

Начало работы с ARDUINO



Загрузка и установка программного обеспечения (ПО) и драйверов

1. Качаем и устанавливаем актуальную версию Arduino IDE с сайта разработчика:



[https://www.arduino.](https://www.arduino.cc)

cc



Download the Arduino IDE



ARDUINO 1.8.13

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

Windows Installer, for Windows 7 and up
Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10 Get

Mac OS X 10.10 or newer

Linux 32 bits

Linux 64 bits

Linux ARM 32 bits

Linux ARM 64 bits

[Release Notes](#)

[Source Code](#)

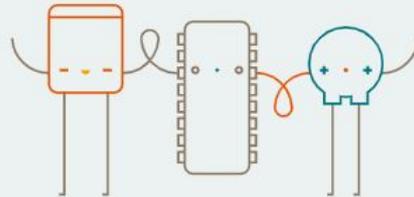
[Checksums \(sha512\)](#)

Загрузка и установка программного обеспечения (ПО) и драйверов

1. Качаем и устанавливаем актуальную версию
Arduino IDE с сайта разработчика:

Contribute to the Arduino Software

Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). [Learn more on how your contribution will be used.](#)



SINCE MARCH 2015, THE ARDUINO IDE HAS BEEN DOWNLOADED **44,920,103** TIMES. (IMPRESSIVE!) NO LONGER JUST FOR ARDUINO AND GENUINO BOARDS, HUNDREDS OF COMPANIES AROUND THE WORLD ARE USING THE IDE TO PROGRAM THEIR DEVICES, INCLUDING COMPATIBLES, CLONES, AND EVEN COUNTERFEITS. HELP ACCELERATE ITS DEVELOPMENT WITH A SMALL CONTRIBUTION! REMEMBER: OPEN SOURCE IS LOVE!

\$3

\$5

\$10

\$25

\$50

OTHER

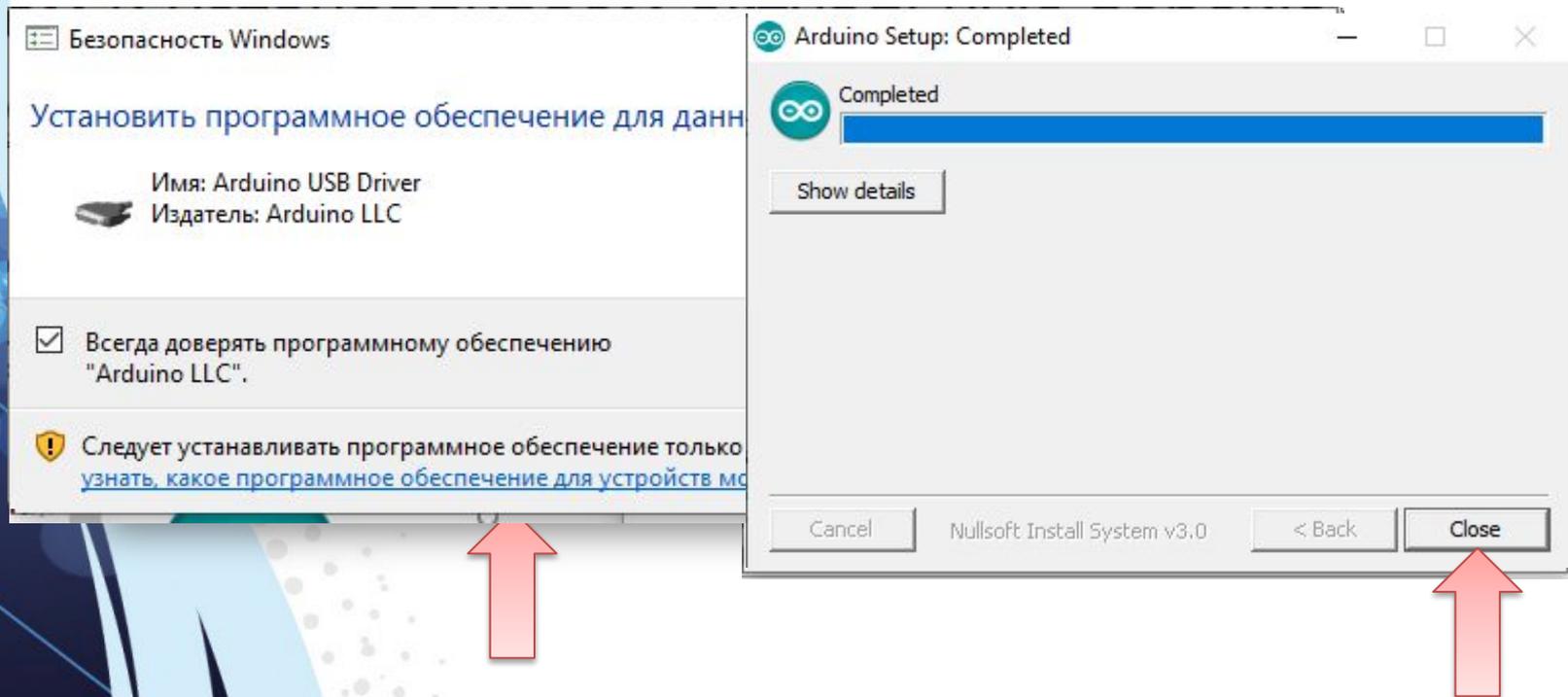
JUST DOWNLOAD

CONTRIBUTE & DOWNLOAD



Загрузка и установка программного обеспечения (ПО) и драйверов

1. Качаем и устанавливаем актуальную версию Arduino IDE с сайта разработчика:



Загрузка и установка программного обеспечения (ПО) и драйверов

2. Качаем и устанавливаем JRE (Java Runtime Environment) с сайта разработчика:



<https://www.java.com>

m

The screenshot shows the Java website interface. At the top, there is a red header with the Java logo on the left, a search bar on the right, and navigation links for 'Загрузить' (Download) and 'Справка' (Help). Below the header, the main content area features a sidebar on the left with 'Справочные ресурсы' (Reference resources) and a list of links. The main content area displays the title '64-разрядная версия Java для Windows' (64-bit version of Java for Windows) in red, followed by the recommendation 'Рекомендуется Version 8 Update 261 (размер файла: 79.19 МВ)' (Version 8 Update 261 is recommended, file size: 79.19 MB) and the release date 'Дата выпуска: 14 июля 2020 г.' (Release date: July 14, 2020). A prominent red button with white text says 'Согласиться и начать бесплатную загрузку' (Agree and start free download), with a red arrow pointing to it from the right. Below the button, there is a paragraph of text: 'Фактом выгрузки Java вы подтверждаете, что прочитали и приняли условия лицензионного соглашения Oracle Technology Network для Oracle Java SE'. At the bottom, a small blue icon is followed by the text: 'После установки Java для применения изменений может потребоваться перезагрузка браузера (необходимо закрыть и снова открыть все окна браузера)' (After installing Java, a browser restart may be required to apply changes (you must close and then reopen all browser windows)).

Справочные ресурсы

- » [Что такое Java?](#)
- » [Удаление предыдущих версий](#)
- » [Отключение поддержки Java](#)
- » [Сообщения об ошибках](#)
- » [Устранение проблем Java](#)
- » [Другие разделы справки](#)

64-разрядная версия Java для Windows

Рекомендуется Version 8 Update 261 (размер файла: 79.19 МВ)
Дата выпуска: 14 июля 2020 г.

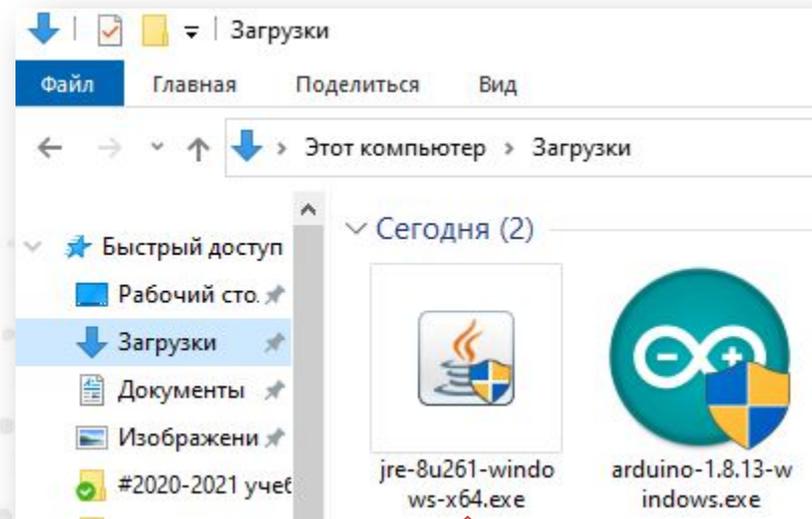
Согласиться и начать бесплатную загрузку

Фактом выгрузки Java вы подтверждаете, что прочитали и приняли условия [лицензионного соглашения Oracle Technology Network](#) для Oracle Java SE

После установки Java для применения изменений может потребоваться перезагрузка браузера (необходимо закрыть и снова открыть все окна браузера).

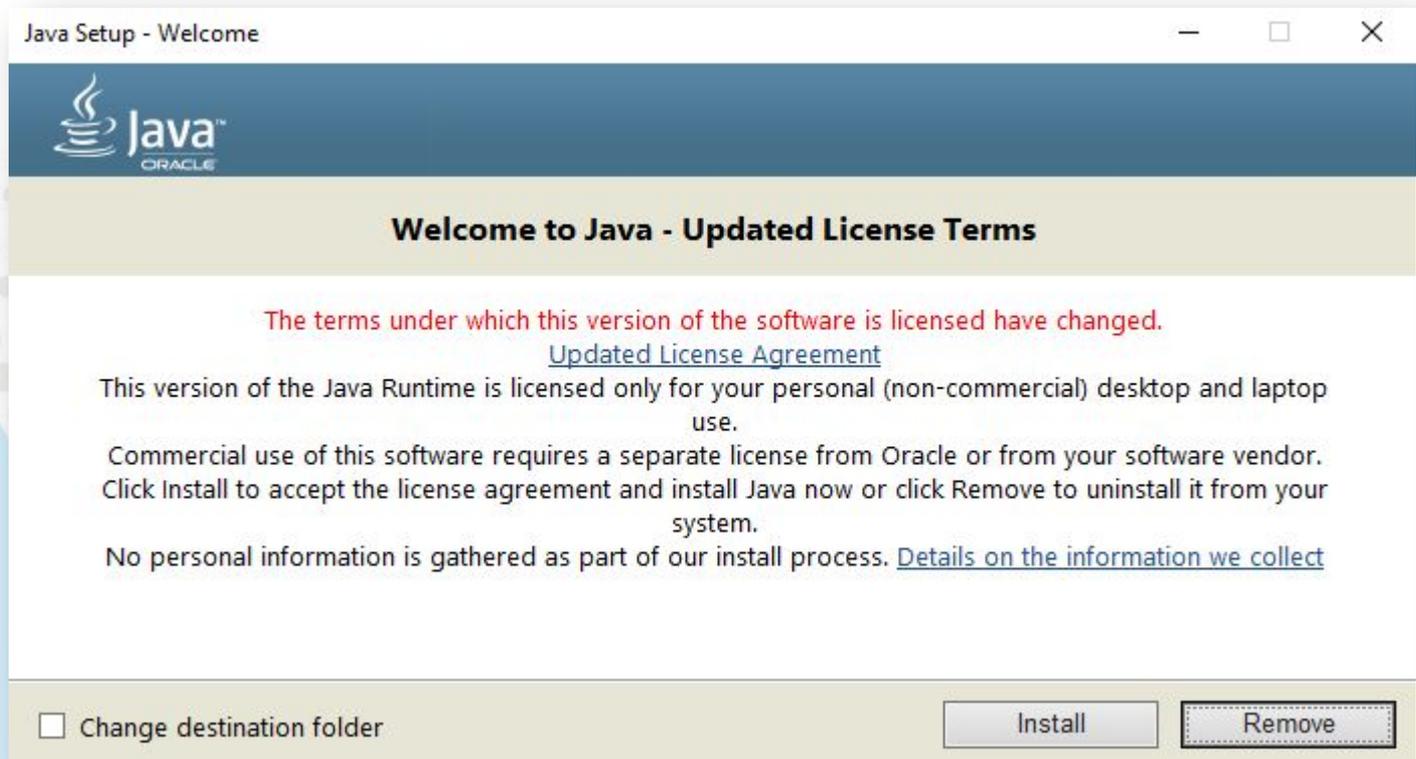
Загрузка и установка программного обеспечения (ПО) и драйверов

2. Качаем и устанавливаем JRE (Java Runtime Environment) с сайта разработчика:



Загрузка и установка программного обеспечения (ПО) и драйверов

2. Качаем и устанавливаем JRE (Java Runtime Environment) с сайта разработчика:



Загрузка и установка программного обеспечения (ПО) и драйверов

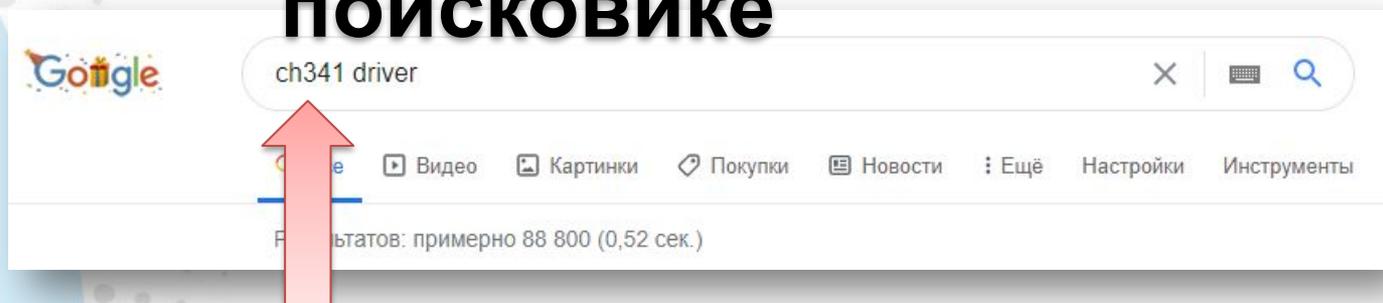
3. При установке Arduino IDE должны автоматически поставиться необходимые для работы драйверы.

На большинстве китайских Arduino-плат стоит контроллер интерфейса USB CH341, для работы с ним нужен отдельный специальный драйвер.

(Китайский контроллер ничем не хуже, он просто дешевле :)

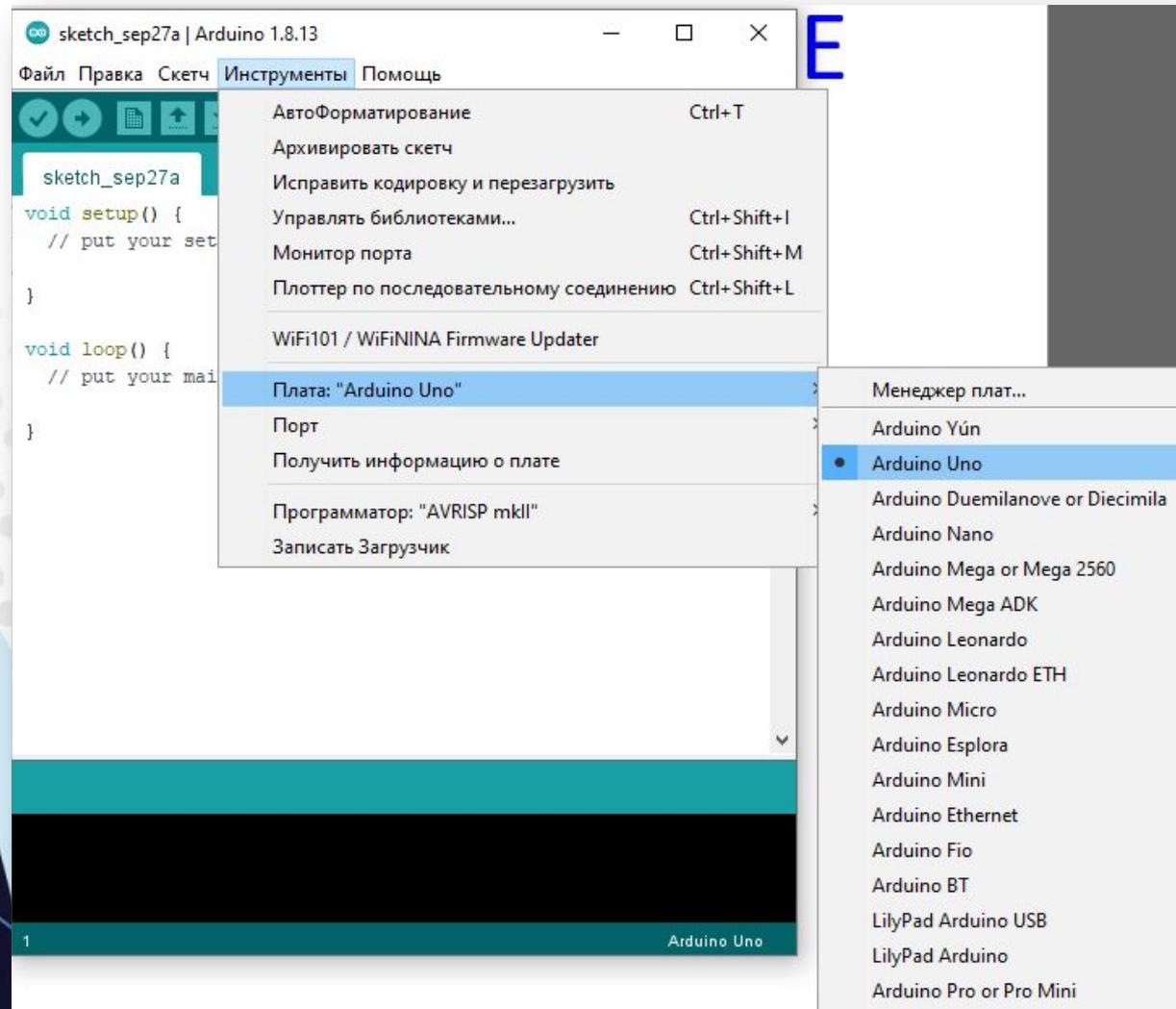
ИЩЕМ В

ПОИСКОВИКЕ



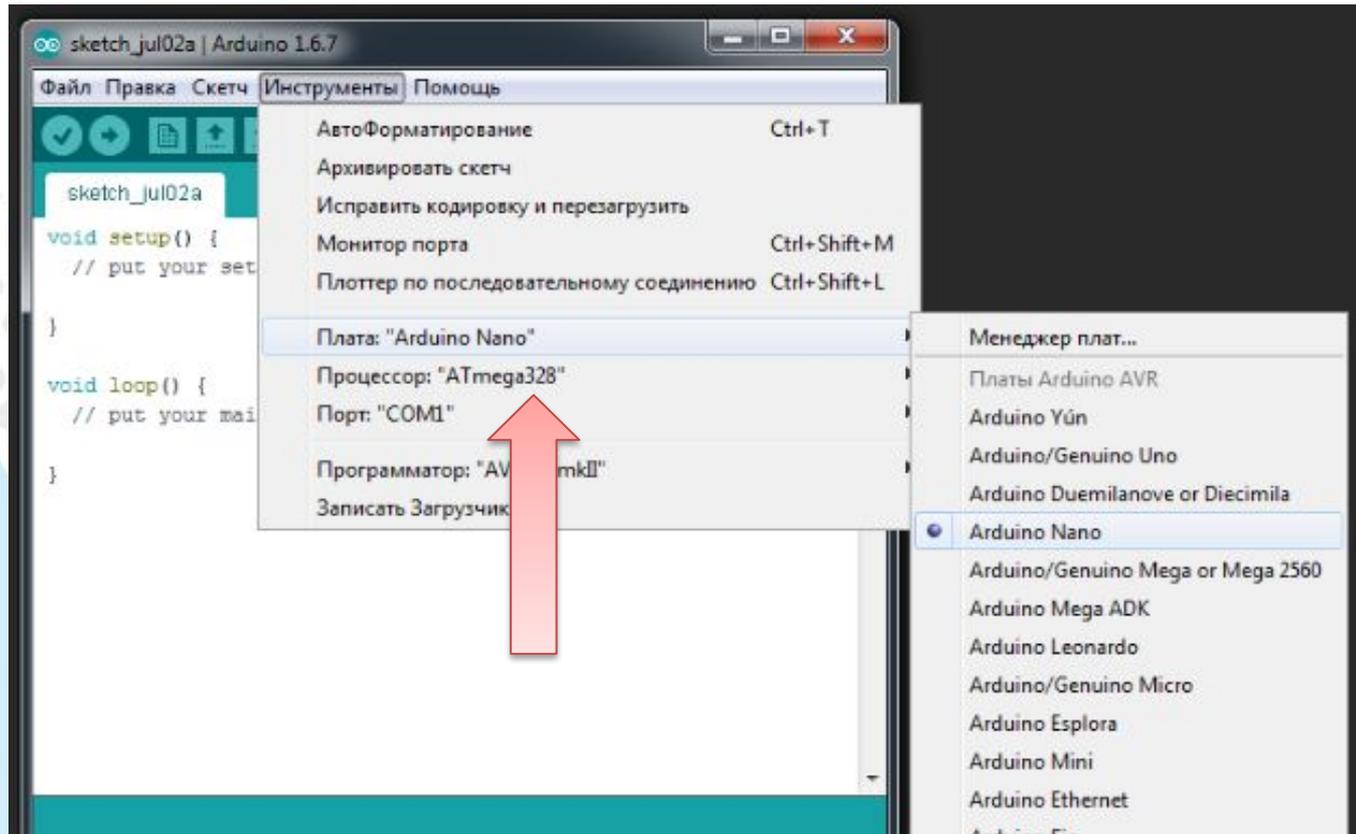
Настраиваем Arduino IDE

Запустить Arduino IDE, выбрать плату
(Инструменты\плата\”ваша плата”)



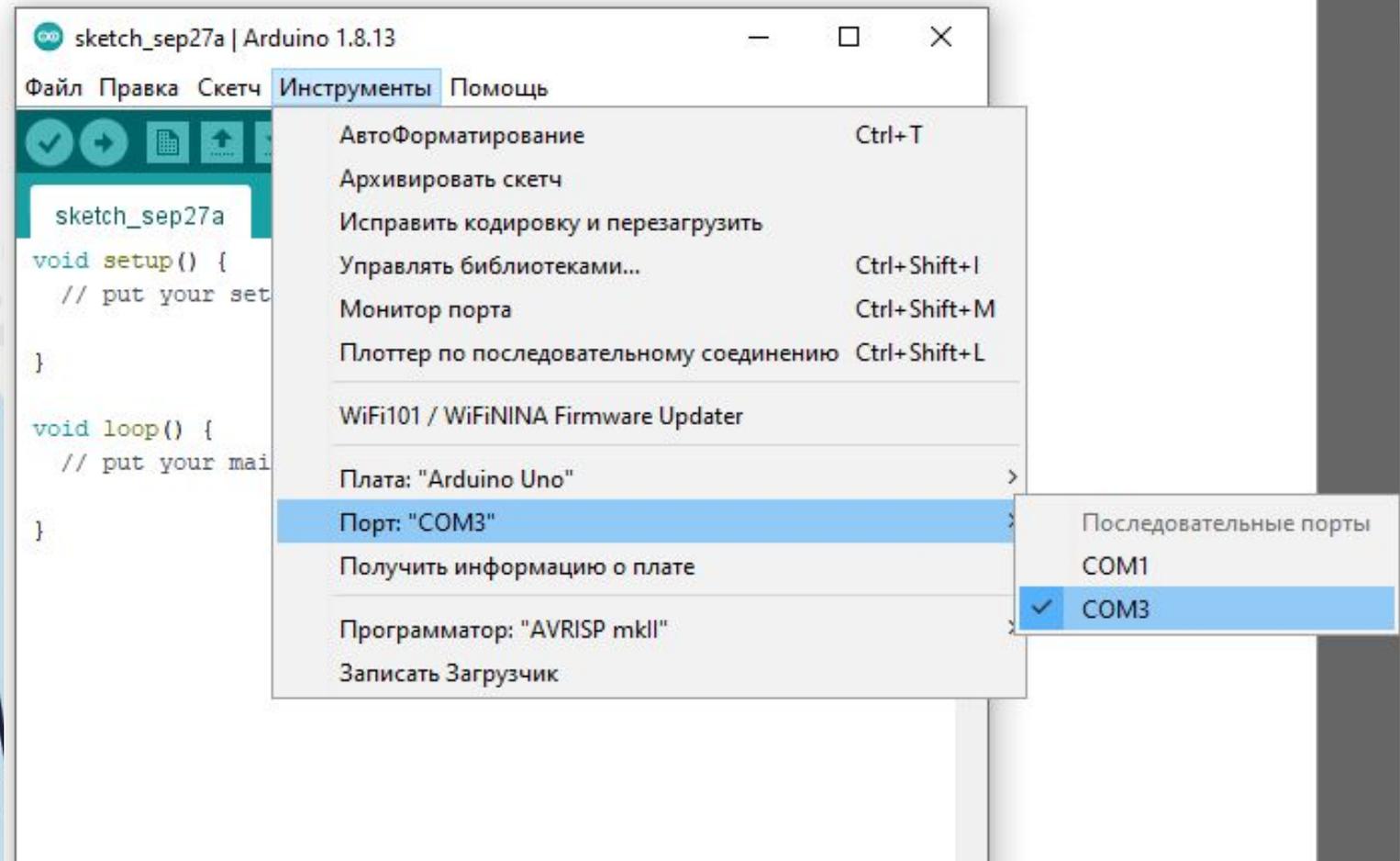
Настраиваем Arduino IDE

Выбрать модель платы/микроконтроллера (загрузчика) Инструменты\Процессор\”**Ваша модель**”. (для Nano)



Настраиваем Arduino IDE

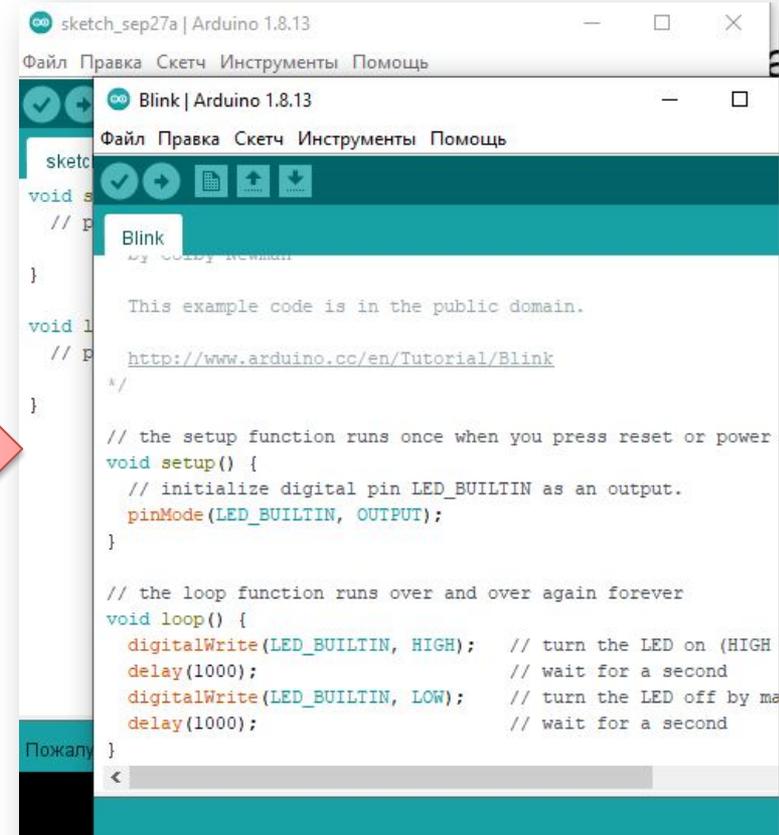
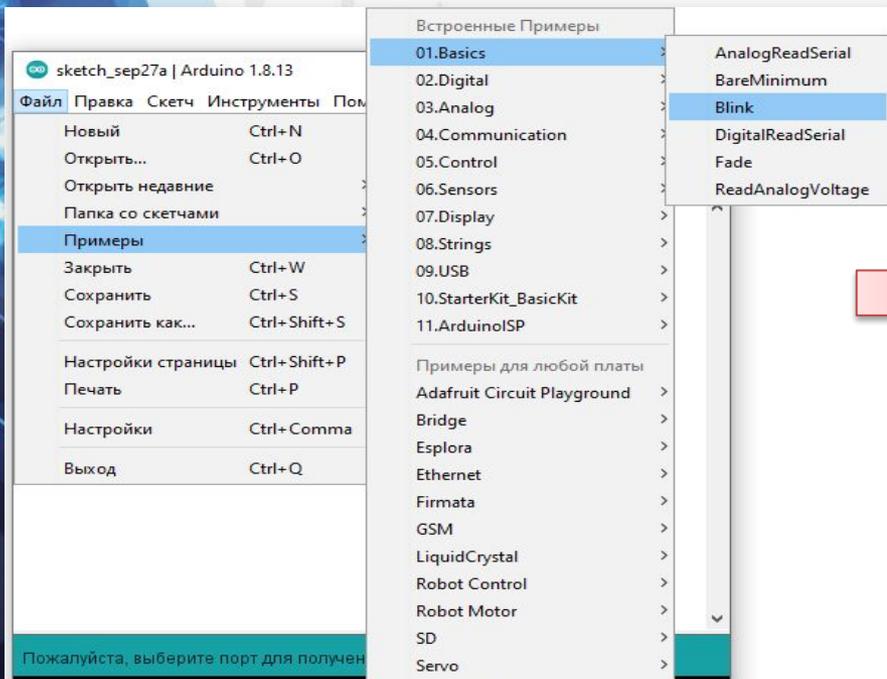
Выбрать порт: инструменты\порт\”COM отличный от COM1, например COM3, COM5...”



Настраиваем Arduino IDE

ВНИМАНИЕ! Как только достанете Arduino из пакетика, сразу прошейте в неё скетч с миганием светодиода (blink.ino)

Таким образом вы узнаете, что Ардуина рабочая *(на тот случай, когда после сборки/пайки она перестанет работать и прошиваться)*



Ответы на частые вопросы:

Ардуину можно прошить только один раз?

- Нет, несколько десятков тысяч раз, всё упирается в ресурс flash памяти. А он довольно большой.

Как стереть/нужно ли стирать старую прошивку при загрузке новой?

- Память автоматически очищается при прошивке. Старая прошивка будет автоматически удалена.

Можно ли записать две прошивки, чтобы они работали вместе?

- Нет, при прошивке удаляются абсолютно все старые данные.

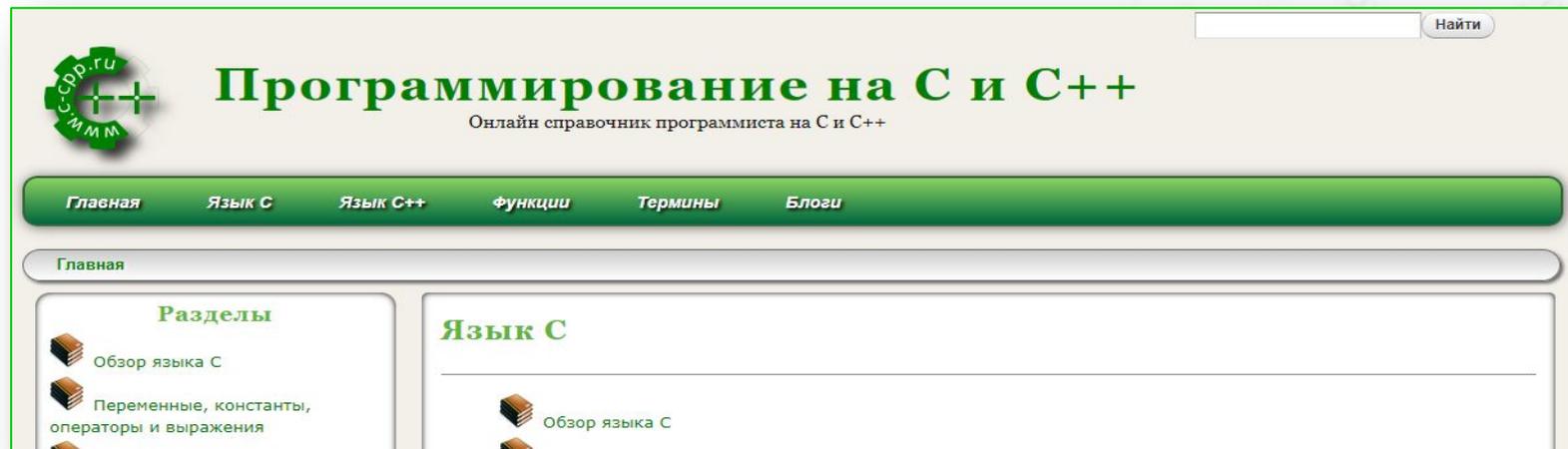
Программирование

микроконтроллера:

Общение с микроконтроллером происходит в письменном виде на языке программирования, язык это очень чёткий, строгий и имеет свой синтаксис и некоторые нормы оформления.

Официальной средой разработки является **Arduino IDE**, где программирование осуществляется на языке **C++** – одном из самых популярных и мощных языков.

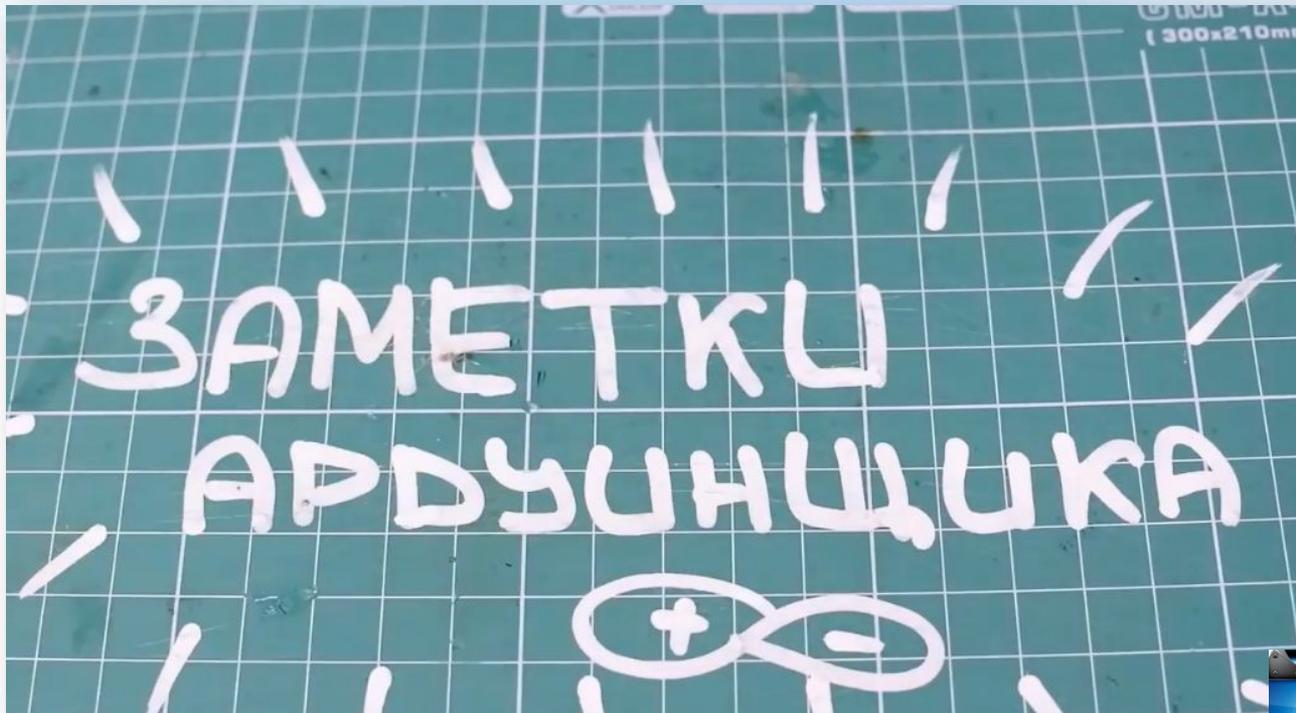
Справочник по языку для любознательных:



The screenshot shows the homepage of the website 'Программирование на C и C++'. The site has a green and white color scheme. At the top left is a logo with 'C++' and 'www.c-cpp.ru'. The main title is 'Программирование на C и C++' with the subtitle 'Онлайн справочник программиста на C и C++'. A navigation bar contains links for 'Главная', 'Язык C', 'Язык C++', 'функции', 'Термины', and 'Блоги'. Below the navigation bar, there are two main content areas. The left area is titled 'Разделы' and lists 'Обзор языка C' and 'Переменные, константы, операторы и выражения'. The right area is titled 'Язык C' and also lists 'Обзор языка C'. A search bar is located in the top right corner.

<http://www.c-cpp.ru/book>

Видеоурок:



6 мин.

**Видео прилагается к
презентации**



Конспект 1.0:

```
myFirstSketch | Arduino 1.6.7
Файл Правка Скетч Инструменты Помощь
myFirstSketch $
11 // подключаются библиотеки, например библиотека servo
12
13 #include <Servo.h>
14
15 // объявляются переменные
16
17 int lolokek;
18
19 void setup() {
20   // всё находящееся здесь выполнится 1 раз при запуске Ардуино
21   // настраиваем датчики и саму Ардуино, готовимся к
22   // выполнению основной программы
23 }
24
25 void loop() {
26   // бесконечный цикл, всё выполняется сверху вниз
27   // идёт чтение значений с датчиков, их обработка
28
Загрузка завершена.
Скетч использует 450 байт (1%) памяти устройства. Всего доступно
Глобальные переменные используют 9 байт (0%) динамической памяти,
27 Arduino Nano, ATmega328 on COM3
```

Конспект 1.0

// Однострочный комментарий

/* Многострочный комментарий

*/ (комментарии не занимают память)

#include - Подключить файл (библиотеку)

void setup() {} - Всё находящееся внутри {}

будет выполнено 1 раз при загрузке Ардуино

void loop() {} - Всё находящееся внутри {}

бесконечно повторяется сверху вниз

Комментарии:

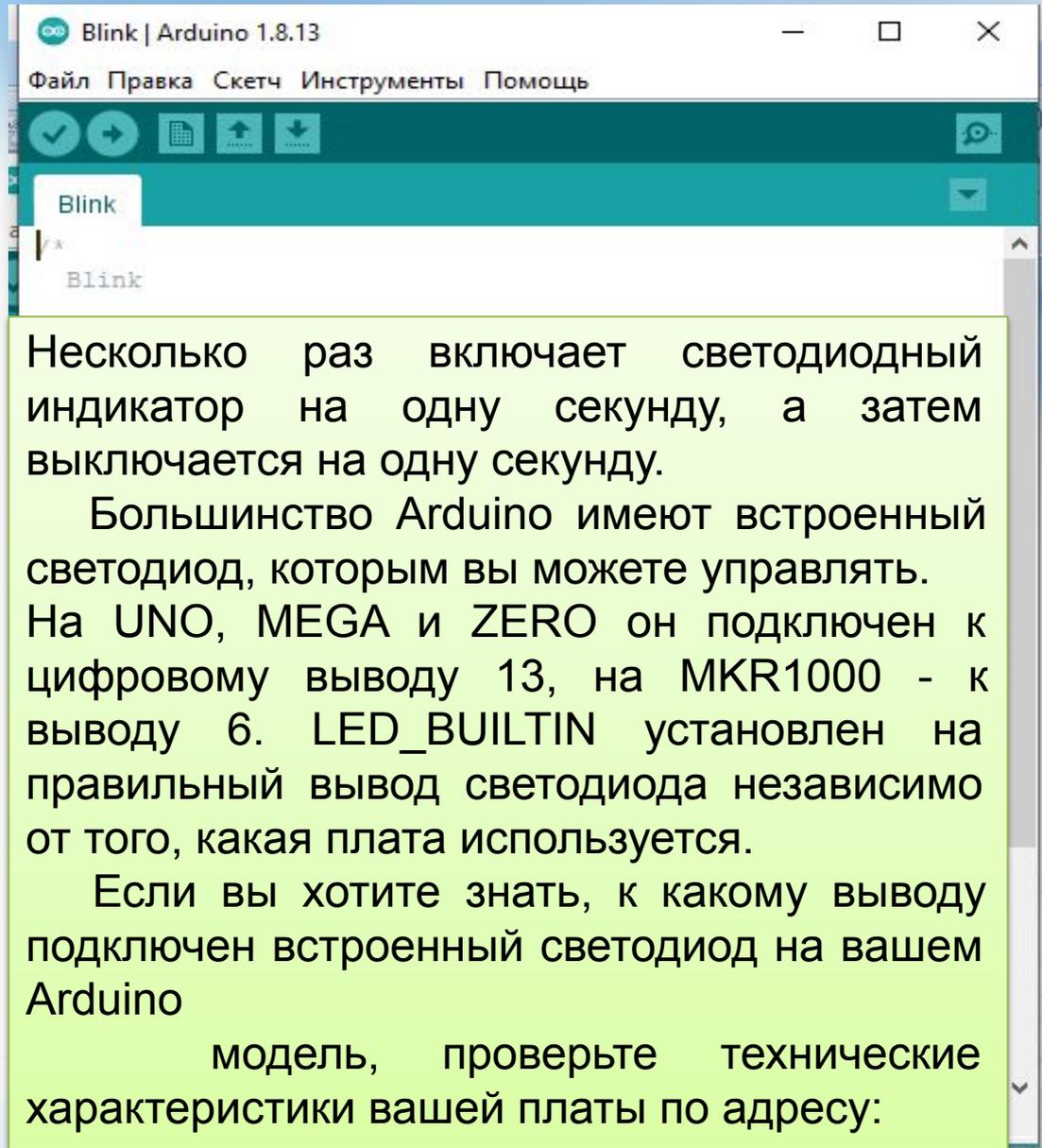
Однострочный комментарий:

`// однострочный комментарий`

`// компилятор меня игнорирует =(`

Многострочный комментарий:

`/* Многострочный комментарий */`

The image shows a screenshot of the Arduino IDE interface. The window title is "Blink | Arduino 1.8.13". The menu bar includes "Файл", "Правка", "Скетч", "Инструменты", and "Помощь". The toolbar contains icons for a checkmark, a right arrow, a grid, an upload arrow, a download arrow, and a gear. The main editor area shows a file named "Blink" with a single line of code: `Blink`. Below the editor, there is a light green text box containing the following text:

Несколько раз включает светодиодный индикатор на одну секунду, а затем выключается на одну секунду.

Большинство Arduino имеют встроенный светодиод, которым вы можете управлять. На UNO, MEGA и ZERO он подключен к цифровому выводу 13, на MKR1000 - к выводу 6. LED_BUILTIN установлен на правильный вывод светодиода независимо от того, какая плата используется.

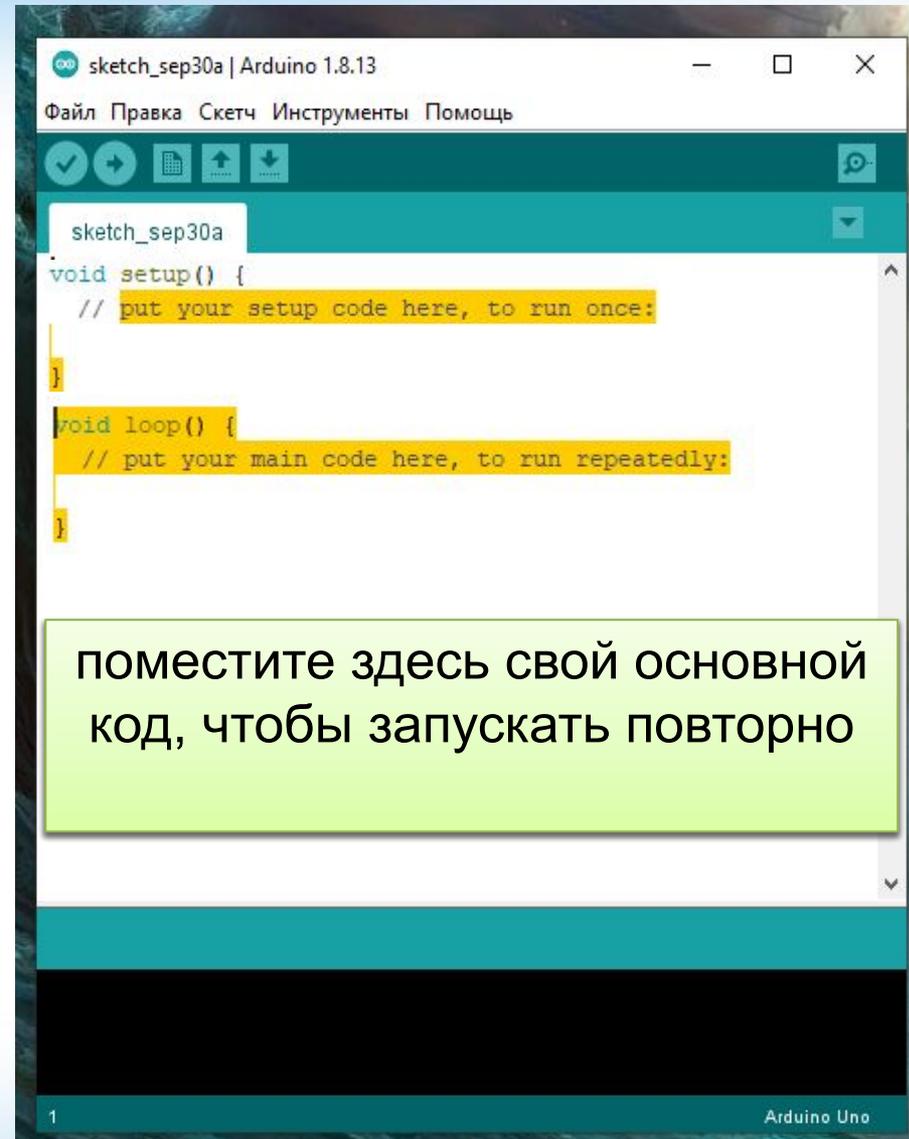
Если вы хотите знать, к какому выводу подключен встроенный светодиод на вашем Arduino

модель, проверьте технические характеристики вашей платы по адресу:

Комментарии:

При запуске Arduino IDE даёт нам заготовку в виде двух обязательных функций: **setup** и **loop**

Код в блоке **setup()** выполняется один раз при каждом запуске микроконтроллера. Код в блоке **loop()** выполняется “по кругу” на всём протяжении работы микроконтроллера, начиная с момента завершения выполнения **setup()**



```
sketch_sep30a | Arduino 1.8.13
Файл Правка Скетч Инструменты Помощь

sketch_sep30a
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

поместите здесь свой основной код, чтобы запускать повторно

1 Arduino Uno

Видеоурок:

```
myFirstSketch | Arduino 1.6.7
Файл Правка Скетч Инструменты Помощь
myFirstSketch $
1 boolean bybool = 1;
2 boolean azaza = 0;
3 boolean logicLOLO = true; // инвертируем !logicLOLO
4
5 char mysymbol = 'a'; // хранится в виде числа согласно ASCII
6
7 byte lolbyte = 150;
8
9 int myvalue 22000;
10
11 unsigned int lolokek = 50000;
12
13 long BIGBIG = 123423425;
14
15 float superTen = 10.25|
16
17
18 void setup() {
19 // put your setup code here, to run once:
20
Загрузка завершена.
Скетч использует 450 байт (1%) памяти устройства. Всего доступно
Глобальные переменные используют 9 байт (0%) динамической памяти,
16 Arduino Nano, ATmega328 on COM3
```

Конспект 1.1

ТИПЫ ДАННЫХ ARDUINO

объявление "тип данных" "название";

или "тип данных" "имя" = "значение";

Название	Вес	Диапазон	Особенность
boolean	1 байт	0 или 1	Логическая переменная, может принимать значения true (1) и false (0)
char	1 байт	-128... 127	Хранит номер символа из таблицы символов ASCII
byte	1 байт	0... 255	
int	2 байта	-32 768... 32 767	
unsigned int	2 байта	0... 65 535	
word	2 байта	0... 65 535	То же самое, что unsigned int.
long	4 байта	-2 147 483 648... 2 147 483 647	- 2 миллиарда... 2 миллиарда
unsigned long	4 байта	0... 4 294 967 295	0... 4 миллиарда...
float	4 байта	-3.4028235E+38... 3.4028235E+38	Хранит числа с плавающей точкой (десятичные дроби). Точность: 6-7 знаков
double	4 байта		То же самое, что float



Видеоурок:

```
myFirstSketch | Arduino 1.6.7
Файл Правка Скетч Инструменты Помощь
myFirstSketch $
7 byte lolbyte = 150;
8
9 int myvalue 22000;
10
11 unsigned int lolokek = 50000;
12
13 long BIGBIG = 123423425;
14
15 float superTen = 10.25;
16 float opopo = 10.0;
17
18
19 void setup() {
20   int celoe_chislo = 10;
21   float des9tka = (float)celoe_chislo / 2.314;
22
23 }
24
25 void loop() {
Загрузка завершена.
Скетч использует 450 байт (1%) памяти устройства. Всего доступно
Глобальные переменные используют 9 байт (0%) динамической памяти,
```

Конспект 1.2

ОСОБЕННОСТИ FLOAT

- 1) Присваивать только значение с точкой, даже если оно целое (10.0)
- 2) Делить тоже только на числа с точкой, даже если они целые (переменная / 2.0)
- 3) При делении целочисленного типа с целью получить число с плавающей точкой, писать (float) перед вычислением!

P.S. Операции с числами типа float занимают гораздо больше времени, чем с целыми!
Если нужна высокая скорость вычислений, лучше применять всякие хитрости =)

Синтаксис:

Тела функций заключаются в фигурные скобки **{ }**

Каждая команда заканчивается точкой с запятой **;**

Метод применяется к объекту через точку **.**

*Пример: **Serial.begin();***

Вызов функции или метода всегда заканчивается скобками **()**

*Пример: **loop()***

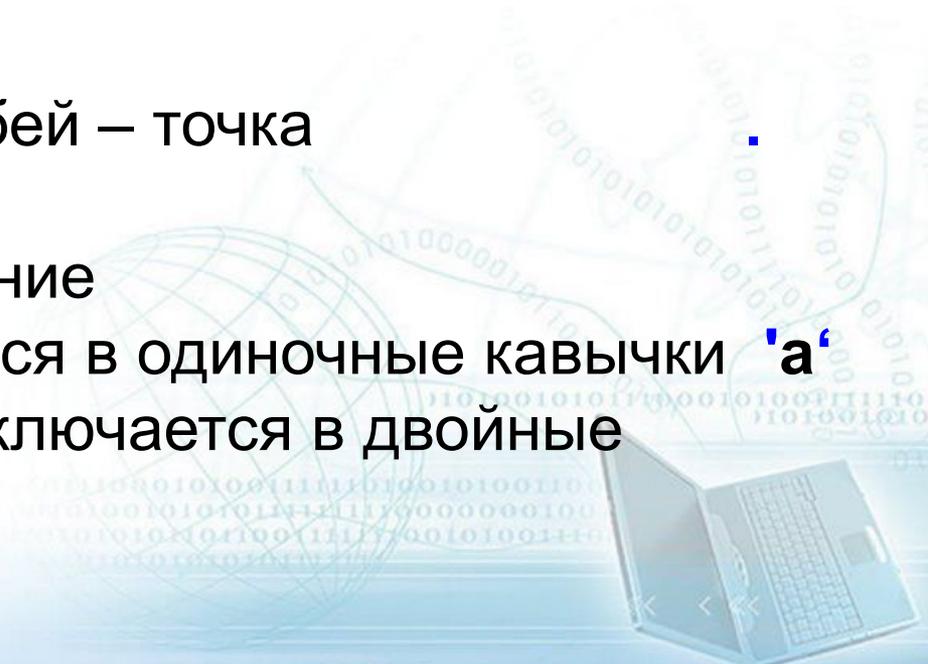
Разделитель десятичных дробей – точка **.**

*Пример: **0.25***

У запятой тут другое применение

Одиночный символ заключается в одиночные кавычки **'а'**

Строка и массив символов заключается в двойные кавычки **"строка"**



Синтаксис:

Регистр имеет значение, т.е. большая буква отличается от маленькой.

Пример: переменные `val` и `Val` – не одно и то же.

