**Преобразование типов**

[Java Syntax](https://javarush.ru/quests/QUEST_JAVA_SYNTAX)

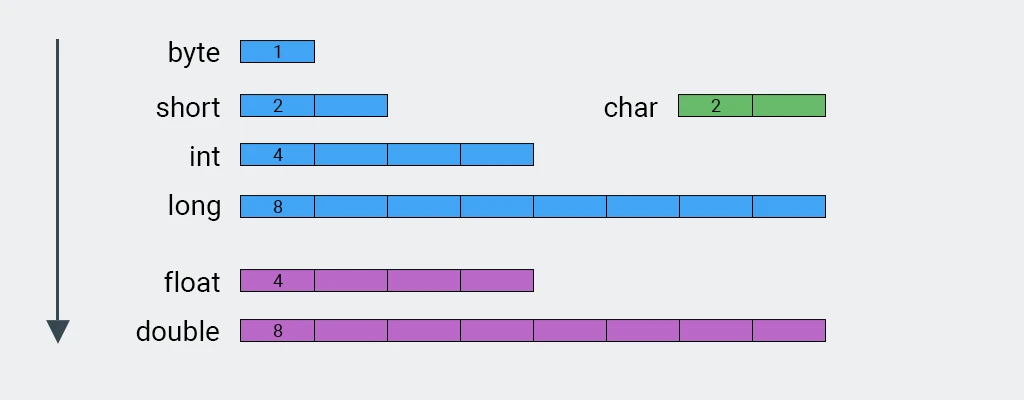
[Уровень 10](https://javarush.ru/quests/lectures?quest=QUEST_JAVA_SYNTAX&level=10), Лекция 3

— А теперь самое интересное. Я расскажу тебе про преобразование типов. Хоть типы переменных всегда неизменны, есть место, где можно проводить преобразование типов. И место это – **присваивание**.

— Можно присваивать друг другу переменные разных типов. При этом значение, взятое из переменной одного типа, будет преобразовано в значение другого типа и присвоено второй переменной.

— В связи с этим, можно выделить два вида преобразования типов: расширение и сужение. Расширение похоже на перекладывание из маленькой корзинки в большую – операция проходит незаметно и безболезненно. **Сужение типа** – это перекладывание из большой корзинки в маленькую: места может не хватить, и что-то придётся выбросить.

— Вот типы, отсортированные по размеру «корзинки»:



— Тут есть пара замечаний:

1. **char** такая же корзинка, как и **short**, но свободно перекладывать из одной в другую нельзя: при перекладывании значения из **short** в **char**, всегда будут теряться значения меньше 0. При перекладывании из char в short будут теряться значения больше 32-х тысяч.

2. При преобразовании из целых чисел в дробные могут отбрасываться самые младшие части числа. Но т.к. смысл дробного числа в том, чтобы хранить приблизительное значение, то такое присваивание разрешается.

— При сужении типа мы должны явно показать компилятору, что мы не ошиблись и отбрасывание части числа сделано намеренно. Для этого используется оператор приведения типа. Это **имя типа в круглых скобочках.**

— Вот как нужно присваивать переменные разных типов:

|  |  |
| --- | --- |
| **Код на Java** | **Описание** |
| byte a = 115;  int b = a; | Расширение типа. Все отлично. |
| int c = 10000;  byte d = (byte) c; | Сужение типа. Нужно явно отбросить лишние байты. |
| int c = 10;  byte d = (byte) c; | Сужение типа. Нужно явно отбросить лишние байты, даже если они равны 0. |
| float f = 10000;  long l = (long) (f \* f);  float f2 = l;  long l2 = (long) f2; | При присваивании к float, происходит расширение типа. При присваивании значения float к long, происходит сужение – необходимо приведение типа. |
| double d = 1;  float f = (float) d;  long l = (long) f;  int i = (int) l;  short s = (short) i;  byte b = (byte) s; | Сужение во всех операция присваивания, кроме первой строки, требует указания явного преобразования типа. |

3

Задача

Java Syntax,  10 уровень,  3 лекция

Набираем код Ӏ Java Syntax: 10 уровень, 3 лекция

Java Syntax: 10 уровень, 3 лекция. Иногда думать не надо, строчить надо! Как ни парадоксально звучит, порой пальцы «запоминают» лучше, чем сознание. Вот почему во время обучения в секретном центре JavaRush вы иногда встречаете задания на набор кода. Набирая код, вы привыкаете к синтаксису и зарабатываете немного материи. А ещё — боретесь с ленью.

— **Оператор приведения типа** нужно указывать перед числом/переменной каждый раз, когда происходит отбрасывание части числа или сужение типа. Он действует только на число/переменную, которое идет непосредственно за ним.

|  |  |
| --- | --- |
| **Код на Java** | **Описание** |
| float f = 10000;  long l = (long) f \* f; | К типу long приводится только одна переменная из двух: умножение long и float дает float. Данный пример кода не будет скомпилирован. |
| float f = 10000;  long l = (long) (f \* f); | Тут все выражение приводится к типу long. Этот пример кода будет скомпилирован. |

— Ясно.