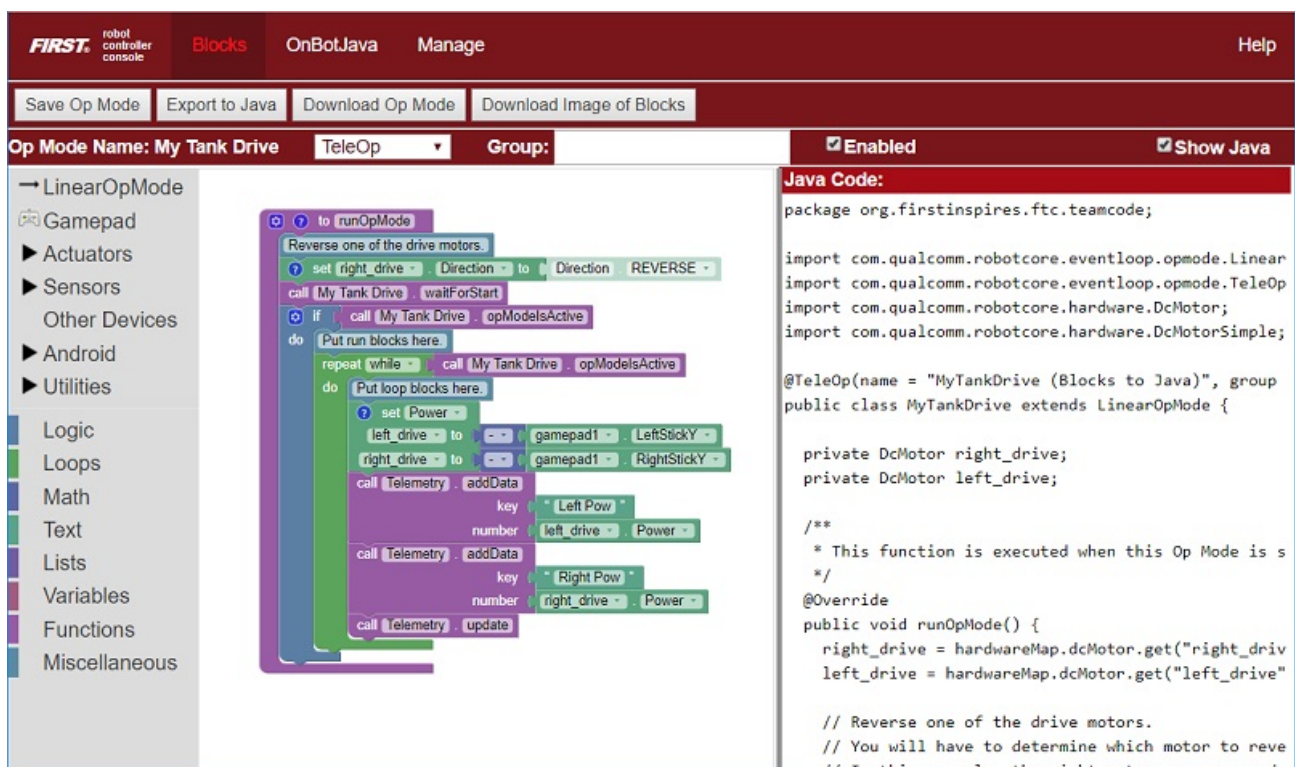


Programming Language Options

When programming within the REV Control System, there are three programming tools to choose from: Blocks, OnBot Java, and Android Studio. The following section highlights basic information, use cases, and access examples for the three compilers.

Blocks

The Blocks Programming Tool is a visual, programming tool that lets programmers use a web browser to create, edit and save their op modes. This tool offers preset snippets of code that can be presented visually, using a drag-and-drop interface.



The Blocks Programming Tool lacks the complexity of the Java based programming tools available, which makes it a great place to start for rookie or novice programmers. Rookie programmers can learn programming logic in an intuitive and easy-to-learn environment. Because the Blocks Programming tool is a web-based interface, where programs are saved directly to the robot, it is easy to access on most devices to make code changes.

Accessing Blocks



This section assumes that you have already gone through the [Configuring Android Device](#) process and that you have JavaScript enabled web browser.

1. Go to WiFi Settings, on a Windows 10 Computer, by clicking on the Wi-Fi symbol.
2. Once the list of available Wi-Fi networks in the vicinity is displayed select the network that matches the name of your WiFi access point.
3. Enter the password you set during the [configuration phase](#).
4. Once connected, open a JavaScript enabled browser (FIRST recommends Google Chrome).
5. Go to IP Address **http://192.168.43.1:8080**
6. At the top of the Robot Controller Console Page, there should be 3 menu options: Blocks, OnBot Java, and Manage. Choose Blocks.
7. Check out the [First Op Mode](#) section to begin coding!



Passwords are case sensitive. If you do not remember your password, check the Program and Manage menu option on *your driver station*.

OnBot Java

A text-based programming tool that lets programmers use a web browser to create, edit and save their Java op modes.


```
1 package org.firstinspires.ftc.teamcode;
2
3 import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
4 import com.qualcomm.robotcore.eventloop.opmode.TeleOp;
5 import com.qualcomm.robotcore.hardware.DcMotor;
6 import com.qualcomm.robotcore.hardware.DcMotorSimple;
7
8 @TeleOp(name = "MyTankDrive (Blocks to Java)", group = "")
9 public class MyTankDrive extends LinearOpMode {
10
11     private DcMotor right_drive;
12     private DcMotor left_drive;
13
14     /**
15      * This function is executed when this Op Mode is selected from the
16      * menu.
17      */
18     @Override
19     public void runOpMode() {
```

Build started at Mon Jun 17 2019 15:45:25 GMT-0400 (Eastern Daylight Time)

Build finished in 2.9 seconds
Build succeeded!


OnBot Java is great for programmers with basic to advanced Java skills who would like to write text-based op modes. OnBot Java shares a web-based interface with the Blocks Programming tool. The web-based model is easy to access on most devices to make code change and reduces the need to have one set device for code changes.

Accessing OnBot Java

 This section assumes that you have already gone through the [Configuring Android Device](#) process and that you have JavaScript enabled web browser.

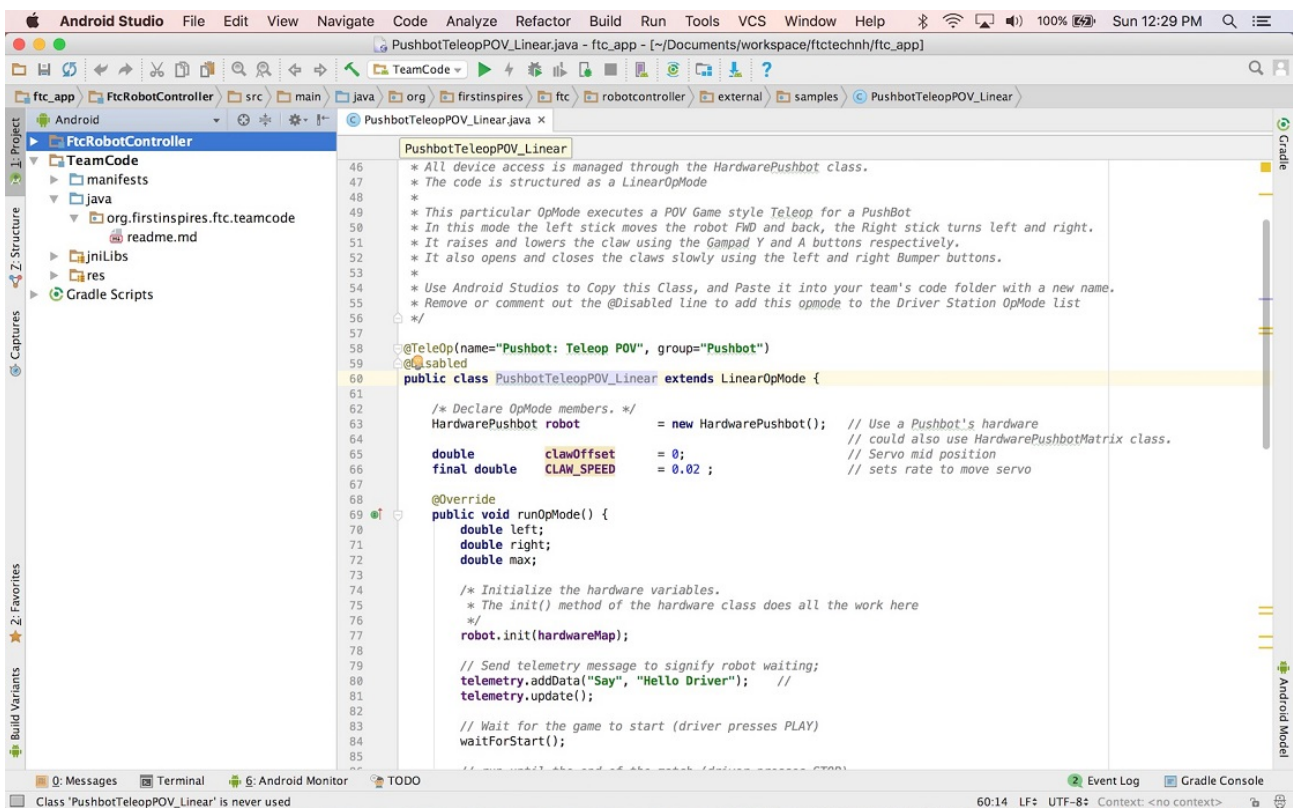
1. Go to WiFi Settings, on a Windows 10 Computer, by clicking on the Wi-Fi symbol.
2. Once the list of available Wi-Fi networks in the vicinity is displayed select the network that matches the name of your WiFi access point.
3. Enter the password you set during the [configuration phase](#).
4. Once connected open a JavaScript enabled browser (FIRST recommends Google Chrome).
5. Go to IP Address **http://192.168.43.1:8080**

- At the top of the Robot Controller Console Page There should be 3 menu options Blocks, OnBot Java, and Manage. Choose OnBot Java
- Check out the **First Op Mode** section to begin coding!

 Passwords are case sensitive. If you do not remember your password check the Program and Manage menu option on *your driver station*.

Android Studio - Java

An advanced integrated development environment for creating Android apps. This tool is the same tool that professional Android app developers use. Android Studio is only recommended for advanced users who have extensive Java programming experience.



```
46  * All device access is managed through the HardwarePushbot class.
47  * The code is structured as a LinearOpMode
48  *
49  * This particular OpMode executes a POV Game style Teleop for a PushBot
50  * In this mode the left stick moves the robot FwD and back, the Right stick turns left and right.
51  * It raises and lowers the claw using the Gampad Y and A buttons respectively.
52  * It also opens and closes the claws slowly using the left and right Bumper buttons.
53  *
54  * Use Android Studios to Copy this Class, and Paste it into your team's code folder with a new name.
55  * Remove or comment out the @Disabled line to add this opmode to the Driver Station OpMode list
56  */
57
58  @TeleOp(name="Pushbot: Teleop POV", group="Pushbot")
59  @Disabled
60  public class PushbotTeleopPOV_Linear extends LinearOpMode {
61
62      /* Declare OpMode members. */
63      HardwarePushbot robot = new HardwarePushbot(); // Use a Pushbot's hardware
64                                                    // could also use HardwarePushbotMatrix class.
65
66      double clawOffset = 0; // Servo mid position
67      final double CLAW_SPEED = 0.02; // sets rate to move servo
68
69      @Override
70      public void runOpMode() {
71          double left;
72          double right;
73          double max;
74
75          /* Initialize the hardware variables.
76           * The init() method of the hardware class does all the work here
77           */
78          robot.init(hardwareMap);
79
80          // Send telemetry message to signify robot waiting;
81          telemetry.addData("Say", "Hello Driver"); //
82          telemetry.update();
83
84          // Wait for the game to start (driver presses PLAY)
85          waitForStart();
86      }
87  }
```

Android Studio allows programmer with an advanced understanding of Java a more powerful development environment to work in. It offers enhanced editing and debugging

features not available with OnBot Java or Blocks. However, Android Studio is not a web-based software and will need a dedicated laptop to run on.

Accessing Android Studio

To learn about how to properly download and work with Android Studio please visit the [FTC Wiki](#).