



L'IOT POUR 5 ANS ET

APPRENDRE À
CODER AVEC
MICRO:BIT



SOMMAIRE

[La carte micro:bit](#)

[Usages](#)

[Exercices](#)

[Conclusion](#)

B B C



micro:bit



Jonathan BARANZINI

Devéloppeur





Thomas CAMI

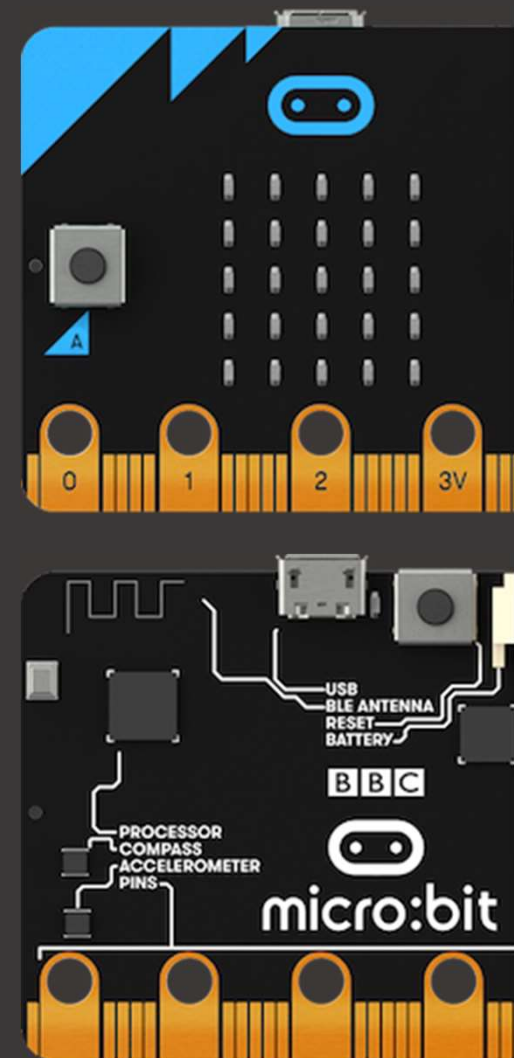
Devéloppeur

A vintage-style silver microphone with a black grille is mounted on a black tripod. The microphone is positioned on the right side of the frame, standing on a mossy log in a dense, sun-dappled jungle. The background is filled with lush green foliage, including large ferns and thick tree trunks, with sunlight filtering through the canopy. A bright green rectangular overlay is located in the bottom-left corner, containing the text 'CARTE MICRO:BIT' in white capital letters.

CARTE MICRO:BIT

LA CARTE MICRO:BIT

	06 / 07 / 2015
nt	BBC (British Broadcasting Corporation), Micro:bit Educational Foundation
	Former les jeunes au développement informatique
ogie	<ul style="list-style-type: none"> • Microcontrôleur 16Mhz, 32 bits • 256ko mémoire flash + 16ko mémoire vive • Matrice led 5 x 5
s	<ul style="list-style-type: none"> • 2 boutons + 1 bouton sensible • 1 thermomètre, 1 boussole • 1 capteur de mouvement 3D • 1 micro + haut-parleur • Ports GPIO (x20) + bluetooth
e d'exploitation	Zephyr OS
es	Bloc, Javascript, Python
ation	<ul style="list-style-type: none"> • USB 5V • Piles (3V)
ires	<ul style="list-style-type: none"> • Lumière : bande led / matrice led, feu tricolore • Moteur : servo-moteur, barrière • Feu tricolore • Capteurs : ultrasons...



A vintage, silver-colored microphone with a ribbed grille is mounted on a black tripod. The microphone is positioned on the right side of the frame, standing on a mossy log. The background is a dense, lush jungle with various green plants, ferns, and tree trunks. Sunlight filters through the canopy, creating a hazy, atmospheric effect. In the bottom left corner, there is a bright green rectangular overlay containing the word "AGES" in white capital letters.

AGES

AGES

ACQUÉRIR LE DÉVELOPPEMENT

Langage block
éophyte

Interface visuelle

Accessible aux
débutants

Documentation



APPRENDRE UN LANGAGE INFORMATIQUE

-  Python
Javascript
Typescript
-  Strict minimum

SE PERFECTIONNER EN ALGO

-  Environnement minimal
-  Faibles capacités (CPU + mémoire)
-  Entretien d'embauche

EXPÉRIMENTER SES IDÉES

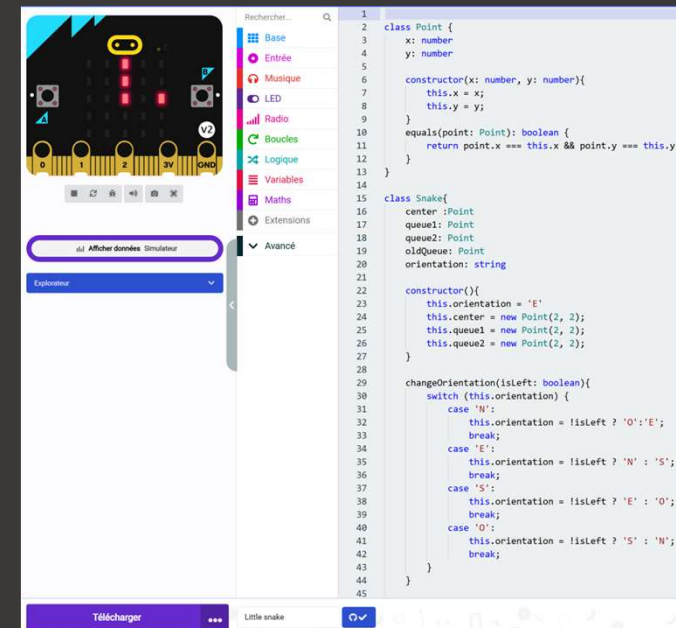
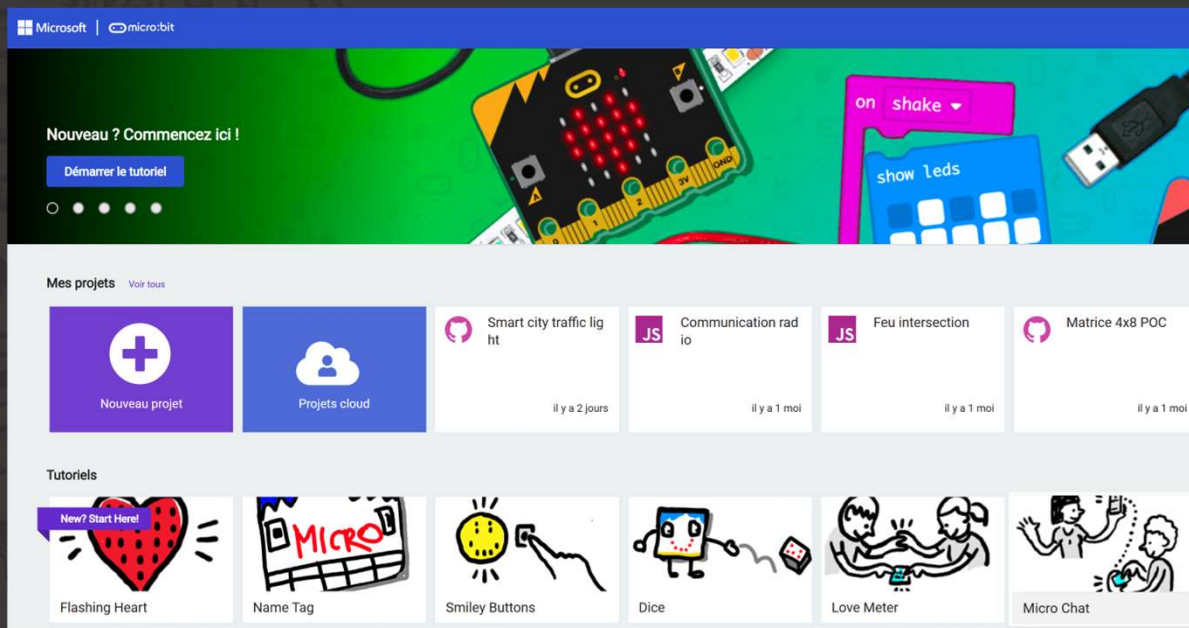
-  Nombreux composants
-  Ports GPIO
-  Pas cher
-  Librairies disponibles

A vintage, silver-colored microphone with a ribbed grille is mounted on a black tripod. The microphone is positioned on the right side of the frame, standing on a mossy log. The background is a dense, lush jungle with various green plants, ferns, and tree trunks. Sunlight filters through the canopy, creating a hazy, atmospheric effect. In the bottom left corner, there is a bright green rectangular overlay containing the text 'RCICES' in white.

RCICES

PLATEFORME DE DÉVELOPPEMENT

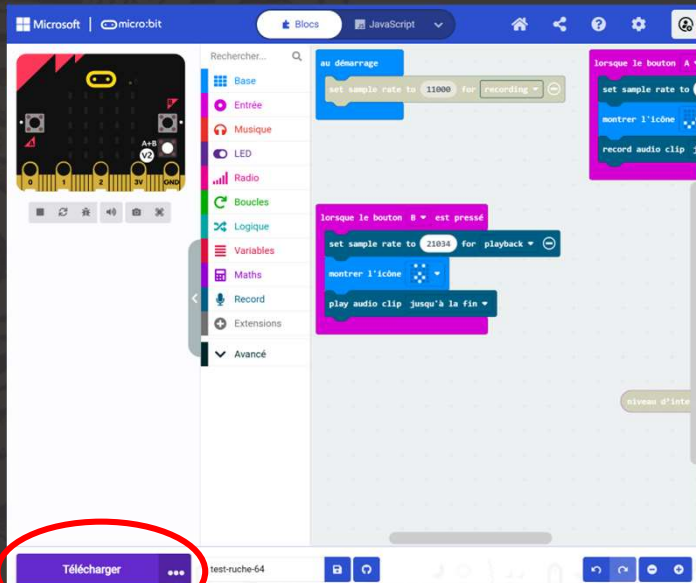
- <https://makecode.microbit.org>
- Compte GitHub pour sauvegarder son code
- Simulateur de carte affiché pour tester le code
- Envoi du code sur la carte en USB
- Passage du bloc au javascript ou python
- Librairie de fonctions et auto-complétion
- Intégration de bibliothèques externes
- Création d'une librairie facilitée



DÉPLOYER SUR LA CARTE

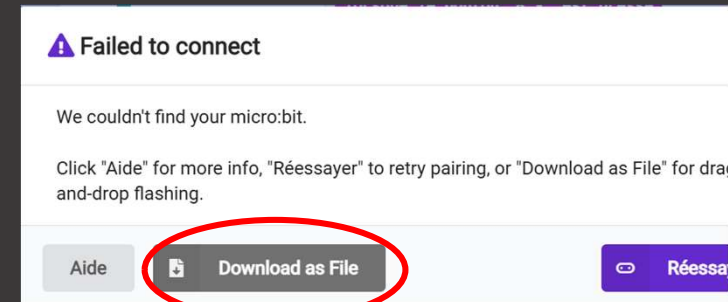
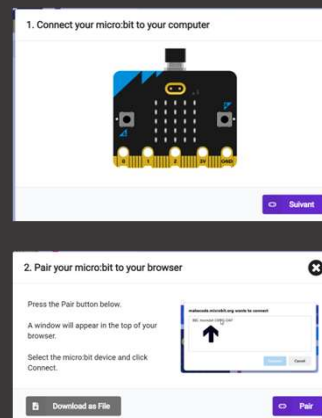
Windows

- Cliquer sur Télécharger
- Suivre les étapes
- Carte reconnue lors de l'appairage
- La carte est à jour



Linux

- Carte non reconnue par la plateforme
- Télécharger le binaire
- Le déplacer directement sur la carte ()
- La carte redémarre et exécute le nouveau



PROGRAMME

<https://jotitan.github.io/microbit-devox-2025>

Découvrir la programmation 

Démo : faire des katas 

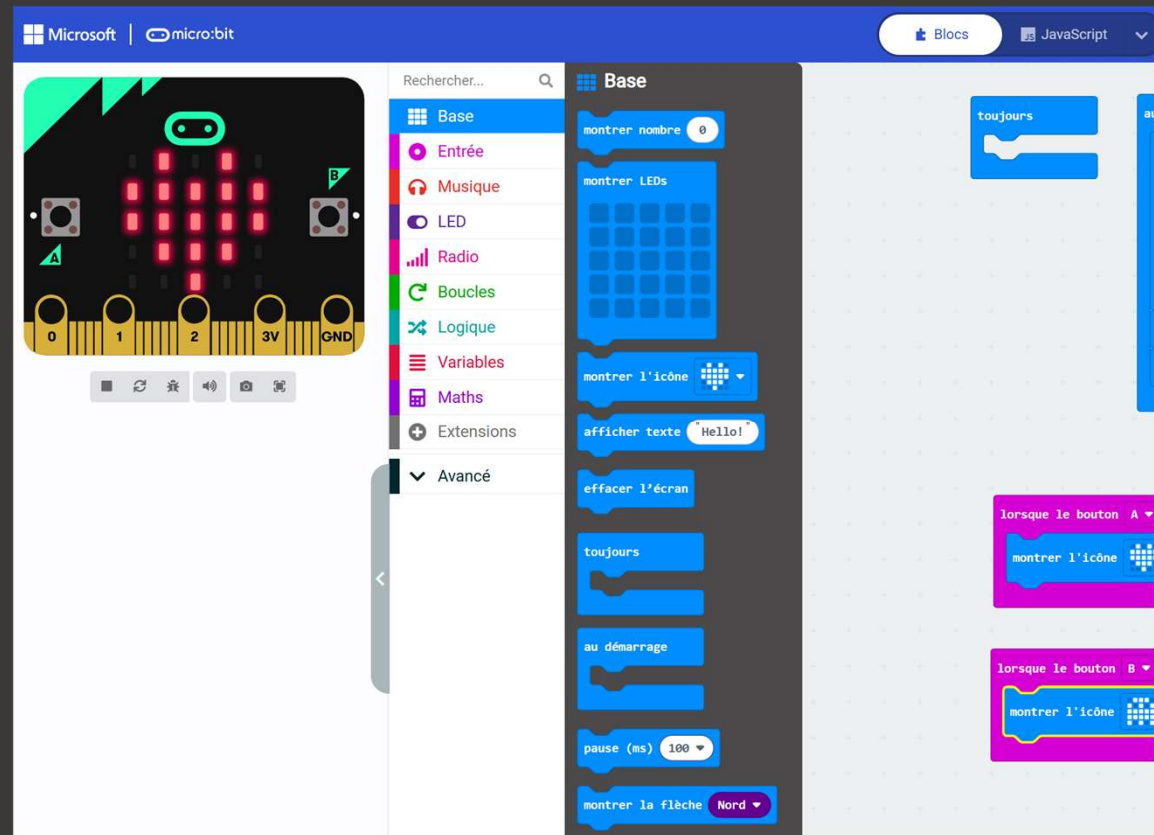
La bille qui roule 

Icônes par radio 


Tu chauffes tu brules  

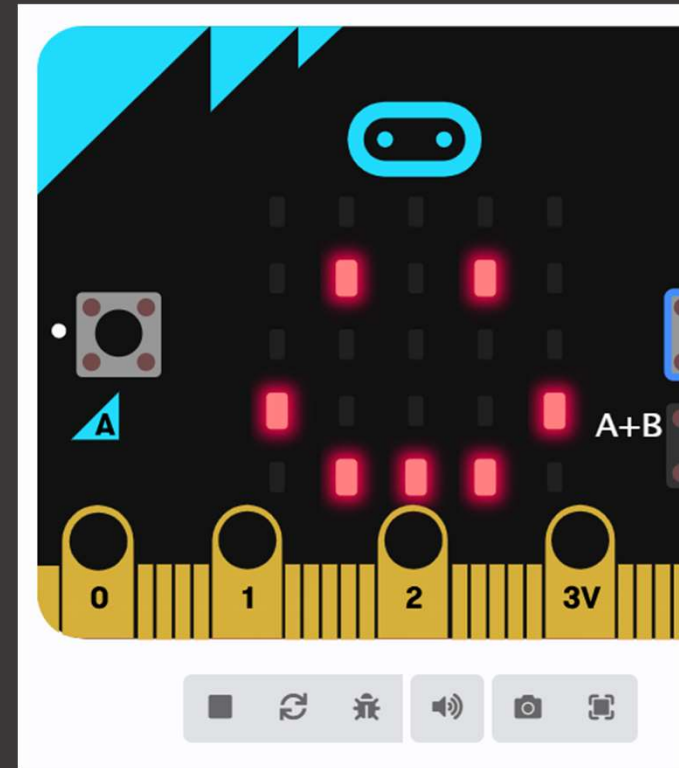
DÉCOUVRIR LA PROGRAMMATION

- **Langage** : bloc
- **Objectif** : manipuler du bloc comme un enfant
- **Exercice** :
 - Au démarrage, afficher une icône de cœur
 - Quand on appuie sur le bouton A, incrémenter un compteur et l'afficher
 - Quand on appuie sur le bouton B, réduire ce compteur et l'afficher
 - Quand on appuie sur les deux boutons, remettre le compteur à 0 et l'afficher



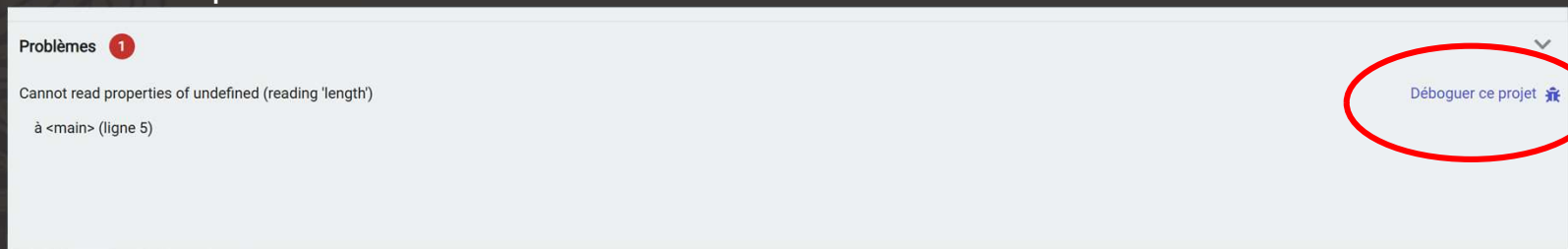
KATA D'ALGORITHMIE / TEST ENTRETIEN

- **Langage** : Javascript
- **Lien** :  <https://github.com/jotitan/microbit-tests-kata>
- **Objectif** : kata court pour s'améliorer en algorithmie
- **Contexte** :
 - La carte micro:bit dispose d'une matrice Led 5x5 permettant de facilement visualiser un tableau
- **Démarrage** :
 - Librairie Github à importer (implémente le bouton A et B)
 - Une fonction à implémenter *loadImplementations*
- **Exercice** :
 - Lire le readme 😊
- **Lancement**
 - Le bouton A affiche une nouvelle entrée
 - Le bouton B lance le test en retournant, si ça marche 😊 + résultat, sinon ☹

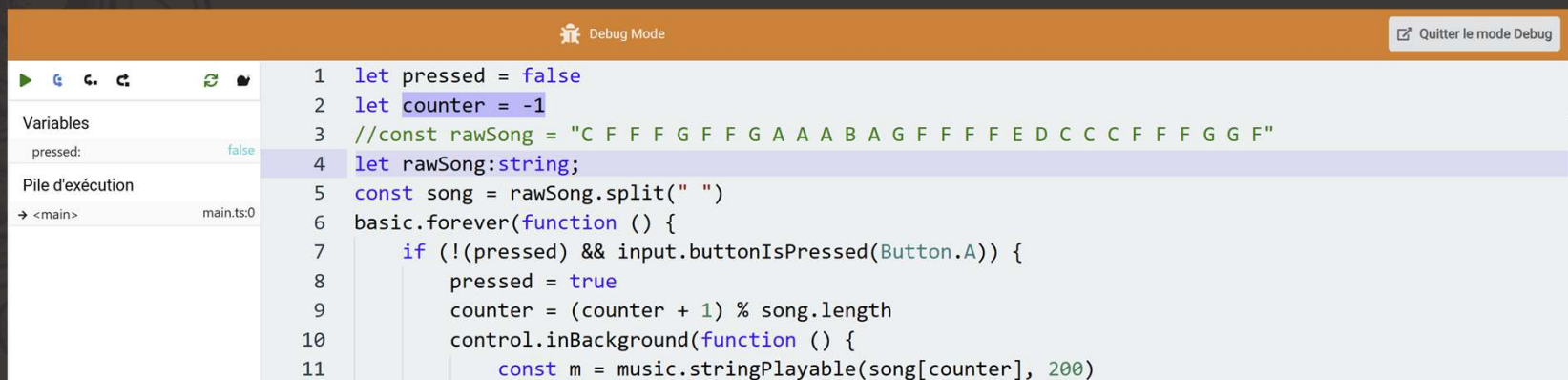


DEBUGGER SUR MICRO:BIT


- Le Classique : F12 et console.log(message);
- Mode débog :
 - Quand ca plante



- L'option « déboguer ce projet apparaît »
- Mode debug normal : points d'arrêt, pile d'exécution et variables



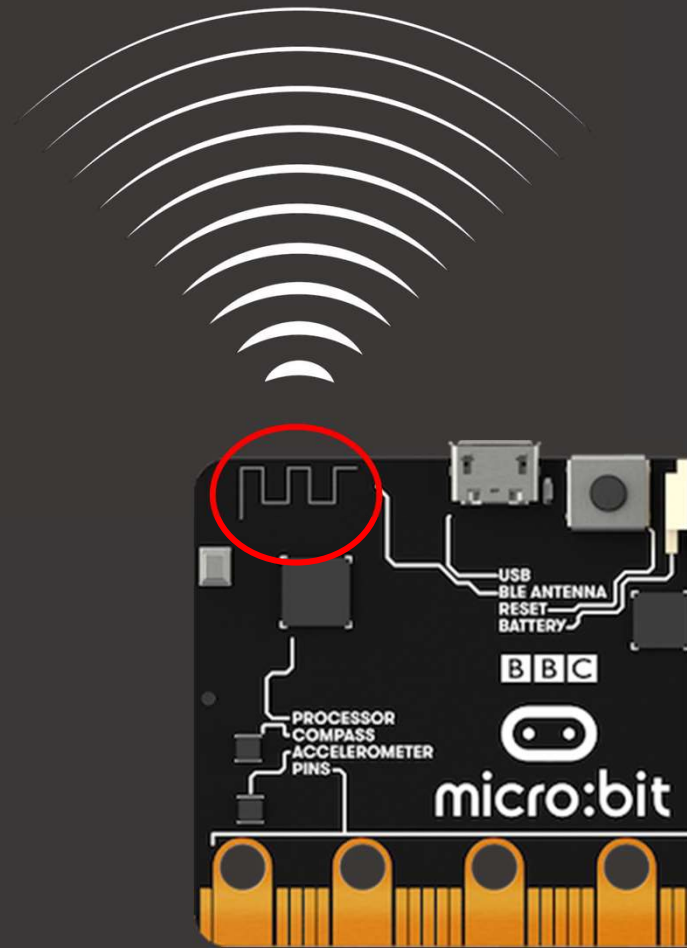
IOT : LA BILLE QUI ROULE

- **Langage** : Javascript
- **Objectif** : Simuler une bille sur un plateau qui bouge
- **Lien** :  <https://github.com/jotitan/microbit-rolling-ball>
- **Exercice** :
 - Utiliser la détection de l'orientation de la carte
 - Etablir




RADIO SUR MICRO:BIT

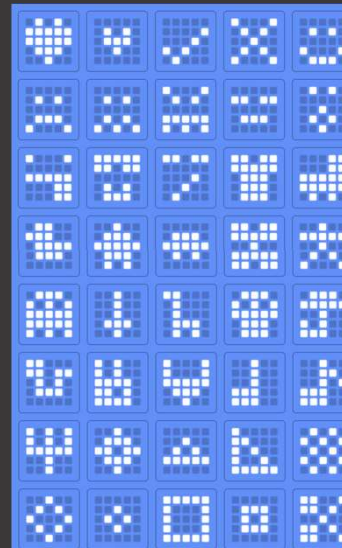
- Fonctionne en radio et bluetooth
- ⚠ Attention aux interférences
- Fréquence
 - Bande de fréquence : entre 2.4Ghz et 2.5Ghz
 - 84 pas de fréquences
 - `radio.setFrequency(freq) // 0-83`
- Group
 - Permet le filtrage des messages
 - 256 valeurs
 - `radio.setGroup(group); // 0-255`
- Fonctions de communication
 - Chaîne de caractères : 19 caractères max
 - Nombre : nombre décimal ou entier
 - Couple chaîne de caractères (8 caractères max) et un nombre
 - Buffer (19 octets max)
 - Emission : `sendXXXXXX` où xxxx est le type
 - Réception : `onReceivedXXXXXX` où xxxx est le type
 - Détail du paquet reçu : `receivedPacket`



Documentation : <https://makecode.microbit.org/reference/radio>

IOT : COMMUNICATION RADIO

- **Langage** : Javascript
- **Objectif** : Faire communiquer les micro:bit par radio. Chacun peut choisir son canal, son icône, et l'envoyer aux autres.
- **Lien** :  <https://github.com/jotitan/microbit-radio-icons>
- **Etapes** :
 - Sélection d'un canal de communication
 - Sélection d'une icône
 - Envoi de l'icône
 - Affichage d'une icône reçue sur le canal



IOT : JEU DU « TU CHAUFFES TU BRULES »

- **Langage** : Javascript
- **Objectif** : Trouver les balises cachées dans la salle
- **Lien** : <https://github.com/jotitan/microbit-burn-cold-game>
- **Exercice** :
 - Plusieurs émetteurs sont présents dans la salle et envoient des messages sur les canaux 4 à 9 toutes les 500 ms
 - Ecrire un récepteur qui utilise la puissance de réception du message pour estimer relativement la distance
 - Afficher sur l'écran des indices pour évaluer la distance de la balise
 - Utiliser les boutons pour changer de groupe radio pour trouver les autres balises





Merci

