

JAVA Starter

Переменные и типы данных.



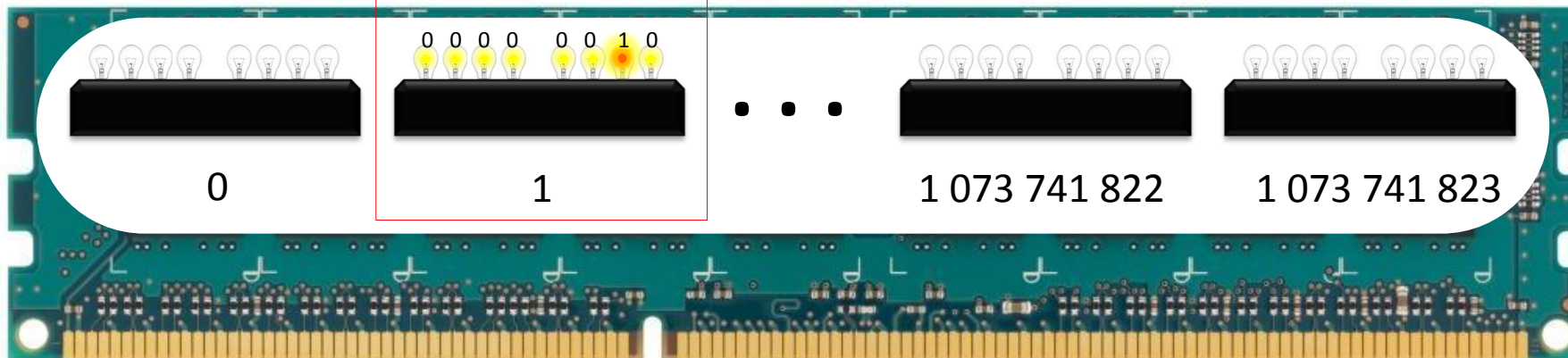
Переменные и типы данных.

Переменная

Variable

Переменная – это область памяти, которая хранит в себе некоторое значение, которое можно изменить.

```
byte a = 2; // 0000 0010 b
```

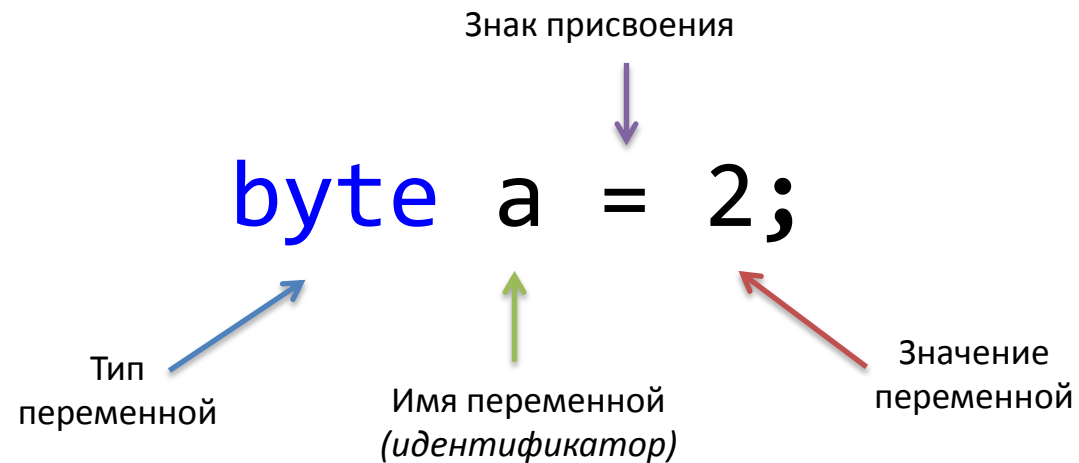


Переменная

Создание переменной

При создании переменной необходимо указать:

- Имя переменной (*идентификатор*)
- Тип переменной
- Начальное значение (*необязательно*)



Инициализация переменной – это первое присвоение ей значения.

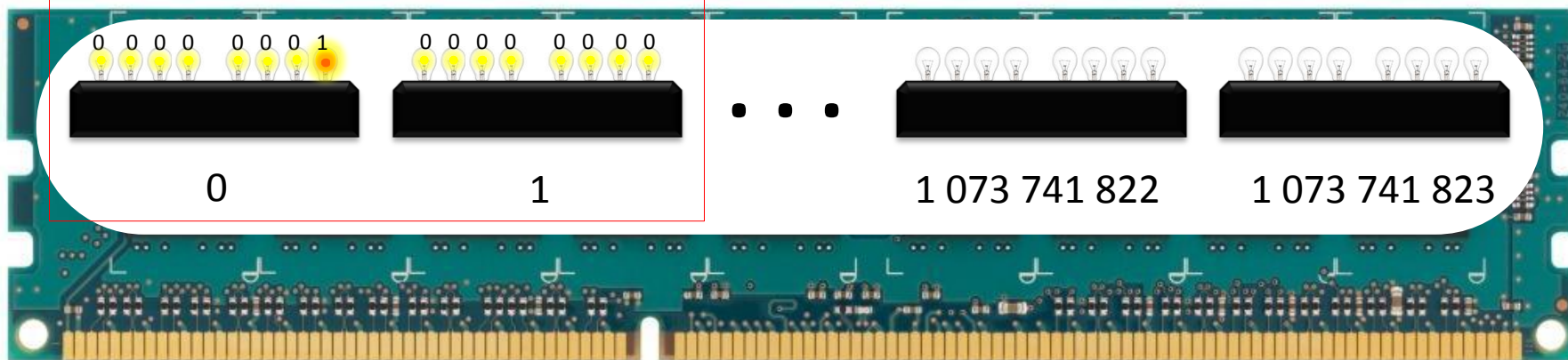
Переменная

Variable

Переменная – это область памяти, которая хранит в себе некоторое значение, которое можно изменить.

```
short a = 256;
```

```
// 0000 0001 0000 0000 b - 0x100
```



Типы данных ОЗУ

Варианты хранения информации в ОЗУ

1 байт = 8 бит `byte` `sbyte` `bool`



2 байта = 16 бит (Машинное слово) `short` `ushort` `char`



4 байта = 32 бита (Двойное машинное слово) `int` `uint` `float`

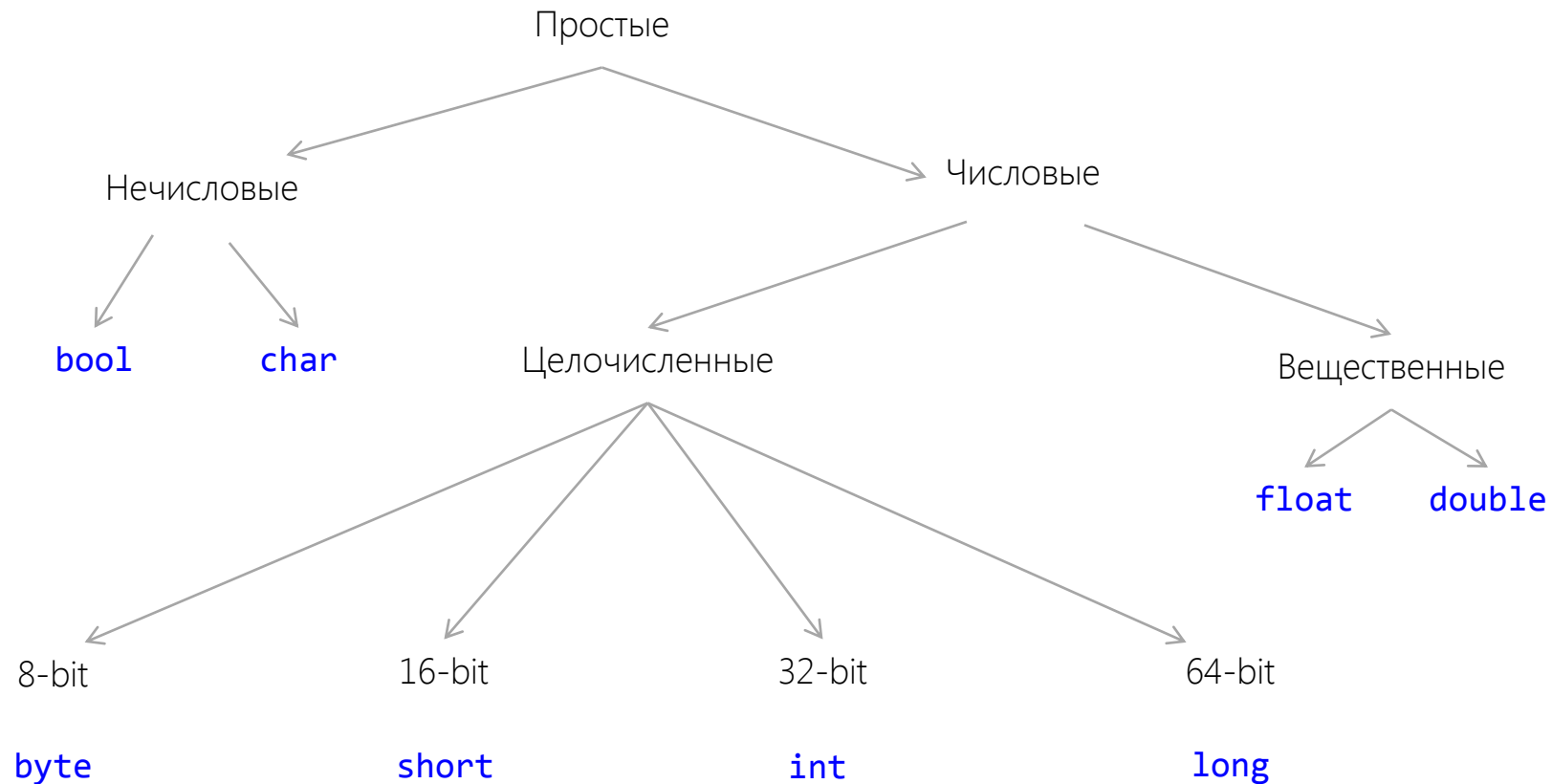


8 байт = 64 бита (Учетверённое машинное слово) `long` `ulong` `double`



Примитивные типы данных

Primitive Data Types



Переменные

Правила именования



Имена переменных должны быть понятны и передавать смысл хранимого значения.

Идентификаторы (имена переменных) могут записываться с помощью символов Unicode, указанных с использованием синтаксиса `\uXXXX`, где `XXXX` — четырехзначный шестнадцатеричный код Unicode символа. Например: `\u005fmyVar`, где `\u005f` - Unicode-код для знака подчеркивания. Такой подход не рекомендован.

`int myVariable = 2;`

Первый символ идентификатора

Допустимые символы	Недопустимые символы
A-Z, a-z, _	0-9, . , , , ? , : , * , > , < , = , & , # , ! , \$ и т.п.

Остальные символы идентификатора

Допустимые символы	Недопустимые символы
A-Z, a-z, _ , 0-9	@ , . , , , ? , : , * , > , < , = , & , # , ! , \$ и т.п.

Переменные

Правила именования

- В идентификаторах допустимо использовать символы алфавита и нижнего подчеркивания:

`myVariable, my_Variable, _MyVariable`

- Использование цифр недопустимо только на первой позиции:

`myVariable1, my1Variable, 1MyVariable`

- Нельзя использовать в качестве идентификаторов зарезервированные ключевые слова:

~~`extends, false, import`~~, `intMyVar`

- Язык JAVA чувствителен к регистру, поэтому если вы напишите их в разном регистре – это будут различные переменные:

`myVariable, MyVariable, myvariable`

Ключевые слова

Keywords

Ключевые слова - это предварительно определенные зарезервированные идентификаторы, имеющие специальные значения для компилятора.

<code>abstract</code>	<code>assert</code>	<code>boolean</code>	<code>break</code>	<code>byte</code>
<code>case</code>	<code>catch</code>	<code>char</code>	<code>class</code>	<code>const</code>
<code>continue</code>	<code>default</code>	<code>do</code>	<code>double</code>	<code>else</code>
<code>enum</code>	<code>extends</code>	<code>final</code>	<code>finally</code>	<code>float</code>
<code>for</code>	<code>goto</code>	<code>if</code>	<code>implements</code>	<code>import</code>
<code>instanceof</code>	<code>int</code>	<code>interface</code>	<code>long</code>	<code>native</code>
<code>new</code>	<code>package</code>	<code>private</code>	<code>protected</code>	<code>public</code>
<code>return</code>	<code>short</code>	<code>static</code>	<code>strictfp</code>	<code>super</code>
<code>switch</code>	<code>synchronized</code>	<code>this</code>	<code>throw</code>	<code>throws</code>
<code>transient</code>	<code>try</code>	<code>void</code>	<code>volatile</code>	<code>while</code>

Ключевые слова нельзя использовать в программе в качестве идентификаторов.



Ключевые слова не могут быть использованы в качестве идентификаторов.

Переменные

Соглашения по именованию

Спецификация языка JAVA рекомендует придерживаться определенных правил (casing conventions) при создании идентификаторов.

Стиль	Описание	Пример
Pascal casing	каждое слово в идентификаторе начинается с большой буквы	MyMethod, Remove
Camel casing	каждое слово, исключая первое, в идентификаторе начинается с большой буквы	myCount, totalDiscount
Uppercase	идентификатор состоит из букв написанных в верхнем регистре	IO, XML



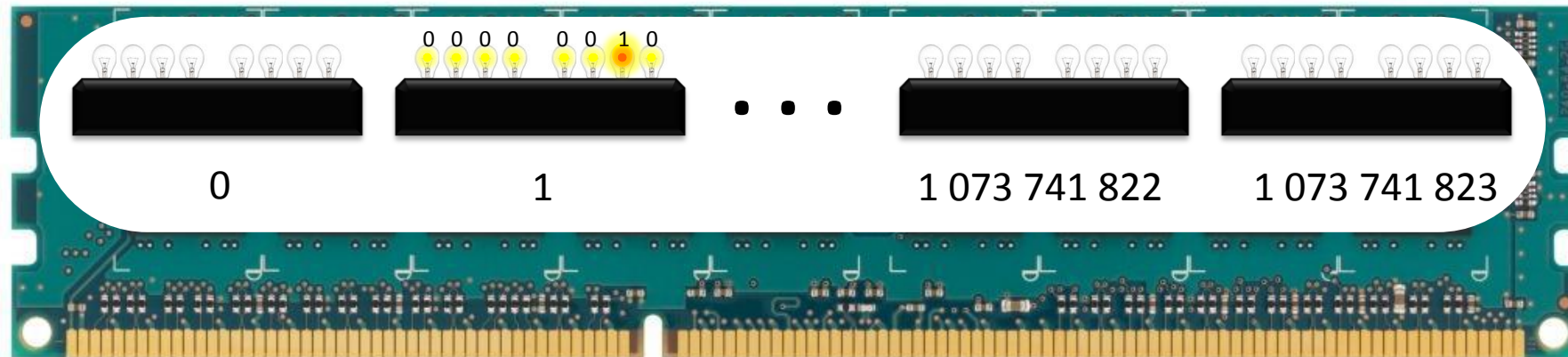
Не рекомендуется пользоваться венгерской нотацией и начинать идентификаторы с символа нижнего подчеркивания.

Константа

Constant

Константа – это область памяти, которая хранит в себе некоторое значение, которое нельзя изменить.

```
const byte a = 2; // 0000 0010 b
```



Попытка присвоить константе новое значение, приводит к ошибке уровня компиляции.

Преобразование значения типа

Casting или Type conversion

Кастинг - это преобразование значения переменной одного типа в значение другого типа.

Явный кастинг (explicit)

Преобразования выполняются только в случае явного указания (в круглых скобках) типа, в который необходимо преобразовать.

Неявный кастинг (implicit)

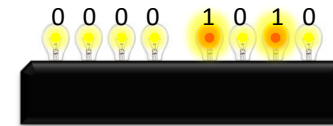
Преобразования выполняются автоматически без потери точности и урезания части исходного значения числа.

Неявный кастинг

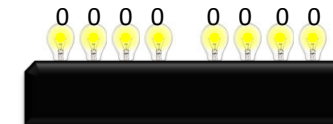
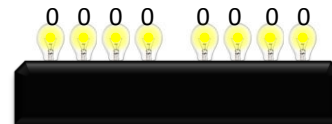
Безопасный

Неявное преобразование типа (безопасное) – преобразование значения меньшего типа в значение большего или целого в вещественное.

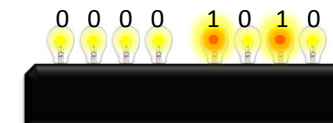
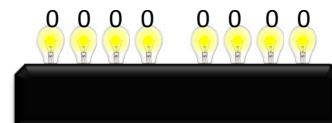
`byte a = 10;`



`short b = 0;`



`b = a;`



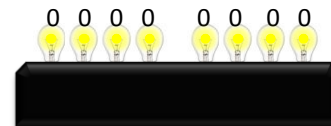
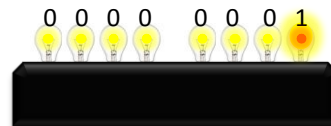
Ячейка заполняется нулевыми значениями или, если число было отрицательным – единицами.

Явный кастинг

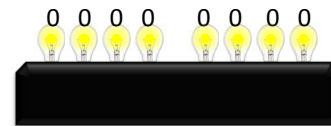
Опасный

Явное преобразование типа (опасное) – преобразование значения большего типа в значение меньшего или вещественного в целое.

```
short a = 256;
```

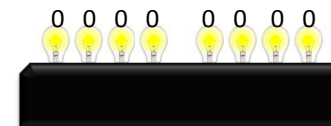


```
byte b = 0;
```



```
b = (byte)a;
```

Оператор явного
преобразования значения типа.



Явный кастинг считается опасным, так как может произойти потеря точности или урезание числа.

Строки

Форматированный вывод

Маркер подстановки
↓
`System.out.printf("Это число %1$s", 1);`
↑
Элемент подстановки

Позиция элемента подстановки, начинается с нуля (0,1,2,3,... и т.д.), указание большей позиции приведет к ошибке.

```
System.out.printf("Это числа по порядку %1$s %2$s ", 1, 2);
```

```
System.out.printf("Вывод наоборот %2$s %1$s ", 1, 2);
```


Проверка знаний

TestProvider.com

TestProvider

Мы помогаем людям оценить себя

Главная Услуги и цены Центр Тестирования Поддержка О нас

Регистрация Войти

Поиск сертификата

Мы в социальных сетях

Тестирование

Языки программирования и информационные технологии

Microsoft

C# ASP.NET MVC JavaScript Patterns OF Design SQL Architecture Guide WCF HTML&CSS XML SEO WPF HTML5&CSS3 JQuery XNA SharePoint GUI for Android Windows Azure Platform Microsoft Patterns&Practices TFS SCRUM ReSharper TDD WWF LINQ Entity Framework Windows Forms Refactoring Microsoft Expression Blend 4 Windows Phone 8 Windows 8 AppStore Visual Studio Tips&Tricks MSF MEF SilverLight AJAX MEF Service Oriented Architecture

Добро пожаловать на TestProvider.com!

Сайт перенесен на новую облачную платформу с использованием системы единой авторизации Single Sign On. Если вы хотите восстановить статистику по предыдущим экзаменам обратитесь в [службу поддержки](#). Для восстановления информации с предыдущей версии сайта, просба написать в службу поддержки Ваш старый и новый логины.

ITVDN PROMETRIC TEST CENTER CyberBionic Microsoft Partner Windows Azure Cloud Partner EBA

TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](https://testprovider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.



JAVA Starter

Q&A