**Элли, переменные и их типы**

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

[Уровень 0](https://javarush.ru/quests/lectures?quest=QUEST_JAVA_SYNTAX&level=0), Лекция 4



В каюту вошла шикарная женщина с розовыми волосами. «Интересно, неужели все человеческие женщины с такими волосами?» — успел подумать Амиго, но быстро смутился под ее взглядом.

— Привет! Меня зовут Эллеонора Керри, можно просто Элли. Я штурман корабля «Galactic Rush».

— Привет, Элли! – смущенно выдавил из себя Амиго. Он не знал почему, но чувствовал, что его щеки горят, как будто пробило маслопровод где-то внутри.

— Я расскажу тебе о самом интересном, что есть в языке Java – о переменных.

— Я готов слушать. А что такое эти переменные?

— Переменные – это специальная штука для хранения данных. Любых данных. Все данные в Java хранятся с помощью переменных. Больше всего переменная по смыслу напоминает коробку.

— Какую коробку?

— Да обычную. Вот, допустим, что ты написал на листке число 13 и положил этот лист в коробку, теперь мы можем говорить, что «коробка хранит значение 13».

— Каждая переменная в Java имеет три важных свойства: тип, имя и значение.

— А можно поподробнее?

— Конечно. Имя используется для того, чтобы отличать одну переменную от другой. Это как надпись на коробке.

— Тип переменной определяет тип значений/данных, которые в ней можно хранить. В коробке для торта храним торт, в коробке для обуви – туфли, и т.д.

— Значение – это некий объект, данные или информация, которая хранится в переменной.

— А можно про тип еще раз?

— Ок. Каждый объект в языке Java имеет свой тип. Например, могут быть такие типы данных: «целое число», «дробное число», «текст», «тип Кот», «тип Дом» и т.д.

— У переменной тоже есть свой тип. Переменная может хранить значения только того же типа, что и она сама.

— В реальной жизни это часто встречается. Разные коробки используются для хранения различных вещей:

[](https://cdn.javarush.ru/images/article/57971b73-8e41-4619-b16a-8b5fcbfcdb47/original.jpeg)

— Чтобы создать переменную используется команда вида: «тип имя».

Примеры:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Создание переменной: сначала тип, затем имя.** | | **Описание** |
| 1 |  | |  |
| int a; | | Создаем переменную по имени a типа int. | | |
| 2 | |  | | |

|  |  |
| --- | --- |
| String s; | Создаем переменную по имени s типа String. |
| 3 |  |

|  |  |
| --- | --- |
| double c; | Создаем переменную по имени c типа double. |

— Два самых часто используемых типа – это целые числа (обозначается словом int) и текст (обозначается словом String).

— А тип double?

— **double** – это дробные (вещественные) числа.

— Ты говорила, что у переменной три свойства: тип, имя и значение. А тут только два. У меня вопрос — а как занести значение в переменную?

— Если вернуться к аналогии с коробками, то представь, что ты взял листик, написал на нем число 42 и положил в коробку. Теперь коробка хранит значение 42.

— Ясно.

— Для занесения значения в переменную существует специальная операция – **операция присваивания**. Она копирует значение из одной переменной в другую. Не переносит, а именно копирует. Как файл на диске. Выглядит это так:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Код** | **Описание** | |
| 1 |  |  | |
| i = 3; | | | В переменную i заносится значение 3. | |
| 2 | | |  | |

|  |  |
| --- | --- |
| a = 1;  b = a+1; | В переменную a заносится значение 1. В переменную b заносится значение 2. |
| 3 |  |

|  |  |
| --- | --- |
| x = 3;  x = x + 1; | В переменную x заносится значение 3. В следующей строчке значение x увеличивается на 1, x теперь равен 4 |

— Для операции присваивания используется **символ равно** «**=**».

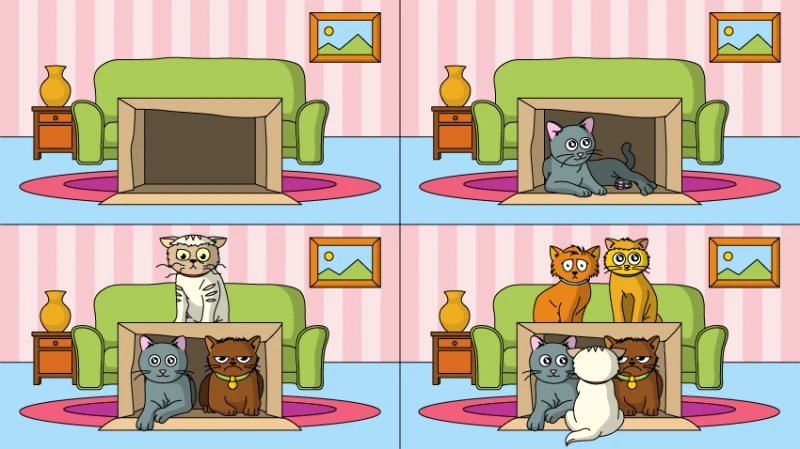
— Еще раз. **Это не сравнивание**. Это именно копирование значения справа от знака равно в переменную, которая слева. Для сравнения в языке Java используется двойное равно «==».

— А я знаю, как положить в переменную кота. Это почти как программа.

— Как поймать кота:

1 Возьмите пустую коробку.

2 Ждите.

[](https://cdn.javarush.ru/images/article/e79fe387-2f74-43d2-ad2f-1cf2449e6e62/original.jpeg)

— Нет, Амиго, в коробку можно запихнуть только одного кота. Тьфу! В переменную можно положить только одно значение.

— Ясно. А можно еще примеров, как создать переменные?

— ОК, повторюсь. Чтобы создать переменную, нужно написать такую команду «тип имя»

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Код** | **Пояснение** | |
| 1 |  |  | |
| String s; | | | Создается переменная s типа String. Такая переменная может хранить тексты. | |
| 2 | | |  | |

|  |  |
| --- | --- |
| int x; | Создается переменная x типа int. Такая переменная может хранить в себе целые числа. |
| 3 |  |

|  |  |
| --- | --- |
| int a, b, c;  int d; | Создаются переменные a, b, c, d типа int. Такие переменные могут хранить целые числа |

— О, теперь все понятно.

— Учти, что нельзя создать две переменных с одинаковыми именами в одном методе.

— А в разных методах?

— В разных методах – можно. Это как коробки стоящие в разных домах.

— А имя переменной может быть любым?

— Почти, но оно не может содержать пробелов, символов +,- и т.д. Лучше всего в имени переменной использовать только латинские символы и цифры.

— Обрати внимание, что **в языке Java играет роль, какие буквы ты напишешь – большие или маленькие**. «int a» – это не то же самое, что и «Int a» .

— Кстати, в Java можно одновременно создавать переменную и присваивать ей значение. Это экономит время и место:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Компактный код** | | | **Длинный код, эквивалентный коду слева** |
| 1 |  | | |  |
| int a = 5;  int b = 6; | |  |

|  |
| --- |
| int a;  a = 5;  int b;  b = 6; |
| 2 |  |

|  |  |
| --- | --- |
| int c = 7;  int d = c+1; |  |

|  |
| --- |
| int c;  c = 7;  int d;  d = c+1; |
| 3 |  |

|  |  |
| --- | --- |
| String s = "I’m Amigo"; |  |

|  |
| --- |
| String s;  s = "I’m Amigo"; |

— Так гораздо компактнее и понятнее.

— На том и стоим.

— Есть два типа, с которыми обязательно нужно познакомиться новичку в языке Java. Это типы int (целые числа) и String (текст/строки).

— Тип **int** позволяет хранить в переменной числа, а также выполнять над ними различные операции: сложение, вычитание, умножение, деление и другие.

|  |  |  |
| --- | --- | --- |
|  | **Код** | **Пояснение** |
| 1 |  |  |
| int x = 1;  int y = x\*2;  int z = 5\*y\*y + 2\*y + 3; | | | | x будет равен 1 y будет равен 2 z будет равен 20+4+3, будет равен 27 |
| 2 | | | |  |

|  |  |
| --- | --- |
| int a = 5;  int b = 1;  int c = (a-b) \* (a+b); | a будет равно 5 b будет равно 1 c будет равно 4\*6, будет равно 24 |
| 3 |  |

|  |  |
| --- | --- |
| int a = 64;  int b = a/8;  int c = b/4;  int d = c\*3; | a будет равно 64 b будет равно 8 c будет равно 2 d будет равно 6 |

— А мне все понятно. Все программирование такое легкое?

— Вообще-то да.

— Это хорошо. Так что там у тебя дальше?

— Тип **String** позволяет хранить текстовые строки.

— Чтобы задать какую-то текстовую строку, в Java надо написать текст строки, а с обеих сторон написать двойные кавычки. Пример:

|  |  |  |
| --- | --- | --- |
|  | **Код** | **Пояснение** |
| 1 |  |  |
| String s = "Amigo"; | | | | s будет содержать текст «Amigo». |
| 2 | | | |  |

|  |  |
| --- | --- |
| String s = "123"; | s будет содержать текст «123». |
| 3 |  |

|  |  |
| --- | --- |
| String s = "123 + 456"; | s будет содержать текст «123 + 456». |

— Я понял. Выглядит не сложно.

— Тогда вот тебе еще интересный факт.

— Строки можно склеивать с помощью знака плюс – «+». Пример:

|  |  |  |
| --- | --- | --- |
|  | **Код** | **Пояснение** |
| 1 |  |  |
| String s = "Amigo" + " is the best"; | | | | s будет содержать текст «Amigo is the best». |
| 2 | | | |  |

|  |  |
| --- | --- |
| String s = ""; | s будет содержать пустую строку – строку вообще без символов. |
| 3 |  |

|  |  |
| --- | --- |
| int x = 333;  String s = "Amigo" + x; | s будет содержать «Amigo333». |

— Так можно еще и складывать строки и числа?

— Да, но учти, что при сложении строк и чисел, всегда получается строка.

— Да я уже догадался по примеру.

— Ну, раз ты у нас такой догадливый, то догадайся, как вывести переменную на экран?

— Э-э-э. Переменную? на экран? Что-то ничего в голову не приходит.

— Тут на самом деле все просто. Чтобы вывести что-то на экран мы используем команду System.out.println(), куда в качестве параметра и передаем то, что мы выводим.

|  |  |  |
| --- | --- | --- |
|  | **Код** | **Вывод на экран** |
| 1 |  |  |
| System.out.**println**("Amigo"); | | | | Amigo |
| 2 | | | |  |

|  |  |
| --- | --- |
| System.out.**println**("Ami"+"go"); | Amigo |
| 3 |  |

|  |  |
| --- | --- |
| String s = "Amigo";  System.out.**println**(s); | Amigo |
| 4 |  |

|  |  |
| --- | --- |
| String s = "Am";  System.out.**println**(s+"igo"); | Amigo |

— Ага! Теперь все стало гораздо понятнее.

— Вот и отлично. Тогда тебе еще три задания.

2

Задача

Java Syntax,  0 уровень,  4 лекция

Комплимент учителю

В секретном центре JavaRush все учителя классные! Студенты отмечают это с самого нулевого уровня, но вслух говорят редко. А доброе слово и кошке приятно, не так ли? Лучшим комплиментом тому, кто учит программированию, будет программа. А если программа ещё и правильную фразу выведет, цены ей не будет! Пишем программу с комплиментом.

2

Задача

Java Syntax,  0 уровень,  4 лекция

Мой юный друг

В 3126 году все знают команду System.out.println() и пользуются ею очень часто. Не только студенты секретного центра JavaRush, даже дети начинают учить программирование именно с этой команды. Давайте напишем программу, в которой будет выведен истинный год рождения моего друга, родившегося 8 лет назад.

1

Задача

Java Syntax,  0 уровень,  4 лекция

Объявляем переменные

В Java нельзя просто так взять — и использовать переменную! Вас не поймут (компиляторы не поймут, ругаться будут, вот увидите!). Так что если уж вы решились что-то делать с переменной, об этом намерении нужно объявить во всеуслышание! Пишем программу, которая объявляет переменные типов int и String.