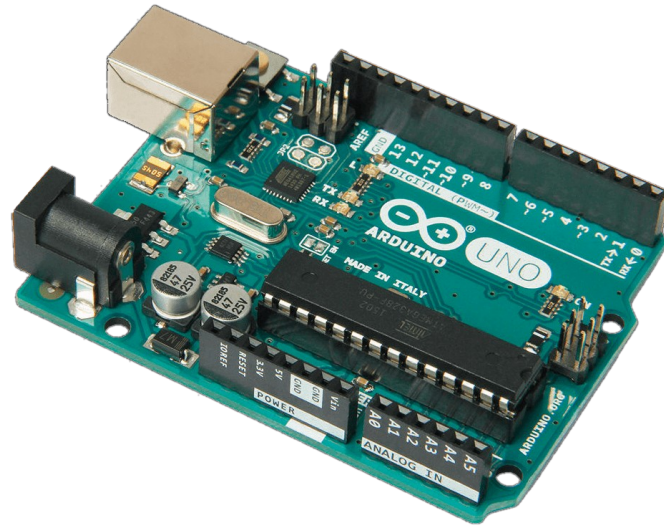
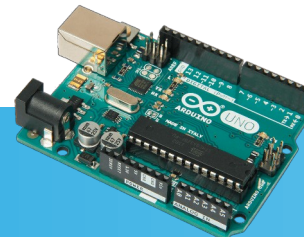
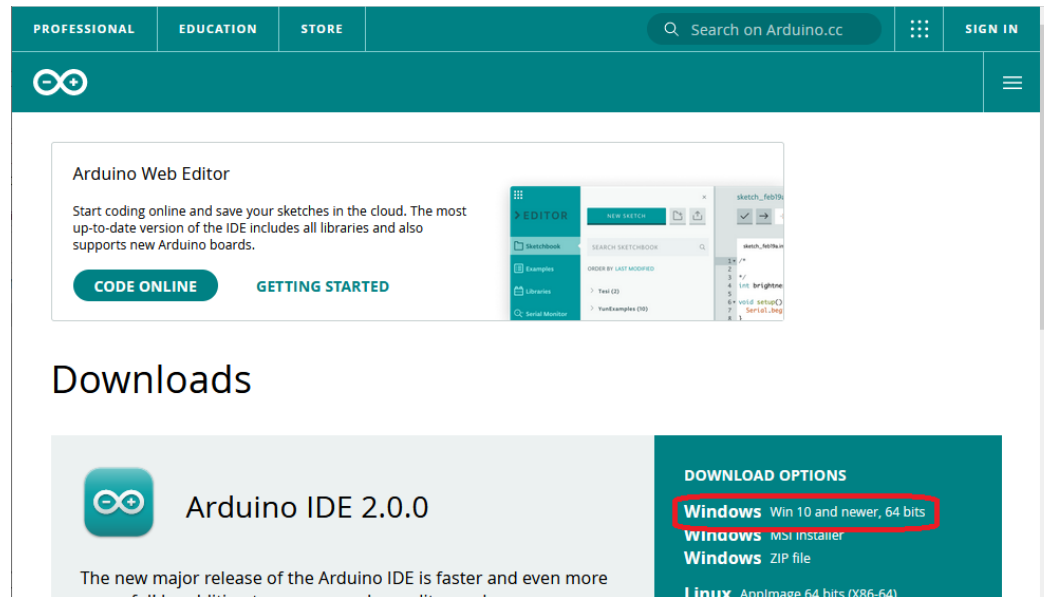


# Arduino workshop #1



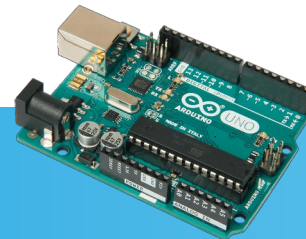
# Installere Arduino udviklingsmiljø

- Gå til <https://www.arduino.cc/>
- Vælg "software" fra top-menuen, og download IDE 2.0.0



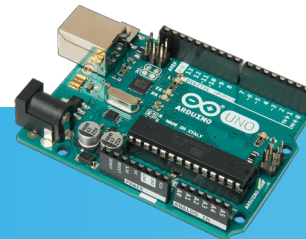
# Udfør installationen

- Når download er færdig startes installationsprogrammet
- Acceptér licensen (I Agree)
- Vælg "Anyone who uses this computer (all users)", tryk "Next" og "Install"
- Når installationen er færdig kan man starte Arduino IDE



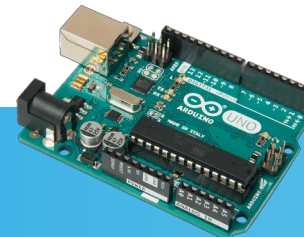
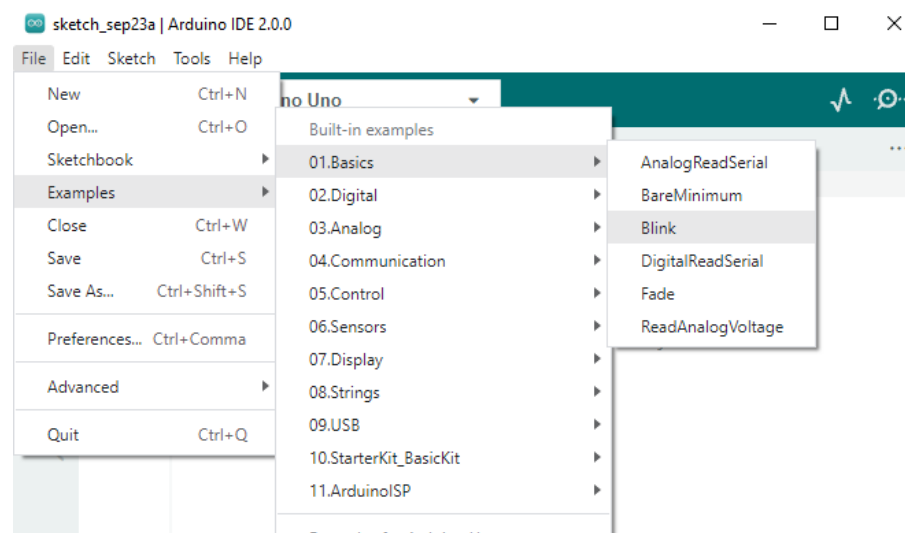
# Forbind Arduino Uno

- Tilslut USB-kabel til Arduino og til PC
- Vælg typen af Arduino:
  - Menuen Tools → Board → Arduino AVR Boards → Arduino Uno
- Vælg COM-port hvor arduino er tilsluttet
  - I menuen Tools → Port vælges den COM-port hvor der står Arduino Uno.



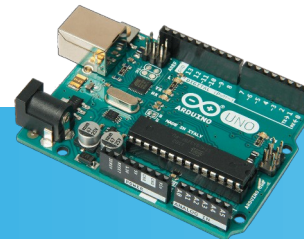
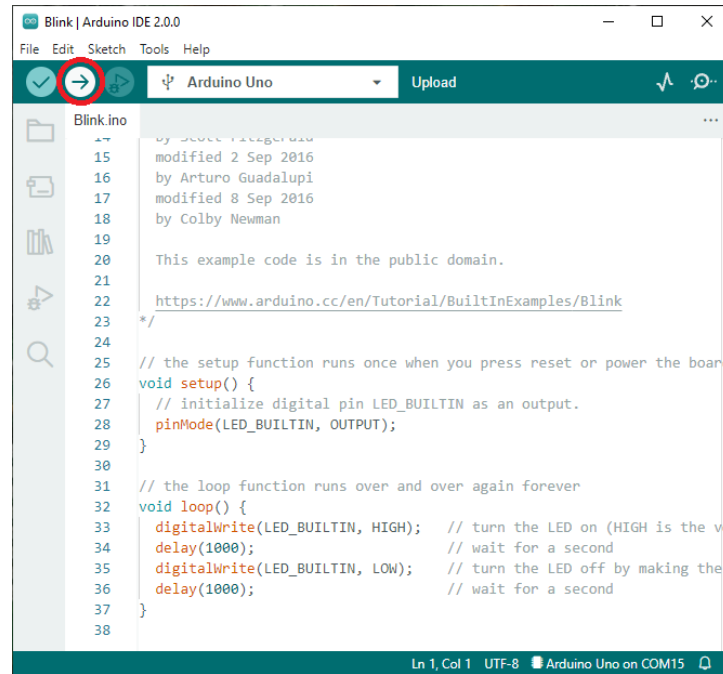
# Første sketch - Blink

- Åbn Blink-eksemplet
  - File → Examples → 01.Basics → Blink



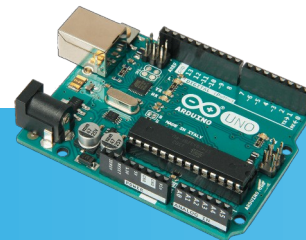
# Upload Blink

- Blink-sketch'en åbner i et nyt vindue. Tryk på upload:



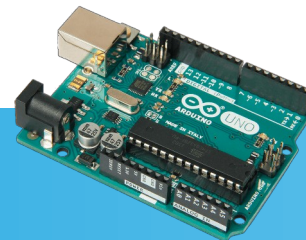
# Læs koden til Blink

- Kommentarer er blokke der starter med `/*` og slutter med `*/`
- Linjer med dbl. skråstreg `//` er resten af linjen kommentar
- Arduino-sketches indeholder (mindst) to funktioner: `setup` og `loop`.
- `setup` kører en gang ved opstart (tænd/reset) af Arduino
- Derefter kører `loop` igen og igen
- Man kan ikke rigtig "afslutte" et Arduino-program



# Funktioner i Arduino / C

- Funktionsdeklaration:
  - retur-type funktionsnavn(argumenter) { funktions-kode }
  - Fx: void setup() { pinMode(LED\_BUILTIN, OUTPUT); }
  - Retur-typen er void
  - Funktionsnavnet er setup
  - Funktionen tager ingen argumenter
  - Funktionen har en enkelt linje kode, som kalder pinMode

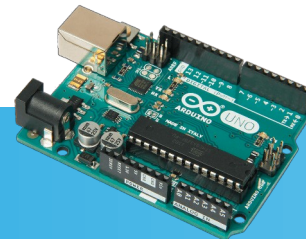




# setup()

- Hvad tror I setup-funktionen i Blink-eksemplet gør?

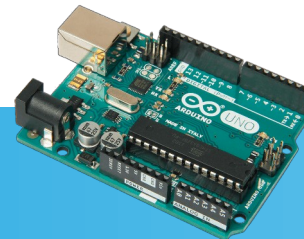
```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}
```



# loop()

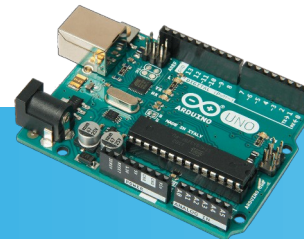
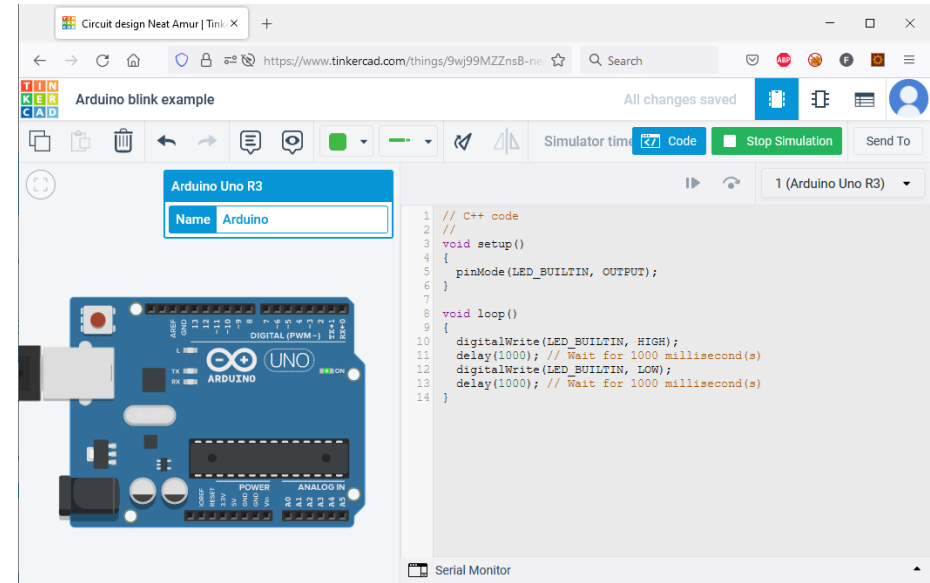
- Hvad tror I loop-funktionen i Blink-eksemplet gør?

```
// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);                     // wait for a second
  digitalWrite(LED_BUILTIN, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);                     // wait for a second
}
```



# Arduino i TinkerCAD

- Man kan eksperimentere med Arduino i TinkerCAD
- Uden at have et board
- Kræver at man opretter login



# Arduino-reference

- Arduino reference: <https://www.arduino.cc/reference/en/>
- Prøv at markere (dobbelklik på) funktionen **digitalWrite** i eksemplet, og tryk på Ctrl+Shift+F

## Functions

For controlling the Arduino board and performing computations.

### Digital I/O

`digitalRead()`  
`digitalWrite()`  
`pinMode()`

### Analog I/O

`analogRead()`  
`analogReference()`  
`analogWrite()`

### Math

`abs()`  
`constrain()`  
`map()`  
`max()`  
`min()`  
`pow()`  
`sq()`  
`sqrt()`

## digitalWrite()

[Digital I/O]

### Description

Write a HIGH or a LOW value to a digital pin.

If the pin has been configured as an OUTPUT with `pinMode()`, its voltage will be set to the corresponding value: 5V (or 3.3V on 3.3V boards) for HIGH, 0V (ground) for LOW.

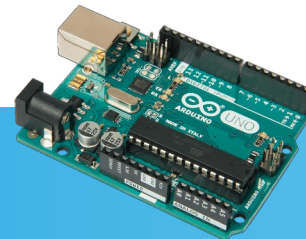
If the pin is configured as an INPUT, `digitalWrite()` will enable (HIGH) or disable (LOW) the internal pullup on the input pin. It is recommended to set the `pinMode()` to `INPUT_PULLUP` to enable the internal pull-up resistor. See the [Digital Pins](#) tutorial for more information.

If you do not set the `pinMode()` to OUTPUT, and connect an LED to a pin,



# Ændre Blink-sketch'en

- Prøv at ændre Blink-sketch'en, så lysdioden blinker 5 gange i sekundet (altså 10 x hurtigere end oprindeligt)
- Når du ændrer i et eksempel skal du gemme sketch'en under et nyt navn
- Til de hurtige: Skriv en sketch, der får lysdioden til at morse SOS
  - S=prik-prik-prik, O=streg-streg-streg
  - Fx prik=300 msec, streg=1000 msec

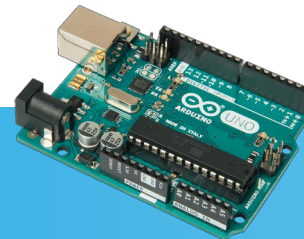


# Eksempel ASCIITable

- Åbn eksemplet ASCIITable
  - File → Examples → 04.Communications → ASCIITable
- Upload sketch til Arduino
- Åbn Serial Monitor (fra menuen Tools eller Ctrl+Shift+M)
- Læs koden og prøv at forstå hvad der sker

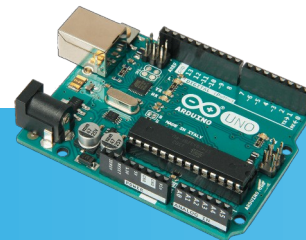
ASCII Table ~ Character Map

```
!, dec: 33, hex: 21, oct: 41, bin: 100001
", dec: 34, hex: 22, oct: 42, bin: 100010
#, dec: 35, hex: 23, oct: 43, bin: 100011
$, dec: 36, hex: 24, oct: 44, bin: 100100
%, dec: 37, hex: 25, oct: 45, bin: 100101
&, dec: 38, hex: 26, oct: 46, bin: 100110
', dec: 39, hex: 27, oct: 47, bin: 100111
```



# Serial.print og println

- Opret en ny sketch (File → New)
- Indsæt Serial.Begin(9600) i setup
- Brug Serial.println i loop-funktionen til at udskrive en tekst (fx dit navn) igen og igen
- Hvad er forskellen på Serial.print og Serial.println?



# Serial Plotter

- Lav en ny sketch
- Indtast programmet til højre
- Forklar hvad programmet gør
- Upload programmet på Arduino
- Check serial monitor
- Fra menuen Tools åbnes Serial Plotter

```
void setup() {  
  Serial.begin(9600);  
}  
  
byte tal = 0;  
void loop() {  
  Serial.println(tal, DEC);  
  tal = tal + 10;  
  delay(100);  
}
```

