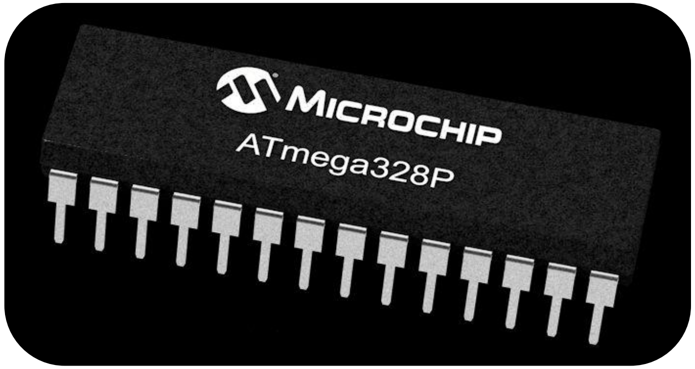
The Arduino IDE and coding in C (part 1)

Introduction to the Arduino IDE (*integrated development environment)*

Based on C++

Latest version ARDUINO IDE 1.8.3 can be downloaded from: https://www.arduino.cc/en/Main/Software



Presentation by Eric S. Clarke 09/06/2017

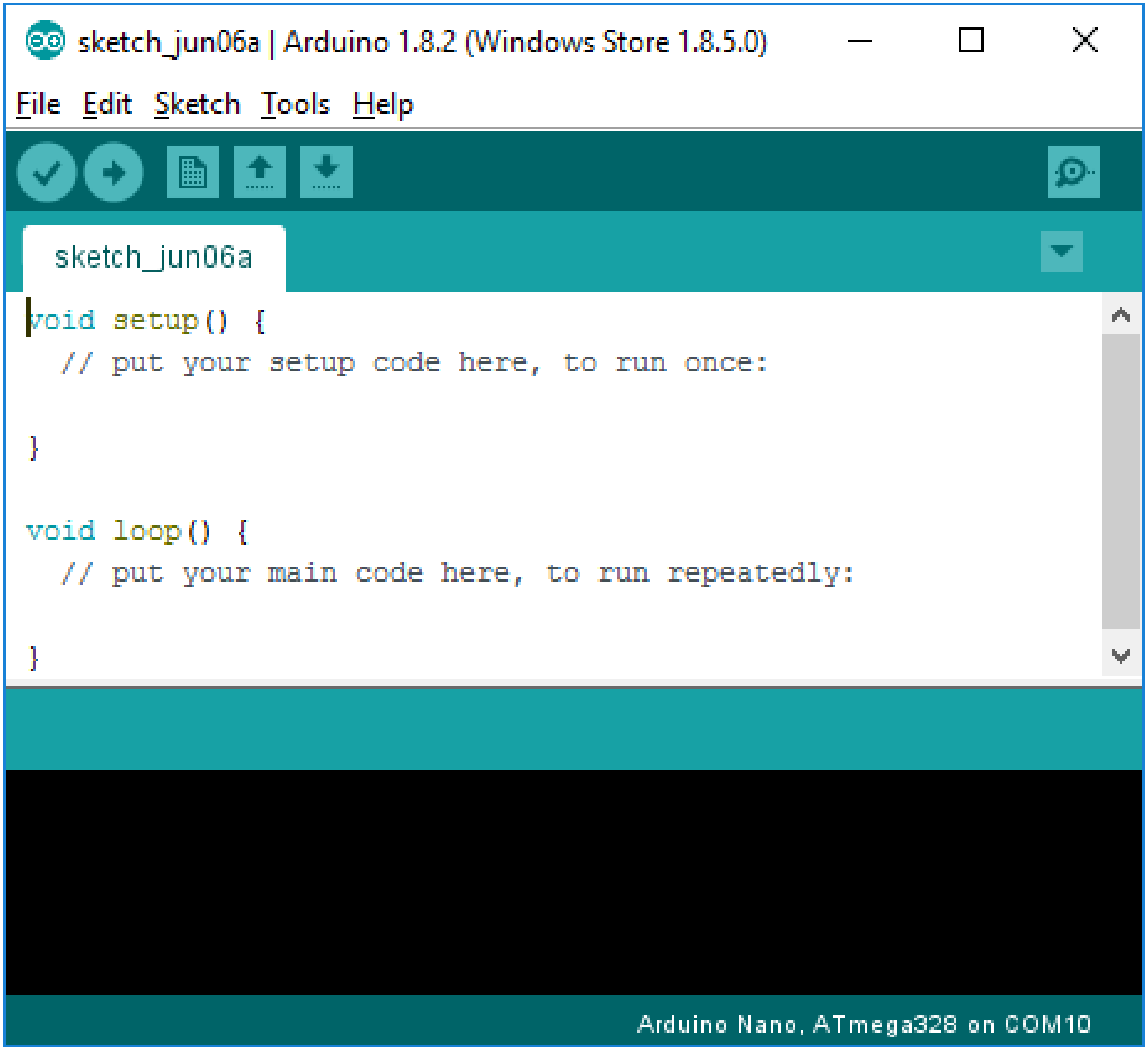
Standard C bare minimum code

```
// any #define here – constants etc
// any #include here – header files etc

Main()
    {
        // put your code here
    }
```

Arduino C bare minimum code

```
//any #define here – constants etc
//any #include here – header files etc
// any global variables here

void setup()
    {
        // put your setup code here to run once:
    }

Void loop()
    {
        // put your main code here to run repeatedly:
    }
```

Arduino C bare minimum code

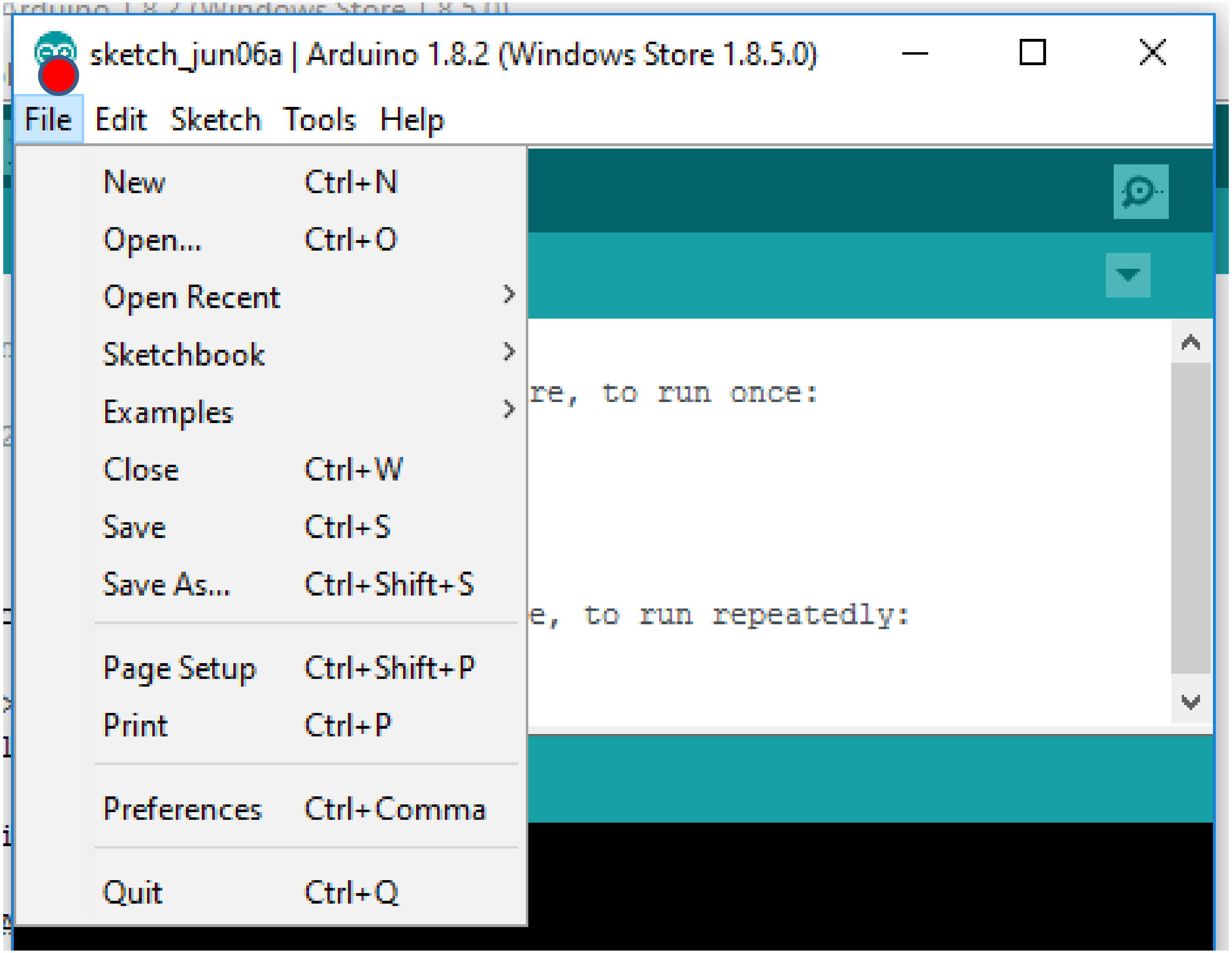


sketch_jun06a | Arduino 1.8.2 (Windows Store 1.8.5.0)
File Edit Sketch Tools Help
sketch_jun06a

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

}

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

}
```

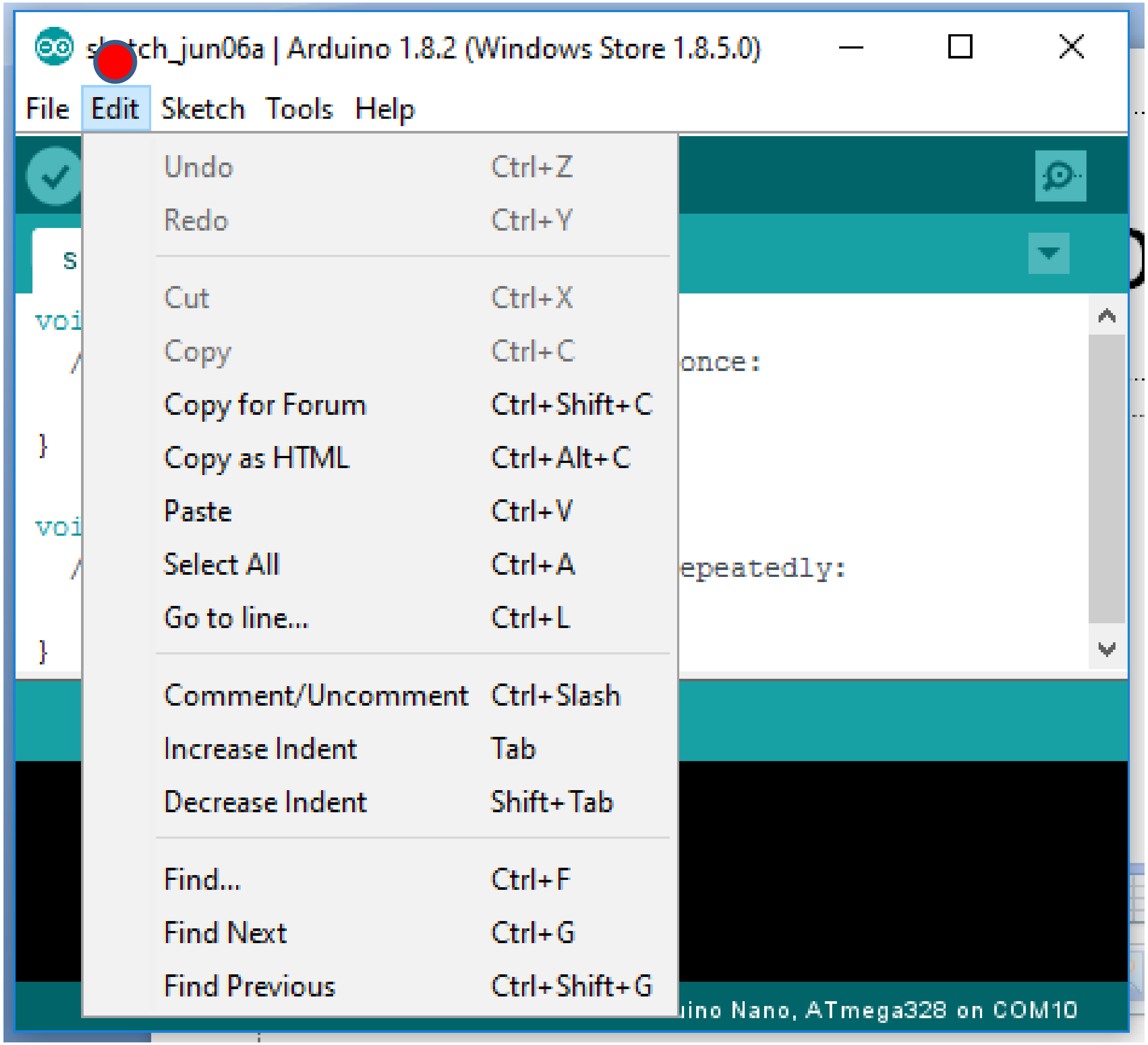
Arduino Nano, ATmega328 on COM10

File Menu

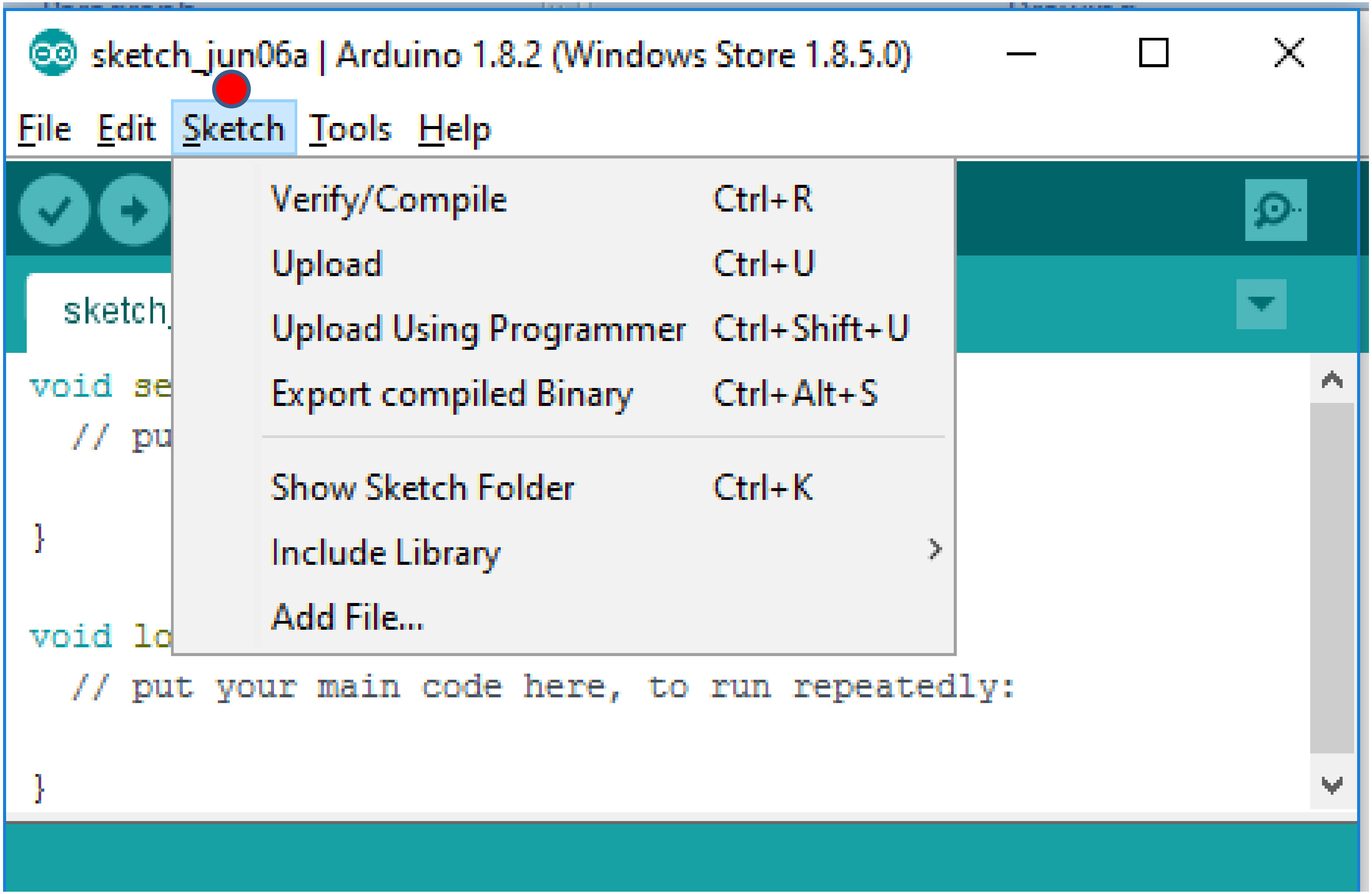


```
re, to run once:

e, to run repeatedly:
```

Edit Menu


sketch_jun06a | Arduino 1.8.2 (Windows Store 1.8.5.0)
File
Edit
Sketch
Tools
Help
Undo Ctrl+Z
Redo Ctrl+Y
Cut Ctrl+X
Copy Ctrl+C
Copy for Forum Ctrl+Shift+C
Copy as HTML Ctrl+Alt+C
Paste Ctrl+V
Select All Ctrl+A
Go to line... Ctrl+L
Comment/Uncomment Ctrl+Slash
Increase Indent Tab
Decrease Indent Shift+Tab
Find... Ctrl+F
Find Next Ctrl+G
Find Previous Ctrl+Shift+G
once:
epeatedly:
ino Nano, ATmega328 on COM10

Sketch or Code Menu

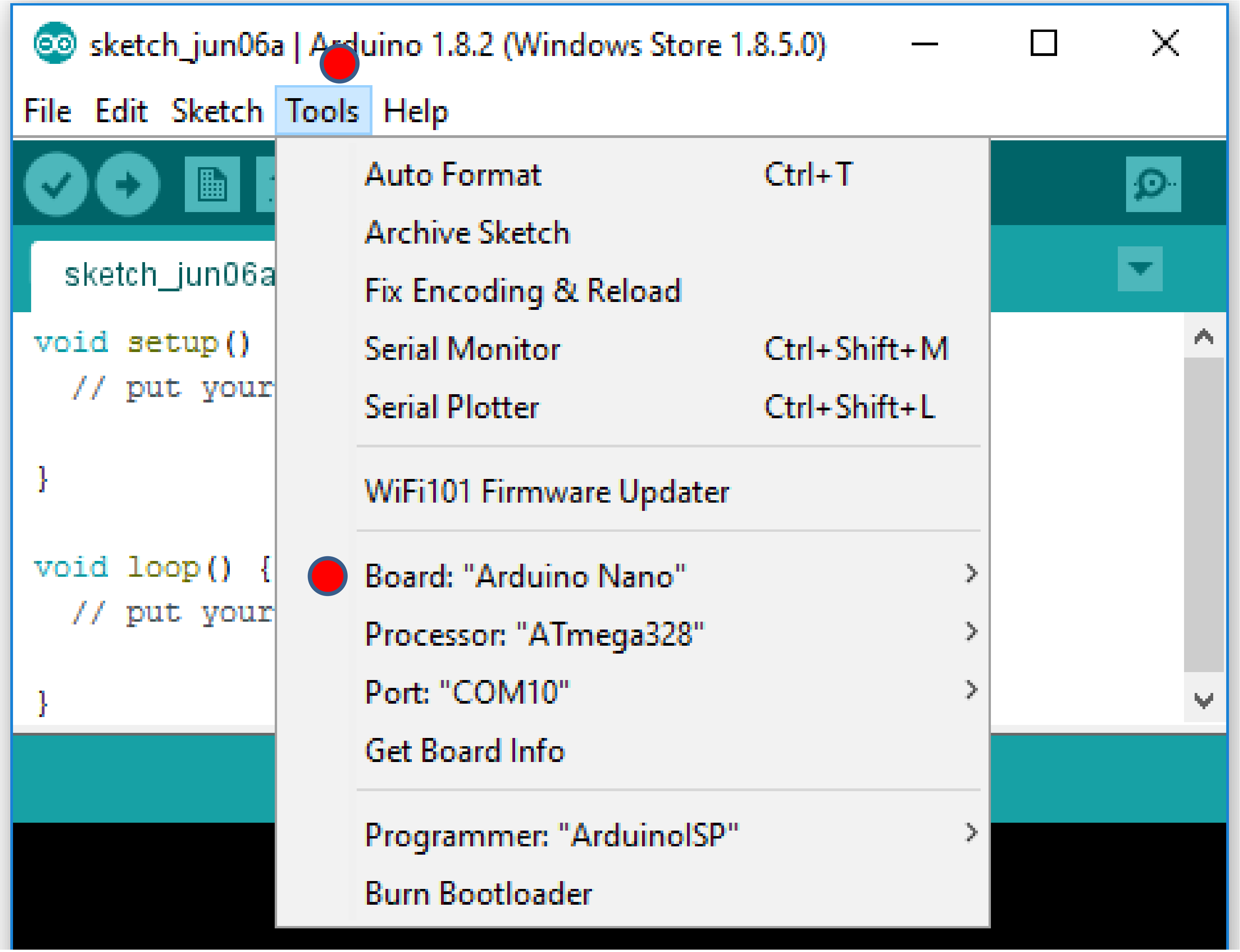


sketch_jun06a | Arduino 1.8.2 (Windows Store 1.8.5.0)

File
Edit
Sketch
Tools
Help
Verify/Compile Ctrl+R
Upload Ctrl+U
Upload Using Programmer Ctrl+Shift+U
Export compiled Binary Ctrl+Alt+S
Show Sketch Folder Ctrl+K
Include Library
Add File...
sketch

```
void se
  // pu

}

void lo
  // put your main code here, to run repeatedly:

}
```

Tools Menu



Board: "Arduino Nano"

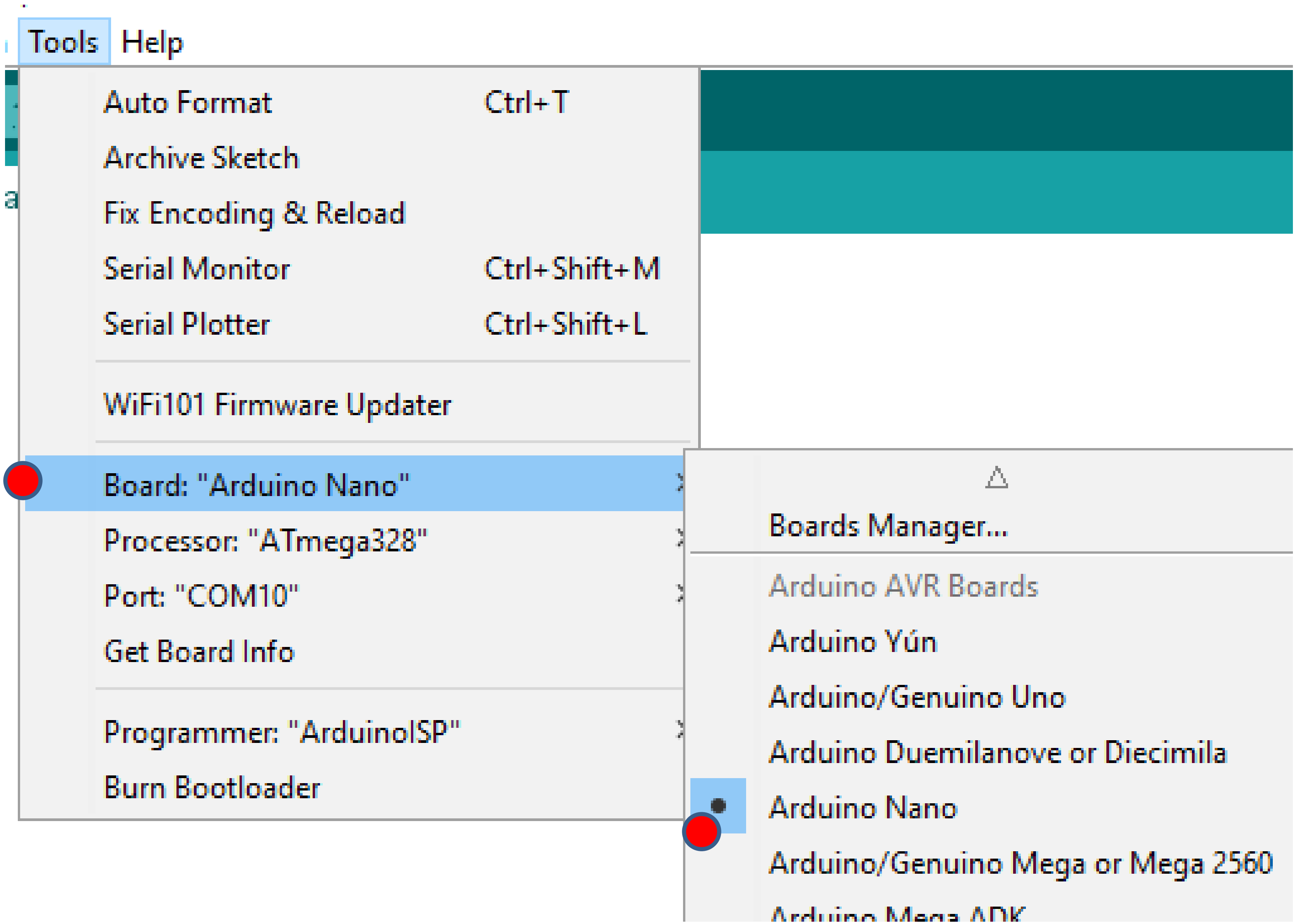
Processor: "ATmega328"

Port: "COM10"

Programmer: "ArduinoISP"

```
void setup()
  // put your

}

void loop() {
  // put your

}
```

Choosing the Arduino board type



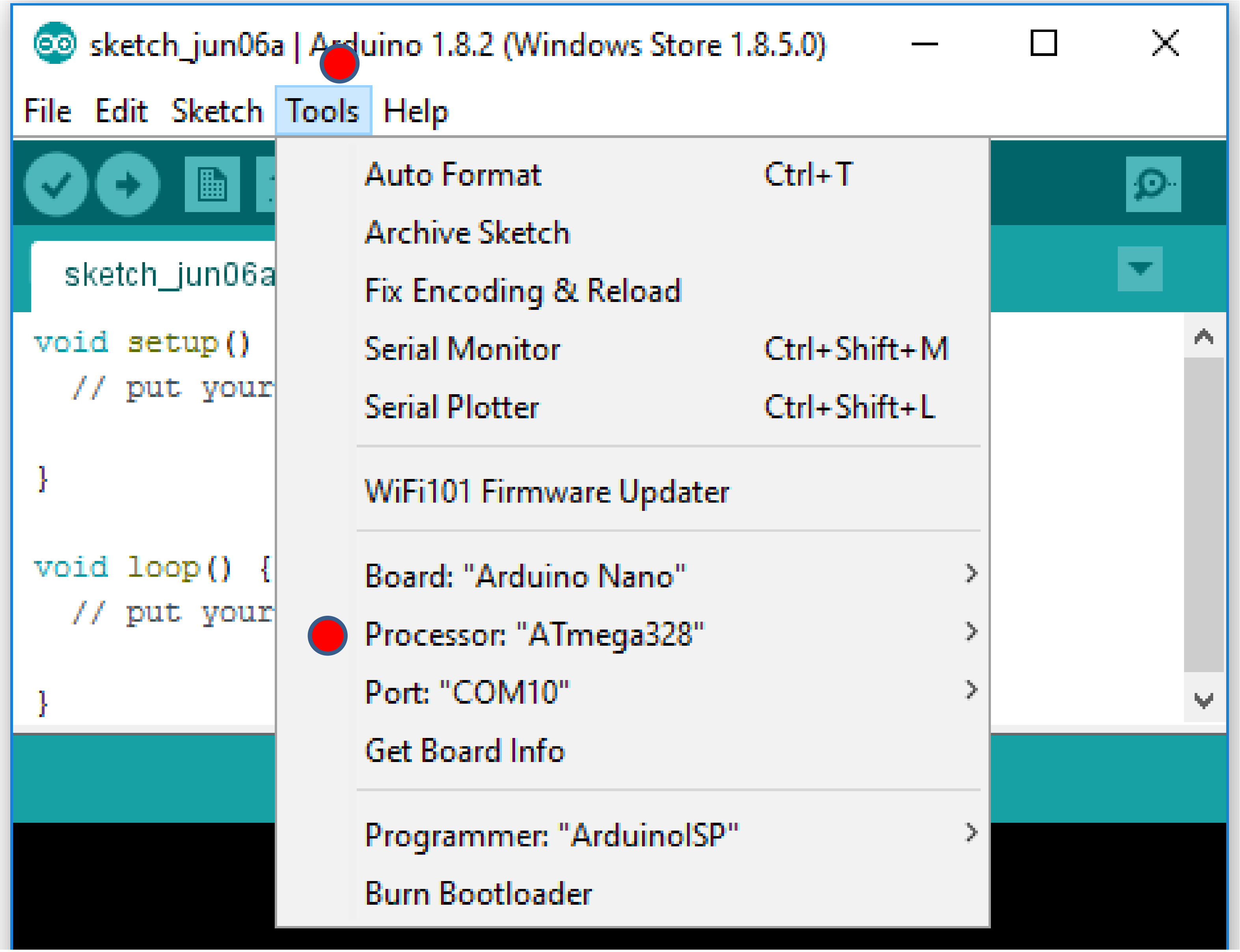
Board: "Arduino Nano"

Processor: "ATmega328"

Port: "COM10"

Programmer: "ArduinoISP"

Tools Menu



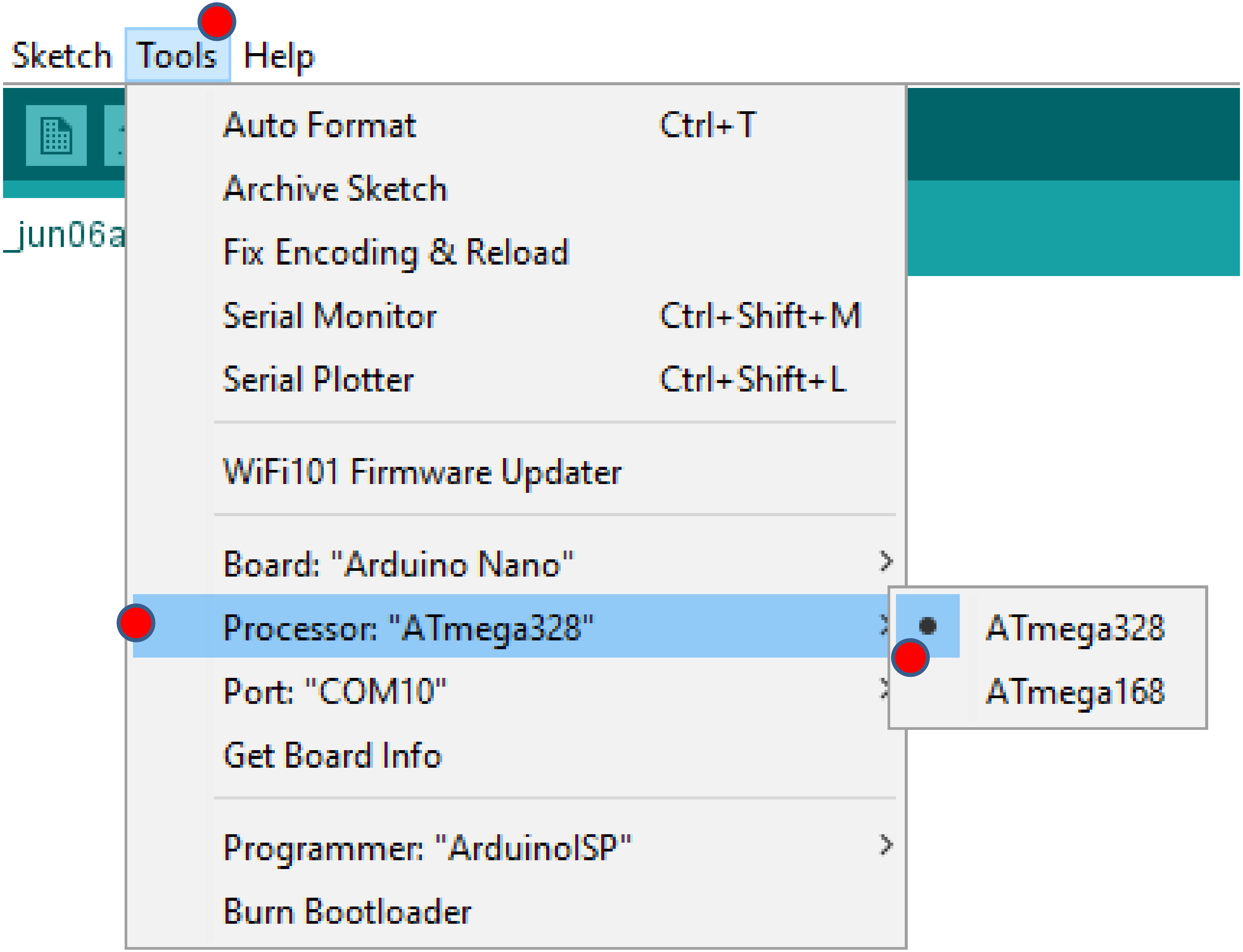
sketch_jun06a | Arduino 1.8.2 (Windows Store 1.8.5.0)

File Edit Sketch Tools Help

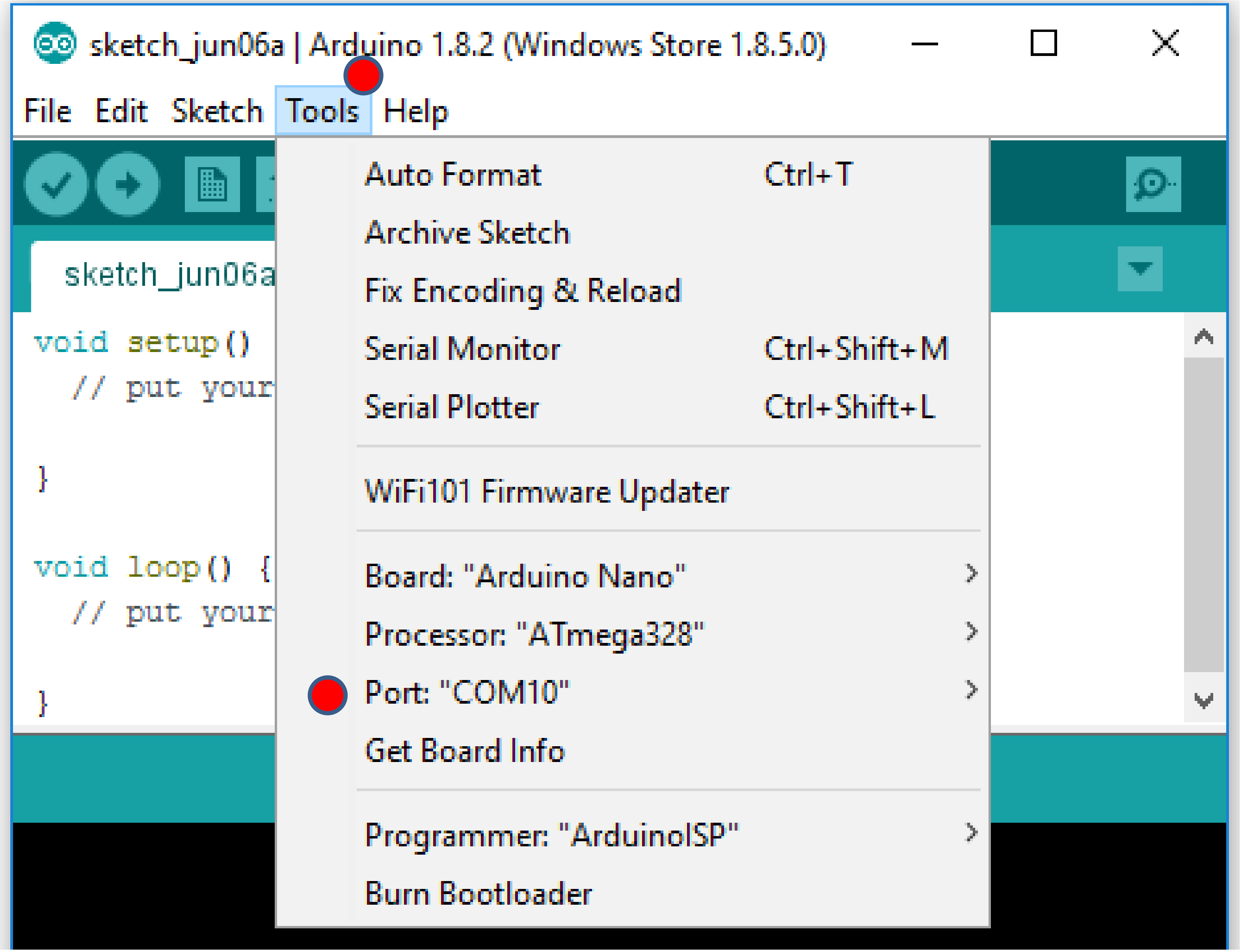
Auto Format	Ctrl+T
Archive Sketch	
Fix Encoding & Reload	
Serial Monitor	Ctrl+Shift+M
Serial Plotter	Ctrl+Shift+L
WiFi101 Firmware Updater	
Board: "Arduino Nano"	›
● Processor: "ATmega328"	›
Port: "COM10"	›
Get Board Info	
Programmer: "ArduinoISP"	›
Burn Bootloader	

```
void setup()
  // put your
}

void loop() {
  // put your
}
```

Choosing the processor type



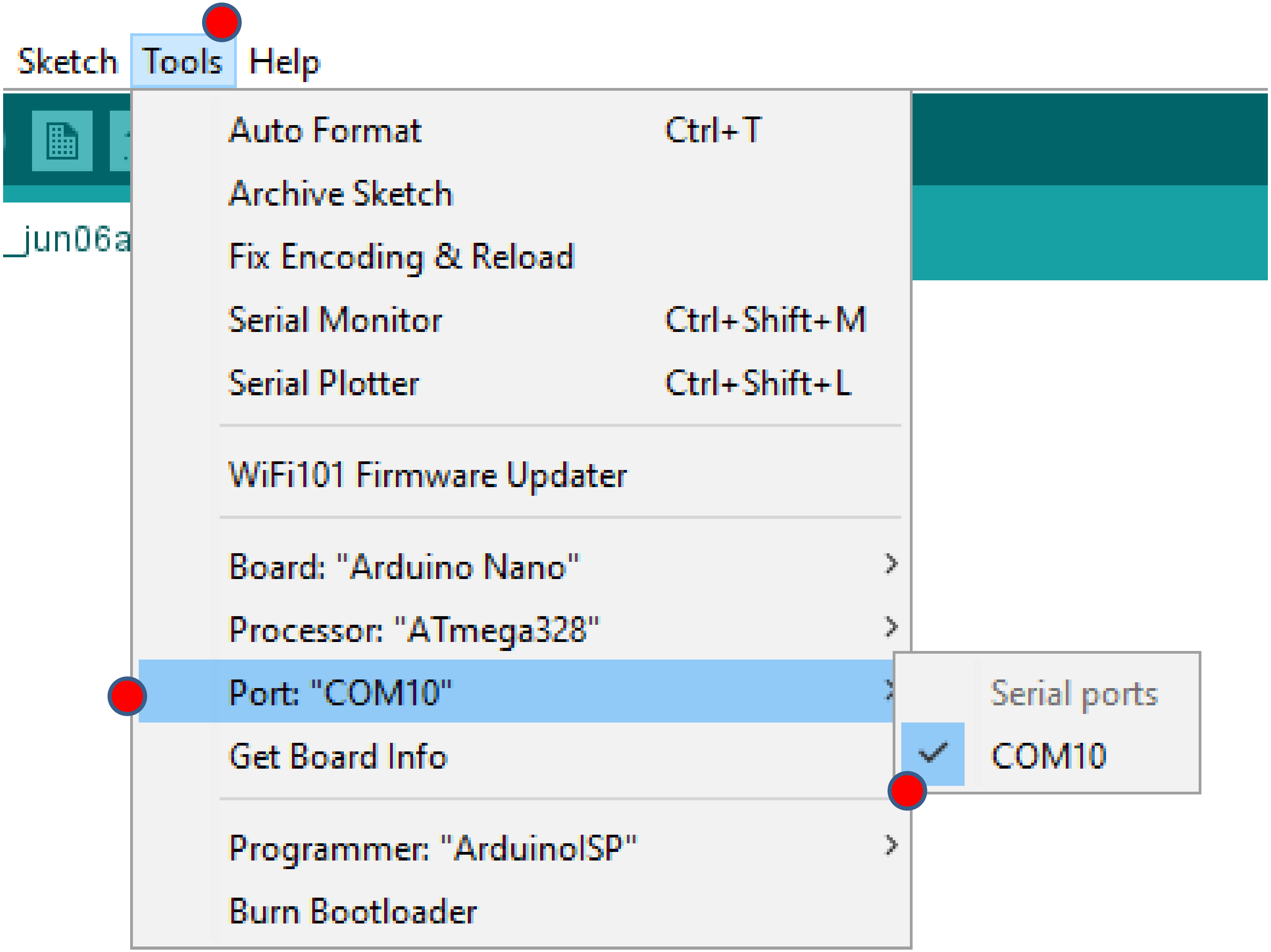
Tools Menu



sketch_jun06a | Arduino 1.8.2 (Windows Store 1.8.5.0)
File Edit Sketch Tools Help
sketch_jun06a
void setup()
// put your
}
void loop() {
// put your
}

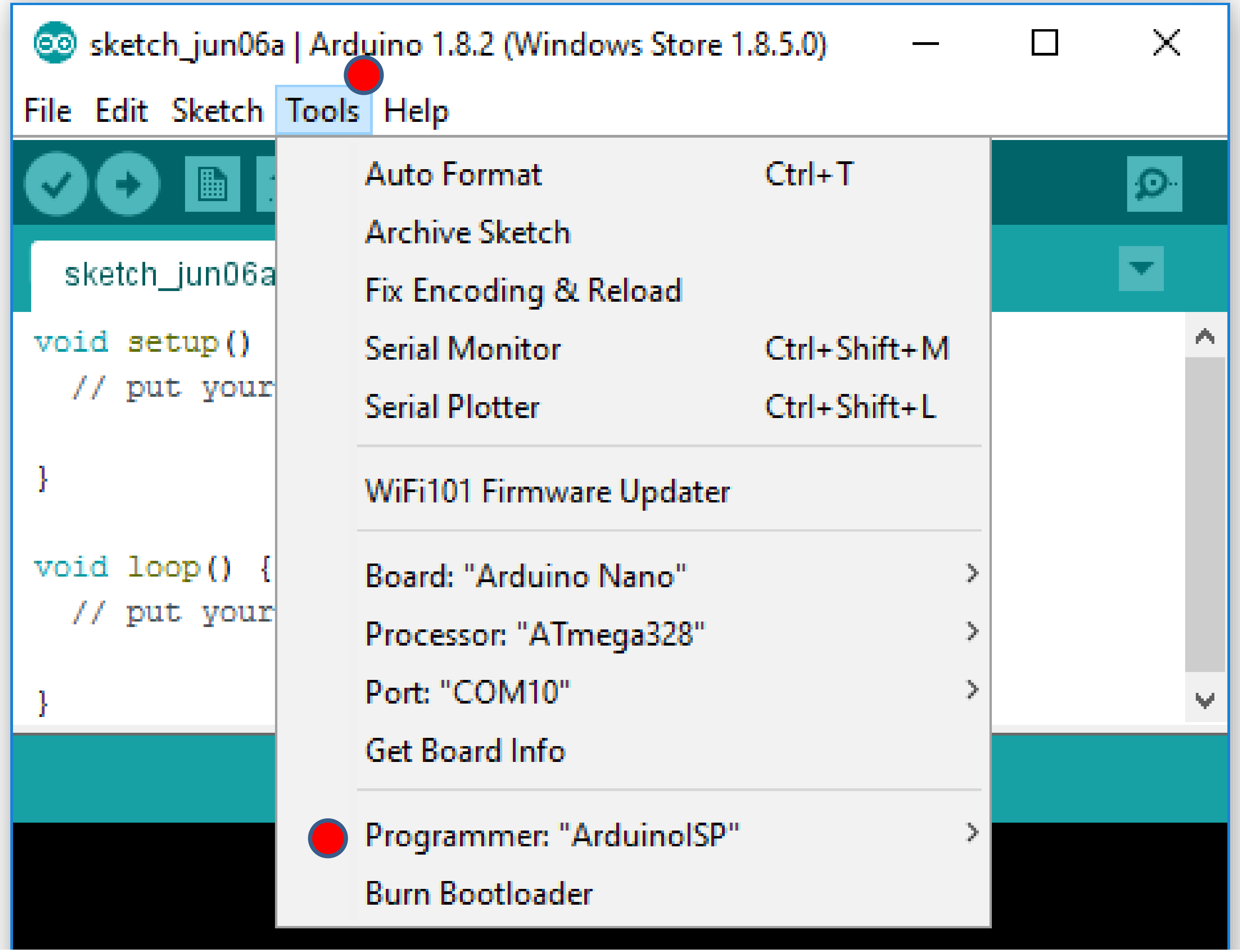
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 Firmware Updater
Board: "Arduino Nano"
Processor: "ATmega328"
Port: "COM10"
Get Board Info
Programmer: "ArduinoISP"
Burn Bootloader

Choosing the USB COM port



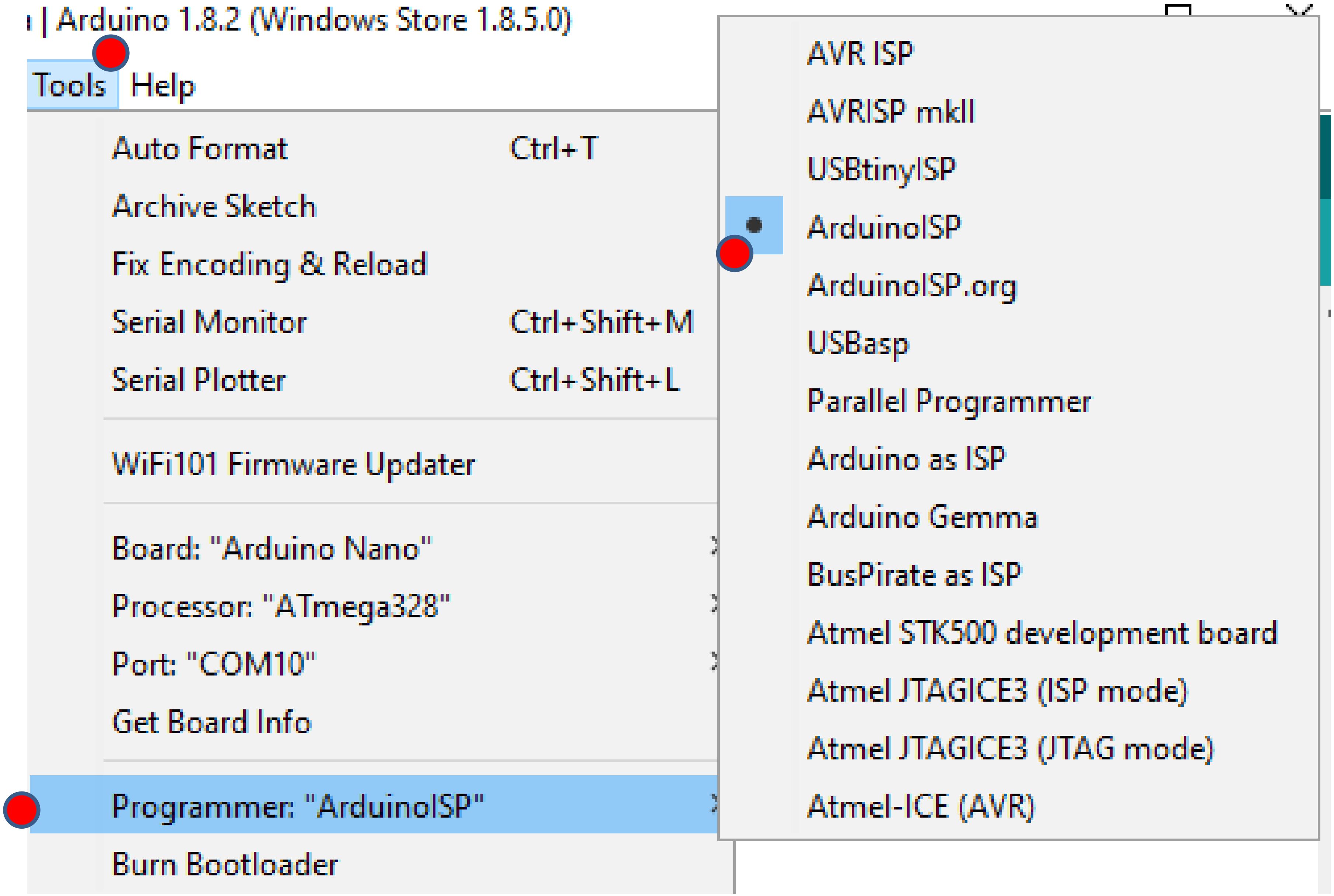
Board: "Arduino Nano"
Processor: "ATmega328"
Port: "COM10"
Programmer: "ArduinoISP"

Tools Menu

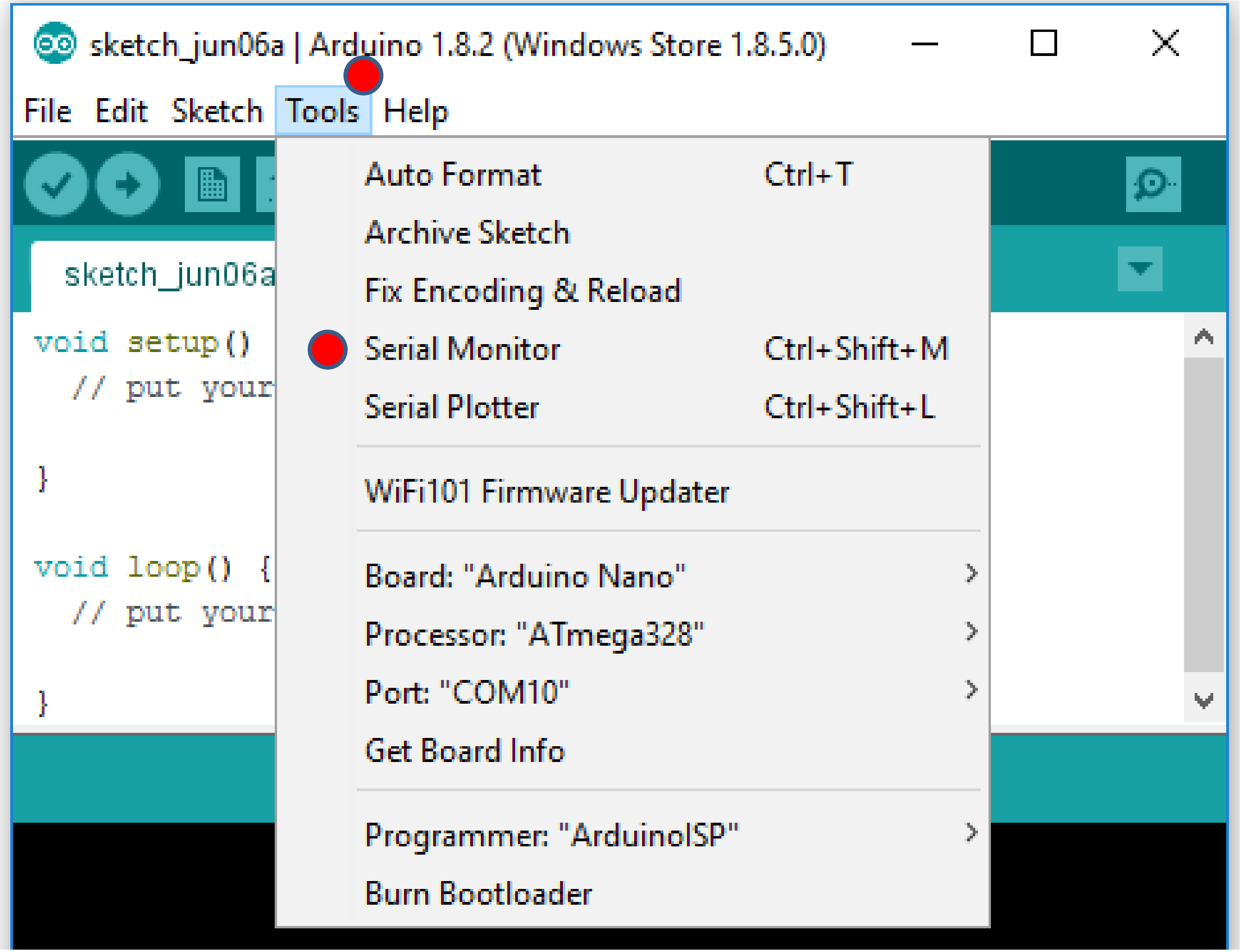


```
void setup()
  // put your

}

void loop() {
  // put your

}
```

Choose programming method



Tools Menu



Board: "Arduino Nano"

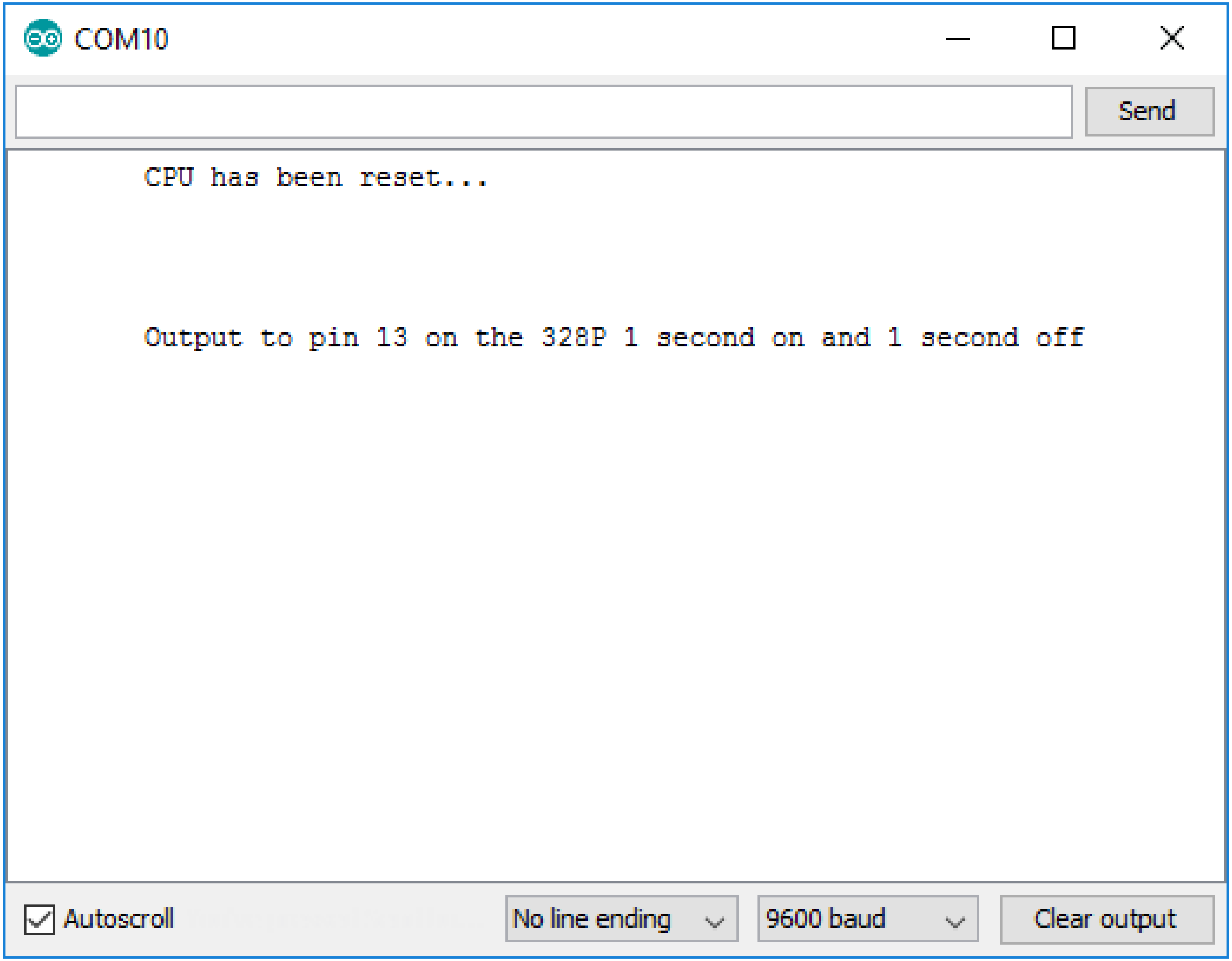
Processor: "ATmega328"

Port: "COM10"

Programmer: "ArduinoISP"

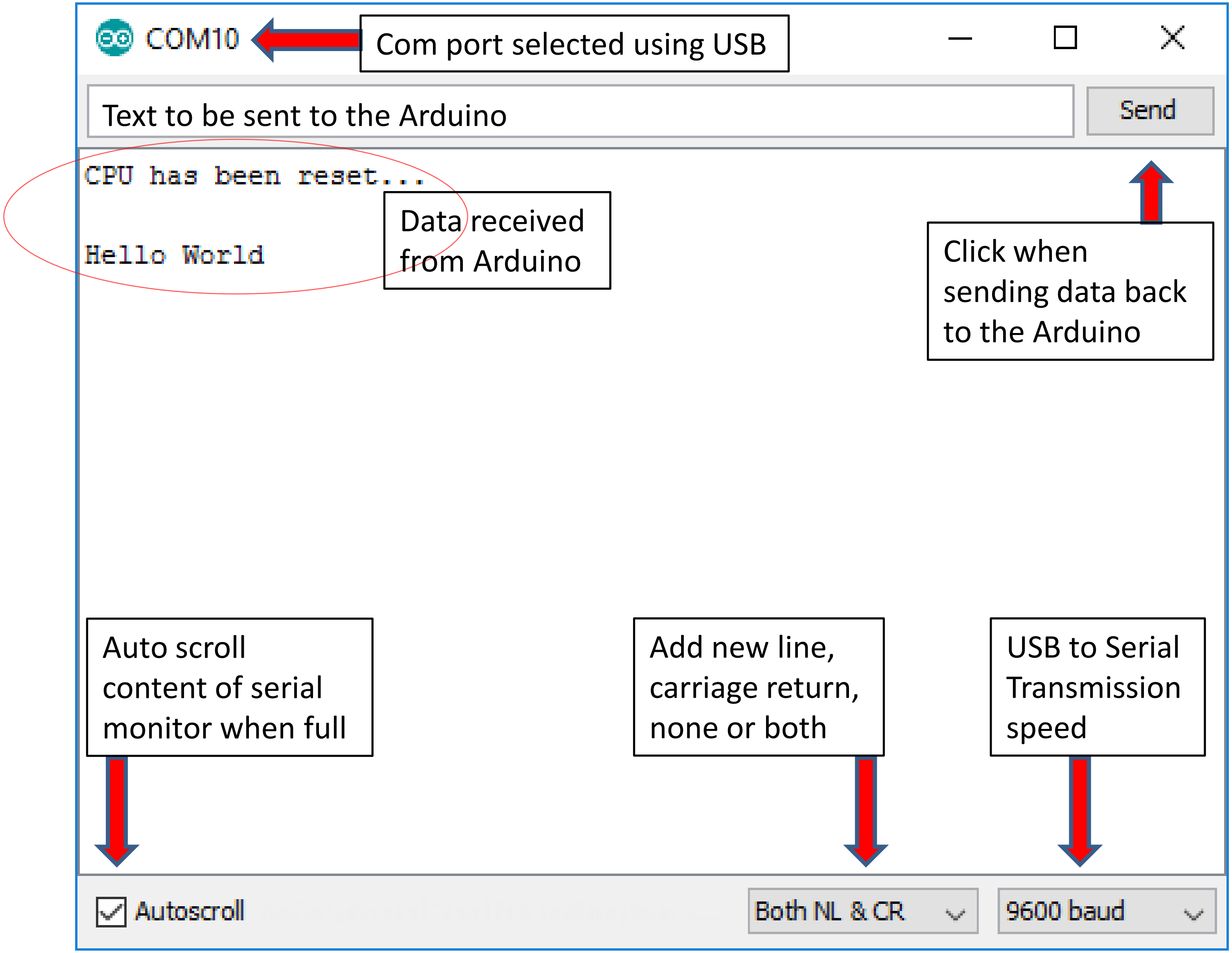
```
void setup()
  // put your

}

void loop() {
  // put your

}
```

Serial Monitor



```
CPU has been reset...



Output to pin 13 on the 328P 1 second on and 1 second off
```

Serial Monitor



Com port selected using USB

Text to be sent to the Arduino

```
CPU has been reset...

Hello World
```

Data received from Arduino

Click when sending data back to the Arduino

Auto scroll content of serial monitor when full

Add new line, carriage return, none or both

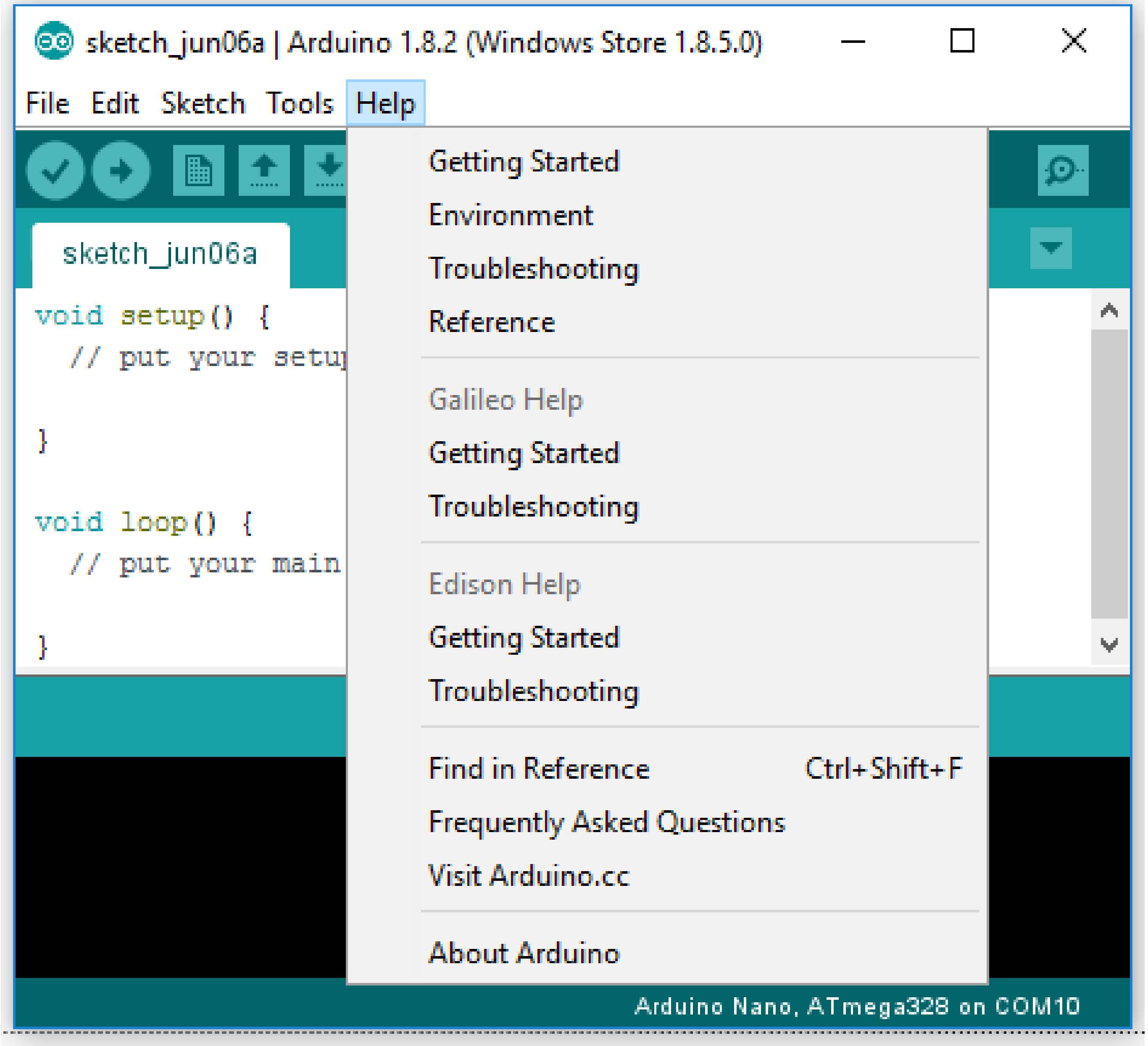
USB to Serial Transmission speed

☑ Autoscroll

Both NL & CR

9600 baud

Help Menu



```
void setup() {
  // put your setu

}

void loop() {
  // put your main

}
```

Arduino Nano, ATmega328 on COM10

Program format in Arduino C

Arduino programs can be divided in four main parts:

comments

structure

values - variables and constants

functions

comments

Comments:

// This is a 1 line comment in C and should be used liberally

/* This is a multi line comment in C and can also be used to disable a block of code temporarily for debugging */

Structure:

setup()

loop()

The structure can also have many functions called from the loop.

Program format in Arduino C

Values – Constants and Variables:

/ warning: C is case sensitive and many programs have failed to function because of a typo in a name e.g. ledpin instead of ledPin */*

Constants:

HIGH | LOW

INPUT | OUTPUT | INPUT_PULLUP

LED_BUILTIN

true | false

integer constants

floating point constants

Also #define constants (e.g. #define MAX_LIMIT 100)

Program format in Arduino C

Variables:

You can create variables, preferably using an invented name that makes it understandable to others. Self documenting code in a sense.

It is surprising how often code written by someone, who then later wishes to modify the program. Only to find they have to spend hours familiarising themselves with the code simply because of poor variable naming and an absence of good comments. Here endeth the sermon ☺

Program format in Arduino C

Variable data types:

Void
Boolean
Char
unsigned char
Byte
Int
unsigned int
word
long
unsigned long
short
float
double

string - char array
String - object
array

Program format in Arduino C

Functions:

Digital I/O

- pinMode()
- digitalWrite()
- digitalRead()

Analog I/O

- analogReference()
- analogRead()
- analogWrite() - PWM

Advanced I/O

- tone()
- noTone()
- shiftOut()
- shiftIn()
- pulseIn()

Time

- millis()
- micros()
- delay()
- delayMicroseconds()

Math

- min()
- max()
- abs()
- constrain()
- map()
- pow()
- sqrt()

Trigonometry

- sin()
- cos()
- tan() Characters

Characters

- isAlphaNumeric()
- isAlpha()
- isAscii()
- isWhitespace()
- isControl()
- isDigit()
- isGraph()
- isLowerCase()
- isPrintable()
- isPunct()
- isSpace()
- isUpperCase()
- isHexadecimalDigit()

Random Numbers

- randomSeed()
- random()

Program format in Arduino C

More Functions:

Bits and Bytes

- lowByte()
- highByte()
- bitRead()
- bitWrite()
- bitSet()
- bitClear()
- bit()

External Interrupts

- attachInterrupt()
- detachInterrupt()

Interrupts

- interrupts()
- noInterrupts()

Communication

- Serial
- Stream

Program format in Arduino C

That Semicolon; ☹

I guess something has to define the end of a statement but …..

Places where semicolons are required:-

After a variable declaration: int a_number = 112;

After a statement: delay(a_number);

Inside a complex **for** loop: for (pos = 0, x = 10, y = 5, z = 0; pos <= 180; pos++, z += 2) { stuff }

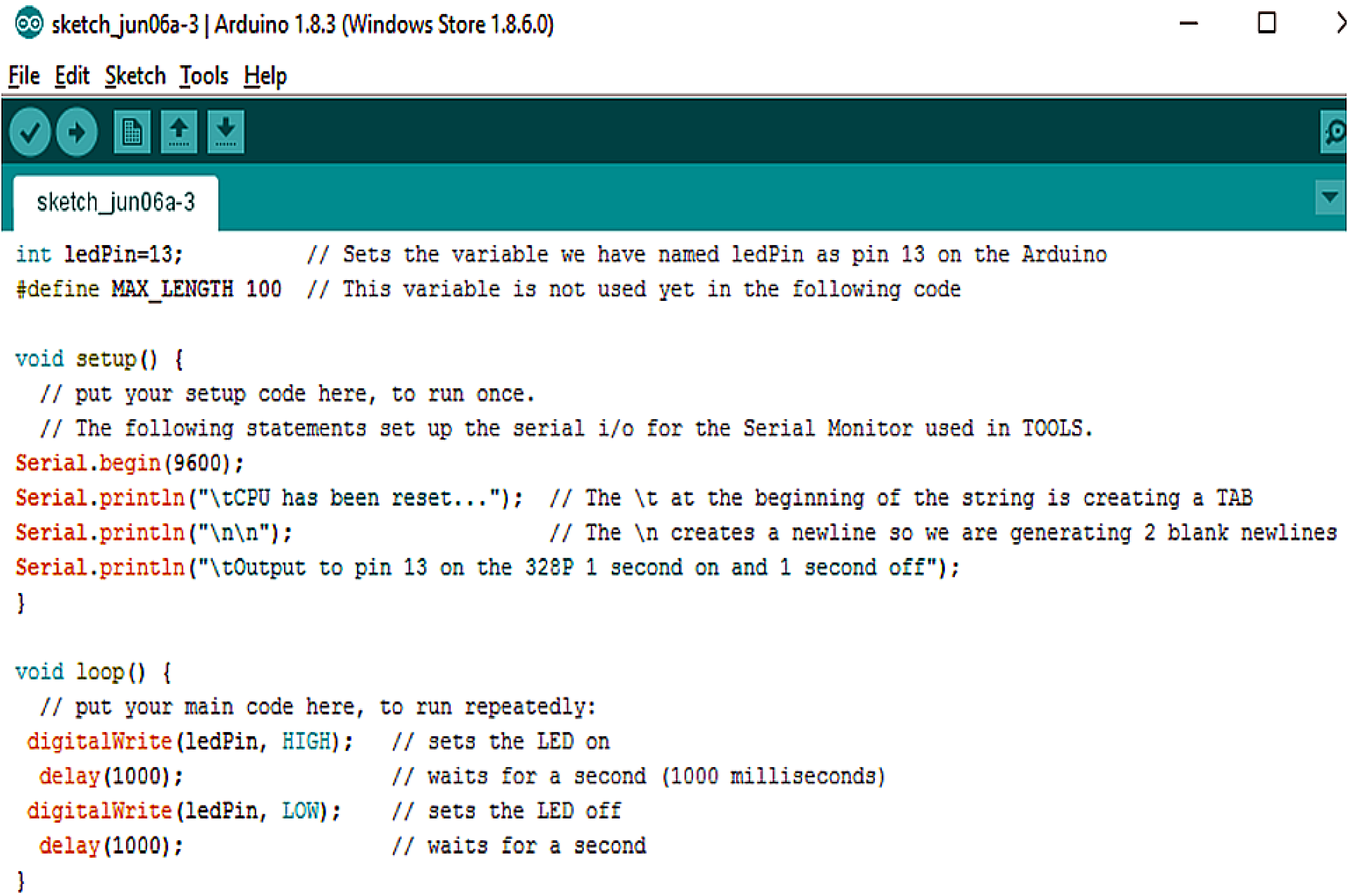
Places where semicolons are NOT required:-

After a function or curly brackets: ***sum_numbers() { do stuff; more stuff; even more; }***

After a #define NUM_CALC 200

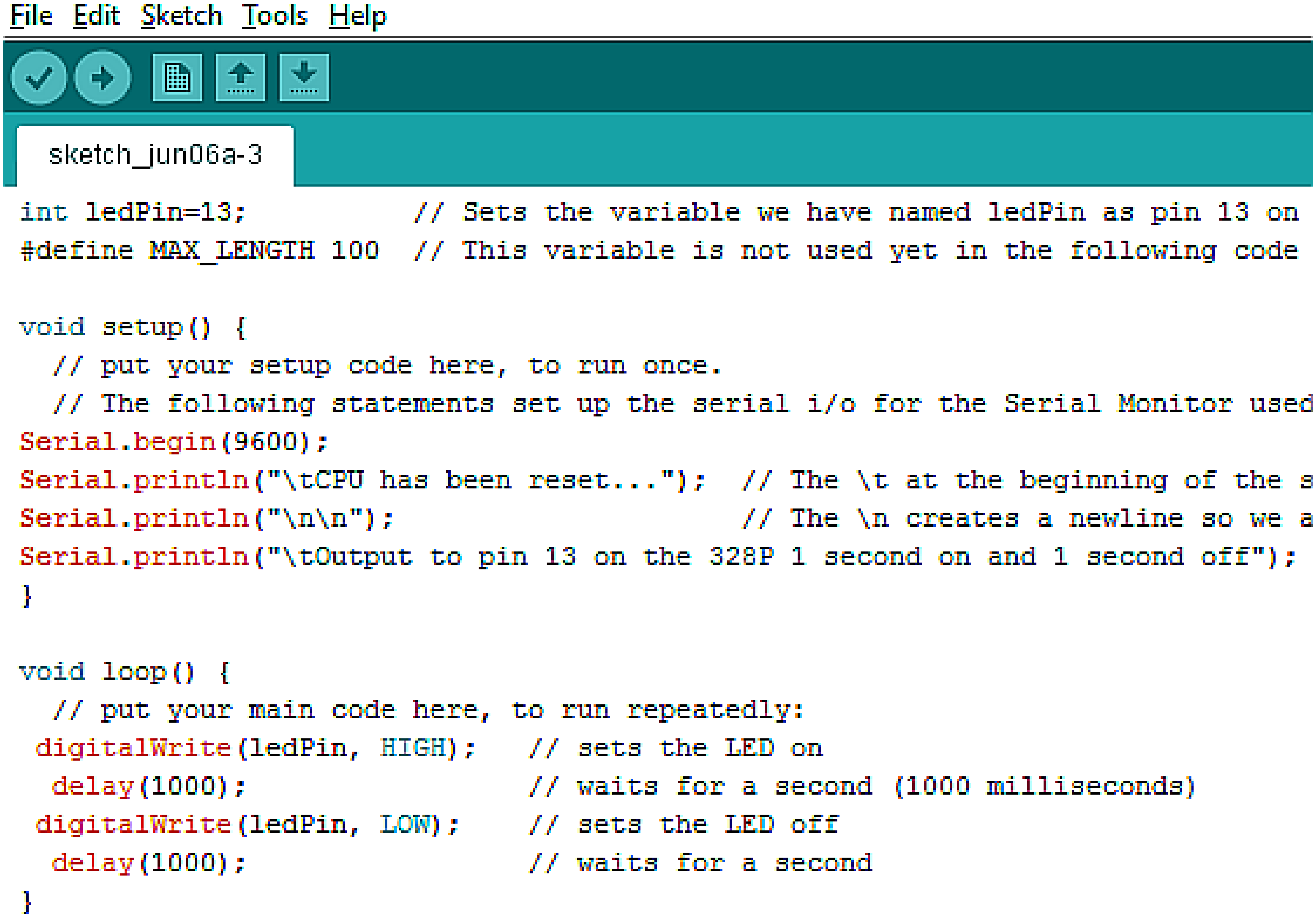
After a #include #ifndef and #define HeaderFileX_h

Program example


sketch_jun06a-3 | Arduino 1.8.3 (Windows Store 1.8.6.0)
File Edit Sketch Tools Help
sketch_jun06a-3

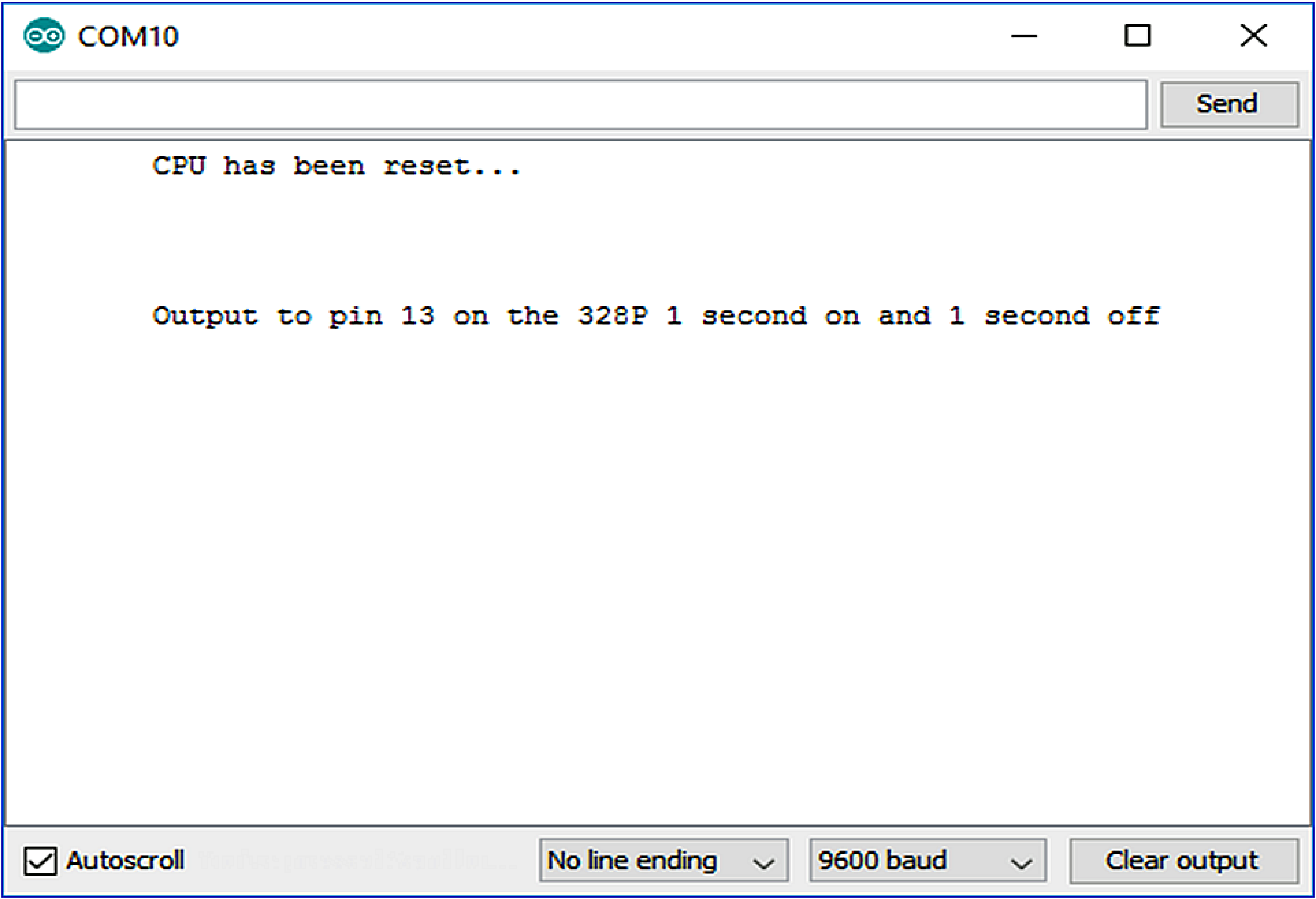
```
int ledPin=13;          // Sets the variable we have named ledPin as pin 13 on the Arduino
#define MAX_LENGTH 100  // This variable is not used yet in the following code

void setup() {
  // put your setup code here, to run once.
  // The following statements set up the serial i/o for the Serial Monitor used in TOOLS.
Serial.begin(9600);
Serial.println("\tCPU has been reset...");   // The \t at the beginning of the string is creating a TAB
Serial.println("\n\n");                      // The \n creates a newline so we are generating 2 blank newlines
Serial.println("\tOutput to pin 13 on the 328P 1 second on and 1 second off");
}

void loop() {
  // put your main code here, to run repeatedly:
 digitalWrite(ledPin, HIGH);   // sets the LED on
  delay(1000);                 // waits for a second (1000 milliseconds)
 digitalWrite(ledPin, LOW);    // sets the LED off
  delay(1000);                 // waits for a second
}
```

Program example



```
int ledPin=13;          // Sets the variable we have named ledPin as pin 13 on
#define MAX_LENGTH 100  // This variable is not used yet in the following code

void setup() {
  // put your setup code here, to run once.
  // The following statements set up the serial i/o for the Serial Monitor used
Serial.begin(9600);
Serial.println("\tCPU has been reset...");  // The \t at the beginning of the s
Serial.println("\n\n");                     // The \n creates a newline so we a
Serial.println("\tOutput to pin 13 on the 328P 1 second on and 1 second off");
}

void loop() {
  // put your main code here, to run repeatedly:
 digitalWrite(ledPin, HIGH);   // sets the LED on
  delay(1000);                 // waits for a second (1000 milliseconds)
 digitalWrite(ledPin, LOW);    // sets the LED off
  delay(1000);                 // waits for a second
}
```

Serial Monitor

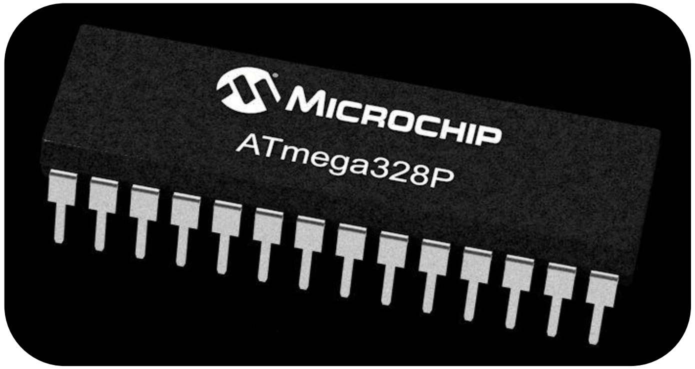


```
CPU has been reset...



Output to pin 13 on the 328P 1 second on and 1 second off
```

The Arduino IDE and coding in C (end of part 1)

Thank you for attending this presentation

Latest version ARDUINO IDE 1.8.3 can be downloaded from:
https://www.arduino.cc/en/Main/Software



Presentation by Eric S. Clarke 09/06/2017