Basic Movement example

Now that you've tested your robot, and know it works, let's try making it move ourselves.

Click on "File", then "Examples", then scroll all the way down the list, until you see "ProtoBotV03". Click on that, then click on "BasicTemplate". This is a clean template file, with no user code in it yet.

```
/*
 *  Template sketch for the Protobot library version 0.3
 *
 *  Written by Jacob Field
 *  Licensed under the MIT License, see license.txt for more details
 */

#include <ProtoBot.h>

ProtoBot robot;

void setup() {

  //Setup the library, init pins
  robot.setupRobot();

  //Now, tell the robot to wait till the left bump sensor is pressed
  robot.waitForBump();

}

void loop() {

  //Code to be run here

}
```

This is where we'll be writing our code, where the highlight is, after the line of code that says "robot.waitForBump()".

Moving the robot Forwards/Backwards:

Type the highlighted code into the arduino editor.

```
void setup() {

  //Setup the library, init pins
  robot.setupRobot();

  //Now, tell the robot to wait till the left bump sensor is pressed
  robot.waitForBump();

  robot.setRight(200);
  robot.setLeft(200);

}
```

Here's how it works:

The "robot.setLeft" and "setRight" functions turn the motors on. The number inside the round "()" brackets are how fast the motors go.

Here, the number is 200, which is almost full speed ahead. The max speed is 255, -255 is full speed backwards, and 0 is stopped.

Try uploading the program, and see what happens!

You might notice the motors never stop. That's because we never told them to stop! How do you think we might stop them?

Here's the code to stop the robot.

Go ahead and add it!

```
void setup() {

  //Setup the library, init pins
  robot.setupRobot();

  //Now, tell the robot to wait till the left bump sensor is pressed
  robot.waitForBump();

  robot.setRight(200);
  robot.setLeft(200);

  delay(1000);

  robot.setRight(0);
  robot.setRight(0);

}
```

Here's how it works:

The "robot.setLeft(0)" and "robot.setRight(0)" lines tell the motors to turn off. But what does the "delay(1000)" do?

The "delay" function tells the robot to wait a certain amount of time. The number inside the round "()" brackets is how long to wait, in milliseconds. 1000 milliseconds is 1 second, 500 is half a second, and so on.

Try changing the number inside the round brackets,

to see if you can get it to drive a certain distance!

Try getting the robot to go backwards!

You can change the speeds to -200 to make it go backwards.

Turning the robot:

To turn, all you need to do is reverse one of the motors. That way, the robot will spin in a circle.

To reverse a motor, just add a “-” before the number inside the round brackets.

```
void setup() {

  //Setup the library, init pins
  robot.setupRobot();

  //Now, tell the robot to wait till the left bump sensor is pressed
  robot.waitForBump();

  robot.setRight(-200);
  robot.setLeft(200);

  delay(1000);

  robot.setRight(0);
  robot.setRight(0);

}
```

How does it work?

By changing the value inside the “delay” function, you can control how long it turns.

By switching which motor is reversed, you switch which way it turns.

```
  robot.setRight(200);
  robot.setLeft(-200);
```

Try turning by switching one motor off, instead!

Look at how this changes how your robot turns. Would this be a better way to turn?