This page is also available in **2 other languages** Change language   English
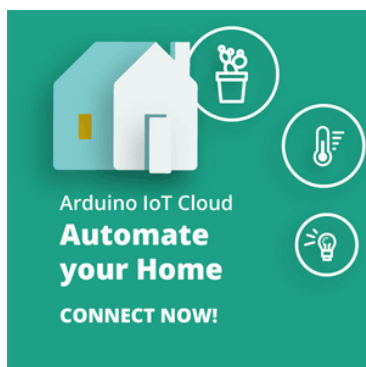
The Arduino Reference text is licensed under a Creative Commons Attribution-Share Alike 3.0 License.

Find anything that can be improved? Suggest corrections and new documentation via GitHub.

Doubts on how to use Github? Learn everything you need to know in this tutorial.

Arduino IoT Cloud
Automate your Home
CONNECT NOW!

Last Revision: 2019/02/21

Last Build: 2021/12/22

EDIT THIS PAGE

Reference > Language > Functions > Random numbers > Random

random()

[Random Numbers]

Description

The random function generates pseudo-random numbers.

Syntax

```
random(max)
random(min, max)
```

Parameters

`min`: lower bound of the random value, inclusive (optional).

`max`: upper bound of the random value, exclusive.

Returns

A random number between min and max-1. Data type: `long`.

Example Code

The code generates random numbers and displays them.

```
long randNumber;

void setup() {
  Serial.begin(9600);

  // if analog input pin 0 is unconnected, random analog
  // noise will cause the call to randomSeed() to generate
  // different seed numbers each time the sketch runs.
  // randomSeed() will then shuffle the random function
  randomSeed(analogRead(0));
}
```

Help

```
randNumber = random(10, 20);
Serial.println(randNumber);

delay(50);
}
```

Notes and Warnings

If it is important for a sequence of values generated by `random()` to differ, on subsequent executions of a sketch, use `randomSeed()` to initialize the random number generator with a fairly random input such as `analogRead()` on an unconnected pin.

Conversely, it can occasionally be useful to use pseudo-random sequences that repeat exactly. This can be accomplished by calling `randomSeed()` with a fixed number, before starting the random sequence.

The `max` parameter should be chosen according to the data type of variable in which the value is stored. In any case, the absolute maximum is bound to the `long` nature of the value generated (32 b 2,147,483,647). Setting `max` to a higher value won't generate an err during compilation, but during sketch execution the numbers generated will not be as expected.

See also

Back to top

Help