

Ravesli [Ravesli](#)


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

Урок №29. Тип данных void

 [Юрий](#) |

- [Уроки C++](#)

|

 Обновл. 2 Сен 2020 |

 36020

[↓](#)  8

Тип void — это самый простой тип данных, который означает «отсутствие любого типа данных». Следовательно, переменные не могут быть типа void:

```
1 void value; // не будет работать, так как переменная не может иметь тип void
```

Тип void, как правило, используется в 3-х случаях:

Использование №1: Указать, что функция не возвращает значение:

```
1 void writeValue(int x) // здесь void означает, что функция не возвращает никакого значения
2 {
3     std::cout << "The value of x is: " << x << std::endl;
4     // Нет стейтмента return, так как тип функции - void
5 }
```

Использование №2: Указать, что функция не имеет никаких [параметров](#) (перешло из языка Си):

```
1 int getValue(void) // здесь void означает, что функция не имеет никаких параметров
2 {
3     int x;
4     std::cin >> x;
5     return x;
6 }
```

Указание типа void как «никаких параметров» является пережитком, сохранившимся еще со времен языка Си. Следующий код равнозначен и более предпочтителен для использования в языке C++:

```
1 int getValue() // пустые скобки означают то же, что и void
```

```
2 | {  
3 |     int x;  
4 |     std::cin >> x;  
5 |     return x;  
6 | }
```

Правило: Используйте пустой список параметров вместо void для указания отсутствия параметров в функции.

Использование №3: Ключевое слово void имеет третий (более продвинутый) способ использования в языке C++, который мы будем рассматривать на [уроке №92](#).

Оценить статью:

★★★★★ (407 оценок, среднее: 4,95 из 5)



[← Урок №28. Инициализация, присваивание и объявление переменных](#)

[Урок №30. Размер типов данных](#)



Комментариев: 8



1. *Константин П:*
[1 июля 2020 в 11:36](#)

Если все-таки разделять понятия "процедура" и "функция" (понимаю, что в С понятие "процедура" отсутствует), то void станет более понятным.

Процедура — это просто последовательность действий.

Функция — это как бы "ответ на вопрос".

"Отсортируй переданный в параметрах массив" — процедура (void).

"Что получится, если перемножить переданные параметры?" — функция.

[Ответить](#)



2. *Андрей:*
[24 февраля 2020 в 17:07](#)

Все конспектировал)
довольно понятно, когда перечитываешь)

[Ответить](#)



3. *Константин:*
[9 марта 2019 в 11:48](#)

Ну представь ветку (яблони), на ней вырастает плод, ты сорвал его и съел (продал). Это и есть void. А не void д. б. бы этот плод обратно в ствол дерева втулить (вернуть).

[Ответить](#)



1. *Иван:*

[12 марта 2019 в 09:57](#)

Плохой пример. Лучше представить ветку, которая не дает плодов.

[Ответить](#)



1. *Константин:*

[24 марта 2019 в 12:18](#)

Нет, хороший пример! Функция ведь производит некое значение. Просто не возвращает его в тело программы, а показывает его пользователю например на экране монитора.

[Ответить](#)



1. *Sasha:*

[8 января 2020 в 20:08](#)

функция void не производит значение



4. *Алексей:*

[29 января 2019 в 14:47](#)

Какой же этот void задрочливый и сложен для понимания (для меня)((((((

[Ответить](#)



1. *Artemiy:*

[18 января 2020 в 15:06](#)

Есть такое по началу....

[Ответить](#)

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.

Отправить комментарий






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