

Ateliers de développement de la station de mesure de l'eau (5 jours)

Mission Otez l'eau

Depuis quelques mois, un changement soudain dans la qualité de l'eau a été signalé dans plusieurs régions du monde, ce qui soulève des inquiétudes. Les premières enquêtes révèlent une contamination progressive de l'eau, liée à des défaillances dans les systèmes de traitement et de rejets des eaux usés par une usine. Face à cette menace environnementale et sanitaire, l'association Savoirs Vivants, et l'IUT de Haguenau, ont été missionnés par les services secrets français pour monter une équipe de roboticiens et de chercheurs pour identifier l'usine à la source de ces contaminations.

En tant que nouveaux membres de ce consortium de chercheurs votre première mission sera de vous trouver un nom d'équipe pour réussir à communiquer facilement avec le reste des autres chercheurs et votre chef d'équipe.

Votre mission consiste à concevoir, assembler, et programmer la **Station AquaLife** pour qu'elle soit capable de détecter les contaminants dans l'eau avec précision et efficacité. Pour cela, vous utiliserez des technologies de pointe en robotique, en informatique et en analyse chimique.

La **Station AquaLife** que vous allez concevoir pour collecter et analyser des échantillons d'eau en temps réel, fournira des données essentielles pour identifier les sources d'eau contaminé et ainsi trouver la source de ces contaminations.

Déroulement du projet :

- **Lundi : Conception 3D** - Initiez-vous au design assisté par ordinateur pour concevoir la pièce manquante de la Station AquaLife, permettant d'afficher les données sur l'écran.
- **Mardi : Assemblage et Pratique** - Construisez le prototype de votre Station AquaLife en assemblant les capteurs et les circuits nécessaires.
- **Mercredi : Programmation Fonctionnelle** - Implémentez le code informatique qui permettra à la Station AquaLife de collecter et d'analyser les données.
- **Jeudi : Tests chimiques** - Testez votre Station AquaLife avec différents échantillons d'eau récoltés autour des usines ciblées.

- **Vendredi : Analyse et Présentation** - Interprétez les données collectées, identifiez les tendances et préparez une présentation de vos résultats. Cette étape sera cruciale pour identifier la source de la contamination.

En participant à la **Mission Otez l'eau**, vous apportez une contribution concrète à la résolution d'un problème environnemental d'envergure. Vous acquerez des compétences en robotique, en analyse de données et en travail d'équipe, tout en aidant à protéger la ressource la plus précieuse de notre planète : l'eau. **Prêts à relever ce défi vital ?**

Préparation des Matériaux / Kits

Station de mesure de l'eau

Liste du matériel de la station : [Liste_materiel.pdf](#)

Connexion des capteurs sur la carte d'extension DRF0762 : [connexion_capteurs_carte_DFR0762.pdf](#)

Arduino

Installation de la carte ESP2 et des bibliothèques pour la station d'eau dans l'IDE Arduino :

[Arduino_Installation_Cartes_Bibliotheques.pdf](#)

Programme Arduino complet de la station d'eau : [programmeEauIncomplet.ino](#)

Ci-dessous, le programme simplifié :

```
/// CODE SIMPLIFIE SANS TDS ET TURBIDITE
// #include <Arduino.h>
#include <U8g2lib.h> // by oliver <olikraus@gmail.com> testé 2.34.22
#include <DHT.h>
#include <OneWire.h>
#include <DallasTemperature.h>

//*****
//          Constante de temps et broches de connexion des capteurs          //
//*****

const int pause_affichage_ms = 100;      // temps de pause d'affichage en milliseconde

// #define TDS_Sensor_Pin_Num A2          // TDS sensor on Analog pin Ax
// #define Turbidity_Sensor_Pin_Num A3    // Turbidity sensor on Analog pin Ax
```

```

#define pH_Sensor_Pin_Num A4          // pH sensor on Analog pin Ax
#define DS18S20_Pin_Num_v2 D2         // DS18B20 sensor on digital pin
Dx                                     <===== Zone d'affectation des capteurs et actionneurs
#define DHT_Sensor_Pin_Num D6         // DHT on Digital pin Dx

//*****//

//Données et configuration pour le capteur DHT
#define ref_DHT DHT22                  // référence du DHT (DHT11, DHT22...)
char Hum_DHT[8], Temp_DHT[8];
DHT dht(DHT_Sensor_Pin_Num, ref_DHT);

// Données pour le capteur pH meter (v2) ==> site DFRobot
const int numMeasure = 10;
char pH_2[8];
float offset_pH2 = 0;    // valeur standard = -1.8 avec la carte pH alimentée en 3.3V et -2.5
avec carte alimentée en 5V

// Donnée TDS meter (v1)
// char TDS_1[8];
// float temperature = 25;

// Données pour le capteur de turbidité
// char Turbi_volt[8], NTU_strg[8];
// float Turbi_val = 0;

// Configuration pour le capteur de température DS18B20 (v2)
OneWire ds(DS18S20_Pin_Num_v2); //on digital pin 2
char Temp_DS_2[8];

// Configuration de l'afficheur SSD1315
U8G2_SSD1306_128X64_NONAME_F_SW_I2C u8g2(/* rotation=*/ U8G2_R0, /* clock=*/ SCL, /* data=*/
SDA, /* reset=*/ U8X8_PIN_NONE);    //Low speed I2C
/*
    U8g2lib Example Overview:
        Frame Buffer Examples: clearBuffer/sendBuffer. Fast, but may not work with all Arduino
boards because of RAM consumption
        Page Buffer Examples: firstPage/nextPage. Less RAM usage, should work with all Arduino
boards.

```

U8x8 Text Only Example: No RAM usage, direct communication with display controller. No graphics, 8x8 Text only.

```
*/

void get_DHT()
{
    float h = 0.0;
    float t = 0.0;

    h = dht.readHumidity();
    dtostrf(h, 5, 1, Hum_DHT);      // conversion digital to string
    t = dht.readTemperature();
    dtostrf(t, 5, 1, Temp_DHT);

    Serial.print("DHT ==>   Humidité : ");
    Serial.print(Hum_DHT);
    Serial.print("   Température (°C) : ");
    Serial.println(Temp_DHT);
}

void get_pH_2()
{
    int buf[10], temp;

    for(int i=0; i<numMeasure; i++)      //Get 10 sample value from the sensor for smooth the
value
    {
        buf[i]=analogRead(pH_Sensor_Pin_Num);
        delay(10);
    }

    for(int i=0; i<numMeasure-1; i++)      //sort the analog from small to large
    {
        for(int j=i+1; j<numMeasure; j++)
        {
            if(buf[i]>buf[j])
            {
                temp=buf[i];
                buf[i]=buf[j];
            }
        }
    }
}
```

```

        buf[j]=temp;
    }
}
}

unsigned long int sumValue = 0; //Store the average value of the sensor feedback

for(int i=2;i<numMeasure-2;i++) //take the average value of 6 center
sample
    sumValue+=buf[i];

float phValue=(float)sumValue*5.0/4096/(numMeasure-4); //convert the analog into
millivolt
// phValue=3.5*phValue + offset_pH2; //convert the millivolt into pH
value avec carte pH alimentée en 5v
    phValue=5.3*phValue + offset_pH2; //convert the millivolt into pH
value avec carte pH alimentée en 3.3v

dtostrf(phValue, 5, 1, pH_2);

Serial.print("pH_2 value: ");
Serial.println(phValue,2);

}

// void get_TDS_1()
// {
//     int TDS_ADC = 0;
//     float TDS_volt = 0;
//     float compensationCoefficient = 1.0+0.02*(temperature-25.0); //temperature
compensation formula: fFinalResult(25^C) = fFinalResult(current)/(1.0+0.02*(fTP-25.0));
//     float compensationVolatge = 0; //temperature compensation
//     float TDS_val = 0;

//     TDS_ADC = analogRead( TDS_Sensor_Pin_Num);

//     TDS_volt = TDS_ADC/4096.0*5.0;
//     compensationVolatge = TDS_volt/comensationCoefficient;
//     TDS_val=(133.42*compensationVolatge*compensationVolatge*compensationVolatge -
255.86*compensationVolatge*compensationVolatge + 857.39*compensationVolatge)*0.5; //convert

```

voltage value to tds value

```
//  dtostrf(TDS_val, 5, 1, TDS_1);

//  Serial.print("TDS_1 valADC (0 à 4095) : ");
//  Serial.print(TDS_ADC);
//  Serial.print("    voltage (V) : ");
//  Serial.print(TDS_volt);
//  Serial.print("    ppm : ");
//  Serial.println(TDS_val);

// }

// void get_Turbidity()
// {
//   int sensorValue = 0;
//   float voltage = 0, NTU = 0;

//   sensorValue = analogRead(Turbidity_Sensor_Pin_Num);    // read the input on analog pin

//   voltage = sensorValue * (5.0 / 4096);                    // Convert the analog reading (which
//   goes from 0 - 4095) to a voltage (0 - 5V)

//   dtostrf(voltage, 5, 1, Turbi_volt);

//   NTU = -1120.4*voltage*voltage + 5742.3*voltage - 4352.9;    //

//   dtostrf(NTU, 5, 0, NTU_strg);

//   Serial.print("Turbidity (volt) : "); // print out the value you read:
//   Serial.print(voltage); // print out the value you read:
//   Serial.print("    Turbidity (NTU) : "); // print out the value you read:
//   Serial.println(NTU); // print out the value you read:

//   Turbi_val = voltage;
// }

void get_TempDS_2()
{
    //returns the temperature from one DS18S20 in DEG Celsius
```

```
byte data[12];
byte addr[8];

if (!ds.search(addr))
{
    //no more sensors on chain,reset search
    Serial.println("No more addresses.");
    Serial.println();
    ds.reset_search();
}

Serial.print("ROM =");
for(int i = 0; i < 8; i++)
{
    Serial.write(' ');
    Serial.print(addr[i], HEX);
}
Serial.println();

if (OneWire::crc8(addr, 7) != addr[7])
{
    Serial.println("CRC is not valid!");
}

// the first ROM byte indicates which chip
byte type_s;
switch (addr[0])
{
    case 0x10:
        Serial.println("  Chip = DS18S20"); // or old DS1820
        type_s = 1;
        break;
    case 0x28:
        Serial.println("  Chip = DS18B20");
        type_s = 0;
        break;
    case 0x22:
        Serial.println("  Chip = DS1822");
        type_s = 0;
```

```

        break;
    default:
        Serial.println("Device is not a DS18x20 family device.");
        break;
    }

    if (addr[0] != 0x10 && addr[0] != 0x28)
    {
        Serial.print("Device is not recognized");
    }

    ds.reset();
    ds.select(addr);
    ds.write(0x44,1);    //start conversion, with parasite power on at the end

    byte present = ds.reset();
    ds.select(addr);
    ds.write(0xBE);      //ReadScratchpad

    for(int i=0; i<9; i++) //weneed9bytes
    {
        data[i]=ds.read();
    }
    ds.reset_search();

    byte MSB=data[1];
    byte LSB=data[0];

    float tempRead=((MSB<<8)|LSB);    //using two's compliment
    float TemperatureSum = tempRead/16;

    // return TemperatureSum;

    dtostrf(TemperatureSum, 5, 1, Temp_DS_2);
    Serial.print("Temperature DS (en °C) : ");
    Serial.println(TemperatureSum);

}

```



```

void setup()
{
    // put your setup code here, to run once:

    Serial.begin(115200);

    Serial.println("*****");
    Serial.println("Setup Serial monitor OK");

    u8g2.begin();
    Serial.println("Setup écran OLED OK");

    dht.begin();

    pinMode(pH_Sensor_Pin_Num, INPUT);          // Set pH sensor pin to input mode
    // pinMode(Turbidity_Sensor_Pin_Num, INPUT); // Set the turbidity sensor pin to input
mode

    Serial.println("End of setup");
    Serial.println("*****");
}

void loop()
{
    // put your main code here, to run repeatedly:

    get_DHT();
    get_pH_2();

    // affichage des données

    u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font

    //*****//
    // écran OLED SSD1315 :
    // fonction U8g2.drawStr(n°colonne,n°ligne,string)

    u8g2.clearBuffer(); // clear the internal memory
    u8g2.drawStr(0,10,"Station eau"); // write something to the internal memory

```

```

u8g2.drawStr(0,25,"Humidity DHT :");    // affichage du texte "Humidity DHT:"
u8g2.drawStr(92,25,Hum_DHT);           // affichage de la valeur de l'humidité
u8g2.drawStr(0,30,"Temp DHT :");       // affichage du texte "Temp DHT:"
u8g2.drawStr(68,30,Temp_DHT);          // affichage de la valeur de température fournie par
le capteur DHT                         <== zone à modifier (remplacer xx par des valeurs)
u8g2.drawStr(0,35,"pH :");              // affichage du texte "pH :"
u8g2.drawStr(20,35,pH_2);              // affichage de la valeur du pH
u8g2.sendBuffer();                     // transfer internal memory to the display

delay(pause_affichage_ms);

//*****//

// get_TDS_1();
// get_Turbidity();
get_TempDS_2();

//*****//

u8g2.clearBuffer();                    // clear the internal memory
// u8g2.drawStr(0,25,"TDS :");           // affichage du texte "TDS :"
// u8g2.drawStr(35,25,TDS_1);            // affichage de la valeur de conductivité TDS
// u8g2.drawStr(0,37,"Turbidity (V) :"); // affichage du texte "Turbidity (V) :"
// u8g2.drawStr(80,37,Turbi_volt);       // affichage de la valeur de turbidité en
volt                                   <== zone à modifier
u8g2.drawStr(0,49,"Temp DS :");         // affichage du texte "Temp_DS :" à la position
u8g2.drawStr(57,49,Temp_DS_2);         // affichage de la valeur de température du
DS18B20
u8g2.sendBuffer();

//*****//

//*****//
//                                     Code pour gérer une LED                                     //
//*****//

// Affciher un message d'alerte selon la valeur de la turbidité

/*

```

```

float seuil = x;                                // valeur du seuil

if (Turbi_val > seuil)                          // comparaison par rapport au seuil
{
    u8g2.drawStr(0,62,"Eau limpide !");
    u8g2.sendBuffer();
}
else
{
    u8g2.drawStr(0,62,"Eau trouble !");
    u8g2.sendBuffer();
}
*/

delay(pause_affichage_ms);

Serial.println();

}

```

Calibration du capteur pH

Programme Arduino pour la calibration du capteur pH : [2024_calibration_materiel_capteur_pH.ino](#)

Procédure de calibration matériel du capteur pH : [Procedure_Calibration_Materiel_Capteur_pH.pdf](#)

Jour 1 - Conception 3D

Mission :

Une dernière pièce est manquante pour la réalisation de la station. Vous allez devoir la concevoir et la produire sur mesure en utilisant un logiciel de conception 3D (TINKERCAD), ce qui permettra d'obtenir le boîtier de la station AquaLife.

Conception Tinkercad T78-12G-PM4

TINKERCAD est un logiciel de conception 3D simple et efficace. Ce tutoriel vous aidera à l'utiliser : <https://innovation.iha.unistra.fr/books/2-fablab-activites-logiciels/page/initiation-3d->

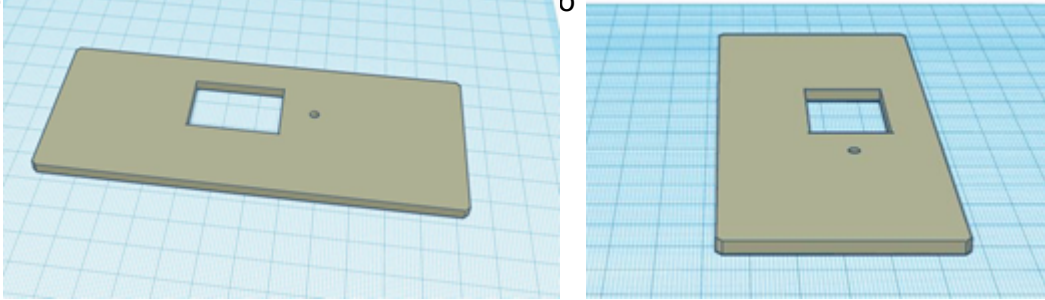
tinkercad

Ouvrir Tinkercad à partir de la barre de recherche Windows de votre ordinateur.

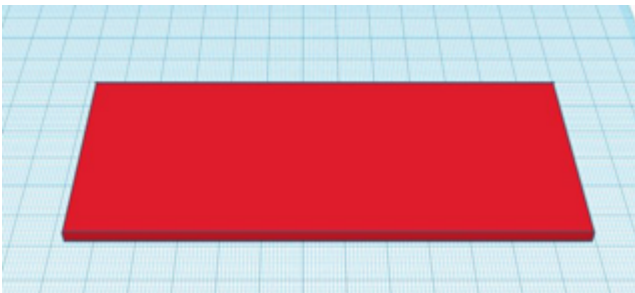
Dans l'onglet **3**, prenez l'outil nommé « boîtes », ensuite faites-le glisser sur le plan de **construction**.

Une fenêtre apparaîtra, vous permettant de régler différents paramètres de cette fenêtre (rayon, étapes, longueur, largeur et hauteur) pour voir ce qui se passe.

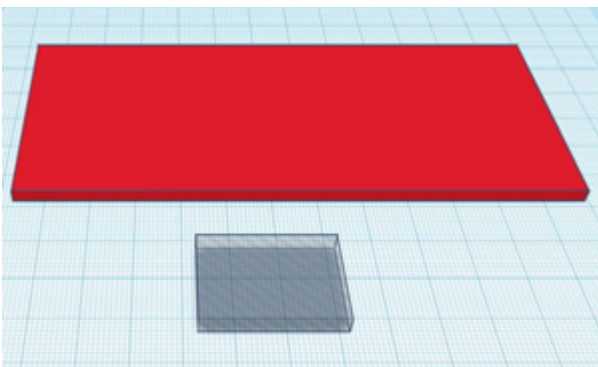
Vous êtes désormais prêt(e) à conceptualiser les pièces nécessaires pour finaliser notre station AquaLife. Celle-ci doit à la fin ressembler à la photo ci-dessous.



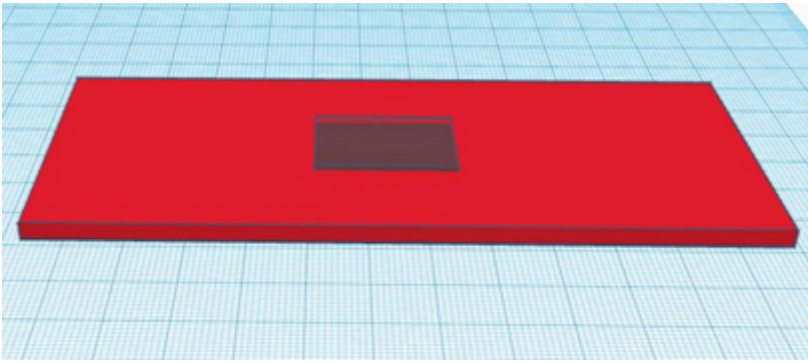
Création de la dernière pièce de la station : Pour commencer, il vous faut un rectangle avec les mesures suivantes : Longueur = 126 mm ; Largeur = 54 mm ; Hauteur = 3 mm.



Nous devons créer un emplacement pour l'afficheur. Pour ce faire, nous avons besoin d'un rectangle de perçage. Donnez-lui les dimensions suivantes : Longueur = 25 mm ; Largeur = 18 mm ; Hauteur = 4 mm.

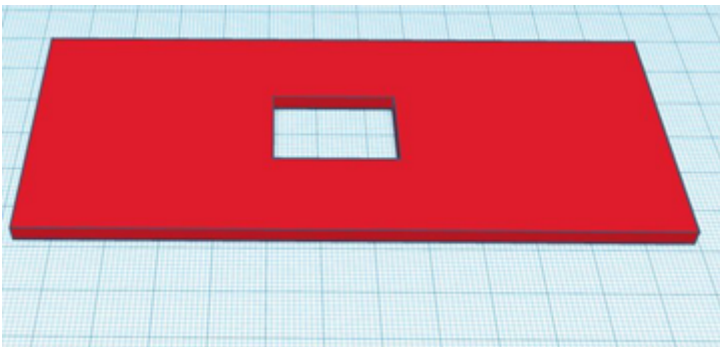


Déplacez-le sur votre rectangle rouge.



Pour assembler les deux, utilisez les raccourcis clavier : Ctrl+A puis Ctrl+G

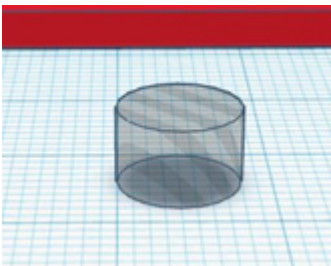
Voici le rendu que vous devriez avoir :



Par la suite, créez l'emplacement pour la vis qui maintiendra votre afficheur.

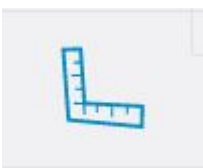
Pour ce faire, créez un cylindre de perçage.

Voici ses dimensions : Diamètre = 8 mm ; Hauteur = 5 mm.

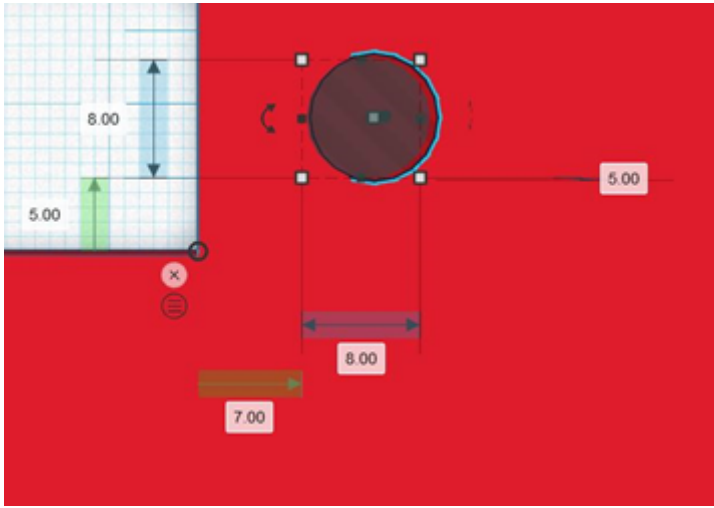


Déplacez-le pour le mettre au bon endroit.

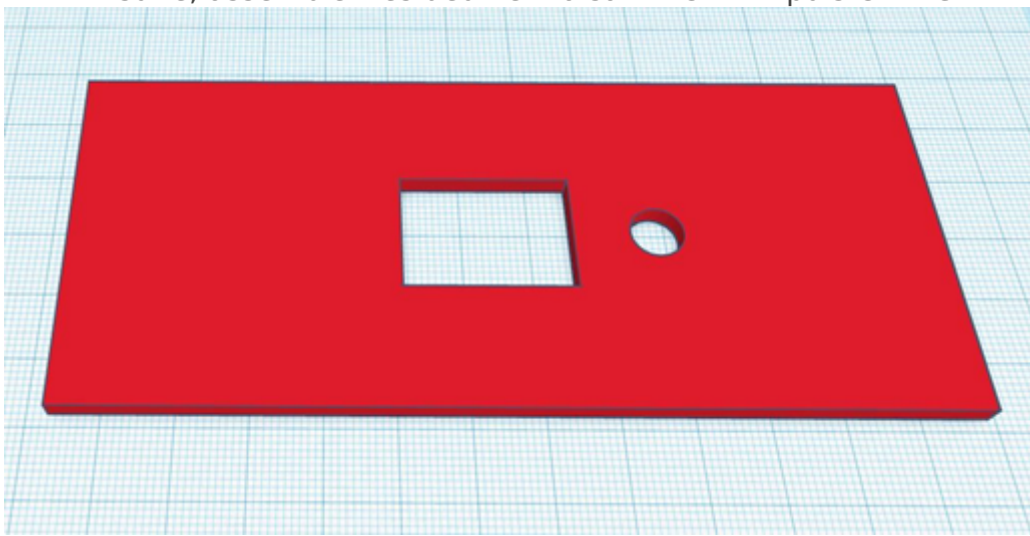
L'outil « règle » sera votre meilleur ami pour placer correctement le cylindre.



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 :



Modifiez seulement les cotes en vert. La cote de gauche sera de 5 mm et celle du bas de 8 mm. Ensuite, assemblez les deux en faisant : Ctrl+A puis Ctrl+G.

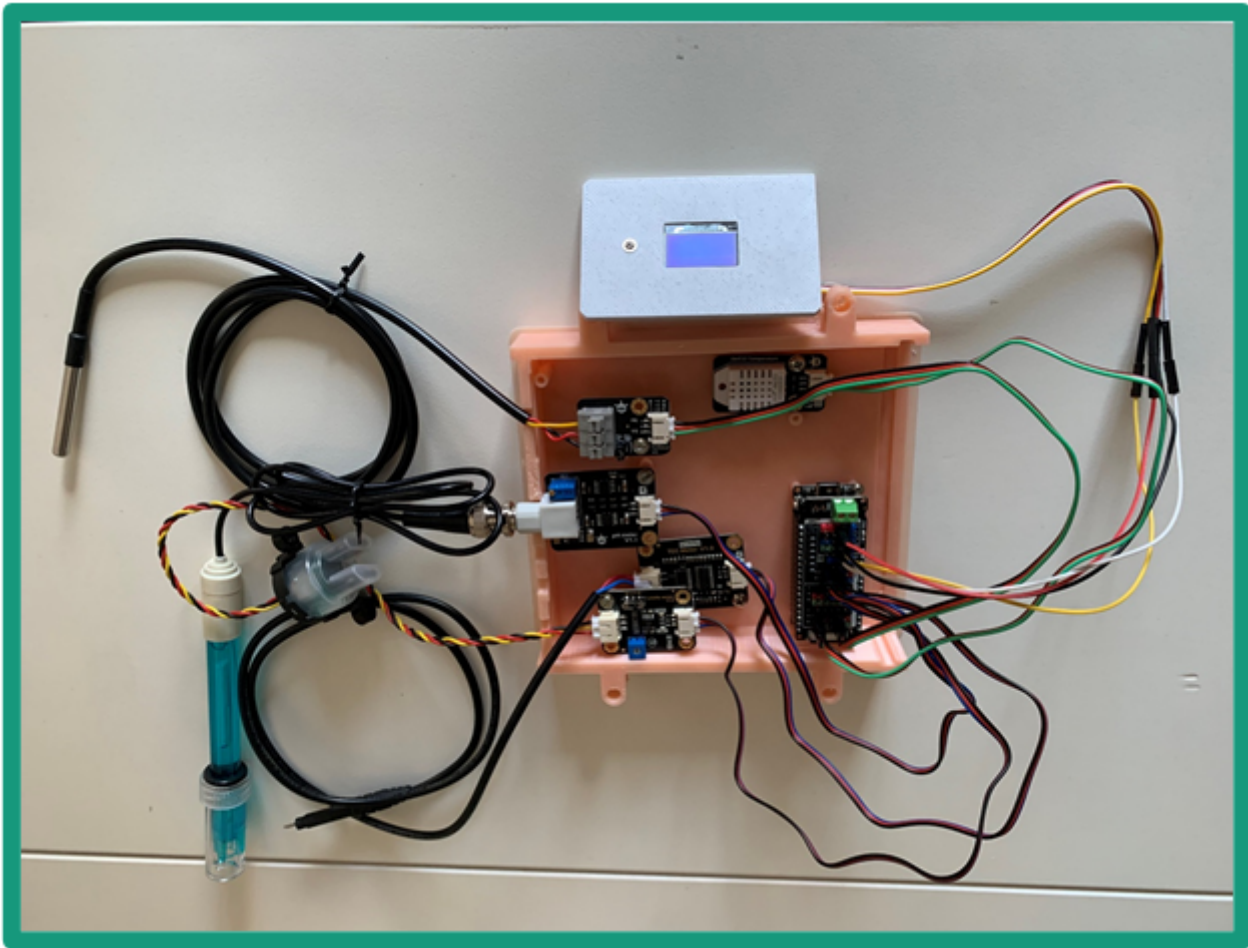


Votre support pour l'afficheur est maintenant prêt !

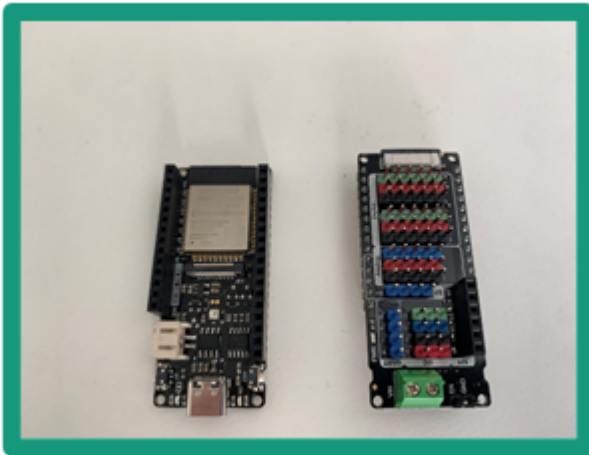
Jour 2 - Assemblage mécanique et électronique

Mission :

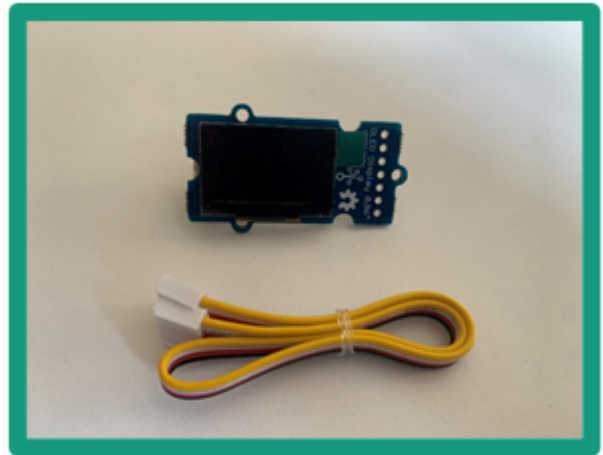
Aujourd'hui, votre objectif est d'assembler les capteurs de votre station AquaLife. Pour réaliser l'assemblage et le montage, vous allez d'abord découvrir les différents capteurs, les fixer à votre station et les câbler à la carte ESP32 en fonction des recommandations de connexion. Ci-dessous une image du montage final :



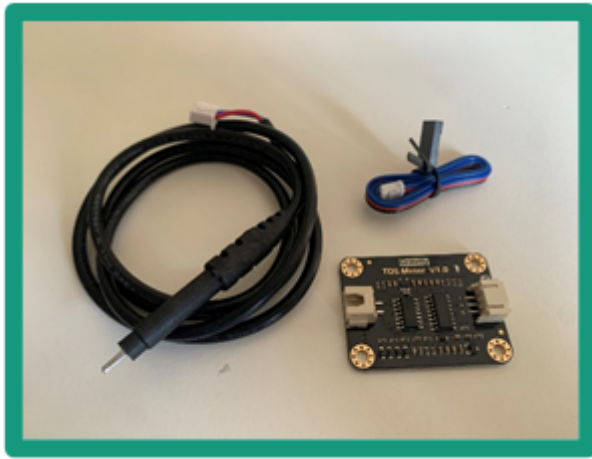
Tutoriel des capteurs :



L'**ESP32** (et son extension) est une carte microcontrôleur très performante qui permet de créer des projets électroniques connectés. Grâce à ses nombreux ports, tu peux y connecter des capteurs, des moteurs, ou d'autres composants pour réaliser des gadgets intelligents, des systèmes de domotique, ou des robots.



Un **afficheur** est un écran qui permet de montrer des informations ou des données provenant du microcontrôleur. On peut l'utiliser pour afficher des textes, des chiffres... C'est très utile pour voir ce que fait le microcontrôleur en temps réel, comme la température d'une pièce, le score d'un jeu, ou l'état d'un système.



Un **conductimètre** est un instrument qui mesure la capacité de l'eau à conduire l'électricité, ce qui nous informe sur la quantité de sels ou d'autres substances dissoutes dans l'eau. Plus l'eau contient de substances dissoutes, mieux elle conduit l'électricité. Cette mesure est très utile pour vérifier la pureté de l'eau.



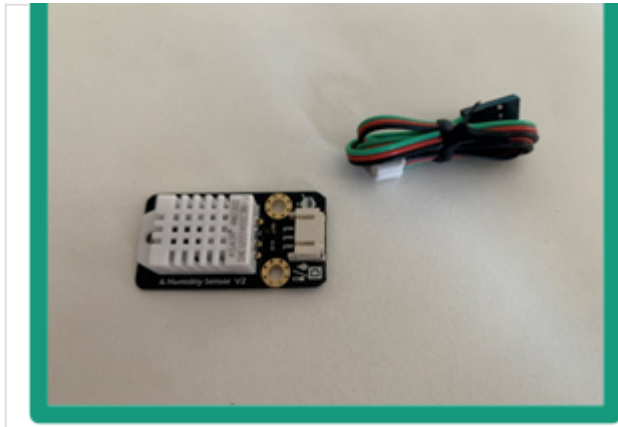
Un **turbidimètre**, permet de mesurer la turbidité, qui sert à évaluer la clarté de l'eau en détectant combien de petites particules sont suspendues dans celle-ci. Plus l'eau contient de particules comme de la terre, des micro-organismes, ou des substances chimiques, plus elle apparaît trouble.



Un **pH-mètre** est un instrument utilisé pour mesurer l'acidité ou la basicité d'une solution. Il donne une valeur appelée "pH", qui varie généralement de 0 à 14. Un pH de 7 est neutre, des valeurs inférieures à 7 indiquent une acidité, et des valeurs supérieures à 7 révèlent une basicité. Le pH-mètre est essentiel, car il aide à comprendre l'équilibre chimique des solutions.



Un **thermomètre** est un instrument qui permet de mesurer la température de l'eau. En mesurant la température, on peut s'assurer que l'eau est dans les conditions optimales, que ce soit pour assurer la santé des organismes aquatiques, garantir la sécurité des baigneurs, ou maintenir l'efficacité des processus industriels.



Le **DHT22** est un capteur qui mesure à la fois la **température et l'humidité de l'air** ambiant. Il fournit des informations précieuses pour contrôler les conditions environnementales. En mesurant l'humidité, le DHT22 aide à surveiller la quantité de vapeur d'eau dans l'air, ce qui est crucial pour maintenir une bonne qualité de l'air. Pour la température, il assure que les conditions soient confortables et sûres.

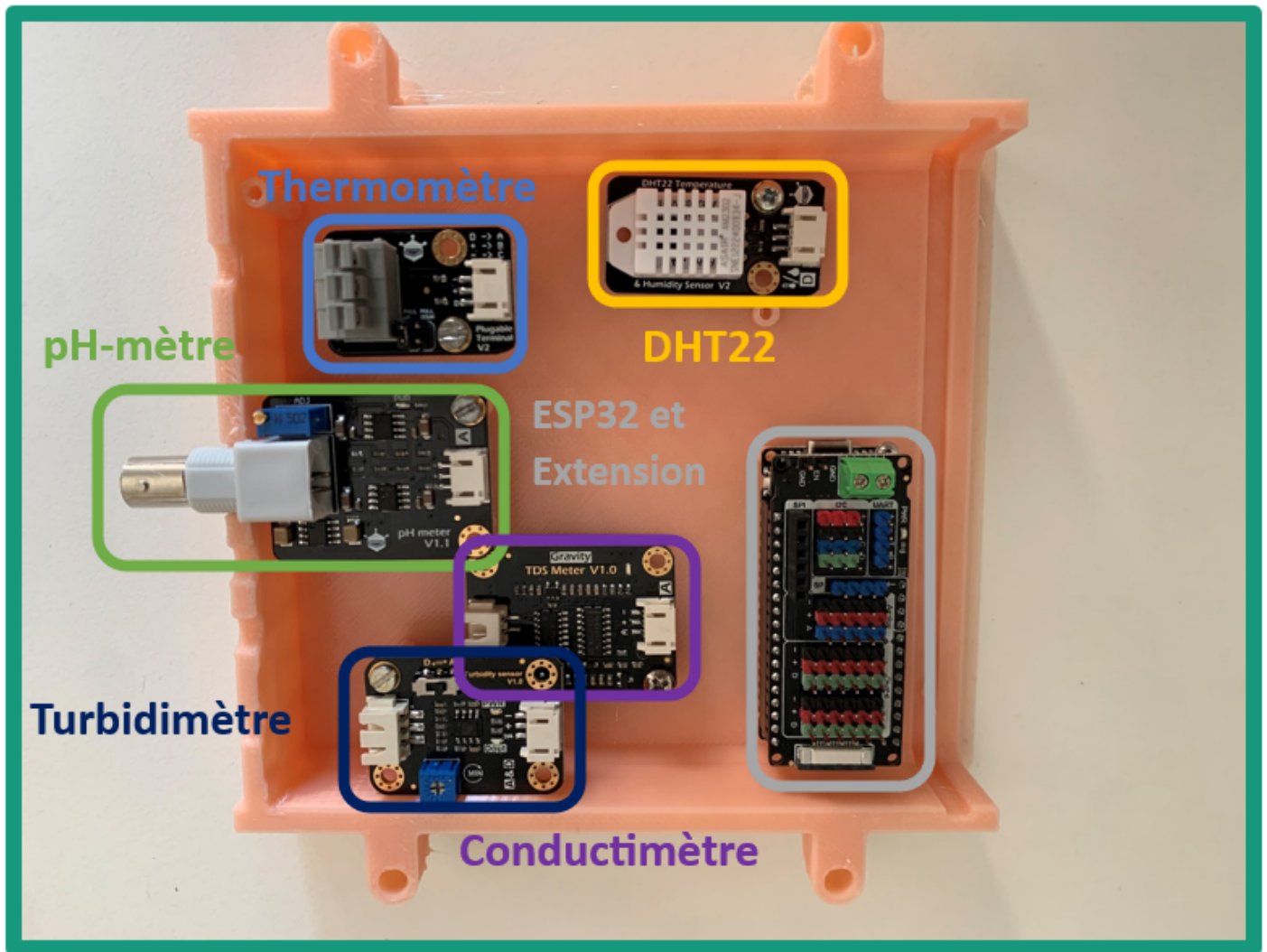
Avec ces informations vous êtes prêts à assembler votre station.

Étape 1 : Aller chercher les composants auprès de l'animateur

- Un afficheur, avec des câbles femelle-femelle de toutes les couleurs.
- Un conductimètre avec sa carte d'interface, ainsi qu'un câble correspondant.
- Un turbidimètre, avec sa carte d'interface, ainsi qu'un câble correspondant.
- Un pH-mètre, avec sa carte d'interface, ainsi qu'un câble correspondant.
- Un thermomètre, avec sa carte d'interface, ainsi qu'un câble correspondant.
- Un DHT22, avec sa carte d'interface, ainsi qu'un câble correspondant. Un ESP32 et son extension, ainsi qu'un câble USB/USB-C pour alimenter votre station à l'ordinateur.

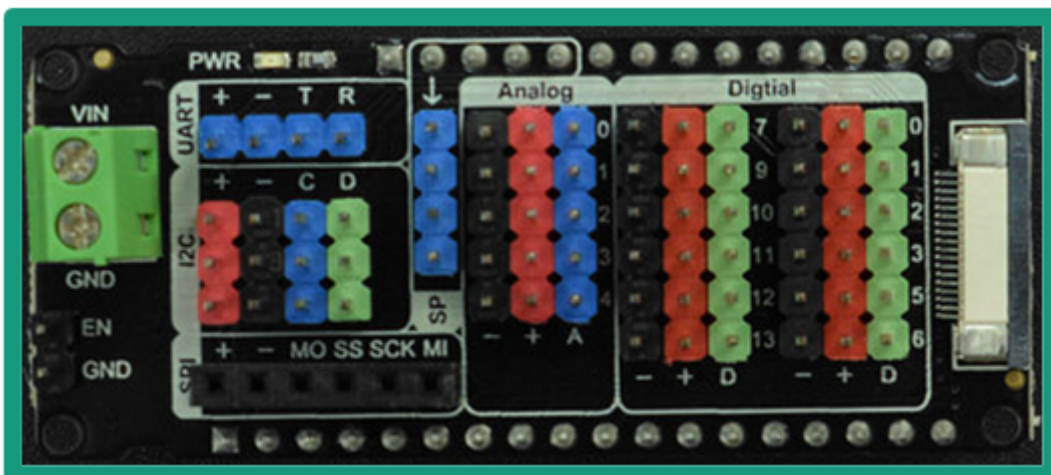
Étape 2 : Fixer les interfaces des capteurs à la station.

Fixer avec des vis adaptées, les interfaces des capteurs dans la boîte de la station comme indiqué ci-dessous.



Étape 3 : Brancher l'afficheur et les capteurs à l'ESP32 via la carte d'extension présentée ci-dessous.

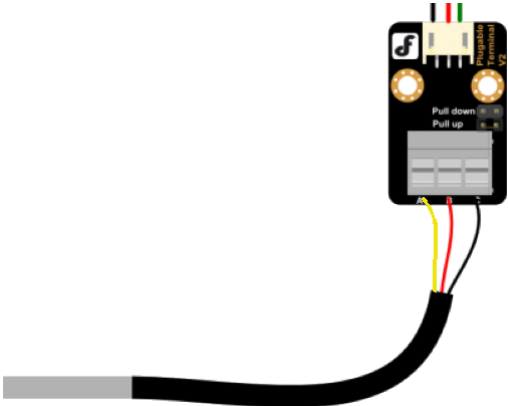
Observer l'organisation de la carte d'extension et repérer les différents signaux.



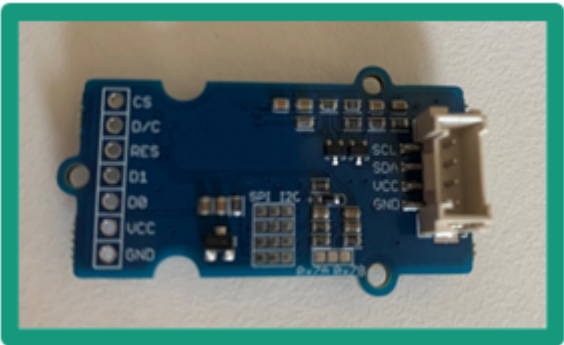
Connecter les capteurs et l'afficheur sur la carte d'extension DFR0762 selon les indications ci-dessous :

Capteurs	Entrée/Sortie	Consigne
Conductimètre	Analog pin 2	Respecter couleur fils
Turbidimètre	Analog pin 3	Respecter couleur fils
pH-mètre	Analog pin 4	Respecter couleur fils
Thermomètre	Digital pin 2	Respecter couleur fils
DHT22	Digital pin 6	Respecter couleur fils

Capteur Température	Entrée/Sortie
Jaune	A
Rouge	B
Noir	C



Afficheur	Entrée/Sortie I2C
SCL	C
SDA	D
VCC	+
GND	-



Jour 3 - Programmation Arduino

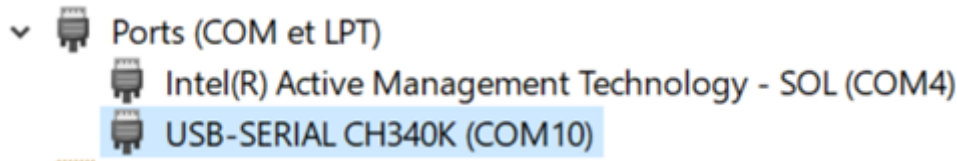
Mission :

Aujourd'hui, votre objectif est de programmer la station AquaLife. Vous allez écrire, modifier et tester le code nécessaire pour contrôler les différents capteurs selon les instructions décrites ci-dessous. Bonne chance.

Étape 1 : Lancement de l'IDE Arduino et configuration de l'application

- Ouvrez l'application Arduino IDE 
- Charger le fichier : **2024_stage_collegiens_v5.ino**

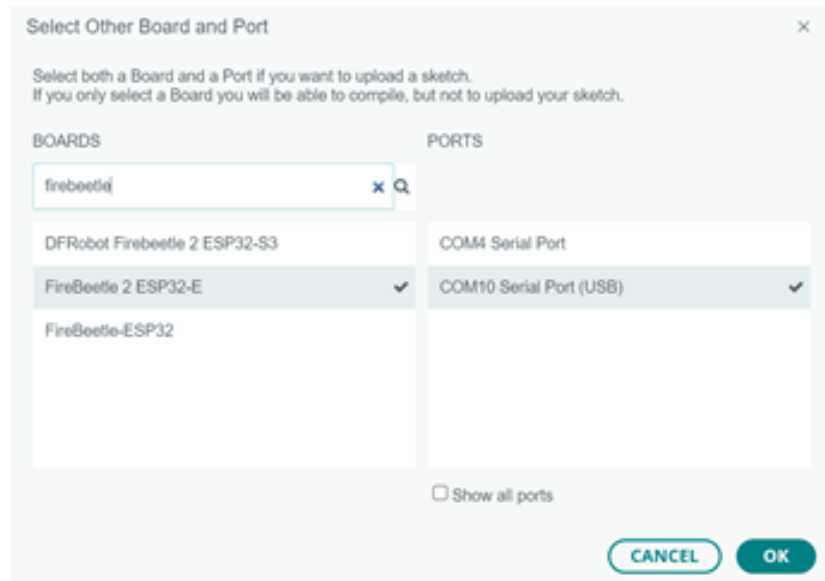
- Connectez votre station ESP32 à votre ordinateur via un câble USB-A/USB-C.
- Vérifiez le port de connexion de votre ESP32 (CH340K), en consultant le gestionnaire de périphériques de votre ordinateur pour identifier le port de communication (dans notre exemple ci-dessous : COM10).



- Sélectionnez ce port de communication dans l'environnement Arduino en utilisant le menu déroulant situé en haut à gauche, comme indiqué dans l'exemple (indiqué comme "Unknown COM10" dans cet exemple).



- Choisissez le modèle de carte approprié pour votre ESP32 dans les options de carte disponibles. Pour cet exemple, sélectionnez "**FireBeetle 2 ESP32-E**" (voir dans l'exemple ci-dessous).



Étape 2 : Assignment des pins des capteurs dans le code

- Dans le code, de la **ligne 13 à 17**, remplacez les marqueurs "**Ax**" et "**Dx**" par les numéros de pin correspondants à votre câblage sur la carte d'extension de l'ESP32. Par exemple, si le pin analogique pour le capteur de TDS est 9, remplacez "**TDS_Sensor_Pin_Num Ax**" par "**TDS_Sensor_Pin_Num A9**".

```
13 #define TDS_Sensor_Pin_Num Ax // TDS sensor on Analog pin Ax
14 #define Turbidity_Sensor_Pin_Num Ax // Turbidity sensor on Analog pin Ax
15 #define pH_Sensor_Pin_Num Ax // pH sensor on Analog pin Ax
16 #define DS18S20_Pin_Num_v2 Dx // DS18B20 sensor on digital pin Dx
17 #define DHT_Sensor_Pin_Num Dx // DHT on Digital pin Dx
```

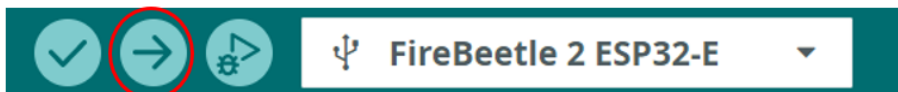
Étape 3 : Compiler et charger le code dans l'ESP32



- Un message d'erreur apparaît, notez les numéros des lignes de code mentionnés.

```
C:\Users\czerw\Desktop\2024_stage_collegiens_v4\2024_stage_collegiens_v4.ino: In function 'void get_pH_2()':
C:\Users\czerw\Desktop\2024_stage_collegiens_v4\2024_stage_collegiens_v4.ino:76:23: error: 'xxxxx' was not declared in this scope
    buf[i]=analogRead(xxxxx);
                      ^~~~~~
                      ~~~~~~
```

- Remplacez les chaînes de caractères **"xxxxx"**, **"yyyyy"** et **"zzzzz"** mentionnées dans les lignes d'erreur par l'identifiant du capteur spécifié. Ces identifiants, comme **"TDS_Sensor_Pin_Num"** pour le turbidimètre, sont indiqués en orange dans les **lignes de code de 13 à 17**.
- Assurez-vous que le capteur concerné par l'erreur est correctement décrit dans une fonction dont le nom est en adéquation avec ce capteur, située dans les lignes précédant l'erreur.
- Recompilez le code pour vérifier que les erreurs ont été corrigées.
Une fois le code compilé sans erreurs, chargez-le dans l'ESP32 en cliquant de nouveau sur le bouton encadré en rouge.



Étape 4 : Correction de l'affichage pour une lecture claire des informations des capteurs

- Modifiez la durée de pause entre chaque séquence d'affichage, qui se trouve à la **ligne 11**. Changez la valeur actuelle de 0,1 seconde à 2 secondes. Notez que cette valeur doit être saisie en millisecondes dans le code.
- Le code pour la première séquence d'affichage est situé entre les **lignes 275 et 291**. Vous remarquerez que les lignes de texte se superposent, rendant difficile la lecture des informations.
- Corrigez ce problème en ajustant le deuxième paramètre (x) dans la fonction d'affichage **u8g2.drawStr(x, y, "texte");** pour bien séparer les informations relatives aux capteurs "Humidity DHT", "Temp DHT", et "pH" sur l'afficheur.

```

275 //*****//
276 // écran OLED SSD1315 :
277 // fonction U8g2.drawStr(n°colonne,n°ligne,string)
278
279 u8g2.clearBuffer();           // clear the internal memory
280 u8g2.drawStr(0,10,"Station eau"); // write something to the internal memory
281 u8g2.drawStr(0,25,"Humidity DHT :"); // affichage du texte "Humidity DHT:"
282 u8g2.drawStr(92,25,Hum_DHT); // affichage de la valeur de l'humidité
283 u8g2.drawStr(0,30,"Temp DHT :"); // affichage du texte "Temp DHT :"
284 u8g2.drawStr(68,30,Temp_DHT); // affichage de la valeur de température fournie par le capteur DHT
285 u8g2.drawStr(0,35,"pH :"); // affichage du texte "pH :"
286 u8g2.drawStr(20,35,pH_2); // affichage de la valeur du pH
287 u8g2.sendBuffer(); // transfer internal memory to the display
288
289 delay(pause_affichage_ms);
290
291 //*****//

```

Étape 5 : Activation d'un message d'alerte sur l'afficheur

- Identifiez dans le code les **lignes 309 à 328** qui permettront d'afficher un message d'alerte en lien avec le capteur de turbidité.
- Actuellement, ces lignes sont en commentaire et ne sont pas prises en compte lors de la compilation. Pour les activer, supprimez les marqueurs de commentaire `/*` et `*/` qui se trouvent respectivement aux **lignes 315 et 328**.
- Relevez la valeur de turbidité affichée sur votre appareil et définissez un seuil inférieur à cette valeur. Modifiez la valeur dans la déclaration (**`float seuil = x;`**) pour y placer votre seuil (supérieur à 0 et inférieur à votre valeur d'affichage).
- Compilez et téléchargez à nouveau le programme dans votre ESP32.
- Testez le fonctionnement du message d'alerte en plaçant une feuille de papier devant le turbidimètre pour simuler une turbidité élevée.

Jour 4 - Tests

Mission :

Aujourd'hui, votre objectif est de tester les différents échantillons d'eau qui nous sont parvenus de trois pays différents avec votre station AquaLife. Vous avez un échantillon d'un cours d'eau de l'**A**ustralie (noté **A**), la **B**résil (noté **B**), le **C**anada (noté **C**) et un cours d'eau en Alsace qui sera le **T**émoin (noté **T**), faites les analyses et compléter le tableau proposé dans le document.

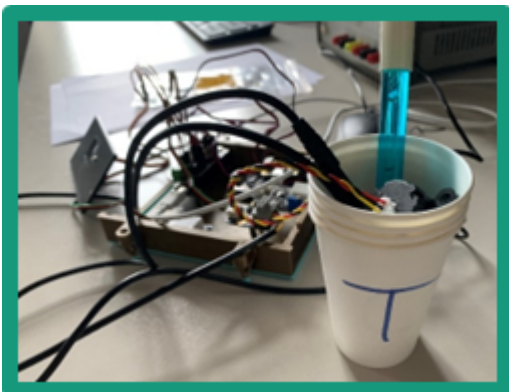
Étape 1 : Collecte et préparation des échantillons d'eau

- Notez et remplissez à moitié quatre gobelets avec des échantillons d'eau provenant de différents pays : **A**ustralie (noté **A**), **B**résil (noté **B**), **C**anada (noté **C**) et le cours d'eau en Alsace qui servira de **T**émoin (noté **T**).
- Avant de verser l'eau dans les gobelets, agitez bien les bouteilles pour homogénéiser les échantillons.
- Utilisez un cinquième gobelet pour rincer vos capteurs après chaque analyse. Remplissez ce gobelet avec de l'eau du robinet. Assurez-vous de changer l'eau de rinçage dans ce gobelet entre chaque type d'eau analysé (A, B, C, et T).



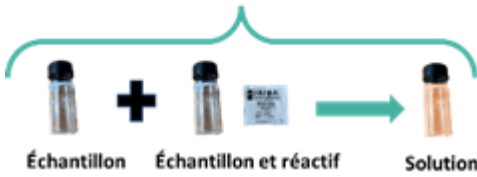
Étape 2 : Analyses physique des échantillons d'eau A, B, C et T

- Pour analyser un échantillon, par exemple l'échantillon A, plongez les capteurs dans le gobelet correspondant. Homogénéisez la solution en remuant doucement le gobelet ou en déplaçant lentement les capteurs dans l'eau.
- Notez les résultats de l'analyse dans le tableau prévu à cet effet (Voir 4ème partie : Compléter le tableau avec vos analyses).
- Après chaque analyse, rincez les capteurs dans le gobelet de rinçage. Répétez la mesure jusqu'à obtenir trois résultats pour chaque capteur et pour chaque échantillon. Ce processus, appelé "triplicatas", est expliqué plus en détail un peu plus loin.
- Pensez à changer l'eau du gobelet de rinçage entre chaque type d'eau analysé (A, B, C et T).



Étape 3 : Analyses chimiques des échantillons d'eau A, B, C et T

- Réalisez l'analyse chimique du Nitrate, à l'aide du photomètre. C'est un appareil qui permet de mesurer la concentration en polluants. Celui-ci mesure la différence entre l'échantillon et la solution (Echantillon avec le réactif), grâce au changement de couleur.
- Pour cela, suivez le protocole illustré dans le schéma ci-dessous.
- Notez les résultats de l'analyse dans le tableau prévu à cet effet un peu plus loin.



Protocole du Photomètre :

1. Prélèvement : À l'aide d'une pipette jaugée, prélevez 10 ml de l'échantillon à analyser.
2. Transfert : Versez l'eau de la pipette jaugée dans une cuvette propre.
3. Calibration : Réalisez un étalonnage en faisant le blanc (zéro) pour ajuster le photomètre avant l'analyse proprement dite.
4. Ajout du réactif : Introduisez le réactif approprié dans l'échantillon.
5. Mélange : Agitez doucement la solution pour bien mélanger le réactif avec l'échantillon.
6. Mesure : Placez la cuvette dans le photomètre et lancez la mesure en appuyant sur le bouton "start".

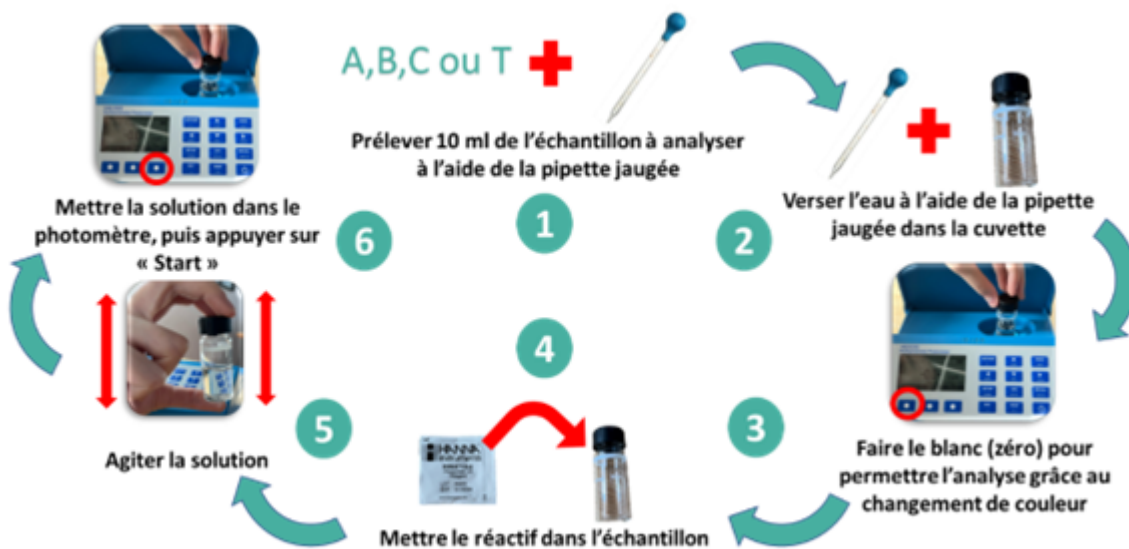


Schéma d'utilisation du photomètre

Étape 4 : Compléter le tableau avec vos analyses

Echantillon	Témoin (T)	Australie (A)	Brésil (B)	Canada (C)
-------------	------------	---------------	------------	------------

pH				
Turbidité				
Conductivité				
Température				
Nitrate				

Tableau des analyses des échantillons des différents pays



Les chercheurs font aussi des analyses en tripliquas, pourquoi ?

Lorsqu'on réalise des analyses en tripliquas, cela signifie qu'on fait la même expérience trois fois. Cela peut sembler répétitif, mais c'est très important, surtout dans la recherche scientifique.

Confirmer les résultats : Imagine que tu essaies une nouvelle recette de cookies avec tes amis. Si tu ne fais qu'un seul cookie et qu'il est parfait, c'est génial, mais tu ne sais pas si c'est parce que tu as suivi la recette à la lettre ou si tu as juste eu de la chance. Si tu fais trois fournées de cookies et qu'elles sont toutes délicieuses, alors tu peux être sûr que la recette est bonne.

Trouver les erreurs : Parfois, une erreur peut se glisser dans une expérience. Si tu ne fais l'expérience qu'une seule fois et qu'il y a une erreur, tu ne le sauras jamais. Mais si tu fais l'expérience trois fois et que les résultats de l'une sont différents des autres, cela peut te signaler qu'il y a eu un problème quelque part. Cela aide à être plus précis.

Moyenne des résultats : Lorsqu'on fait une expérience plusieurs fois, on peut calculer la moyenne des résultats. Cela permet d'avoir une idée plus claire de ce qui se passe réellement. Par exemple, si un jour il fait 25°C, un autre 28°C et un autre 26°C, la température moyenne te donne une idée plus stable du temps qu'il fait réellement.

En faisant des analyses en tripliquas, les scientifiques s'assurent que leurs découvertes sont solides et fiables. C'est comme vérifier ton travail en mathématiques ; plus tu vérifies, plus tu es sûr de ton résultat !

Jour 5 - Analyse et Présentation

Mission :

Aujourd'hui, votre objectif est de traiter les données des différents échantillons prélevés dans les cours d'eau de l'**A**ustralie (noté **A**), la **B**résil (noté **B**), le **C**anada (noté **C**) et le **T**émoin Alsace (noté **T**). Pour cela, vous devez traiter vos données à l'aide d'un fichier Excel, interpréter vos résultats pour apporter vos conclusions et enfin compléter un PowerPoint pour présenter vos résultats.

Étape 1 : Traitement des données à l'aide d'un fichier Excel

- Complétez le tableau Excel à votre disposition avec les données récoltées la veille par votre station AquaLife.
- Calculer la moyenne de vos tripliquas en utilisant la formule entre crochet [=MOYENNE(Cellule1; Cellule2; Cellule3)]

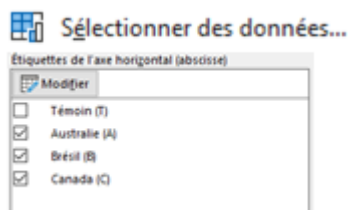
Rappel : une Cellule = Colonne + Ligne

Exemple : Cellule1 pour le pH du Témoin est B2 qui contient la

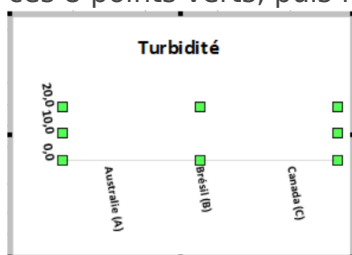
	A	B
1	Echantillon	Témoin (T)
2		7,6
3		7,3
4		7,2
5	Moyenne pH	=MOYENNE(B2; B3; B4)
6		

valeur 7,6 dans l'exemple.

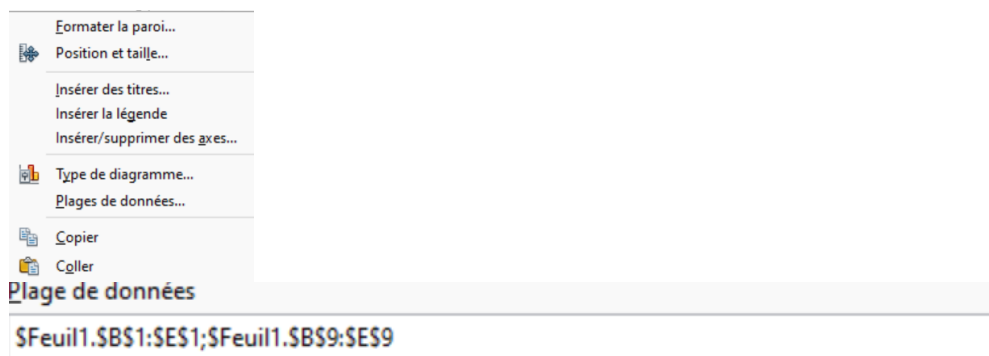
- Les graphiques pour le pH, la conductivité, la turbidité et la concentration en Nitrate apparaissent. Attention plusieurs erreurs se sont glissées à l'intérieur et il faut les corriger. Le graphique pour le pH est correct. Utilisez ce format comme modèle pour corriger les autres graphiques, en suivant les instructions ci-dessous :
 - o Pour le graphique de la turbidité, il manque l'échantillon Témoin. Pour l'ajouter, faites un clic droit sur le graphique et allez dans "**Sélectionner les données...**" et cochez la case correspondant au Témoin (T).



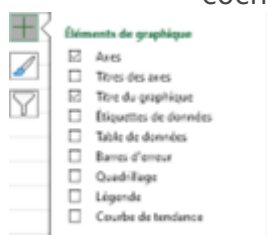
- **ATTENTION** si vous êtes sur open office cliquer sur le graphique jusqu'à ce que vous ayez ces 8 points verts, puis faites un clic droit.



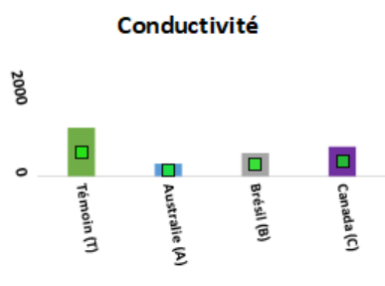
- Cliquez sur "plage de données" et remplacer chacun des deux "C" par un "B" .



- o Pour le graphique de la conductivité, il manque les étiquettes des valeurs de vos échantillons. Pour les ajouter, faites un clic gauche sur le graphique et allez dans "**Eléments de graphique**" en haut à droite du graphique pour cocher la case "**étiquette de données**" pour les ajouter.




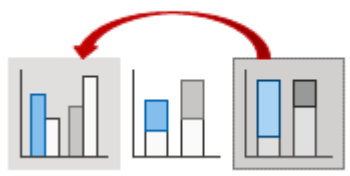
- **ATTENTION** si vous êtes sur open office cliquer sur le graphique jusqu'à ce que vous ayez ces 4 points verts, puis faites un clic droit.



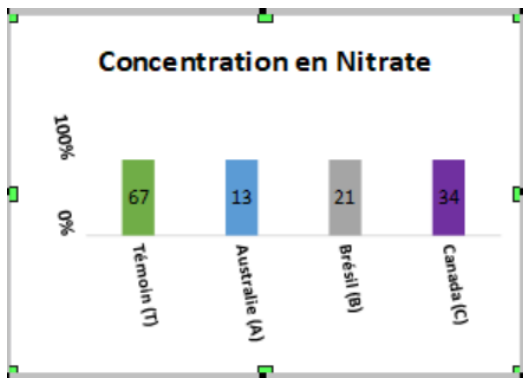
- Cliquez sur "Insérer des étiquettes de données".

- o Le graphique de la concentration en Nitrates est au format pourcentage (%), ce qui ne permet pas de distinguer les différences entre les échantillons. Faites un clic droit sur le graphique pour aller dans "**Modifier le type de graphique**" et choisissez "**Histogramme groupé**".

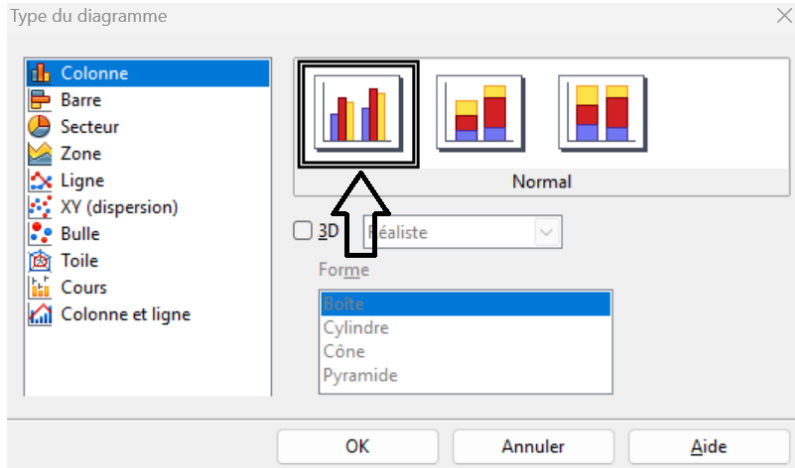
 **Modifier le type de graphique...**



- **ATTENTION** si vous êtes sur open office cliquer sur le graphique jusqu'à ce que vous ayez ces 8 points verts, puis faites un clic droit.



- Cliquez sur "Type de diagramme" et sélectionner le premier qui se nomme "Normal".



Étape 2 : Interprétation et conclusion

- Compléter le tableau ci-dessous, pour cela ajouter des - / + /++ dans chaque ligne en fonction des informations présentées par la suite sur le pH, la turbidité, la conductivité et la concentration en nitrate de vos échantillons.

Allez lire l'Exemple dans la section pH en bas de la page.

- Faites le **Bilan** en comptant le nombre de + pour chaque échantillon. Celui qui en a le plus est probablement le cours d'eau le plus contaminé.

Attention, il n'y a pas de bonnes ou mauvaises réponses, c'est à vous d'interpréter vos résultats !

Echantillon	Australie (A)	Brésil (B)	Canada (C)
pH			
Turbidité			
Conductivité			
Nitrate			
Bilan			

- **Le pH**, c'est comme une échelle qui nous dit si l'eau est acide, neutre ou basique. Pour les rivières et les lacs, le mieux, c'est que leur pH soit entre 6,5 et 8,5. C'est comme être

juste au milieu, ni trop acide, ni trop basique, ce qui est parfait pour les plantes et les animaux qui vivent dans l'eau.

- o **Si le pH est en dessous de 6,5**, l'eau est trop acide, ça peut être difficile pour les poissons et d'autres créatures aquatiques de vivre, grandir et se reproduire normalement. Cela peut arriver à cause de la pollution de l'air qui tombe avec la pluie.
- o **Si le pH est au-dessus de 8,5**, l'eau est trop basique, ça peut aussi causer des problèmes aux poissons et à leur environnement, comme irriter leur peau ou changer la façon dont certains nutriments et toxines se comportent dans l'eau. Cela peut être dû à des rejets d'usines.

Exemple : Si j'ai un pH compris entre 6,5 et 8,5 alors je vais noter **-** dans mon tableau ; en revanche, si j'ai un pH compris entre 8,5 et 10, alors je vais noter **+**, car celui-ci s'éloigne un peu de la normal, mais si j'ai un pH > 10 alors **++**, car celui-ci s'éloigne fortement de la normal.

Enfin, si j'ai un pH compris entre 5 et 6,5, alors je vais noter **+**, car celui-ci s'éloigne un peu de la normal, mais si j'ai un pH < 5 alors **++**, car celui-ci s'éloigne fortement de la normal.

- **La turbidité**, correspond au trouble présent dans l'eau.
 - o **Si la turbidité est au-dessus du niveau normal (Témoin)**, cela signifie généralement que l'eau est très claire. Cela peut être bon pour la pénétration de la lumière et aide à maintenir un écosystème sain pour la photosynthèse des plantes aquatiques.
 - o **Si la turbidité est en dessous du niveau normal (Témoin)**, l'eau devient plus trouble. Cela peut être causé par des activités humaines comme le défrichement des terres ou la construction, qui déplacent beaucoup de terre. Une eau trouble peut créer plusieurs problèmes pour les animaux aquatiques comme des difficultés respiratoires (Les particules fines dans l'eau peuvent bloquer ou endommager les branchies des poissons, rendant difficile pour eux de respirer).
- **La conductivité**, est une mesure qui nous dit combien l'eau peut conduire l'électricité. Pour les rivières et les lacs, une conductivité normale peut varier entre 50 et 1500 microsiemens par centimètre ($\mu\text{S}/\text{cm}$).
 - o **Si la conductivité est en dessous du niveau normal (50-1500)**, cela peut signifier que l'eau est très pure. Bien que cela puisse sembler positif, une faible conductivité peut parfois limiter la disponibilité de certains nutriments essentiels pour la vie aquatique, affectant ainsi les organismes qui dépendent de ces nutriments.
 - o **Si la conductivité est au-dessus du niveau normal (50-1500)**, cela peut être dû à des activités humaines telles que l'agriculture, où des fertilisants riches en minéraux sont utilisés, ou des effluents industriels. Une conductivité élevée peut poser des problèmes pour les espèces aquatiques.

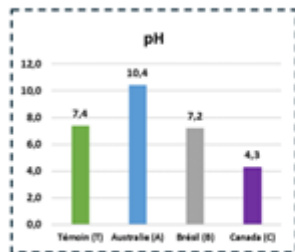
- **La concentration en nitrate**, est un indicateur important de la santé d'un cours d'eau, car les nitrates sont une forme de nutriment essentiel pour la croissance des plantes, mais peuvent être problématiques en grande quantité. Normalement, une concentration en nitrate dans un cours d'eau devrait être inférieure à 1000 microgramme par litre (µg/L) pour être considérée comme sûre pour la plupart des écosystèmes aquatiques.
 - o **Si la concentration en nitrate est en dessous du niveau normal**, cela peut signifier qu'il y a peu de sources de pollution par les nitrates à proximité.
 - o **Si la concentration en nitrate est au-dessus du niveau normal**, cela peut être un signe de pollution comme les déchets agricoles ou les eaux usées non traitées. Des niveaux élevés de nitrates peuvent entraîner plusieurs problèmes écologiques, notamment des effets sur la santé des organismes aquatiques et **risque pour la santé humaine**.

Étape 3 : Mise en forme des résultats sur PowerPoint

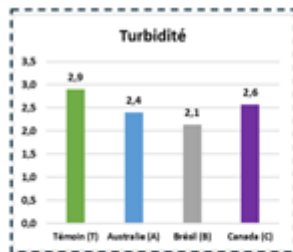
- Vous devez compléter la présentation PowerPoint en ajoutant vos graphiques dans les emplacements prévus : **pH, turbidité, conductivité et concentration en nitrate**. Pour cela faite un copier-coller de vos graphiques que vous avez créé de votre fichier Excel. Attention pour garder le bon code couleur, vous devez faire copier **Ctrl+C** et un collage spécial image avec le raccourci **Ctrl+Alt+V**.
- Pour finir, remplacer les **XXX** par le nom du Pays qui a le cours d'eau le plus pollué d'après vos résultats.

Félicitation vous avez réussi votre mission et identifié le cours d'eau le plus pollué parmi les échantillons d'eau !!

Vous devez avoir obtenu une présentation qui ressemble à celle ci-dessous :



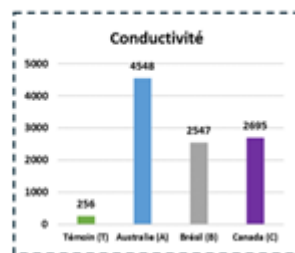
Comparatif du pH



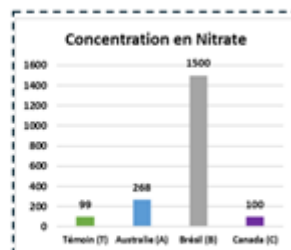
Comparatif de la turbidité



La source d'eau la plus contaminée
par l'usine est celle du XXX.



Comparatif de la conductivité



Comparatif de
la concentration Nitrates



Nouvelle correction : programmeEauCompleto.ino

Ancien code : 2024_stage_collegiens_v5_Station_Eau.ino

Revision #40

Created 22 April 2024 13:36:55 by admin_idf

Updated 2 July 2025 07:50:21 by admin_idf