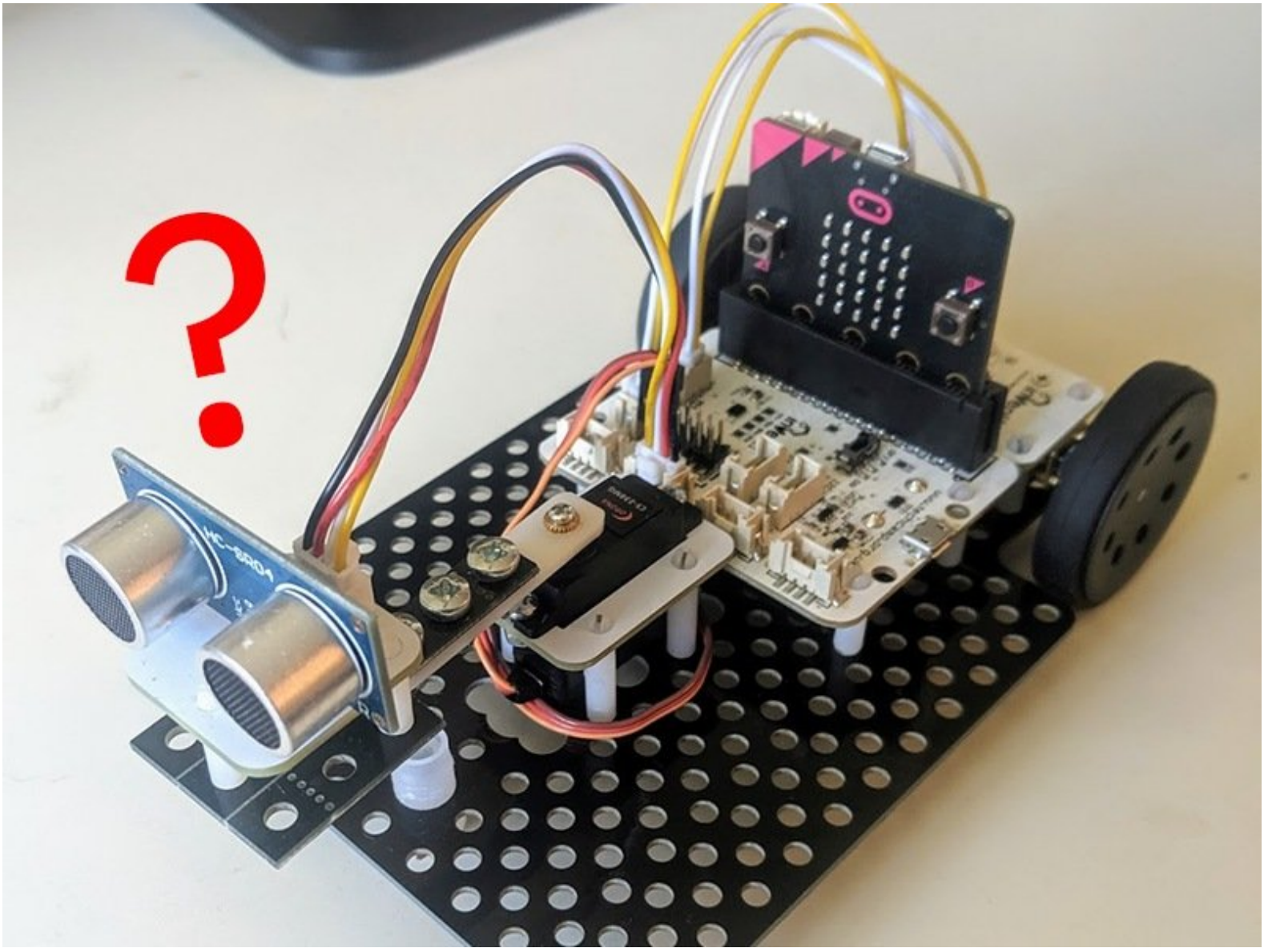
4 - Smarter Driving

Use the servo sonar module to improve our intelligent driving!



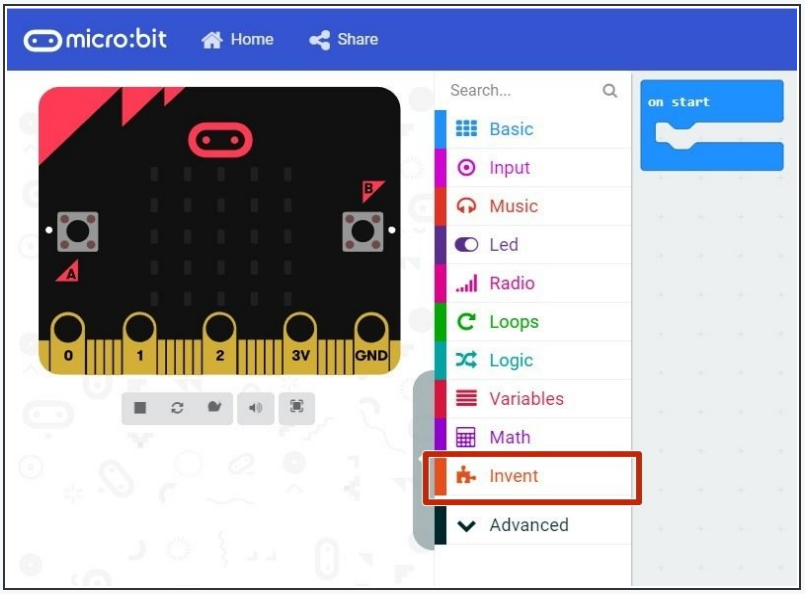
INTRODUCTION

Use the servo sonar module to improve our intelligent driving!

Step 1

Accessing the Invent modules

- In this project, we'll be using your robot's motors to create a self-driving car!
- To access the motor libraries, go to https://www.techcamp.org.uk/invent (https://www.techcamp.org.uk/invent) . This will give you access to our Invent modules, which includes a module for controlling the motors.
- You'll know you're in the right place if you can see the Invent namespace on the right-hand side.

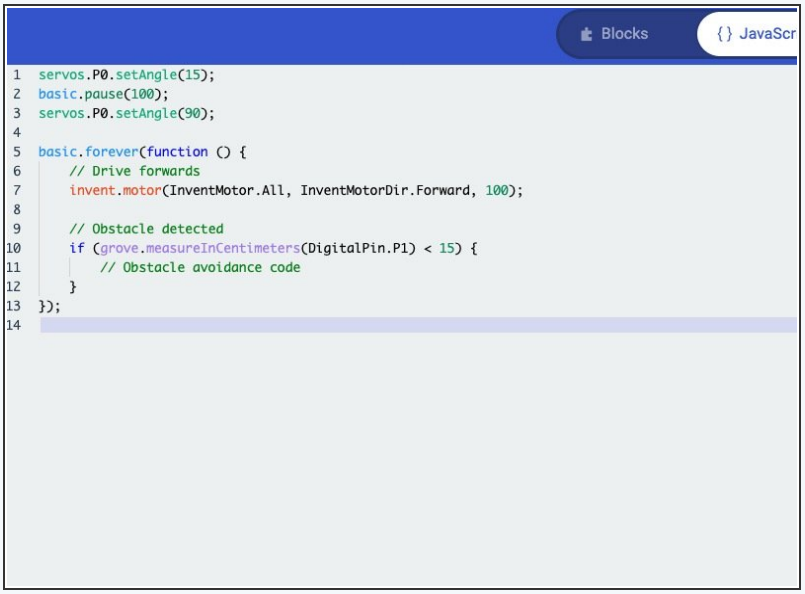


Step 2

Starting our code

- In this project, we want to develop a car that looks left and right when it detects an obstacle, and then goes in the direction with more clearance. Let's get started!
- First, we reset the servo position to be looking straight ahead in the first 3 lines.
- Next in the forever loop, we tell the car to drive forwards. We then have an if statement, which will run if there's an obstacle in front of the sonar.

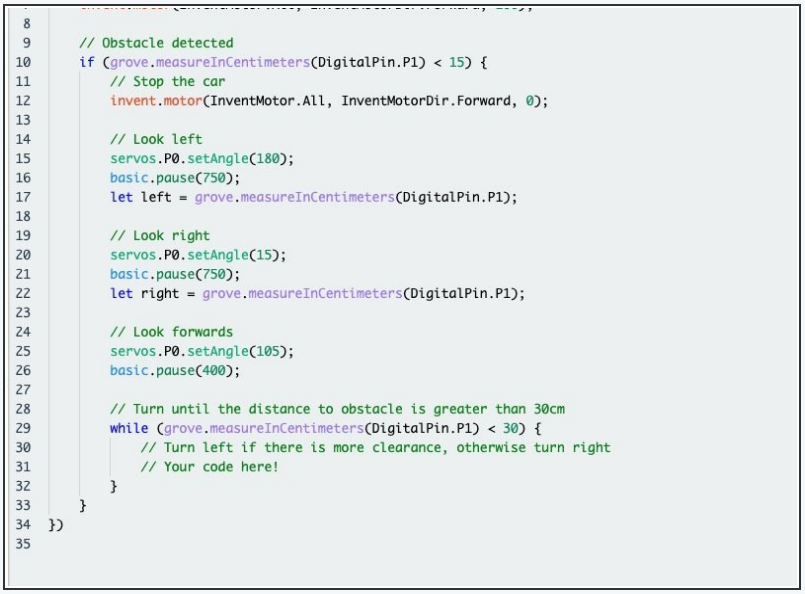
In TypeScript, comments are denoted with // rather than # like in Python.



Step 3

Writing the obstacle avoidance code

- First, we need to stop the car if there's an obstacle so we don't ram into it!
- Next, we use the servo to look left and take the sonar reading, and then the same for the right.
- Now that we have our measurements, we can look straight ahead again.



```
    // Obstacle detected
    if (grove.measureInCentimeters(DigitalPin.P1) < 15) {
        // Stop the car
        invent.motor(InventMotor.All, InventMotorDir.Forward, 0);

        // Look left
        servos.P0.setAngle(180);
        basic.pause(750);
        let left = grove.measureInCentimeters(DigitalPin.P1);

        // Look right
        servos.P0.setAngle(15);
        basic.pause(750);
        let right = grove.measureInCentimeters(DigitalPin.P1);

        // Look forwards
        servos.P0.setAngle(105);
        basic.pause(400);

        // Turn until the distance to obstacle is greater than 30cm
        while (grove.measureInCentimeters(DigitalPin.P1) < 30) {
            // Turn left if there is more clearance, otherwise turn right
            // Your code here!
        }
    }
})
```

Step 4

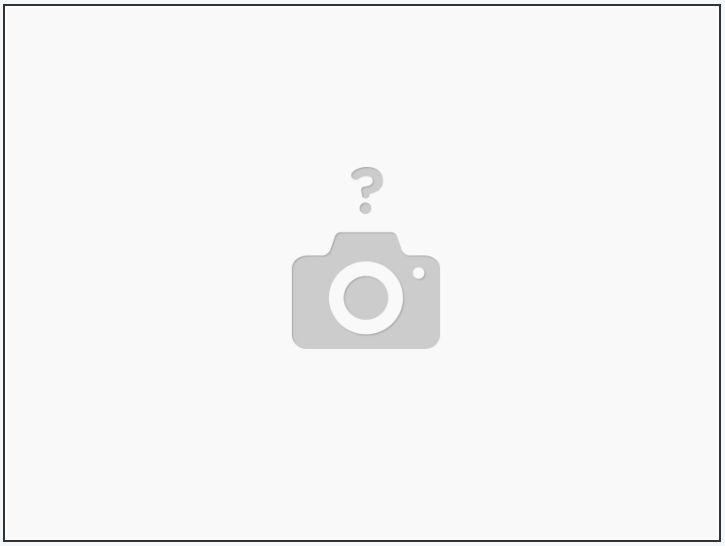
Challenge: finish the code!

- You'll notice we left the contents of the while loop empty!
- Can you write the code that will make the car turn left if there's more space, or right if not?

```
    // Obstacle detected
    if (grove.measureInCentimeters(DigitalPin.P1) < 15) {
        // Stop the car
        invent.motor(InventMotor.All, InventMotorDir.Forward, 0);

        // Look left
        servos.P0.setAngle(180);
        basic.pause(750);
        let left = grove.measureInCentimeters(DigitalPin.P1);

        // Look right
        servos.P0.setAngle(15);
        basic.pause(750);
        let right = grove.measureInCentimeters(DigitalPin.P1);

        // Look forwards
        servos.P0.setAngle(105);
        basic.pause(400);

        // Turn until the distance to obstacle is greater than 30cm
        while (grove.measureInCentimeters(DigitalPin.P1) < 30) {
            // Turn left if there is more clearance, otherwise turn right
            // Your code here!
        }
    }
})
```

Step 5

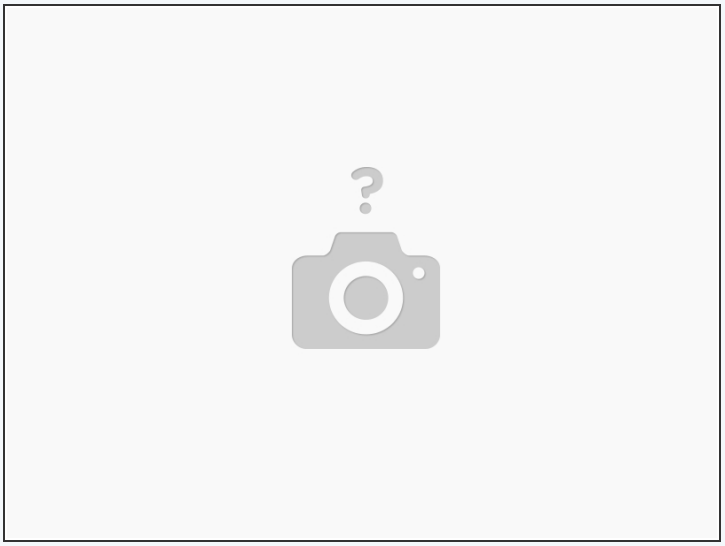
Building the car



- On your board from the sonar project, attach the two motors if they're not already installed!

Step 6

Testing it out!



- Program your micro:bit and turn on the car!
- You should find that when it encounters an obstacle, it will stop, look both ways, and then turn in the direction with more clearance. If so, you're done!