**Outline**

Learn the Arduino web based IDE and basic Arduino programming by implementing the basic blink program and modifying it to blink external LEDs.

**Objectives**

* Use constants and variables,
* Explain the difference between syntax, logic, and run-time errors in computer programs;
* Demonstrate the ability to correct syntax, logic, and run-time errors in computer programs;
* Design a simple program from a program template or skeleton (e.g., teacher-supplied skeleton, Help facility code snippet);
* Use Help documentation as a guide to designing and writing programs.
* Use the features of a software development environment to debug programs and create functioning computer programs;
* Work independently, using the Help function, to resolve syntax issues while programming;
* Work independently, using reference materials (e.g., code snippets, sample programs, APIs, tutorials), to design and write functioning computer programs.
* Describe the functions and features of a software development environment and use it to write and run a computer program;

**Materials**

* Arduino Web Development Environment (IDE)

**Level 0: Create an Arduino Account**

On-line Resource: <https://create.arduino.cc/>

NOTE: Always use Chrome for the Web IDE.

1. Go to the website and Select “Arduino Web Editor” and “Create a New Account”. Follow instructions.
2. Check your personal email to confirm your account. Then login and Select “Arduino Web Editor” and agree to terms & conditions (read this first).
3. Note: DO NOT install the plug-in if requested to do so. (Just cancel all dialog boxes.)

**Level 0**

Complete

**Level 1: Understanding the Blink Example**

1. Follow the editor tour and the tutorial at: <https://create.arduino.cc/projecthub/Arduino_Genuino/getting-started-with-the-arduino-web-editor-4b3e4a>
2. Select the menu item “Examples” → “Basic” → “Blink”.
3. Make sure the sketch code is loaded into the editor window.

NOTE: The following steps must be done using the teacher laptop at the front of the classroom.

1. Connect the Arduino board to your PC using the USB cable.
2. Log in to your Arduino Create account.
3. Upload and run the Blink sketch code on the Arduino board.

**Level 1**

**Complete**

**Level 2: Using External Documentation**

On-Line Resource: <https://www.arduino.cc/en/Reference/HomePage>

1. Read the on-line documentation to become familiar with the Arduino IDE and command language.
2. Identify each program command and program statement in your blink program.

The functions in the program are digitalWrite(), pinMode(), setup(), loop(), void and delay.

1. Use the on-line documentation to make notes on each command so you understand and can explain every part of your program.

digitalWrite()

* It is when you select a pin to be used and give it an output of high or low. This is if it is configured as an output.
* If configured was a input, it will enable (high) or disable (low) the internal pull up in the input pin.

pinMode()

* It would configure a pin as a input or output.

delay

* It would pause a program for the amount of time in milliseconds which is set in the parameter.
* 1000 milliseconds in a second.

Setup()

* This is a function called to start the sketch and can only run once.

Loop()

* Loops parts of the program.

Void

* Void is used only in function declarations.

1. Explain the difference between a constant and a variable.

* Constant is a qualifier variable that would change a variable. It is defined as const and you cannot assign a variable to a constant variable. The value cannot be changed.
* Variables are a data type that stores a value and it is likely to change as the program runs.

1. Explain the following:
   1. Syntax error

* Syntax error is an incorrect character or string.
  1. Logic error
* Logic error is when there is a mistake in the program’s code that is incorrect or unexpected.
  1. Run-time error
* Run-time error is when there is an error that occurs while the program is running.

**Level 3: Extend the Blink Pattern**

1. Modify the blink program to produce two short blinks followed by one long blink.
2. Use a variable to control the blink time.
3. Upload and run the Extended Blink sketch code on the Arduino board and show it to your teacher.

**Level 3 Code**

void setup() {

pinMode(LED\_BUILTIN, OUTPUT);

}

void loop() {

digitalWrite(LED\_BUILTIN, HIGH);

delay(200);

digitalWrite(LED\_BUILTIN, LOW);

delay(200);

digitalWrite(LED\_BUILTIN, HIGH);

delay(200);

digitalWrite(LED\_BUILTIN, LOW);

delay(200);

digitalWrite(LED\_BUILTIN, HIGH);

delay(1000);

digitalWrite(LED\_BUILTIN, LOW);

delay(200);

}

**Level 4: Add External LED**

Useful resources in the “Arduino” folder on the class repository:

* + Getting started guide.pdf
  + Public\_materials→Ebook→Arduino book.pdf
  + Lessons→Lesson1-LED blink

1. Add a second external LED (using components on the prototype board) and make it blink in sync with the on-board LED.
2. Modify your program so that the external LED blinks following a different pattern than your on-board LED.
3. Upload and run the External LED Blink sketch code on the Arduino board and show it to your teacher.

**Level 4 Code**

Question 1

void setup() {

pinMode(LED\_BUILTIN, OUTPUT);

pinMode(13, OUTPUT)

}

void loop() {

digitalWrite(LED\_BUILTIN, HIGH);

digitalWrite(13, HIGH)

delay(200);

digitalWrite(LED\_BUILTIN, LOW);

digitalWrite(13, LOW)

delay(200);

digitalWrite(LED\_BUILTIN, HIGH);

digitalWrite(13, HIGH)

delay(200);

digitalWrite(LED\_BUILTIN, LOW);

digitalWrite(13, LOW)

delay(200);

digitalWrite(LED\_BUILTIN, HIGH);

digitalWrite(13, HIGH)

delay(1000);

digitalWrite(LED\_BUILTIN, LOW);

digitalWrite(13, LOW)

delay(200);

}

Question 2

void setup() {

pinMode(LED\_BUILTIN, OUTPUT);

pinMode(12, OUTPUT);

}

void loop()

{

digitalWrite(LED\_BUILTIN, HIGH);

delay(200);

digitalWrite(12, HIGH);

digitalWrite(LED\_BUILTIN, LOW);

delay(100);

digitalWrite(12, LOW);

digitalWrite(LED\_BUILTIN, HIGH);

delay(200);

digitalWrite(LED\_BUILTIN, LOW);

digitalWrite(12, HIGH);

delay(200);

digitalWrite(LED\_BUILTIN, HIGH);

digitalWrite(12, LOW);

delay(1000);

digitalWrite(LED\_BUILTIN, LOW);

delay(200);

}