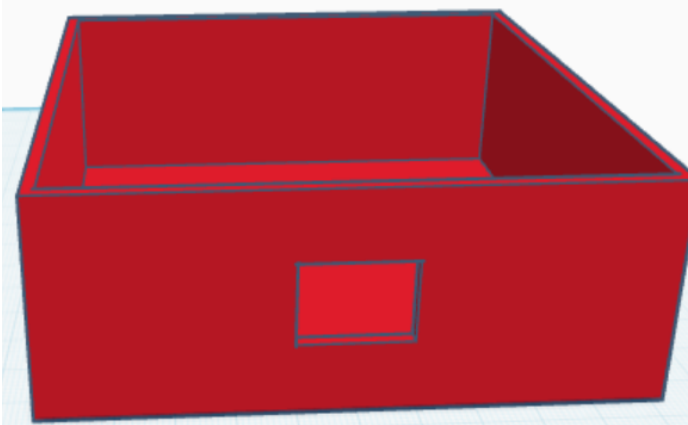
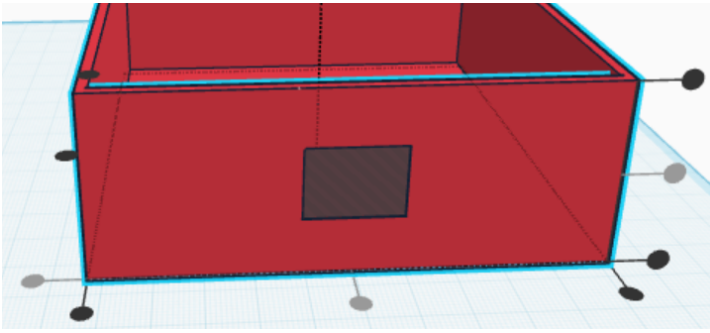
- Sélectionner le 3ème pour le mettre dans la boîte et placer des cylindre dans les 4 trous du shield ( essayer de les placer dans un coin).



#### **Création du trou pour le Capteur :**

- Créer un cube de 20 x 20 x 15 mm.
- Puis venir percer une paroi latéral de la boîte avec l'outil aligner comme précédemment.

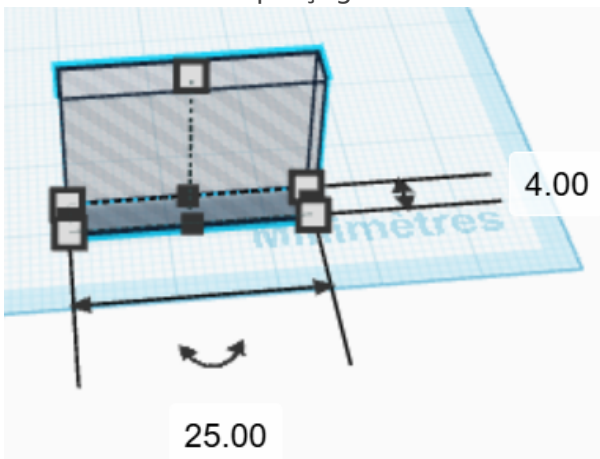




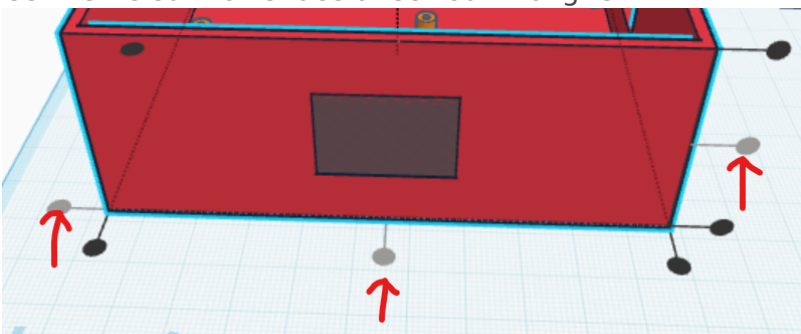
- Placer le trou à l'emplacement souhaité pour le passage du capteur.

### Création du trou pour l'afficheur :

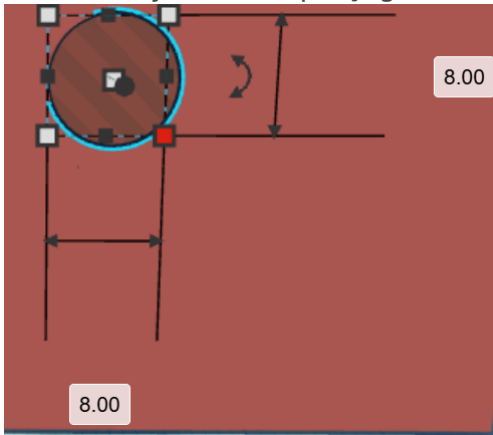
- Créer un cube de perçage de dimensions 25 x 4 x 18 mm.



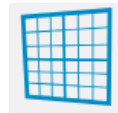
- Centrer le sur votre face avec l'outil "aligner"



- Créer un cylindre de perçage en le plaçant sur la face de dimensions 8 x 8 x 5 mm

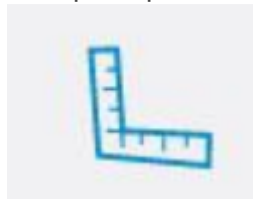


- Le rentrer dans la paroi de 3mm à l'aide de la flèche noir

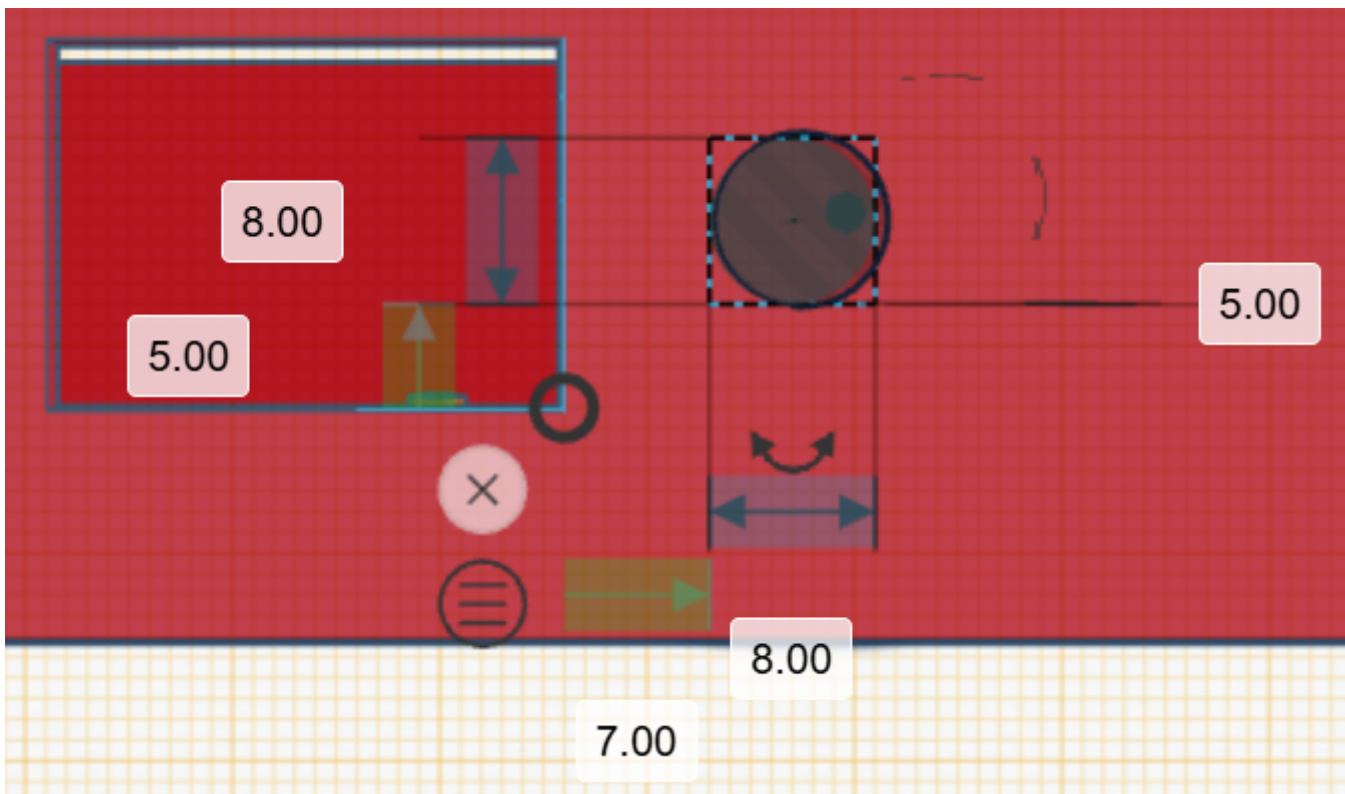


- Placer un nouveau plan sur votre face à l'aide de cet outil : il suffit juste de cliquer dessus puis de cliquer sur votre face.

L'outil « règle » sera votre meilleur ami pour placer correctement le cylindre.



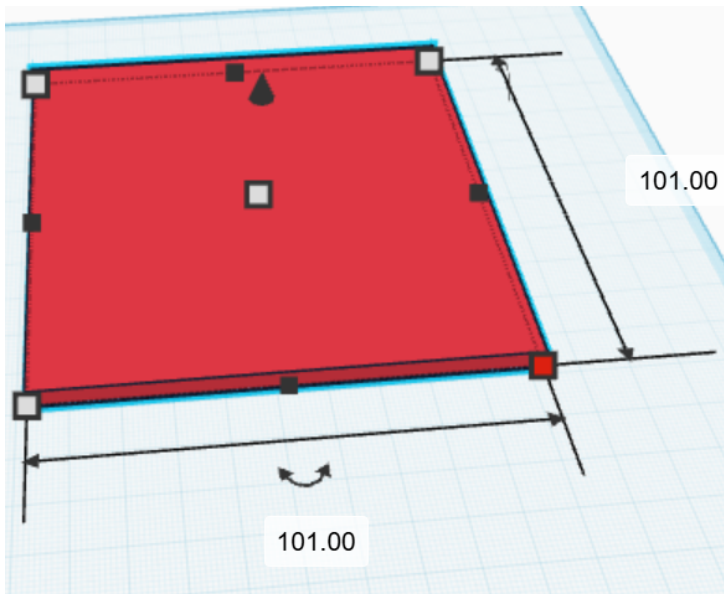
Placez cet outil dans le coin inférieur droit de l'emplacement de l'afficheur. Une fois placé, des cotes apparaîtront, comme sur l'image suivante (ne pas oublier de cliquer sur le cylindre), puis les modifier pour avoir les mêmes que sur l'image :



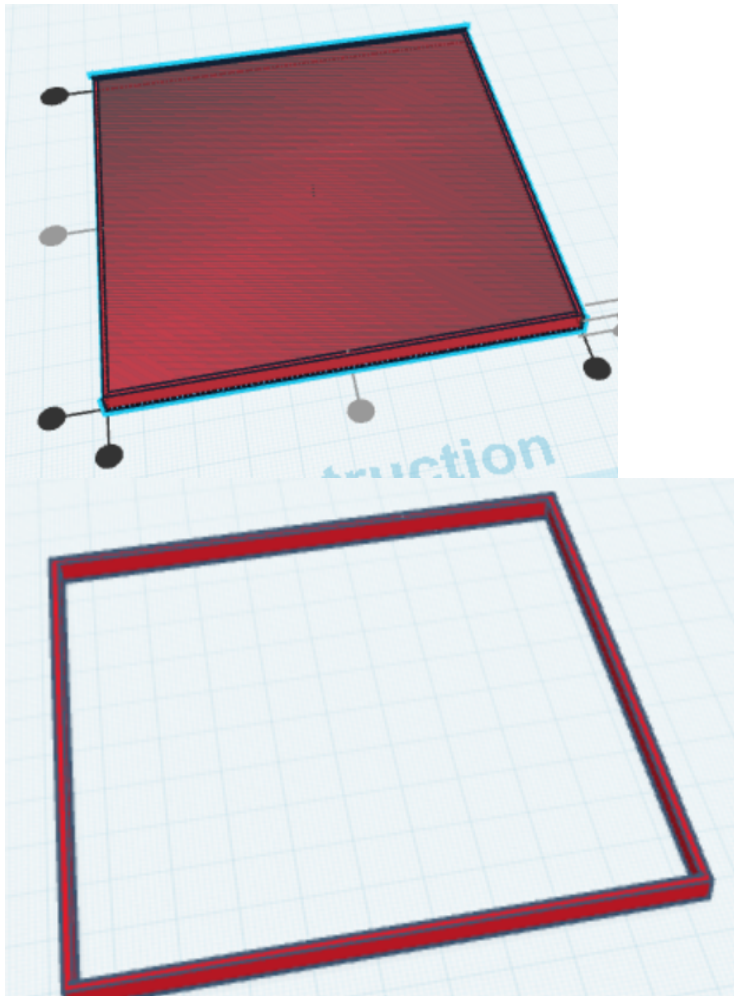
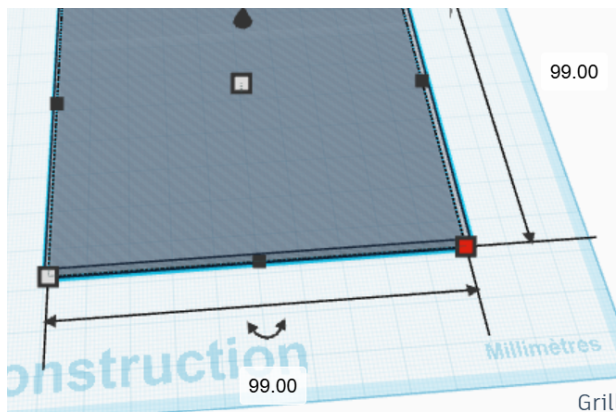
- Regrouper maintenant le cylindre avec la boîte
- Replacer votre plan sur le sol

### Conception du Capot :

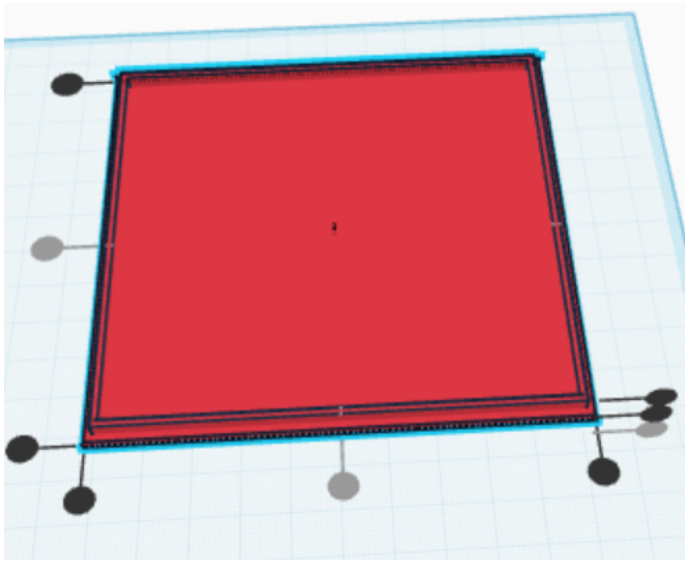
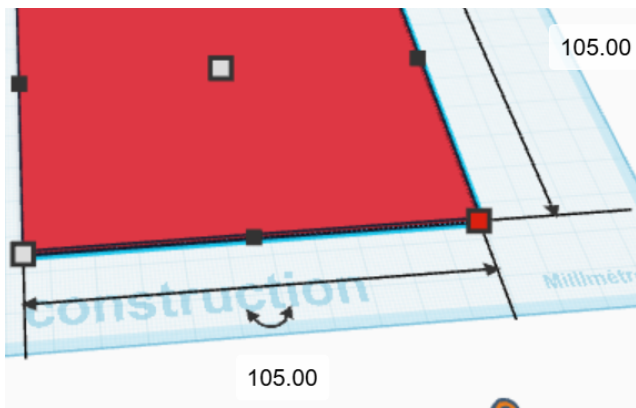
- Créer un nouveau cube de 101 x 101 x 4 mm.



- Ajouter un deuxième cube de 99 x 99 x 4 mm en mode perçage et venir percer ce dernier au premier grâce à l'outil "perçage" en le centrant avec l'outil "aligner".



- Puis ensuite ajouter un cube en dessous de 105 x 105 x 1 mm toujours en le centrant



#### **Ajustements Finaux :**

- Vérifier l'alignement du capot avec le boîtier principal.
- Ajuster la position du capot pour qu'il puisse se fermer correctement.
- Réviser les dimensions et l'agencement global pour assurer la cohérence du design.

#### **Validation et Export :**

- Vérifier la modélisation en 3D pour s'assurer de la conformité avec les spécifications.
- Exporter le modèle au format souhaité pour une utilisation ultérieure (STL, etc.).

## **3. Impression 3D**

Afin de comprendre de qu'est l'impression 3D vous pouvez visiter les pages présente dans la catégorie suivante :

<https://innovation.iha.unistra.fr/books/2-fablab-formation-machines-logiciels/chapter/impression-3d-fdm>

Et pour savoir comment lancer une impression sur l'une de nos machines, veuillez vous référer à la page suivante :

<https://innovation.iha.unistra.fr/books/2-fablab-formation-machines-logiciels/page/lancer-une-impression-avec-bambu-studio>

# Phase 2 - Impression 3D

## Phase 2 - Impression 3D

### Objectifs :

- Acquérir une introduction à la préparation de modèles pour l'impression 3D.
- Maîtriser l'utilisation du logiciel Cura.
- Préparer un modèle 3D pour l'impression avec les paramètres appropriés.

### Compétences techniques :

- Utilisation générale de la préparation pour l'impression 3D.
- Utilisation spécifique du logiciel Cura.

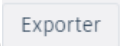

### Savoir-être et compétences transversales :

- Capacité à suivre des instructions détaillées.

### Déroulement de la séance :

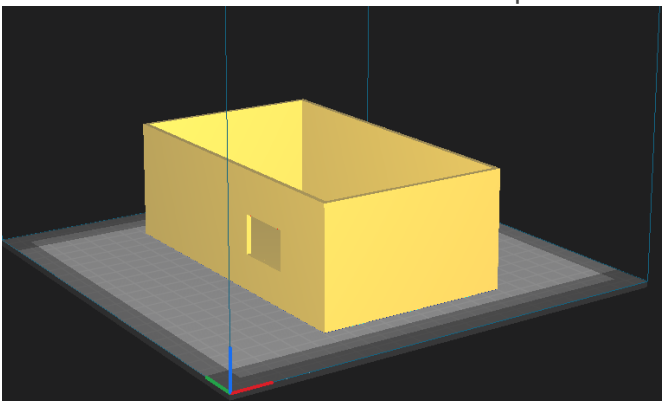
- Consignes de sécurité et précautions matérielles.
- Travail individuel.
- Présentation de l'activité du jour, rappel des objectifs.

### 1.Exportation de la boîte :

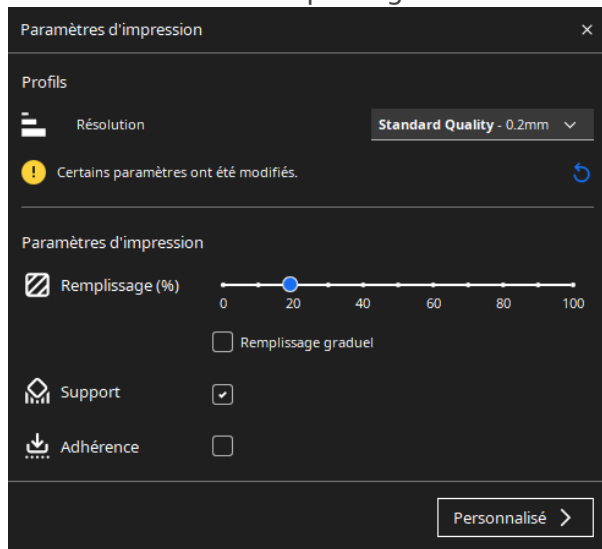
- Pour exporter un fichier de Tinkercad il suffit de cliquer sur ce bouton  puis sur celui la  et le voilà qu'il se retrouve dans vos téléchargement.

### 2.Préparation du fichier pour l'impression :

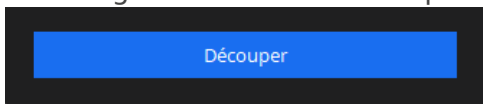
- Dans un premier temps lancez le logiciel "Ultimaker Cura".
- Cherchez votre fichier dans votre explorateur et faite le glisser dans Cura.



- Par la suite configurez les paramètres d'impression :
  - Sélection d'une qualité standard.
  - Activation des supports pour les parties surplombantes.
  - Mettre à 20% de remplissage.



- Ajustements éventuels des paramètres pour optimiser l'impression.
- Pour la génération du code G pour l'imprimante 3D appuyez sur ce bouton



- Maintenant vous avez plus qu'à transférer ce fichier sur la carte SD de l'imprimante utilisée et lancer l'impression.

## Conclusion / Rangement :

- Récapitulation des points clés de la séance.
- Rangement du matériel en fin de séance.

Dans cette phase, les participants vont apprendre à préparer leur modèle 3D pour l'impression en utilisant le logiciel Cura. Ils vont importer leur fichier STL, ajuster les paramètres d'impression, générer le code pour l'imprimante 3D et exporter le fichier prêt pour l'impression. Si des questions se posent ou si vous avez besoin de plus d'informations, n'hésitez pas à demander !



# Fonctionnement et programmation des composants

----- Début de la nouvelle page

## **Objectif :**

Comprendre le fonctionnement des composants du projet et apprendre à les programmer

## **1. Introduction à l'Arduino Nano**

Qu'est-ce qu'un Arduino ?

<https://innovation.iha.unistra.fr/books/robotique-educative/page/quest-ce-que-larduino>

## **2. Capteur de température et d'humidité (DHT22)**

Qu'est ce qu'un capteur de température et d'humidité (DHT22) et comment il fonctionne ?

<https://innovation.iha.unistra.fr/books/robotique-educative/page/fonctionnement-et-programmation-du-capteur-de-temperature-et-dhumidite-dht22/edit>

## **3. Ecran OLED**

Qu'est ce qu'un écran OLED et comment il fonctionne ?

## Objectifs

### Compétences techniques:

- Programmation pour stocker les valeurs de température et d'humidité dans la mémoire de l'Arduino Nano.

## Déroulement de la séance

### 1. Introduction:

- Présentation des éléments nécessaires pour la programmation.
- Importance de la programmation dans le projet de station météo.

### 2. Programmation:

- Commençons par la programmation sur Mblock.

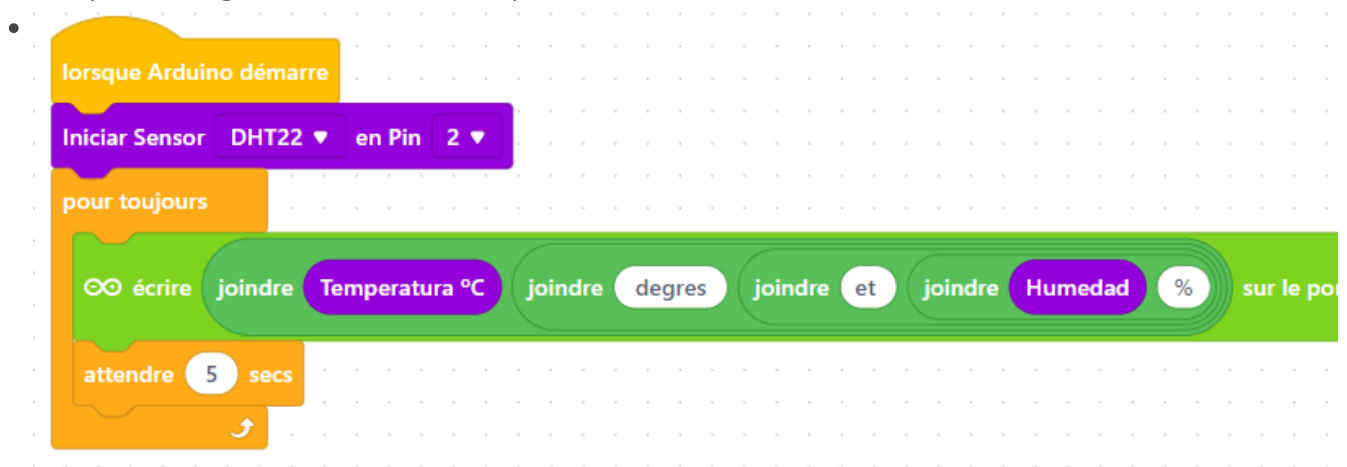


- N'oubliez pas d'installer l'extension pour le capteur en cliquant sur ce bouton <sup>extension</sup>

, puis taper dans la barre de recherche "DHT22" et ensuite juste besoin de l'ajouter.



- Maintenant vous avez juste besoin de faire le code qui permet de mesuré la température grâce au DHT 22 et qui la renvoie sur le moniteur série.



- Avant de téléverser le code n'oubliez pas de décocher ces 2 cases ci-dessous qui se situe dans le moniteur série.

☐ Envoyer en hexadécimal ☐ Réception hexadécimale

- Pour la suite nous allons passer sur le logiciel "Arduino IDE" car il nous offre plus de possibilités.
- Dans un premier temps télécharger la librairie Arduino DHT de Adafruit

<https://www.arduino.cc/reference/en/libraries/dht-sensor-library/>

- bien installer la dépendance Adafruit Unified Sensor

#### DHT sensor library by Adafruit

1.4.6 installed

Arduino library for DHT11, DHT22, etc Temp & Humidity Sensors  
Arduino library for DHT11, DHT22, etc Temp & Humidity Sensors  
[More info](#)

1.4.6 ▼

REMOVE

```
#include <SoftwareSerial.h>
#include <DHT.h>
#include <EEPROM.h>

#define DHTPIN 9
#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);

const int EEPROM_SIZE = 512;           // Taille de l'EEPROM en octets
const int ENTRY_SIZE = sizeof(float) * 2; // Taille d'une entrée (température + humidité)
const int NUM_ENTRIES = EEPROM_SIZE / ENTRY_SIZE; // Nombre total d'entrées possibles

int currentAddress = 0; // Adresse actuelle pour stocker
bool storageFull = false; // Indique si l'EEPROM est pleine

unsigned long previousMillis = 0; // Dernier temps de stockage
const unsigned long interval = 3600000; // Intervalle d'une heure (en millisecondes)

void setup() {
  Serial.begin(9600);
  dht.begin();
  Serial.println("Démarrage du système...");
}
```

```

void loop() {
    unsigned long currentMillis = millis();

    // Vérifie si une heure s'est écoulée
    if (!storageFull && currentMillis - previousMillis >= interval) {
        previousMillis = currentMillis;

        // Lecture des données du capteur
        float temperature = dht.readTemperature();
        float humidity = dht.readHumidity();

        if (!isnan(temperature) && !isnan(humidity)) {
            // Vérifie si suffisamment d'espace reste dans l'EEPROM
            if (currentAddress + ENTRY_SIZE <= EEPROM_SIZE) {
                // Stocker la température
                EEPROM.put(currentAddress, temperature);
                currentAddress += sizeof(float);

                // Stocker l'humidité
                EEPROM.put(currentAddress, humidity);
                currentAddress += sizeof(float);

                // Affichage des données
                Serial.print("Données stockées à l'adresse ");
                Serial.print(currentAddress - ENTRY_SIZE);
                Serial.print(": Température = ");
                Serial.print(temperature);
                Serial.print(" °C, Humidité = ");
                Serial.println(humidity);
            } else {
                storageFull = true; // Indiquer que l'EEPROM est pleine
                Serial.println("EEPROM pleine. Arrêt du stockage.");
            }
        } else {
            Serial.println("Erreur de lecture du capteur DHT22.");
        }
    }
}

```

- Ce code va nous permettre de stocker la température et l'humidité dans la mémoire de notre Arduino nano et ça toute les heures.
- après avoir stocker les température et l'humidité il faut les récupérer, voici le code :

```
#include <EEPROM.h>

const int EEPROM_SIZE = 512;          // Taille totale de l'EEPROM en octets
const int ENTRY_SIZE = sizeof(float) * 2; // Taille d'une entrée (température +
humidité)
const int NUM_ENTRIES = EEPROM_SIZE / ENTRY_SIZE; // Nombre total d'entrées
(température + humidité)

void setup() {
  Serial.begin(9600);
  Serial.println("Lecture des données stockées dans l'EEPROM...");

  for (int i = 0; i < NUM_ENTRIES; i++) {
    int address = i * ENTRY_SIZE;

    // Lecture des données de l'EEPROM
    float storedTemperature, storedHumidity;
    EEPROM.get(address, storedTemperature);
    EEPROM.get(address + sizeof(float), storedHumidity);

    // Vérification et affichage des données
    Serial.print("Entrée ");
    Serial.print(i + 1);
    Serial.print(" à l'adresse ");
    Serial.print(address);
    Serial.print(": ");

    if (!isnan(storedTemperature) && !isnan(storedHumidity)) {
      Serial.print("Température = ");
      Serial.print(storedTemperature);
      Serial.print(" °C, Humidité = ");
      Serial.print(storedHumidity);
      Serial.println(" %");
    } else {
      Serial.println("Valeurs non valides ou non initialisées.");
    }
  }
}
```

```
    }  
}  
  
void loop() {  
    // Pas besoin de code dans loop pour cette démonstration  
}
```

### 3. Validation:

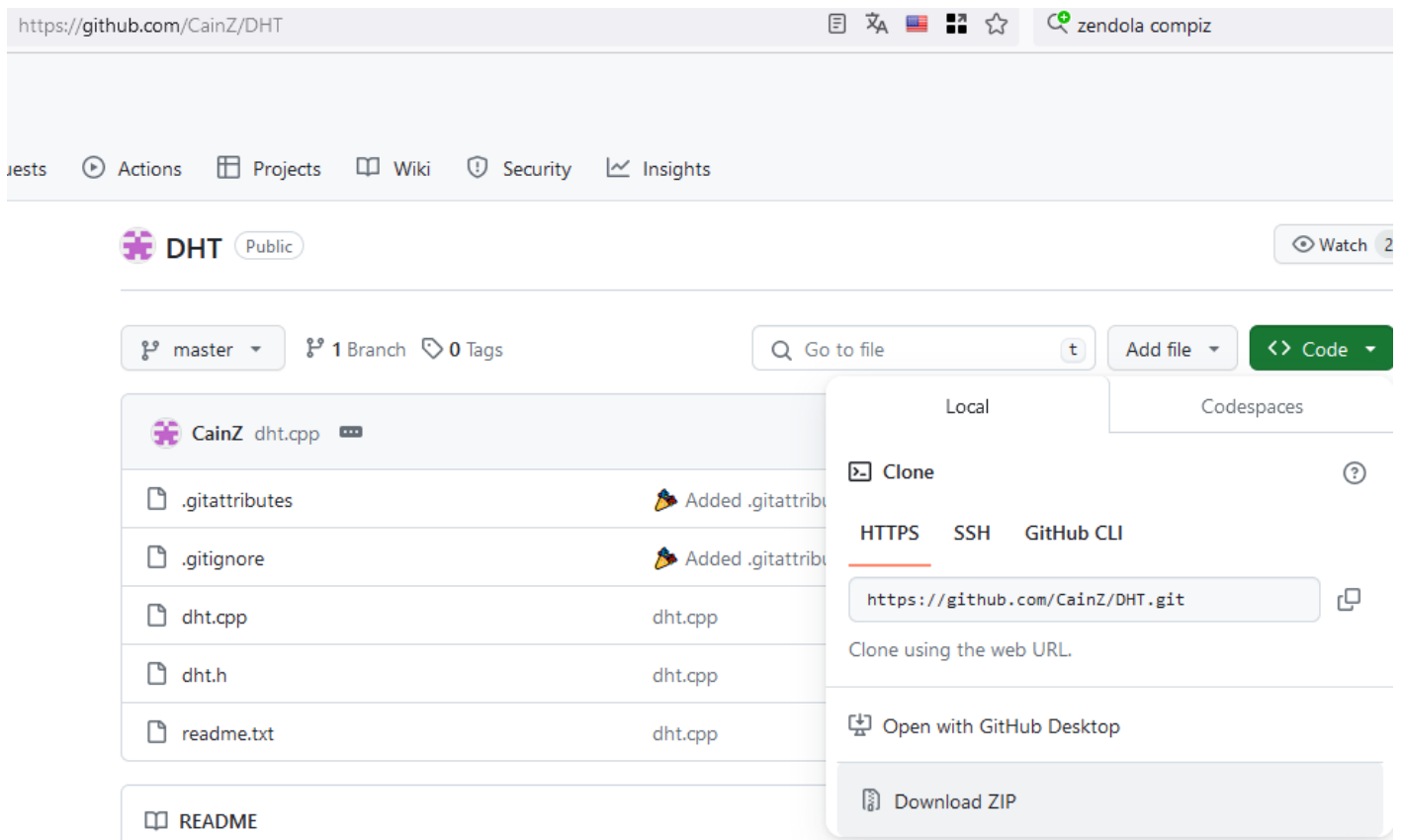
- Vérification du programme pour s'assurer du bon fonctionnement.
- Test du stockage des valeurs dans la mémoire EEPROM.

### 4. Conclusion:

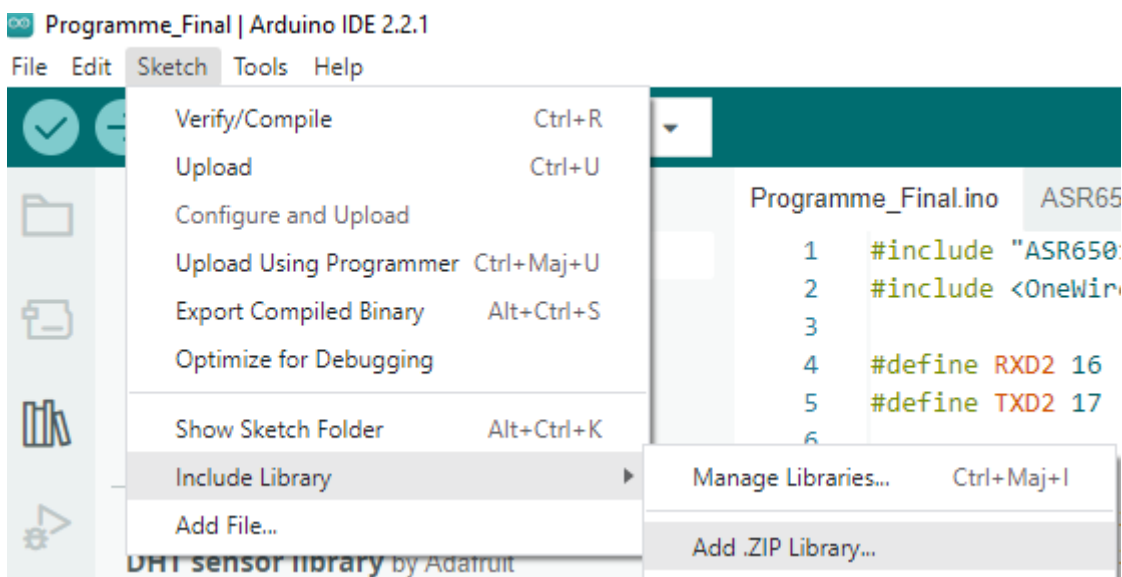
- Récapitulation des points clés de la séance.
- Réponses aux questions éventuelles.

## Résolution de problèmes

- Si `dht.h` n'est pas reconnu quand vous vérifiez/compilez le fichier, essayer une librairie adaptée à la marque de votre DHT22. Par exemple pour DFRobot :
  - [https://wiki.dfrobot.com/DHT22\\_Temperature\\_and\\_humidity\\_module\\_SKU\\_SEN0137#target\\_2](https://wiki.dfrobot.com/DHT22_Temperature_and_humidity_module_SKU_SEN0137#target_2)
- Cliquer sur Code > Télécharger ZIP <https://github.com/CainZ/DHT>



- Ajouter la librairie à la main dans ArduinoIDE



- Vérifier la bonne installation de la librairie, dans "contributed libraries"

Include Library

Add File...

1

}

2

DHT dht(DHTPIN, DHTTYPE);

3

4

int adresseEEPROM = 0; // Adresse de l'EEPROM

5

int adresseDecimaleEEPROM = 0; // Adresse de l'EEPROM pour la température

6

int adresseHumiditeEEPROM = 0; // Adresse de l'EEPROM pour l'humidité

7

int adresseDecimaleHumiditeEEPROM = 0; // Adresse de l'EEPROM pour l'humidité

8

9

void loop() {

10

// Mesurer la température et l'humidité

11

float temperature = dht.readTemperature();

12

float humidite = dht.readHumidity();

13

14

// Vérifier si les mesures sont valides

15

if (!isnan(temperature) & !isnan(humidite)) {

16

// Stocker la température et l'humidité dans l'EEPROM

17

EEPROM.write(adresseEEPROM, temperature);

18

EEPROM.write(adresseDecimaleEEPROM, (int)(temperature \* 10));

19

EEPROM.write(adresseEEPROM, humidite);

20

EEPROM.write(adresseDecimaleHumiditeEEPROM, (int)(humidite \* 10));

21

}

22

}

Manage Libraries...

Add .ZIP Library...

Arduino libraries

Arduino\_BuiltIn

EEPROM

Ethernet

Firmata

HID

Keyboard

LiquidCrystal

Mouse

SD

Servo

SoftwareSerial

SPI

Stepper

TFT

Wire

Contributed libraries

Adafruit Unified Sensor

DHT sensor library



# Phase 4 - Mesure et Analyse des Données de Température avec le Capteur DHT22

## Phase 4 - Mesure et Analyse des Données de Température avec le Capteur DHT22

### Objectifs de la séance :

- Apprendre à utiliser des stations de mesure de température équipées de capteurs DHT22 et d'une carte Arduino Nano.
- Collecter des données de température à différents endroits sur une période donnée.
- Analyser les données recueillies grâce à Excel et réaliser des courbes de tendance.
- Comprendre l'impact des conditions environnementales sur la température.

### Compétences techniques :

- **Utilisation de capteurs et d'Arduino Nano** : Installation, branchement et paramétrage des stations de mesure.
- **Manipulation de données avec Excel** : Importer des fichiers de données, organiser les informations en tableaux, tracer des graphiques (courbes de tendance, histogrammes, etc.).
- **Traitement des données** : Analyser les écarts de température selon les endroits et périodes de mesure.

### Savoir-être :

- **Rigueur** : Suivre les étapes de mesure et d'analyse de façon précise et méthodique.

- **Travail en équipe** : Collaborer pour installer les stations, récolter les données et partager les résultats.
- **Autonomie** : Être capable d'installer la station de mesure et d'effectuer des relevés de manière indépendante.

## Compétences transversales :

- **Résolution de problèmes** : Interpréter les anomalies ou incohérences dans les données et adapter les mesures.
  - **Communication scientifique** : Présenter de manière claire et synthétique les résultats sous forme de graphiques et de conclusions.
  - **Esprit critique** : Analyser les données avec discernement pour en tirer des conclusions pertinentes.
- 

## Déroulement de la séance :

### 1. Introduction

- Présentation des objectifs de la séance.
- Rappel du fonctionnement du capteur DHT22 et de l'Arduino Nano.
- Explication des points de mesure : définir les différents endroits où seront placées les stations (intérieur, extérieur, pièces fermées, etc.).

### 2. Installation des stations de mesure

- Mise en place des stations avec le capteur DHT22 à différents endroits prédéfinis.
- Vérification de la connectivité et du bon fonctionnement de chaque station.
- Lancement de la collecte des données (durée de mesure à définir : quelques heures ou une journée complète).

### 3. Collecte des données

- Chaque participant vérifie régulièrement les stations et enregistre les données (extraction via l'Arduino).
- Récupération des température via l'Arduino IDE, puis exportation sur Excel ou Open Office.

```
Donnée stockée à l'adresse 0: Température = 21.30 °C, Humidité = 49.90 %
Donnée stockée à l'adresse 8: Température = 21.40 °C, Humidité = 50.40 %
Donnée stockée à l'adresse 16: Température = 21.50 °C, Humidité = 49.50 %
Donnée stockée à l'adresse 24: Température = 21.40 °C, Humidité = 49.20 %
Donnée stockée à l'adresse 32: Température = 21.40 °C, Humidité = 49.30 %
Donnée stockée à l'adresse 40: Température = 21.40 °C, Humidité = 49.40 %
Donnée stockée à l'adresse 48: Température = 21.30 °C, Humidité = 49.40 %
Donnée stockée à l'adresse 56: Température = 21.30 °C, Humidité = 49.50 %
Donnée stockée à l'adresse 64: Température = 21.30 °C, Humidité = 49.50 %
Donnée stockée à l'adresse 72: Température = 21.30 °C, Humidité = 49.90 %
Donnée stockée à l'adresse 80: Température = 21.30 °C, Humidité = 49.70 %
Donnée stockée à l'adresse 88: Température = 21.30 °C, Humidité = 49.80 %
Donnée stockée à l'adresse 96: Température = 21.30 °C, Humidité = 49.80 %
Donnée stockée à l'adresse 104: Température = 21.30 °C, Humidité = 50.30 %
Donnée stockée à l'adresse 112: Température = 21.10 °C, Humidité = 51.60 %
```

Donnée stockée à l'adresse 0: Température = 21.30 °C, Humidité = 49.90 %	21.30	49.90
Donnée stockée à l'adresse 8: Température = 21.40 °C, Humidité = 50.40 %	21.40	50.40
Donnée stockée à l'adresse 16: Température = 21.50 °C, Humidité = 49.50 %	21.50	49.50
Donnée stockée à l'adresse 24: Température = 21.40 °C, Humidité = 49.20 %	21.40	49.20
Donnée stockée à l'adresse 32: Température = 21.40 °C, Humidité = 49.30 %	21.40	49.30
Donnée stockée à l'adresse 40: Température = 21.40 °C, Humidité = 49.40 %	21.40	49.40
Donnée stockée à l'adresse 48: Température = 21.30 °C, Humidité = 49.40 %	21.30	49.40
Donnée stockée à l'adresse 56: Température = 21.30 °C, Humidité = 49.50 %	21.30	49.50
Donnée stockée à l'adresse 64: Température = 21.30 °C, Humidité = 49.50 %	21.30	49.50
Donnée stockée à l'adresse 72: Température = 21.30 °C, Humidité = 49.90 %	21.30	49.90
Donnée stockée à l'adresse 80: Température = 21.30 °C, Humidité = 49.70 %	21.30	49.70
Donnée stockée à l'adresse 88: Température = 21.30 °C, Humidité = 49.80 %	21.30	49.80
Donnée stockée à l'adresse 96: Température = 21.30 °C, Humidité = 49.80 %	21.30	49.80
Donnée stockée à l'adresse 104: Température = 21.30 °C, Humidité = 50.30 %	21.30	50.30
Donnée stockée à l'adresse 112: Température = 21.10 °C, Humidité = 51.60 %	21.10	51.60

- Pour récupérer les température facilement il y a cette formule =DROITE(GAUCHE(A1; NBCAR(A1)-23); 5).
- Pour récupérer la valeur de l'humidité vous pouvez utiliser cette formule =DROITE(GAUCHE(A1; NBCAR(A1)-2); 5) si vous avez collé votre texte dans la case A1, ensuite si vous êtes sur Exel n'oubliez pas de les convertir en valeurs.

Coller

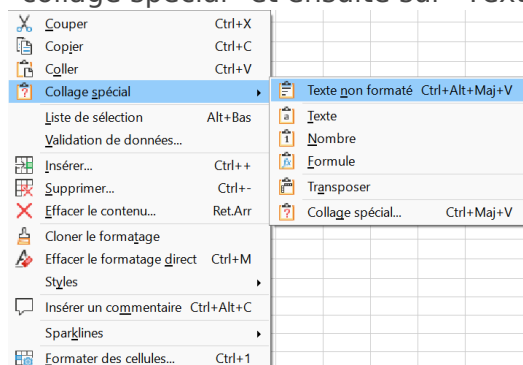
Options de collage :



- Collage spécial...

Recherche intelligente

- **Attention si vous êtes sur Libre Office**, il faut copier vos valeur puis les coller en "collage spécial" et ensuite sur "Texte non formaté".

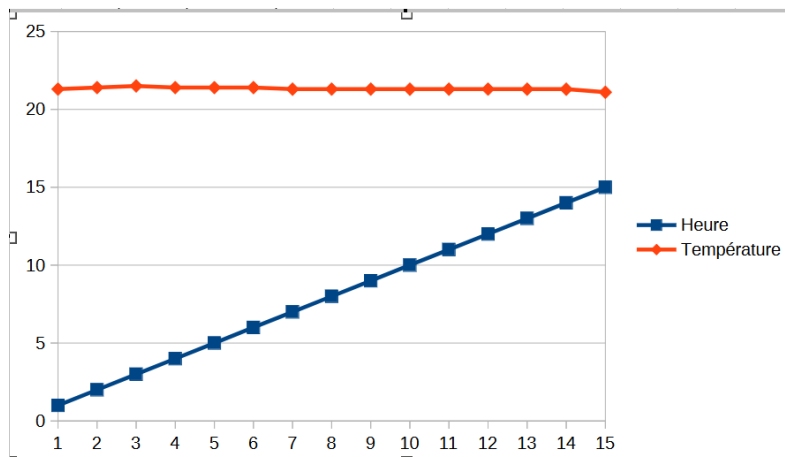


Et ensuite appuyer sur "OK".

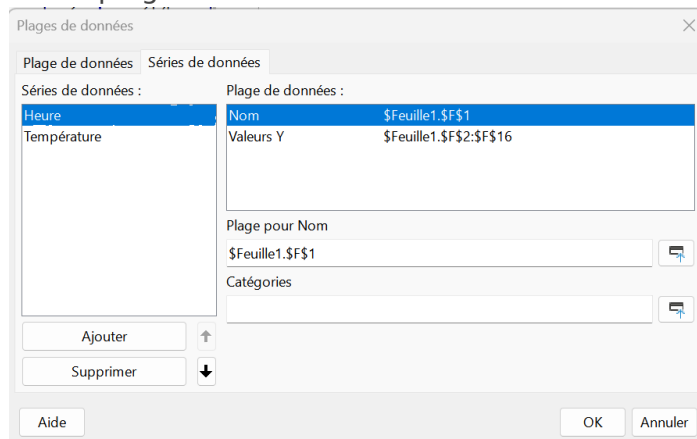
- Faire de même pour les valeurs de l'humidité.

#### 4. Analyse des données sur Excel

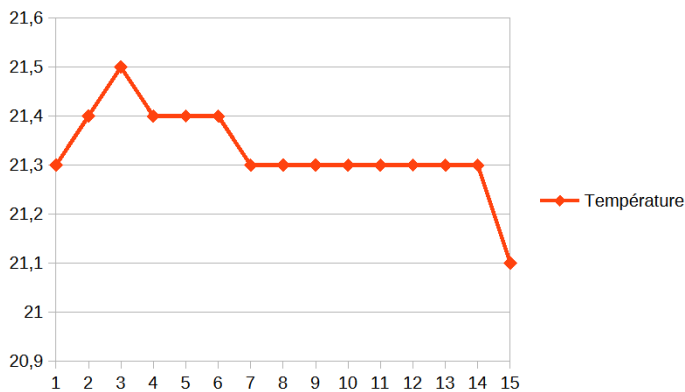
- Importation des données dans Excel ou Libre Office.
- Structuration des informations sous forme de tableau.
- Création des graphiques : courbes de tendance pour visualiser les écarts de température selon les points de mesure.
- Pour créer un graphique sur Excel sélectionnez votre tableau et allez dans "Insertion" puis "Graphiques Recommandés".
- Sur Libre Office sélectionnez vos valeurs puis allez dans "Insertion", "Diagrammes", "Ligne", et prenez le 2ème.



- Après avoir obtenu l'image ci-dessus, faites un clic droit sur le graphique et ensuite dans "plage de données" allez dans "séries de données".



- Et vous avez juste à supprimer les données des heures.



- **Sur Excel** n'oubliez pas de changer l'échelle des température en mettant la température la plus basse à 0.
- Analyse des résultats : interpréter les différences entre les différents points de mesure.
- Faire la même chose pour l'humidité.

## 5. Conclusion et échanges

- Chaque participant présente ses résultats sous forme de graphiques.
- Discussion autour des facteurs qui peuvent influencer les écarts de température (environnement, exposition au soleil, ventilation, etc.).
- Synthèse des principaux apprentissages de la séance.