# Exercise 1: Use a LED for Morse code

## Equipment

For this exercise you will need:

- 1 x Arduino Uno
- 1 x LED
- 1 x Resistor ($\sim 60\Omega - 220\Omega$)
- Wires

Remember to select your USB port.
Tools→Port

## Reading

Chapter 2 - 5

## Setup

Open the "Blink" program for inspiration (In Arduino IDE: File → Examples → 01.Basics → Blink).

- Connect the LED to any digital pin on the Uno. Use a resistor to limit the current going through the LED.
- Use pinmode(PIN, OUTPUT) to setup the pin for output (Change PIN for the name of the digital pin you picked).
- Define your constants
  const uint8_t ledPin = <value>; //e.g. D6
  const int timeUnit = <value>; //e.g. 500

## Questions & Exercises

**1a:** Morse code uses 5 "bits" for representing numbers 0-9, how many bits are needed if you would use binary? Why is this not possible in Morse code?
**1b:** What is the value of **a** after the loop?

```
int a = 1;
for(int i = 0; i < 5; i++){
    a += a;
}
```

Now it is time to make a program that executes Morse Code using the LED.
**1c:** Make a program that morses "SOS". You should follow the international requirement for morse code shown in figure 1.
**1d:** Try setting ledPin = LED_BUILTIN, what happens?
**1e:** Update the program to morse your name
**1f:** Finally, update the program to morse your name using for-loops

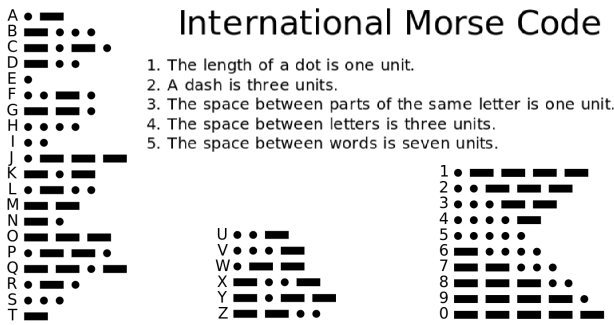


Figure 1: Morse code

**FOR-loops**

For-loops are useful for repeating something a known number of times.

```
for(int i = 0; i < 10; i++){
 //this loop will run ten times
 }
```

## Hint

Use digitalWrite(ledPin, HIGH) to turn on the LED the wait some time with delay(timeUnit) do the same to turn a LED off, just set the pin LOW instead of HIGH