

Ravesli [Ravesli](#)

- [Уроки по C++](#)
- [OpenGL](#)
- [SFML](#)
- [Qt5](#)
- [RegExr](#)
- [Ассемблер](#)
- [Купить .PDF](#)

Урок №5. Компиляция вашей первой программы

👤 [Юрий](#) |

- [Уроки C++](#)

|

📅 Обновлено. 17 Сен 2020 |

👁 119862

🗨 70

Перед написанием нашей первой программы мы еще должны кое-что узнать.

Оглавление:

1. [Теория](#)
2. [Пользователям Visual Studio](#)
3. [Пользователям Code::Blocks](#)
4. [Пользователям командной строки](#)
5. [Пользователям веб-компиляторов](#)
6. [Пользователям других IDE](#)
7. [Если компиляция прошла неудачно](#)
8. [Заключение](#)

Теория

Во-первых, несмотря на то, что код ваших программ находится в файлах .cpp, эти файлы добавляются в **проект**. Проект содержит все необходимые файлы вашей программы, а также сохраняет указанные вами настройки вашей **IDE**. Каждый раз, при открытии проекта, он запускается с того момента, на котором вы остановились в прошлый раз. При компиляции программы, проект говорит **компилятору и линкеру**, какие файлы нужно скомпилировать, а какие связать. Стоит отметить, что файлы проекта одной IDE не будут работать в другой IDE. Вам придется создать новый проект (в другой IDE).

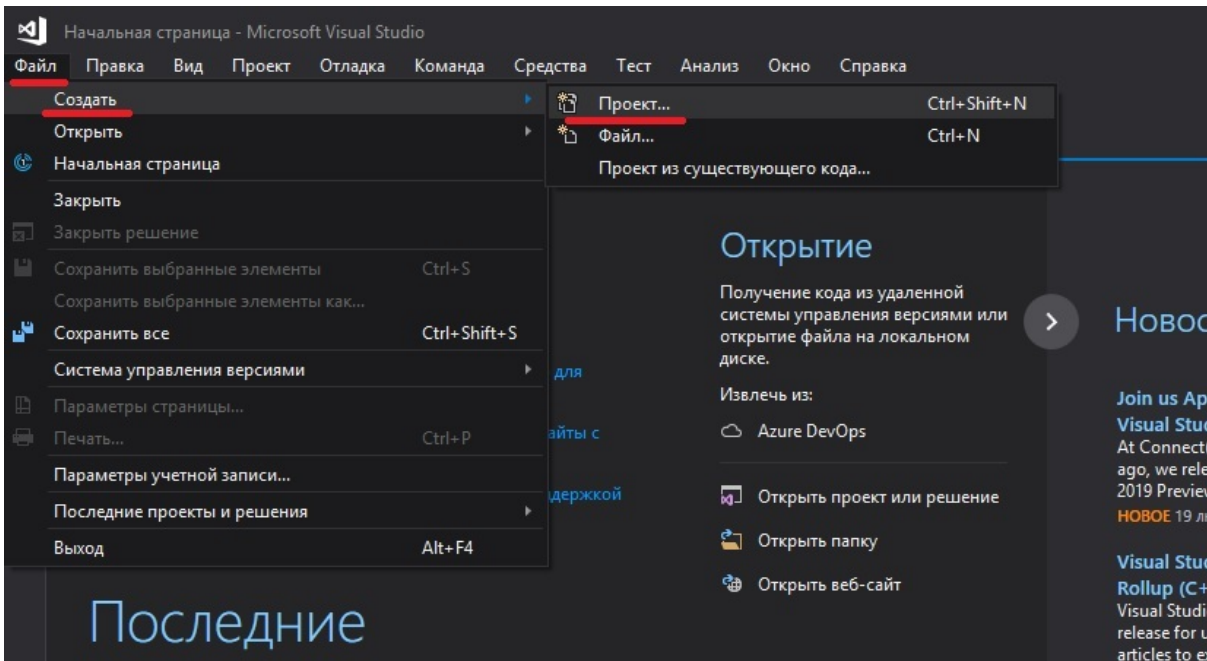
Во-вторых, есть разные типы проектов. При создании нового проекта, вам нужно будет выбрать его тип. Все проекты, которые мы будем создавать на данных уроках, будут **консольного типа**. Это означает, что они запускаются в консоли (аналог командной строки). По умолчанию, консольные приложения не имеют графического интерфейса пользователя — **GUI** (сокр. от «*Graphical User Interface*») и компилируются в автономные исполняемые файлы. Это идеальный вариант для изучения языка C++, так как он сводит всю сложность к минимуму.

В-третьих, при создании нового проекта большинство IDE автоматически добавят ваш проект в рабочее пространство. **Рабочее пространство** — это своеобразный контейнер, который может содержать один или несколько связанных проектов. Несмотря на то, что вы можете добавить несколько проектов в одно рабочее пространство, все же рекомендуется создавать отдельное рабочее пространство для каждой программы. Это намного упрощает работу для новичков.

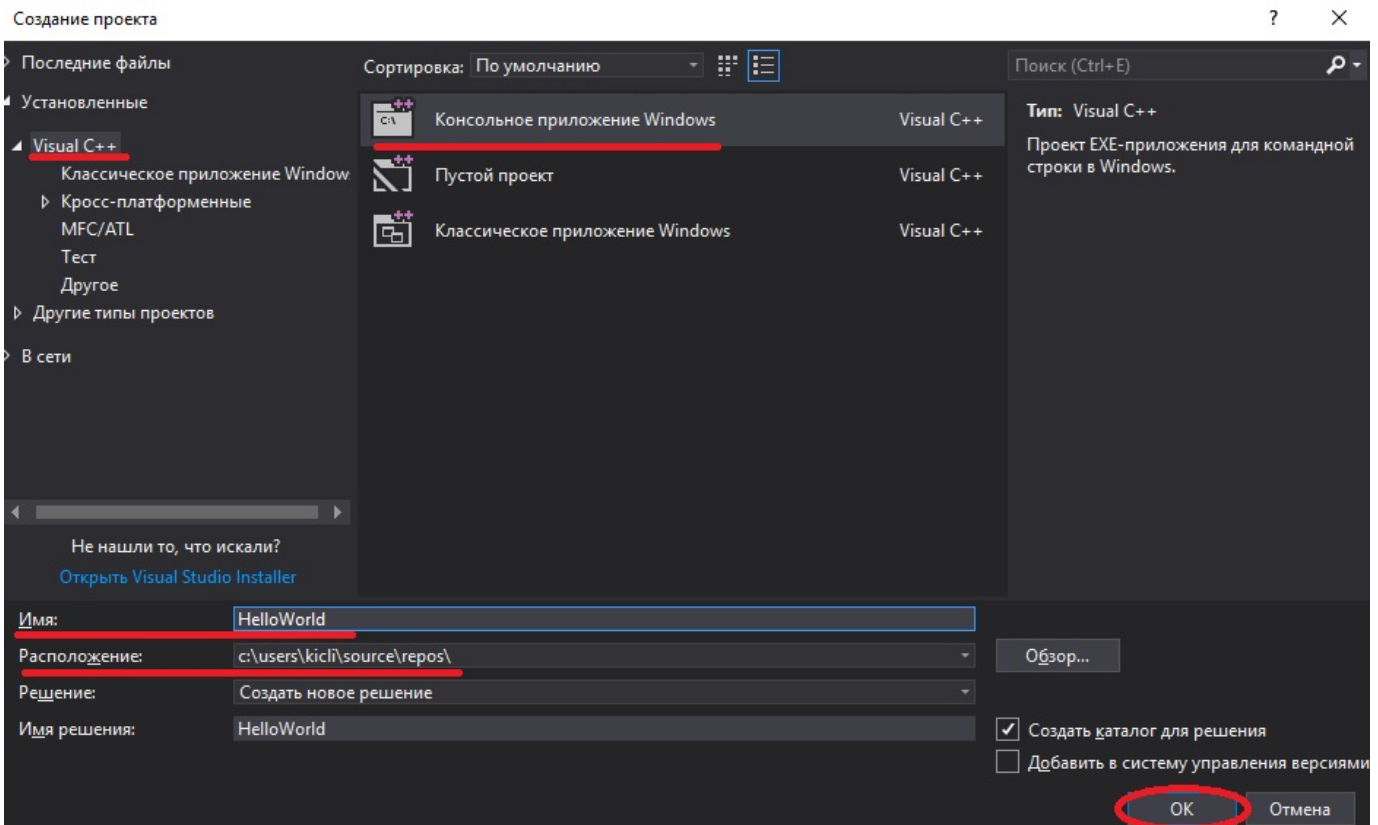
Традиционно, первой программой на новом языке программирования является всеми известная программа «Hello, world!». Мы не будем нарушать традиции 😊

Пользователям Visual Studio

Для создания нового проекта в Visual Studio 2019, вам нужно сначала запустить эту IDE, затем выбрать "Файл" > "Создать" > "Проект":



Дальше появится диалоговое окно, где вам нужно будет выбрать "Консольное приложение Windows" из вкладки "Visual C++" и нажать "ОК":

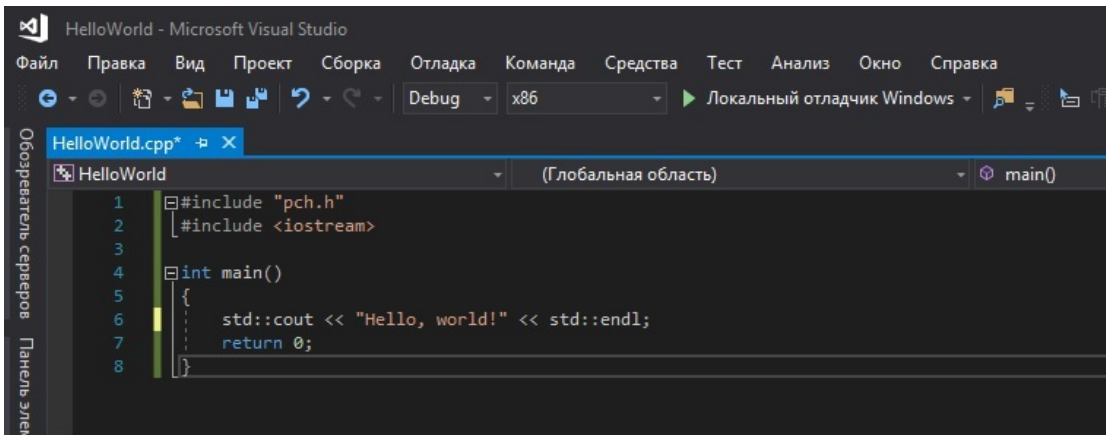


Также вы можете указать имя проекта (любое) и его расположение (рекомендую ничего не менять) в соответствующих полях.

В текстовом редакторе вы увидите, что уже есть некоторый текст и код — удалите его, а затем напечатайте или скопируйте следующий код:

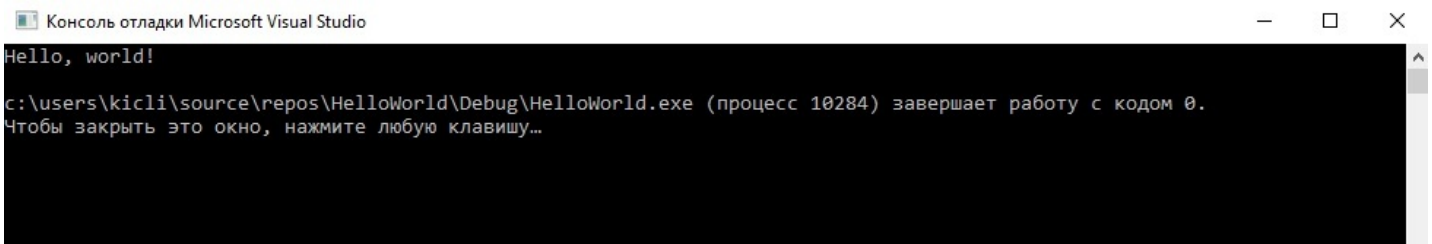
```
1 #include <iostream>
2
3 int main()
4 {
5     std::cout << "Hello, world!" << std::endl;
6     return 0;
7 }
```

Вот, что у вас должно получиться:



ВАЖНОЕ ПРИМЕЧАНИЕ: Строка `#include "pch.h"` требуется только для пользователей Visual Studio 2017. Если вы используете Visual Studio 2019 (или более новую версию), то не нужно писать эту строку вообще.

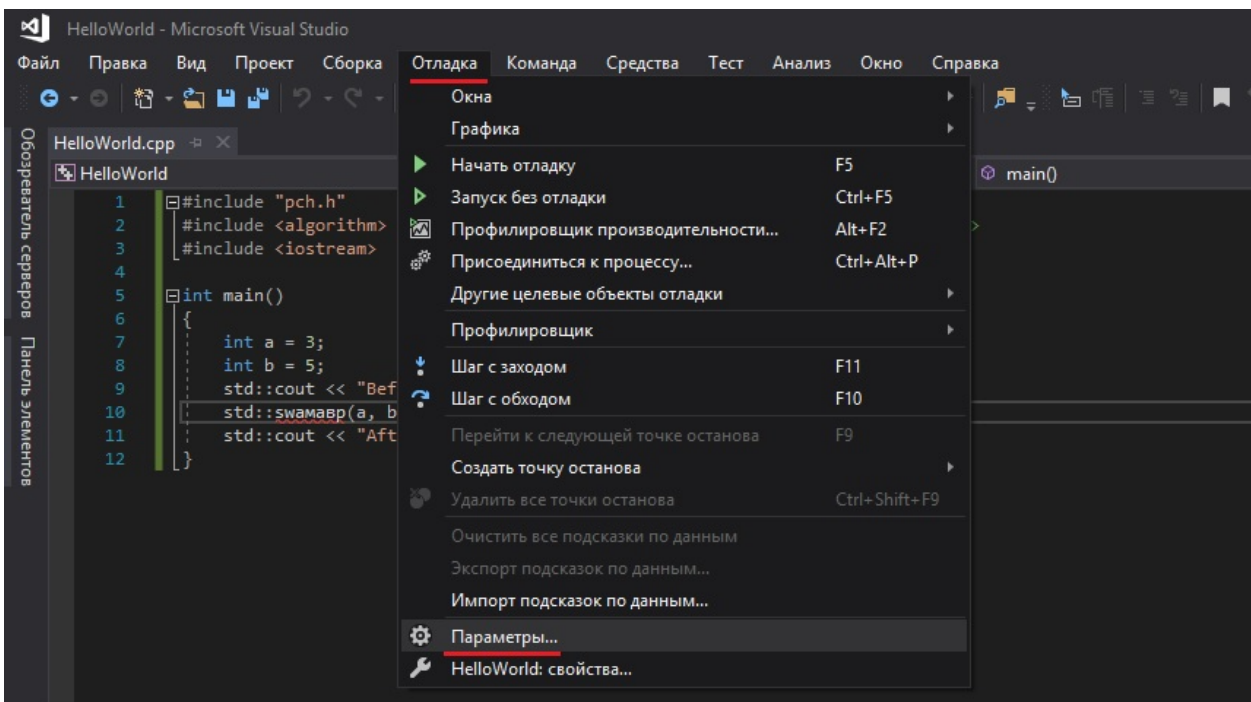
Чтобы запустить программу в Visual Studio, нажмите комбинацию `Ctrl+F5`. Если всё хорошо, то вы увидите следующее:



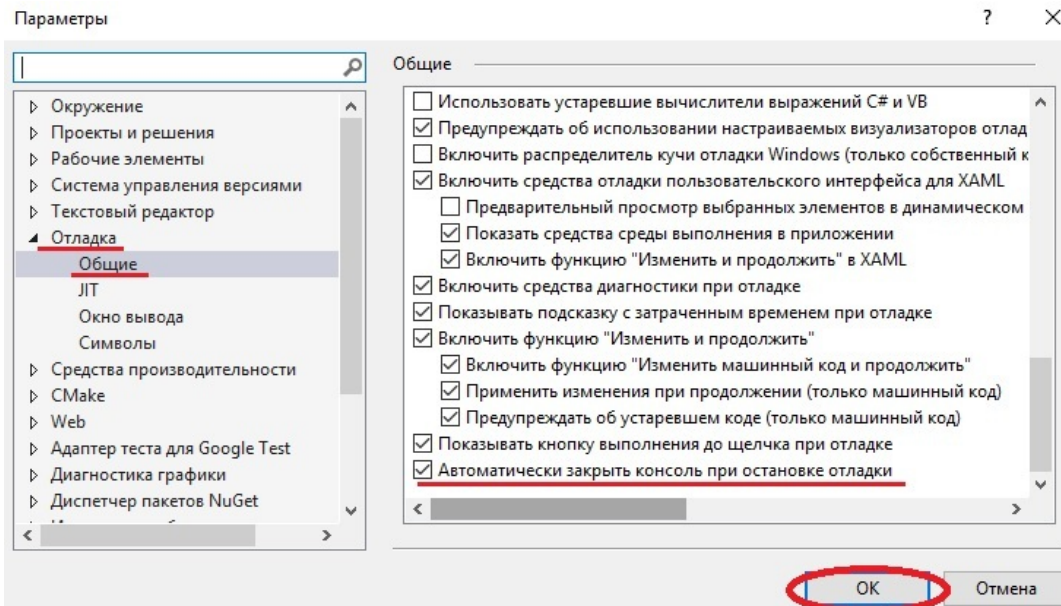
Это означает, что компиляция прошла успешно и результат выполнения вашей программы следующий:

Hello, world!

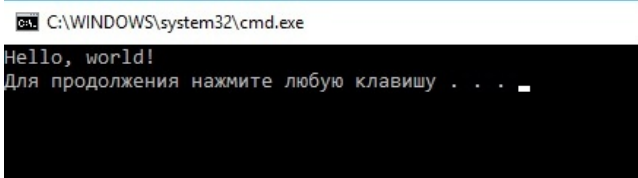
Чтобы убрать строку «...завершает работу с кодом 0...», вам нужно перейти в "Отладка" > "Параметры":



Затем "Отладка" > "Общие" и поставить галочку возле "Автоматически закрыть консоль при остановке отладки" и нажать "ОК":



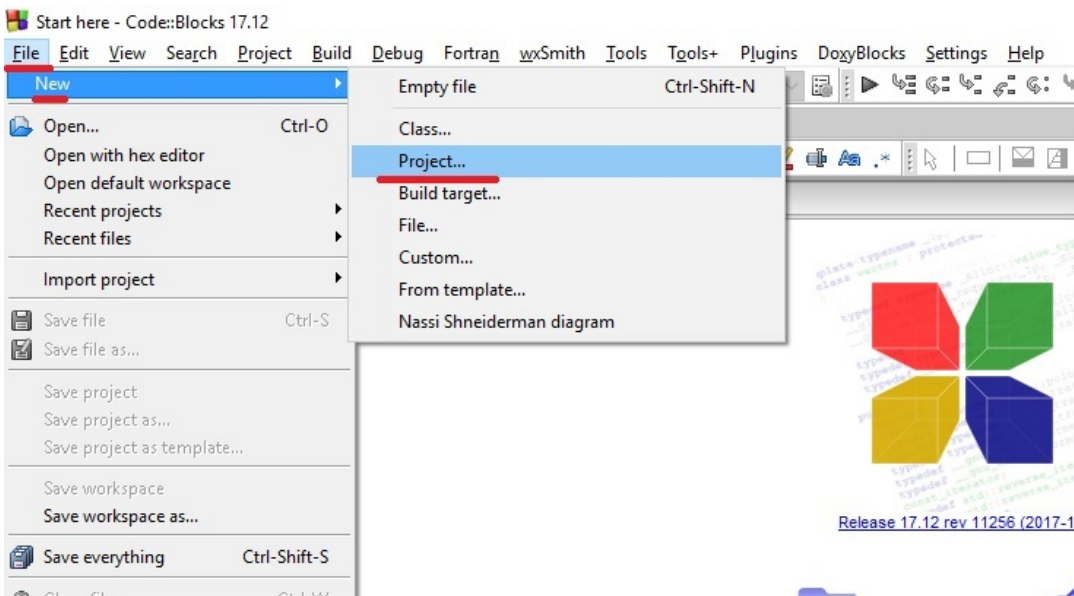
Тогда ваше консольное окно будет выглядеть следующим образом:



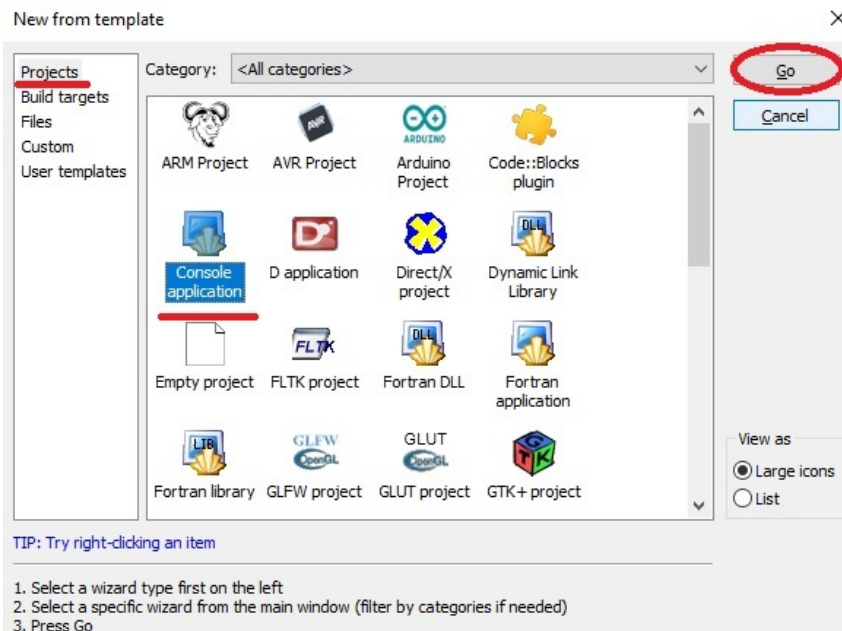
Готово! Мы научились компилировать программу в Visual Studio.

Пользователям Code::Blocks

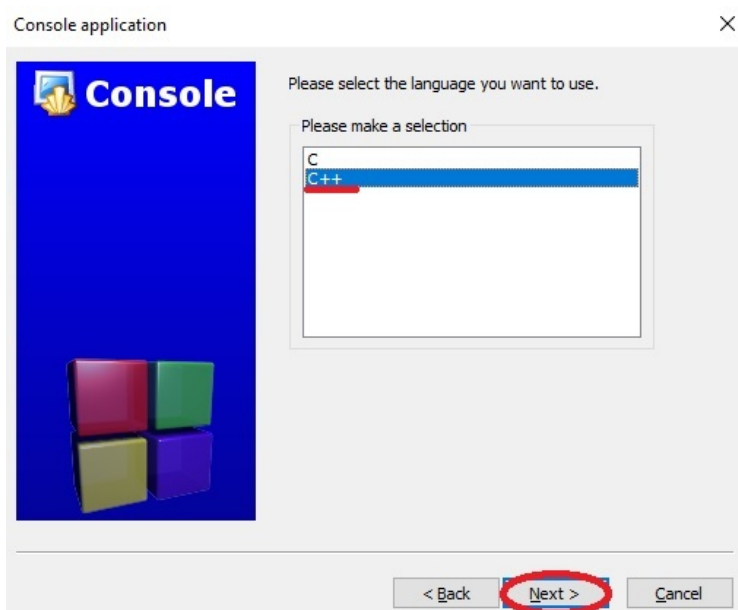
Чтобы создать новый проект, запустите Code::Blocks, выберите "File" > "New" > "Project":



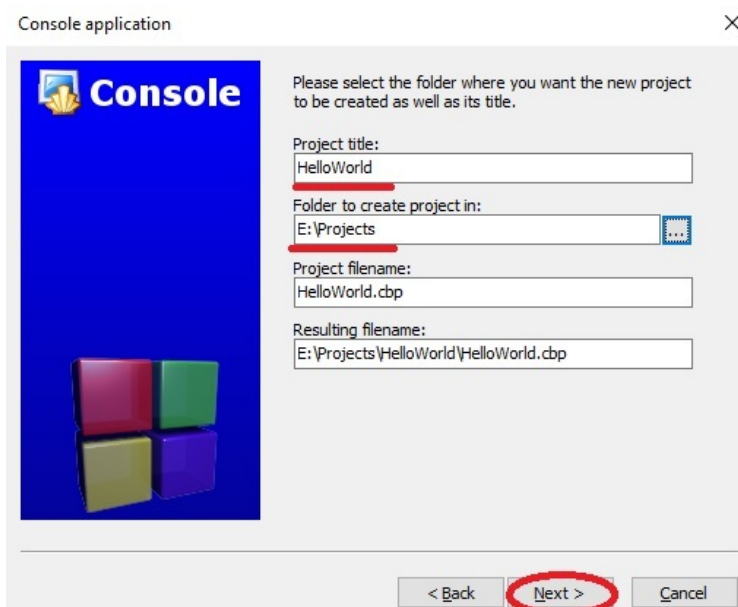
Затем появится диалоговое окно, где вам нужно будет выбрать "Console application" и нажать "Go":



Затем выберите язык "C++" и нажмите "Next":

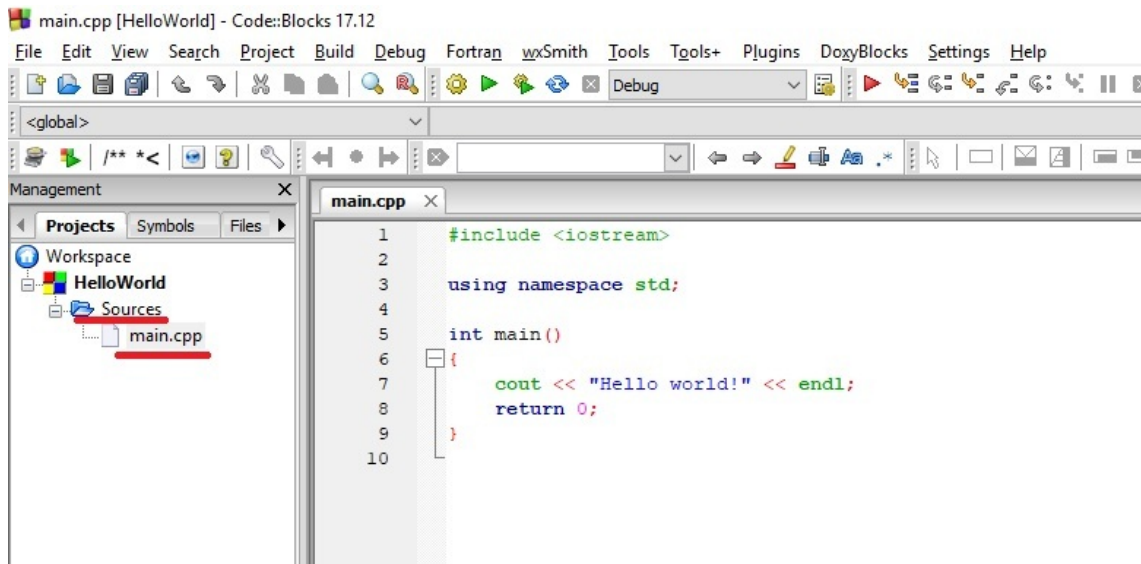


Затем нужно указать имя проекта и его расположение (можете создать отдельную папку *Projects*) и нажать "Next":



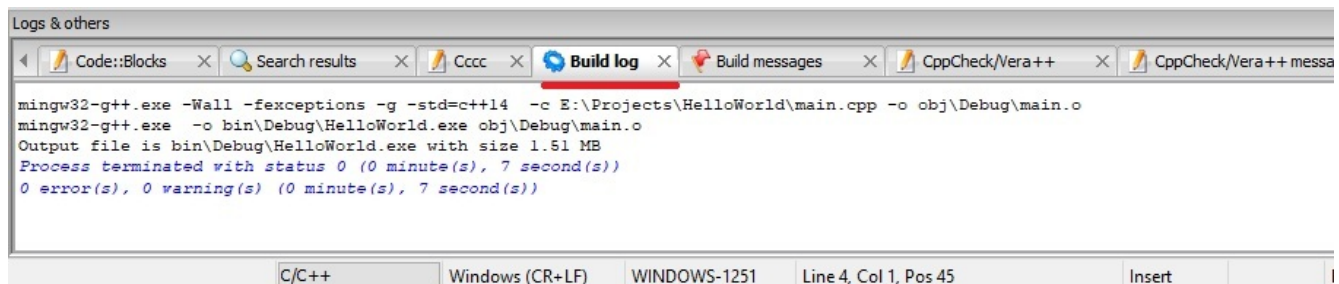
В следующем диалоговом окне нужно будет нажать "Finish".

После всех этих манипуляций, вы увидите пустое рабочее пространство. Вам нужно будет открыть папку *Sources* в левой части экрана и дважды кликнуть по *main.cpp*:



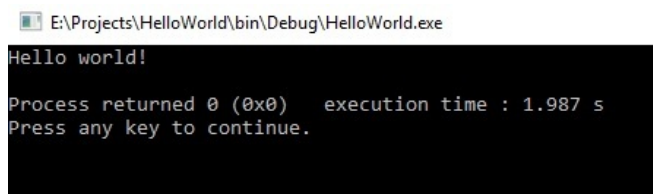
Вы увидите, что программа «Hello, world!» уже написана!

Для того, чтобы скомпилировать ваш проект в Code::Blocks, нажмите **Ctrl+F9**, либо перейдите в меню "Build" и выберите "Build". Если всё пройдет хорошо, то вы увидите следующее во вкладке "Build log":



Это означает, что компиляция прошла успешно!

Чтобы запустить скомпилированную программу, нажмите **Ctrl+F10**, либо перейдите в меню "Build" и выберите "Run". Вы увидите следующее окно:



Это результат выполнения вашей программы.

Пользователям командной строки

Вставьте следующий код в текстовый файл с именем *HelloWorld.cpp*:

```
1 #include <iostream>
2
3 int main()
4 {
5     std::cout << "Hello, world!" << std::endl;
6     return 0;
7 }
```

В командной строке напишите:

```
g++ -o HelloWorld HelloWorld.cpp
```

Эта команда выполнит компиляцию и линкинг файла *HelloWorld.cpp*. Для запуска программы напишите:

HelloWorld

Или:

./HelloWorld

И вы увидите результат выполнения вашей программы.

Пользователям веб-компиляторов

Вставьте следующий код в рабочее пространство:

```
1 #include <iostream>
2
3 int main()
4 {
5     std::cout << "Hello, world!" << std::endl;
6     return 0;
7 }
```

Затем нажмите "Run". Вы должны увидеть результат в окне выполнения.

Пользователям других IDE

Вам нужно:

Шаг №1: Создать консольный проект.

Шаг №2: Добавить файл .cpp в проект (если нужно).

Шаг №3: Вставить следующий код в файл .cpp:

```
1 #include <iostream>
2
3 int main()
4 {
5     std::cout << "Hello, world!" << std::endl;
6     return 0;
7 }
```

Шаг №4: Скомпилировать проект.

Шаг №5: Запустить проект.

Если компиляция прошла неудачно (а.к.а. «О Боже, что-то пошло не так!»)

Всё нормально, без паники. Скорее всего, это какой-то пустяк.

- ➔ Во-первых, убедитесь, что вы написали код правильно: без ошибок и опечаток. Сообщение об ошибке компилятора может дать вам ключ к пониманию того, где и какие ошибки случились.
- ➔ Во-вторых, просмотрите [Урок №7](#) — там есть решения наиболее распространенных проблем.
- ➔ Если всё вышесказанное не помогло — «загуглите» проблему. С вероятностью 90% кто-то уже сталкивался с этим раньше и нашел решение.

Заключение

Поздравляем, вы написали, скомпилировали и запустили свою первую программу на языке C++! Не беспокойтесь, если вы не понимаете, что означает весь этот код, приведенный выше. Мы детально всё это рассмотрим на следующих уроках.

Оценить статью:

★★★★★ (633 оценок, среднее: 4,88 из 5)



← [Урок №4. Установка IDE \(Интегрированной Среды Разработки\)](#)



Комментариев: 70



1. Рома:

[6 октября 2019 в 14:39](#)

Скиньте ссылку на Visual Studio 2017 Community не могу найти по ссылке переходил выходит версия 2019 года

[Ответить](#)



1. Юрий:

[6 октября 2019 в 15:16](#)

На оф.ресурсе Майкрософта ссылки на эту версию уже нет(

[Ответить](#)



1. Рома:

[7 октября 2019 в 16:55](#)

Спасибо уже не надо. И да там есть на оф. сайте Microsoft VS 2019 <https://visualstudio.microsoft.com/ru/downloads/> пролистав ниже можно найти предыдущие версии <https://visualstudio.microsoft.com/ru/vs/older-downloads/> после нужно оформить БЕСПЛАТНУЮ подписку и там выйдут куча версий и там уже искать ВСЕ ЭТО БЕСПЛАТНО ЕСЛИ НЕ БРАТЬ ЗА ДЕНЬГИ.

[Ответить](#)



2. оля:

[17 июля 2019 в 05:15](#)

Здравствуй, Юрий. Я уже оставляла сообщение. Сейчас открыла, а его нет. Почему так? В общем у меня ошибка error с 2065 и с 2039. Может быть это оттого, что я неправильно скачала программное обеспечение? сначала я поставила галочку только возле 'Пакета sdk для windows 10', все остальные галочки убрала. Запустила установщик. Программа установилась. Потом я поняла и зашла заново, поставила недостающие галочки, нажала на клавишу "Изменить", программа снова закачалась. Все ли правильно я сделала?

[Ответить](#)



1. Юрий:

[18 июля 2019 в 12:59](#)

Привет. В уроке №4 ведь всё подробно расписано, как устанавливать и какие галочки ставить — есть скрины. Для того, чтобы проверить, всё ли верно запустилось — пишите программу «Hello, world!». Если работает, значит, что всё гуд.

[Ответить](#)



3. Сергей:

[9 июля 2019 в 18:02](#)

День добрый. У меня отсутствует в библиотеке — "pch.cpp"
И при написании всего кода выше - "Hello world!" — выдает ошибку.
а если не приписывать — include "pch.cpp", все работает корректно.
Обязательно ли его прописывать?

Microsoft visual studio 2019

[Ответить](#)



1. Юрий:

[10 июля 2019 в 18:00](#)

Не обязательно прописывать.

[Ответить](#)

4. *Арте́м:*[22 июня 2019 в 14:50](#)

А обязательно оставлять какое-то пространство между командами?

[Ответить](#)1. *Slav:*[29 июля 2019 в 02:41](#)

нет, необязательно. Но для того, чтобы легче читать код и Вам и другим людям существует "практика отсупов."

<https://tproger.ru/translations/stanford-cpp-style-guide/>

[Ответить](#)5. *Нубик в программировании:*[19 июня 2019 в 14:32](#)

Классные уроки!!!

[Ответить](#)6. *Владимир:*[15 мая 2019 в 09:27](#)

Попытался сделать через командную строку а мне пишут "g++ не является командой исполняемой программой или пакетным файлом"

[Ответить](#)1. *Mario Cassar:*[30 сентября 2019 в 11:56](#)

g++ — Линуксовый компилятор...

[Ответить](#)7. *Анастасия:*[26 апреля 2019 в 22:57](#)

Здравствуй! Установила по ссылке из 4го урока Visual Studio 2019. После перезагрузки компьютера (она требовалось установщиком) не нашла новой иконки на рабочем столе. После долгих поисков нужной иконки в новых установленных папках, в итоге запустила программу через установщик... Затем пришлось зарегистрировать учётную запись и зайти под ней.

Первую программу удалось запустить по инструкции успешно, НО только после удаления строчки "#include <pch.h>", хотя Вы пишете, что эта строчка для работы в VS обязательна. У меня есть два вопроса:

1) Как сделать иконку для запуска Visual Studio 2019 на рабочем столе?

2) Может быть, с новыми версиями VS отпала необходимость в строчке "#include <pch.h>"? С этой строчкой первая программа не запускается, fatal error C1083: Не удастся открыть файл включения: pch.h: No such file or directory

[Ответить](#)1. *Вадим:*[30 мая 2019 в 15:36](#)

Тоже после удаления этой строчки, все запустилось

[Ответить](#)2. *максим:*[30 мая 2019 в 18:24](#)

Если Вы не знаете таких простых вещей у меня вопрос. Зачем вы учите C++?

[Ответить](#)1. *Вадим:*[31 мая 2019 в 01:01](#)

Ага, это из той серии... А зачем ты идешь на работу, если у тебя даже опыта нет? Не за тем ли, чтобы его получить?
P.S. Очень важно научиться, не отвечать вопросом на вопрос.

[Ответить](#)



2. Анастасия:

[5 июня 2019 в 10:07](#)

Максим, Вы про иконку? Да, я и вправду не знаю, как её найти, откуда вытащить. А Вы знаете? Так подскажите! Я из-за этой "простой" трудности стала пользоваться другой IDE (DEV C++)
Не думаю, что эта "простая" вещь связана с моей мотивацией изучать C++. В остальном почти всё идёт гладко, я уже на 55 уроке, а Вы?

[Ответить](#)



1. Niik:

[21 сентября 2019 в 19:52](#)

"Пуск"Все программы VS Community запуск



3. Юрий:

[6 июня 2019 в 14:58](#)

Максим, давайте не будем говорить о простых вещах и о том, кто их должен знать по умолчанию. Процесс изучения чего-либо подразумевает появление вопросов/проблем и их решение.

[Ответить](#)



4. loser:

[1 августа 2019 в 14:13](#)

Максим, вот приходишь ты в первый раз в первый класс, а училка те говорит "Таких простых вещей не знаешь, зачем тогда учишь?"

[Ответить](#)



3. Анастасия:

[5 июня 2019 в 13:43](#)

Откопала-таки ярлык Visual Studio, чтобы вытащить его на рабочий стол. Для тех, кто столкнулся с такой же проблемой, вот путь к ярлыку, если устанавливали по инструкции:
C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\IDE
файл называется devenv и имеет тип "Приложение"

[Ответить](#)



1. Юрий:

[6 июня 2019 в 14:51](#)

Я тоже не мог найти этот ярлык никак при установке именно новых версий Visual Studio. Спасибо вам за решение.

[Ответить](#)



2. Valery:

[9 августа 2019 в 16:06](#)

Ярлык для Visual Studio 2019 (и не только) — это очень просто:
Пуск — Мои Программы (это принципиально) — правый клик на Visual Studio 2019 — Отправить (ярлык на рабочий стол).
Всё! Ярлык на месте, т. е. на рабочем столе, но с невнятным названием. Исправляем: правый клик на ярлыке — Переименовать — вместо devenv.exe пишите — Visual Studio 2019 (или то, что Вам захочется) — ОК (Enter). Успешной работы!

[Ответить](#)



8. Бабкен:

[3 апреля 2019 в 17:30](#)

Здравствуйте, Юрий. Огромное спасибо вам за лекции!

Я скачал себе Visual studio 2019 года. При создании консольного приложения у меня отсутствует файл pch.h и программа работает без него. Я видел комментарий, связанный с тем, что человек запускал пустой проект, но я запускаю консольное приложение. Может это связано с тем, что в 2019 году они обновили версию и убрали данный файл?

[Ответить](#)



1. *Никита:*

[10 апреля 2019 в 19:18](#)

Чтобы это решить надо зайти в Проект->Свойства->C/C++->Все параметры И параметр "Предварительно откомпилированный заголовок" поставить на Не использовать предварительно....

[Ответить](#)



1. *Алексей:*

[26 апреля 2019 в 22:40](#)

У меня была такая же проблема. Выяснилось, что VS 2019 ни в какую не хочет сам создавать файлы pch.cpp и pch.h, создал вручную (ничего не писал в обоих) — заработало... Вот только вопрос, это вообще нормально, что пришлось их вручную создавать?

[Ответить](#)



2. *Дмитрий:*

[17 мая 2019 в 21:40](#)

Добрый день, также проблема. На Visual Studio 2019 не создаются Файлы pch.cpp и pch.h, если не добавлять строку #include "pch.h", то все работает нормально, при добавлении пришлось это Файлы создавать вручную. Изменения в свойствах Предварительно откомпилированный заголовок — На не использовать предварительно ... Ничего не дало. Интересно как — то решается эта проблема?

[Ответить](#)



1. *Юрий:*

[6 июня 2019 в 15:11](#)

Если у вас всё работает без добавления строчки #include <pch.h>

То совершать лишние телодвижения не нужно.



9. *Dweeni:*

[23 февраля 2019 в 15:13](#)

Связано с Visual Studio 2017.

Когда хочу создать проект консольный его просто нету для си++.
как не искал не смог найти что делать?

[Ответить](#)



10. *Эзи:*

[17 февраля 2019 в 20:12](#)

ДА! у меня получилось на DevC++))

[Ответить](#)



11. *Олег:*

[4 января 2019 в 11:54](#)

Юрий, спасибо за науку. Такой вопрос — все прошло успешно, никто не ругался при сборке, все рк. Но когда запускаю — окно всплывает на долю секунды и пропадает, даже прочитать ничего не успеваешь. Это в чем проблема?

[Ответить](#)



1. *Юрий:*

[4 января 2019 в 16:58](#)

Ответ есть в уроке №7.

[Ответить](#)



12. *Pifagores:*
[11 ноября 2018 в 14:48](#)

Спасибо, всё здорово!

Только в C++ для WIN 7x32 клавиша запуск "F5",

А подключаемый файл командой `#include "pch.h"`,
вместо `#include "stdafx.h"`

[Ответить](#)

R

1. *Юрий:*
[13 ноября 2018 в 13:26](#)

В версии Visual Studio 2017:

```
1 #include "stdafx.h"
```

Заменяли на:

```
1 #include "pch.h"
```

[Ответить](#)



1. *Аиот:*
[12 декабря 2018 в 07:19](#)

```
1 #include "pch.h"
```

это меня спасло спасибо!

[Ответить](#)



2. *Евген:*
[14 января 2019 в 16:14](#)

Я в VS 2017 пишу на пустом проекте и у меня не требует pch.h, проекты работают. Может я делаю что-то не так?

[Ответить](#)

R

1. *Юрий:*
[15 января 2019 в 20:28](#)

Всё верно. Вы выбрали создать "Пустой проект", а не "Консольное приложение". В "Пустом проекте" подключение предварительно скомпилированного заголовка pch.h не требуется.



13. *илья:*
[3 октября 2018 в 19:31](#)

мой visual studio community 2017 внезапно стал платным, подскажите что делать?

[Ответить](#)



1. *Константин:*
[24 октября 2018 в 21:51](#)

Илья, а ты регистрировался на сайте Microsoft-а при скачивании загрузочного файла? Я вот только надумал этот Visual Community 2017 скачать и на твой вопрос наткнулся — сразу подумал: это если я зарегистрируюсь, то они и до моей ОС доберутся. И вообще, народ! Это только Илья предложили раскошиться или кто-то ещё был "удивлён"?!

[Ответить](#)



1. *Денис (MegaTech):*
[14 ноября 2018 в 23:12](#)

Кхм, если он брал не Community версию VS или Community, но не регался — с него попросят. Если Комьюнити — зарегайся, если не Комьюнити — ставь её или плати.

[Ответить](#)



2. *Геннадий:*
[18 мая 2019 в 23:37](#)

Платить!

[Ответить](#)



14. *Кох:*
[18 сентября 2018 в 17:46](#)

MVS 2017.
 вместо "stdafx.h" используется "pch.h".
 мне жалко мои потерянные нервы

[Ответить](#)



1. *Дмитрий:*
[20 сентября 2018 в 14:41](#)

Спасибо тебе большое , я уже второй день не мог понять в чём проблема

[Ответить](#)



2. *Pifagores:*
[11 ноября 2018 в 14:55](#)

Согласен, у меня ещё вместо F7 на выполнение клавиша F5

[Ответить](#)



15. *Artem:*
[3 сентября 2018 в 23:31](#)

Дружище ,огромное спасибо!Перепробовал кучу книг,статей и прочих источников для «чайников» ,но ничего понятного для меня на находилось.Не было понятной структуры ,что и для чего.Подписался ,надеюсь дальше будет все с таким же формате.Еще раз спасибо,если и дальше так пойдёт ,придётся донатить тебе ,ибо не донатить совесть не позволит ,за такой понятный материал))

[Ответить](#)



1. *Юрий:*
[3 сентября 2018 в 23:41](#)

Спасибо, друг, что читаешь 😊 Сам искал понятный ресурс с четкой структурой, когда удалось найти — решил поделиться с Рунетом.

[Ответить](#)



16. *Константин:*
[31 августа 2018 в 06:46](#)

Юра, пожалуйста, распиши как создавать новый workspace. Спасибо.

[Ответить](#)



17. *Евгений:*
[6 июля 2018 в 10:14](#)

Офигеть, наконец то кто-то нормально рассказал и учёл все возможные варианты. Спасибо, на данный момент автор — лучший в рунете (на моей памяти)

[Ответить](#)



1. *Юрий:*

[10 июля 2018 в 00:29](#)

Хорошими знаниями нужно делиться 😊

[Ответить](#)

18. Даниил:

[7 июня 2018 в 11:02](#)

У меня нет Win32. Вроде бы я всё устанавливал, правильно. С чем это связано? Можно ли использовать обычный файл?

[Ответить](#)

1. Юрий:

[7 июня 2018 в 19:08](#)

В обновлениях Visual Studio Win32 убрали. Ориентируйтесь на то, чтобы было Console Application и C++ — это самое главное.

[Ответить](#)

1. Станислав:

[18 ноября 2018 в 19:56](#)

Здравствуйте. Наткнулся на такую проблему. У меня нету пункта win 32 и Console Application. Только консольное приложение windows. Подскажите что делать.

[Ответить](#)

1. Юрий:

[18 ноября 2018 в 21:10](#)

Привет, выбирать "Консольное приложение Windows".



19. SV:

[29 мая 2018 в 15:59](#)

пробую использовать Code::Blocks

программа из примера компилируется, запускается.

Потом закрываю консольное окно и пробую редактировать текст Hello world. Пробую снова компилировать — ошибка. Жду некоторое время(несколько минут) — снова компилирую — компилируется нормально. Если редактировать и сразу компилировать — то компилируется нормально, а если пробовать запускать — то после нормального запуска и последующей попытки редактирования — ошибка. и надо ждать несколько минут.

Проблема связана с консолью? Что то где то ждёт не закрытое\не завешенное?

[Ответить](#)

1. Юрий:

[29 мая 2018 в 16:20](#)

Вы после редактирования сохраняете изменения? И когда вы запустили и программа выполнилась — вы после этого останавливаете режим отладки?

[Ответить](#)

1. SV:

[30 мая 2018 в 08:21](#)

Я по разному пробовал (с поправкой только начал вообще знакомиться с предметом и средой). И сохранять пытался, и стирать, но всё это работает пока не запущу программу на выполнение. После того как она выполнилась(окно закрываю!) — редактирование и компиляция без ошибок возможны только по истечении некоторого времени.

[Ответить](#)

1. Юрий:

[30 мая 2018 в 23:44](#)

Тогда ничего не могу подсказать по решению этой проблемы. Обратитесь на форумы по программированию с вашим вопросом.

20. *ctcu:*[30 марта 2018 в 09:56](#)

Добрый день!

Несколько дней ушло на не найденный ответ по ошибке:
 не удалось загрузить импортированный файл проекта
 C:\Users\...\AppData\Local\Microsoft\MSBuild\v4.0\
 Microsoft.Cpp.Win32.user.props Отсутствует корневой элемент
 Где можно посмотреть?

[Ответить](#)1. *Юрий:*[30 марта 2018 в 11:45](#)

«Система проектов Visual C++ основана на MSBuild, значения хранятся в XML-файле проекта по умолчанию .props и .targets файлов, что (для Visual Studio 2015) находятся в ВашДиск\Program Files(x86)\MSBuild\Microsoft.Cpp\v4.0\V140 и в пользовательских PROPS-файлов, которые требуется добавить. Настоятельно рекомендуется НЕ изменять файлы вручную» —
msdn.microsoft.com/ru-ru/library/.

Попробуйте удалить весь свой Visual Studio и установить заново в точном соответствии с [уроком 4](#).

[Ответить](#)21. *Андрей Оганесян:*[21 марта 2018 в 00:10](#)

Здравствуй, такая проблема: всё написал, скомпилировал, но при запуске в консоли выдаётся "Для продолжения нажмите любую клавишу . . ." и никакого hello world..(Помогите может я что-то не так сделал?)

[Ответить](#)1. *Юрий:*[21 марта 2018 в 01:10](#)

Привет, смотрите [урок 7](#).

[Ответить](#)22. *Natig:*[20 февраля 2018 в 17:01](#)

на visual 2017 не как не хочет компелировать, выдает ошибку windowsproject4.cpp(8): fatal error C1010: непредвиденный конец файла во время поиска

[Ответить](#)1. *Юрий:*[20 февраля 2018 в 17:19](#)

Скорее всего, вы не добавили stdafx.h:

```
1 | #include "stdafx.h"
```

Смотрите [Урок 7. Самые распространенные проблемы в C++](#).

[Ответить](#)1. *Koh:*[18 сентября 2018 в 17:48](#)

pmtscr.com/kvua5u

изменено название файла

[Ответить](#)23. *Pavel:*

[28 ноября 2017 в 23:22](#)

Спасибо за описание. Программа запустилась и работает (правда пришлось повозиться с установкой компилятора, писал, что `can't find compiler executable in your search path (GNU GCC compiler)`). Вопрос решил. Но если вместо «Hello world» написать что-либо на кириллице — то на экране видна абракадабра. В чем проблема? Использую Code::Blocks

[Ответить](#)

1. Юрий:

[29 ноября 2017 в 01:35](#)

Проблема в кодировке. Для корректного отображения кириллицы в консольном окне — добавьте заголовочный файл:

```
1 #include <locale>
```

и пропишите следующую строчку кода в `main`:

```
1 setlocale(LC_CTYPE, "rus");
```

Либо, если вы еще планируете вводить данные на кириллице, тогда подключите заголовочный файл:

```
1 #include <Windows.h>
```

и пропишите в `main`:

```
1 SetConsoleCP(1251); SetConsoleOutputCP(1251);
```

[Ответить](#)

24. Jaan:

[19 ноября 2017 в 20:01](#)

Многообещающее начало изучения сложной для меня темы такой как `с++` очень много вопросов как и положено новичку. Некоторые моменты хотелось бы поподробнее получить в виде объяснения. В общем пока я доволен вашим сайтом. Спасибо!

[Ответить](#)

1. Юрий:

[19 ноября 2017 в 21:16](#)

Пожалуйста, читайте — учитесь.

[Ответить](#)

1. Новичок:

[28 ноября 2017 в 18:00](#)

Как создать первую программу есть, но нет объяснения, что мы, первом делом подключаем библиотеки, что такое `std` и что такое `cout`, почему в конце стоит `return 0`, и было бы не плохо добавить вариации кода... По мне, так этот код выглядит легче:

```
1 #include
2 using namespace std;
3
4 int main()
5 {
6     cout << "Hello, World!\n";
7     return 0;
8 }
```

Директива `using`, дает непосредственный доступ к элементам из пространства имен `std`.

Но есть и такой вариант:

```
1 #include
2 using namespace std;
3 using std::cout;
4 using std::endl;
5
6 int main()
7 {
8     cout << "Hello, World!" << endl;
```

```
9 | return 0;  
10| }
```

[Ответить](#)

1. *Юрий:*
[28 ноября 2017 в 18:15](#)

Этот урок посвящен первой компиляции первой программы. Что такое return, cout, std — рассказывается детально в следующих соответствующих уроках.

Добавить комментарий

Ваш E-mail не будет опубликован. Обязательные поля помечены *

Имя *

Email *

Комментарий

☐ Сохранить моё Имя и E-mail. Видеть комментарии, отправленные на модерацию

☐ Получать уведомления о новых комментариях по электронной почте. Вы можете [подписаться](#) без комментирования.






[TELEGRAM](#)  [КАНАЛ](#)

Электронная почта



[ПАБЛИК](#) 

ТОП СТАТЬИ

-  [Словарь программиста. Сленг, который должен знать каждый кодер](#)
-  [Урок №1. Введение в программирование](#)
-  [70+ бесплатных ресурсов для изучения программирования](#)
-  [Урок №1: Введение в создание игры «Same Game»](#)
-  [Урок №4. Установка IDE \(Интегрированной Среды Разработки\)](#)

- [Ravesli](#)
- - [О проекте](#) -
- - [Пользовательское Соглашение](#) -
- - [Все статьи](#) -
- Copyright © 2015 - 2020