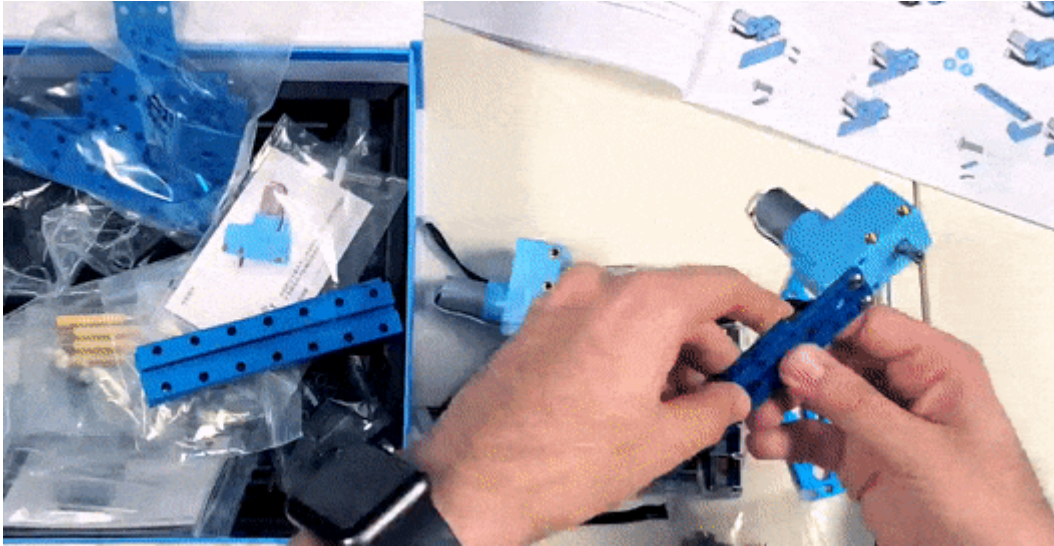


## التحكم بالروبوت عبر الأوامر الصوتية

سنقوم في هذا المشروع باستخدام مستشعر الصوت الموجود في اللوحة الرئيسية Auriga لروبوت Ranger لبرمجة روبوت يستجيب للأوامر الصوتية



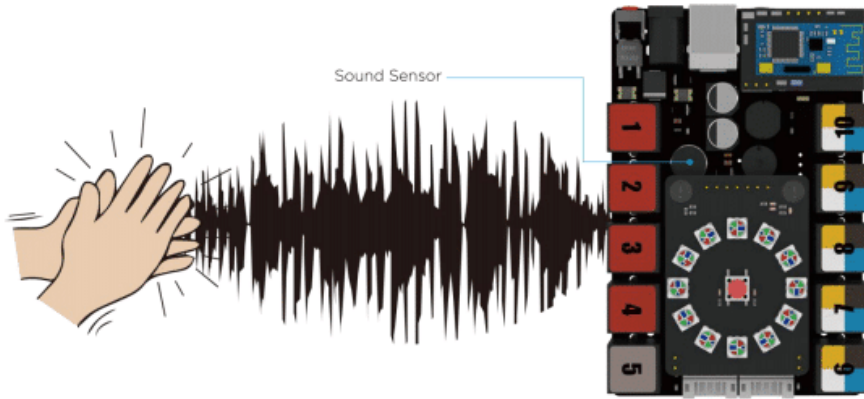
الأدوات المطلوبة



mBot Ranger Kit

## Sound Sensor مستشعر الصوت

يمكن أن نقول بأن مستشعر الصوت هو عبارة عن مايكروفون صغير يقوم بتحويل الصوت إلى إشارة كهربائية، ويعطي قيمة تتراوح بين 0-1023. تختلف القيمة باختلاف المحيط الذي نقوم بتشغيل الروبوت فيه لا يستطيع مستشعر الصوت التمييز بين الكلمات لكن يستطيع التمييز بين الأصوات المرتفعة والمنخفضة التي تقع في المدى الخاص به. كلما كانت شدة الصوت أعلى كانت الإشارة الكهربائية أكثر.

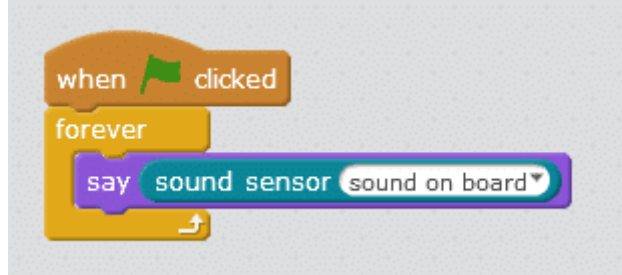


### تحديد مدى قراءة المستشعر

قبل أن نقوم بكتابة البرنامج نحتاج أولاً إلى عمل اختبار صغير لمعرفة المدى الذي يقرأ فيه المستشعر، والقراءات التي يعطيها عند

التصفيق أو الكلام أو أي أصوات أخرى.  
نقوم أولاً بتوصيل الروبوت إلى الحاسوب عن طريق الكيبل.

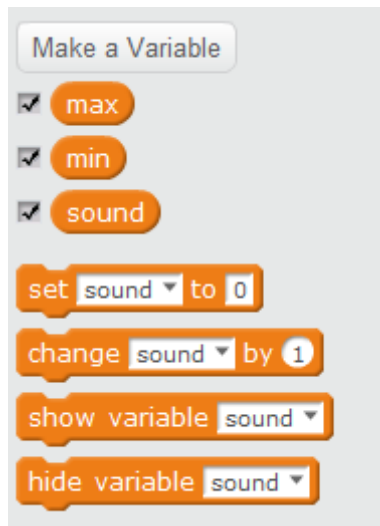
قم بإنشاء متغير Sound  
اختيار الأمر say وتعيينه على مستشعر الضوء  
ووضع الأمر داخل حلقة forever



ستظهر قراءة المستشعر على الشاشة



نلاحظ أن القيمة تتغير بسرعة كبيرة جداً وبهذا يصعب تحديد أعلى قيمة وأقل قيمة، لذا سنضيف متغيرين آخرين  
Max ليمثل أعلى قيمة، و Min ليمثل أقل قيمة



سنقوم بتعيين المتغيرين على قراءة مستشعر الصوت

```
set max to sound sensor sound on board
set Min to sound sensor sound on board
```

ونقوم كذلك بتعيين المتغير Sound لقراءة مستشعر الصوت

```
forever
  set Sound to sound sensor sound on board
```

في البداية ستكون  $Max = Min$   
داخل حلقة forever ستتغير قيمة المتغير Sound  
ولتحديد أعلى وأقل قيمة نستخدم if ..then  
بحيث إذا كانت قيمة sound أكبر من قيمة max  
يتم إعادة تعيين قيمة max إلى هذه القيمة كأعلى قيمة  
ويتم إعادة تعيين قيمة min إلى أقل قيمة تمت قراءتها في المتغير Sound

```
when green flag clicked
  set max to sound sensor sound on board
  set Min to sound sensor sound on board
  forever
    set sound to sound sensor sound on board
    if sound > max then
      set max to sound
    if sound < Min then
      set Min to sound
```

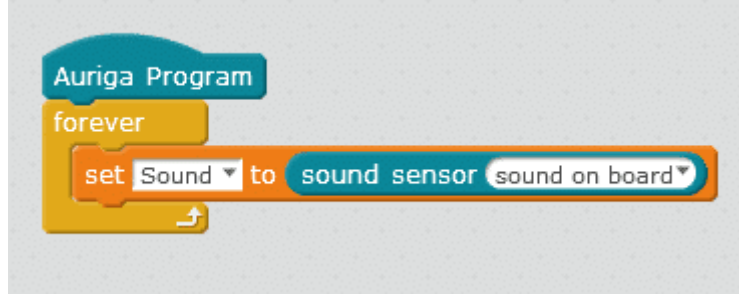
## الكود البرمجي

سنقوم ببرمجة الروبوت بحيث يتراجع للوراء عند التصفيق أو إعطائه أي أمر صوتي

سنأخذ بالاعتبار القيم التي حصلنا عليها من الكود السابق، لاستخدامها في كتابة كود الروبوت الجديد

نقوم بفتح صفحة جديدة في برنامج mBlock

وإنشاء متغير Sound وتعيينه لقراءة مستشعر الصوت في اللوحة الرئيسية



نقوم بإضافة شرط if then

بحيث إذا كانت قراءة مستشعر الصوت أعلى من قيمة معينة يتحرك الروبوت للخلف

لتحديد هذه القيمة نعود إلى القيم التي حصلنا عليها من الكود السابق

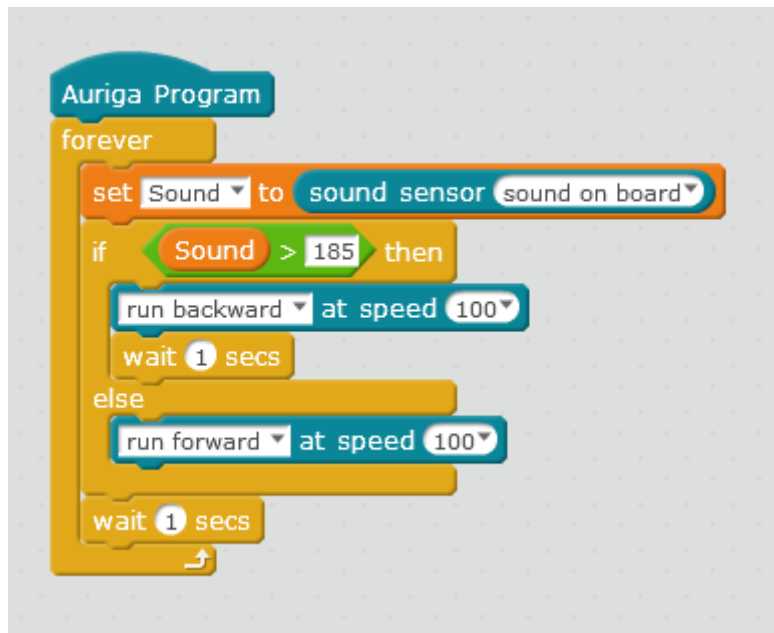


نلاحظ في الصورة أن قيمة  $\text{max} = 184$  وهذه القيمة تم تسجيلها قبل إصدار أي صوت

مما يعني أنه عند التصفيق مثلا ستكون قيمة المستشعر أكبر من هذه القيمة



إذا كانت قيمة القراءة أقل، يتم تنفيذ الأوامر بعد else وهو التحرك للأمام



```
Auriga Program
forever
  set Sound to sound sensor sound on board
  if Sound > 185 then
    run backward at speed 100
    wait 1 secs
  else
    run forward at speed 100
  wait 1 secs
```

The image shows a Scratch script for a program named "Auriga Program". It starts with a "forever" loop. Inside the loop, the first block is "set Sound to sound sensor sound on board". This is followed by an "if" block with the condition "Sound > 185". If the condition is true, the script executes "run backward at speed 100" followed by "wait 1 secs". If the condition is false, it executes "run forward at speed 100". After the "if" block, there is a "wait 1 secs" block. The loop ends with a small arrow icon pointing back to the start of the "forever" loop.

قم بتجربة إضافة أوامر برمجية تقوم بإضاءة LED عند التصفيق

لمزيد من المعلومات حول الأوامر البرمجية الأخرى قم بالرجوع لدرس جولة حول الأوامر البرمجية