**Language Reference**

Arduino programs can be divided in three main parts: *structure*, *values* (variables and constants), and *functions*.

|  |  |  |
| --- | --- | --- |
| **Structure**   * [setup](http://arduino.cc/en/Reference/Setup)() * [loop](http://arduino.cc/en/Reference/Loop)()   **Control Structures**   * [if](http://arduino.cc/en/Reference/If) * [if...else](http://arduino.cc/en/Reference/Else) * [for](http://arduino.cc/en/Reference/For) * [switch case](http://arduino.cc/en/Reference/SwitchCase) * [while](http://arduino.cc/en/Reference/While) * [do... while](http://arduino.cc/en/Reference/DoWhile) * [break](http://arduino.cc/en/Reference/Break) * [continue](http://arduino.cc/en/Reference/Continue) * [return](http://arduino.cc/en/Reference/Return) * [goto](http://arduino.cc/en/Reference/Goto)   **Further Syntax**   * [;](http://arduino.cc/en/Reference/SemiColon) (semicolon) * [{}](http://arduino.cc/en/Reference/Braces) (curly braces) * [//](http://arduino.cc/en/Reference/Comments) (single line comment) * [/\* \*/](http://arduino.cc/en/Reference/Comments) (multi-line comment) * [#define](http://arduino.cc/en/Reference/Define) * [#include](http://arduino.cc/en/Reference/Include)   **Arithmetic Operators**   * [=](http://arduino.cc/en/Reference/Assignment) (assignment operator) * [+](http://arduino.cc/en/Reference/Arithmetic) (addition) * [-](http://arduino.cc/en/Reference/Arithmetic) (subtraction) * [\*](http://arduino.cc/en/Reference/Arithmetic) (multiplication) * [/](http://arduino.cc/en/Reference/Arithmetic) (division) * [%](http://arduino.cc/en/Reference/Modulo) (modulo)   **Comparison Operators**   * [==](http://arduino.cc/en/Reference/If) (equal to) * [!=](http://arduino.cc/en/Reference/If) (not equal to) * [<](http://arduino.cc/en/Reference/If) (less than) * [>](http://arduino.cc/en/Reference/If) (greater than) * [<=](http://arduino.cc/en/Reference/If) (less than or equal to) * [>=](http://arduino.cc/en/Reference/If) (greater than or equal to)   **Boolean Operators**   * [&&](http://arduino.cc/en/Reference/Boolean) (and) * [||](http://arduino.cc/en/Reference/Boolean) (or) * [!](http://arduino.cc/en/Reference/Boolean) (not)   **Pointer Access Operators**   * [\* dereference operator](http://arduino.cc/en/Reference/Pointer) * [& reference operator](http://arduino.cc/en/Reference/Pointer)   **Bitwise Operators**   * [&](http://arduino.cc/en/Reference/BitwiseAnd) (bitwise and) * [|](http://arduino.cc/en/Reference/BitwiseAnd) (bitwise or) * [^](http://arduino.cc/en/Reference/BitwiseAnd) (bitwise xor) * [~](http://arduino.cc/en/Reference/BitwiseXorNot) (bitwise not) * [<<](http://arduino.cc/en/Reference/Bitshift) (bitshift left) * [>>](http://arduino.cc/en/Reference/Bitshift) (bitshift right)   **Compound Operators**   * [++](http://arduino.cc/en/Reference/Increment) (increment) * [--](http://arduino.cc/en/Reference/Increment) (decrement) * [+=](http://arduino.cc/en/Reference/IncrementCompound) (compound addition) * [-=](http://arduino.cc/en/Reference/IncrementCompound) (compound subtraction) * [\*=](http://arduino.cc/en/Reference/IncrementCompound) (compound multiplication) * [/=](http://arduino.cc/en/Reference/IncrementCompound) (compound division) * [&=](http://arduino.cc/en/Reference/BitwiseCompoundAnd) (compound bitwise and) * [|=](http://arduino.cc/en/Reference/BitwiseCompoundOr) (compound bitwise or) | **Variables**  **Constants**   * [HIGH](http://arduino.cc/en/Reference/Constants) | [LOW](http://arduino.cc/en/Reference/Constants) * [INPUT](http://arduino.cc/en/Reference/Constants) | [OUTPUT](http://arduino.cc/en/Reference/Constants)| [INPUT\_PULLUP](http://arduino.cc/en/Reference/Constants) * [true](http://arduino.cc/en/Reference/Constants) | [false](http://arduino.cc/en/Reference/Constants) * [integer constants](http://arduino.cc/en/Reference/IntegerConstants) * [floating point constants](http://arduino.cc/en/Reference/Fpconstants)   **Data Types**   * [void](http://arduino.cc/en/Reference/Void) * [boolean](http://arduino.cc/en/Reference/BooleanVariables) * [char](http://arduino.cc/en/Reference/Char) * [unsigned char](http://arduino.cc/en/Reference/UnsignedChar) * [byte](http://arduino.cc/en/Reference/Byte) * [int](http://arduino.cc/en/Reference/Int) * [unsigned int](http://arduino.cc/en/Reference/UnsignedInt) * [word](http://arduino.cc/en/Reference/Word) * [long](http://arduino.cc/en/Reference/Long) * [unsigned long](http://arduino.cc/en/Reference/UnsignedLong) * [short](http://arduino.cc/en/Reference/Short) * [float](http://arduino.cc/en/Reference/Float) * [double](http://arduino.cc/en/Reference/Double) * [string](http://arduino.cc/en/Reference/String) - char array * [String](http://arduino.cc/en/Reference/StringObject) - object * [array](http://arduino.cc/en/Reference/Array)   **Conversion**   * [char()](http://arduino.cc/en/Reference/CharCast) * [byte()](http://arduino.cc/en/Reference/ByteCast) * [int()](http://arduino.cc/en/Reference/IntCast) * [word()](http://arduino.cc/en/Reference/WordCast) * [long()](http://arduino.cc/en/Reference/LongCast) * [float()](http://arduino.cc/en/Reference/FloatCast)   **Variable Scope & Qualifiers**   * [variable scope](http://arduino.cc/en/Reference/Scope) * [static](http://arduino.cc/en/Reference/Static) * [volatile](http://arduino.cc/en/Reference/Volatile) * [const](http://arduino.cc/en/Reference/Const)   **Utilities**   * [sizeof](http://arduino.cc/en/Reference/Sizeof)() | **Functions**  **Digital I/O**   * [pinMode](http://arduino.cc/en/Reference/PinMode)() * [digitalWrite](http://arduino.cc/en/Reference/DigitalWrite)() * [digitalRead](http://arduino.cc/en/Reference/DigitalRead)()   **Analog I/O**   * [analogReference](http://arduino.cc/en/Reference/AnalogReference)() * [analogRead](http://arduino.cc/en/Reference/AnalogRead)() * [analogWrite](http://arduino.cc/en/Reference/AnalogWrite)() - *PWM*   **Due only**   * [analogReadResolution](http://arduino.cc/en/Reference/AnalogReadResolution)() * [analogWriteResolution](http://arduino.cc/en/Reference/AnalogWriteResolution)()   **Advanced I/O**   * [tone](http://arduino.cc/en/Reference/Tone)() * [noTone](http://arduino.cc/en/Reference/NoTone)() * [shiftOut](http://arduino.cc/en/Reference/ShiftOut)() * [shiftIn](http://arduino.cc/en/Reference/ShiftIn)() * [pulseIn](http://arduino.cc/en/Reference/PulseIn)()   **Time**   * [millis](http://arduino.cc/en/Reference/Millis)() * [micros](http://arduino.cc/en/Reference/Micros)() * [delay](http://arduino.cc/en/Reference/Delay)() * [delayMicroseconds](http://arduino.cc/en/Reference/DelayMicroseconds)()   **Math**   * [min](http://arduino.cc/en/Reference/Min)() * [max](http://arduino.cc/en/Reference/Max)() * [abs](http://arduino.cc/en/Reference/Abs)() * [constrain](http://arduino.cc/en/Reference/Constrain)() * [map](http://arduino.cc/en/Reference/Map)() * [pow](http://arduino.cc/en/Reference/Pow)() * [sqrt](http://arduino.cc/en/Reference/Sqrt)()   **Trigonometry**   * [sin](http://arduino.cc/en/Reference/Sin)() * [cos](http://arduino.cc/en/Reference/Cos)() * [tan](http://arduino.cc/en/Reference/Tan)()   **Random Numbers**   * [randomSeed](http://arduino.cc/en/Reference/RandomSeed)() * [random](http://arduino.cc/en/Reference/Random)()   **Bits and Bytes**   * [lowByte](http://arduino.cc/en/Reference/LowByte)() * [highByte](http://arduino.cc/en/Reference/HighByte)() * [bitRead](http://arduino.cc/en/Reference/BitRead)() * [bitWrite](http://arduino.cc/en/Reference/BitWrite)() * [bitSet](http://arduino.cc/en/Reference/BitSet)() * [bitClear](http://arduino.cc/en/Reference/BitClear)() * [bit](http://arduino.cc/en/Reference/Bit)()   **External Interrupts**   * [attachInterrupt](http://arduino.cc/en/Reference/AttachInterrupt)() * [detachInterrupt](http://arduino.cc/en/Reference/DetachInterrupt)()   **Interrupts**   * [interrupts](http://arduino.cc/en/Reference/Interrupts)() * [noInterrupts](http://arduino.cc/en/Reference/NoInterrupts)()   **Communication**   * [Serial](http://arduino.cc/en/Reference/Serial) * [Stream](http://arduino.cc/en/Reference/Stream)   **USB (Leonardo and Due only)**   * [Keyboard](http://arduino.cc/en/Reference/MouseKeyboard) * [Mouse](http://arduino.cc/en/Reference/MouseKeyboard)   . |