

# Типы данных. Зачем они нужны?

Предполагается, что Вы всё ещё помните с прошлого шага, что такое переменная. Если не помните, вернитесь и посмотрите. А мы продолжаем.

Кроме имени и области в памяти, любая переменная имеет такую важную характеристику как **тип данных**, которые будут храниться в этой переменной. Постараюсь пояснить.

Как уже говорилось ранее, программы работают с различными данными, а не только с числами. И если для человека записи "x=3", "x=3.0" и "x = три" несут одинаковую смысловую нагрузку, то для компьютера это не так очевидно. Мы понимаем, что в каждой из этих записей говорится, что x равен трём. Компьютер же пока не настолько умен, чтобы это осознать. Ему нужно точно знать, с какими данными он будет работать. И дело даже не в том, число это или слово, ему важно даже то, целое это число или вещественное. Поэтому все переменные имеют дополнительную характеристику -- тип данных.

Наверное, вам известно, что в памяти компьютера все числа хранятся в виде последовательности 0 и 1. Так вот, для того чтобы записать любое число только ноликами и единицами существуют определённые правила. Эти правила довольно сильно отличаются для целых чисел и для вещественных чисел. В памяти компьютера числа "3" и "3.0" будут записаны совершенно по-разному. Для тех, кто заинтересовался представлением чисел в памяти компьютера, в конце урока я дам ссылки на дополнительные материалы по этой теме.

Кроме того, зачастую компьютеры гораздо быстрее и лучше могут производить различные действия с целыми числами, чем с вещественными. Если вы не забросите этот курс, то на одном из следующих уроков мы в этом убедимся. Раньше, когда компьютеры занимали целую комнату, это была безусловная истина, а теперь этот факт уже нужно проверять.

Подведём итог. Каждая переменная должна иметь конкретный тип данных. И в эту переменную можно сохранить данные только этого типа.

Возвращаясь к аналогии с ящиками, можем думать, что существуют различные ящики. Одни ящики подходят только для целых чисел, другие - только для вещественных чисел, третьи - только для хранения букв. И нельзя в ящик для букв положить вещественное число. Точнее положить можно, но тогда ваша программа работать не будет. Очень важно следить за этим делом и не путать мух с котлетами. Порядок должен быть в этом деле.

## Базовые типы данных в языке Си.

Ниже выписаны некоторые (не все!) базовые типы данных, которые есть в языке Си, и которые вы будете использовать в ваших программах чаще всего.

- целые числа – тип int,
- вещественные числа – тип double

- и символы – тип char.

Есть и другие типы данных, но мы пока о них говорить не будем, чтобы не забивать свою память различными нюансами, которые на начальном этапе не так уж важны.

Важно другое. Необходимо понимать, что тип данных для переменной нужно выбирать в зависимости от того, что мы собираемся в этой переменной хранить. Например, если бы мы считали количество посещений какой-то страницы на сайте, то для этого мы использовали переменную типа int, а вот для хранения курса доллара целые числа уже бы не подошли. Думаю, у банка возникли бы серьёзные проблемы после использования такой программы.

## Практика

Решите предложенные [задачи с автоматической проверкой](#).

Исследовательские задачи для хакеров:

1. Под переменные различных типов данных выделяется различное количество памяти. В стандарте языка определены только соотношения между этими размерами. Ваша задача самостоятельно изучить как работает функция sizeof(); и выяснить размеры типов данных на вашем компьютере. (выполнять после прохождения всего второго урока)

## Дополнительные материалы.

1. Язык Си является языком со статической слабой явной типизацией. Хотите узнать что это такое? Читайте статью [о типизации в различных языках программирования](#)
2. В уроке обещал подобрать годный материал о том, как хранятся различные числа в памяти компьютера. Но беглый поиск по сети не дал удовлетворительных результатов. Поэтому вот статья о том, как хранятся вещественные числа в компьютере и какие с ними происходят казусы. [Ссылка](#)

Интернет версия: [http://youngcoder.ru/less2/typy\\_dannyh.php](http://youngcoder.ru/less2/typy_dannyh.php)