

Introduction to Programming - Robotics Course

1. Opening the IDE

When cadets first open the IDE, they will see this:

```
void setup() {  
  // put your setup code here, to run once:  
  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
  
}
```

Describe that all lines with a `//` are comments and will not be executed by the Arduino.

2. Serial Monitor

Now begin the Serial Monitor, this is how we will view our code

```
void setup() {  
  // put your setup code here, to run once:  
  Serial.begin(9600);  
  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
  
}
```

Discuss two things here:

1. `9600` is the baud rate or how fast you can communicate over Serial Comms. `9600` means we can send 9,600 bits of information per second
2. Every line of Arduino code must end in a `;`, this tells the Arduino to execute the command

3. Printing to the Serial Monitor

Now enter this line into the loop function:

`Serial.println("Hello World");` the code should now look like:

```
void setup() {  
  // put your setup code here, to run once:  
  Serial.begin(9600);  
  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
  Serial.println("Hello World");  
  
}
```

`Serial.println()` is an Arduino function that will print a line of text onto the Serial Monitor. `ln` here means it will print a new line after the text, like pressing `enter` on a keyboard.

We can place any text between the `" "`

Get cadets to upload and view the result, you should see `Hello World` being printed each program loop:

```
Hello World  
Hello World  
Hello World  
Hello World  
Hello World  
...
```

4. Adding Delays

We can add delays to make the printing more readable:

```
void loop() {  
  // put your main code here, to run repeatedly:  
  Serial.println("Hello World");  
  delay(500);  
  
}
```

This `delay()` function pauses the Arduino for a number of milliseconds, `delay(500);` pauses for 500 milliseconds or 0.5 seconds.

5. Freeplay

Get cadets to have a play:

- Changing the contents of the `Serial.println()` line
- Changing the duration of the `delay()` function
- Adding more prints and delays