

# Строка в Java. Приведение (casting) ТИПОВ ДАННЫХ





TEL-RAN  
by Starta Institute

1

# ПОВТОРЕНИЕ ИЗУЧЕННОГО

# Повторение

- Переменные в Java
- Типы переменных
- Локальные переменные
- Переменные экземпляра
- Статические переменные
- Сравнение переменных
- Примеры типов переменных
- Сканер



2

# ВОПРОСЫ ПО ПОВТОРЕНИЮ

# Введение

- Введение в ссылочные типы данных, Строка
- `String.length`, `String.charAt()`, `String.substring()`
- Символьные переменные
- Приведение (casting) переменных



3

# ОСНОВНОЙ БЛОК

# Ссылочные типы данных, строка

Как передаются переменные в java:

- Прimitives типы данных — это predefined типы данных Java
  - при сохранении примитивного типа данных значения будут присвоены стеку
  - когда переменная копируется, создается другая копия переменной
- Не примитивные или ссылочные типы данных
  - не примитивные типы создаются пользователями в Java
  - при сохранении переменная будет сохранена в стеке, а исходный объект будет сохранен в куче

# Строки

- String в Java — это объекты, которые внутренне поддерживаются массивом символов.
- Поскольку массивы неизменяемы (не могут расти), строки также неизменны.
- Когда вносятся изменения в строку, создается совершенно новая строка.
- Когда объект String создается как литерал, объект будет создан в пуле констант String. Это позволяет JVM оптимизировать инициализацию строкового литерала.
- С помощью оператора new. В случае динамического выделения строк им назначается новая ячейка памяти в куче. Эта строка не будет добавлена в пул констант String.





# String.length, String.charAt(), String.substring()

- Метод `length()` класса `Java String` находит длину строки.
- Метод `charAt(i)` класса `Java String` возвращает значение `char` с заданным номером индекса.

Номер индекса начинается с 0 и доходит до  $n-1$ , где  $n$  — длина строки.

- Метод `substring()` класса `Java String` возвращает часть строки.

Мы передаем числовые позиции `beginIndex` и `endIndex` в методе подстроки `Java`, где `beginIndex` является включающим, а `endIndex` — исключаящим.

`beginIndex` начинается с 0, `endIndex` начинается с 1.



# Преобразование Char в Int

Символы описываются в языке Java char типом.

Символы преобразуются по таблице кодировки UTF-16.

Это все буквы, числа и специальные символы существующие на нашей планете.

Тип char является псевдо-целочисленным типом, поэтому значения этого типа можно задавать в виде числа.

Char to Int

Способ 1: использование значений ASCII

Способ 2: использование метода `valueOf()` класса `String`

Способ 3: использование метода `getNumericValue()` класса `Character`





TEL-RAN  
by Starta Institute

# 4

## ЗАДАНИЕ ДЛЯ ЗАКРЕПЛЕНИЯ

Сделайте преобразование CHAR → INT

1. Using ASCII value
2. Using `valueOf()`
3. Using `getNumericValue()`

Проанализируйте вывод

# Приведение переменных

В Java числовые типы данных совместимы друг с другом.

Но автоматическое преобразование из числового типа в символьный или логический не поддерживается.

**char и boolean несовместимы друг с другом**

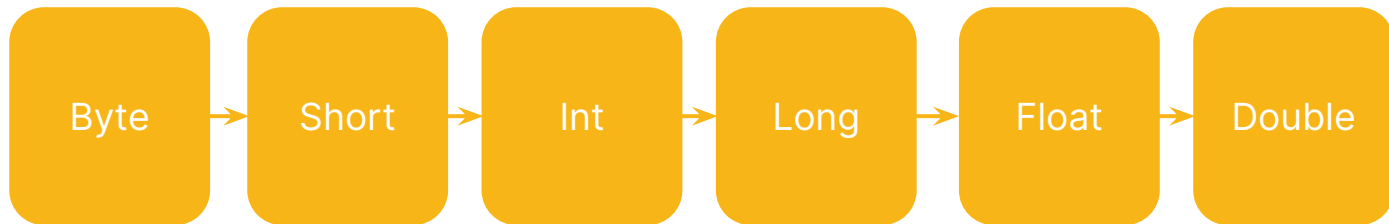
Data type	Bits In Memory
boolean	1
byte	8 (1 byte)
char	16 (2 bytes)
short	16(2 bytes)
int	32 (4 bytes)
long	64 (8 bytes)
float	32 (4 bytes)
double	64 (8 bytes)

# Преобразование типов

Расширяющее преобразование имеет место, когда два типа данных преобразуются автоматически.

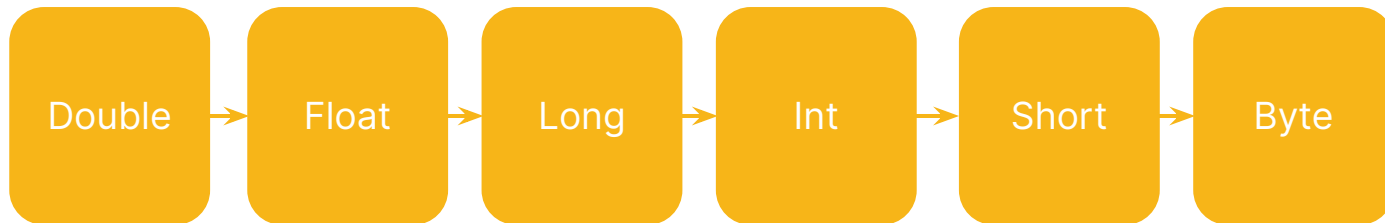
Это происходит, когда:

- два типа данных совместимы
- мы присваиваем значение меньшего типа данных большему типу данных



# Явное преобразование

- Если мы хотим присвоить значение большего типа данных меньшему типу данных, то выполняем явное приведение типов или сужение.
- Это полезно для несовместимых типов данных, где нельзя выполнить автоматическое преобразование.
- Здесь целевой тип указывает желаемый тип для преобразования указанного значения.
- При применении явных преобразований мы можем столкнуться с потерей данных.



# Преобразования при операциях

- если один из операндов операции относится к типу `double`, то и второй операнд преобразуется к типу **`double`**
- если предыдущее условие не соблюдено, а один из операндов операции относится к типу **`float`**, то и второй операнд преобразуется к типу **`float`**
- если предыдущие условия не соблюдены, один из операндов операции относится к типу **`long`**, то и второй операнд преобразуется к типу **`long`**
- иначе все операнды операции преобразуются к типу **`int`**





# Экспресс-опрос

- **Вопрос 1.**

Из чего состоит строка в Java?

- **Вопрос 2.**

Что может произойти при применении явного преобразования числа?



5

# ВОПРОСЫ ПО ОСНОВНОМУ БЛОКУ

6

# ЗАДАНИЕ ДЛЯ ЗАКРЕПЛЕНИЯ

## ЗАДАНИЕ

1. Произвести `int` → `long` путем “расширения”
2. Произвести `int` → `byte` путем “сужения”
3. `double` → `int`
4. `int` → `double`
5. Распечатайте результат в консоль

Для теста используйте число `int` = 127, а затем с числом 128  
Проанализируйте вывод



TEL-RAN  
by Starta Institute

7

# ПРАКТИЧЕСКАЯ РАБОТА

# Практическое задание 1

1. Создать новую строку: "I study Java"
2. Вывести в консоль длину строки
3. Вывести в консоль часть строки "Java" используя substring



# Реализация задания 1

```
public static void main(String[] args) {  
    String str="I study Java";  
    System.out.println(str.length());  
    String substr = str.substring( beginIndex: 8); // Starts with 0 and goes to end  
    System.out.println(substr);  
}
```



TEL-RAN  
by Starta Institute

8

# ОСТАВШИЕСЯ ВОПРОСЫ



# Домашнее задание

№ 1

1. Создайте строку через new - I study Basic Java!
2. Напишите метод, который принимает в качестве параметра строку, передайте в этот метод строку, которую создали в п.1
3. Распечатать пред-последний символ строки. Используем метод `String.charAt()`.
4. Проверить, содержит ли ваша строка подстроку "Java". Используем метод `String.contains()`.
5. Вырезать строку Java с помощью метода `String.substring()`.
6. Заменить все символы "а" на "о".
7. Преобразуйте строку к верхнему регистру.
8. Преобразуйте строку к нижнему регистру.

# Домашнее задание

№ 2

1. Создайте методы с математическими операциями  $+$ ,  $-$ ,  $*$ ,  $/$ 
  - а. Каждый метод принимает два числа в параметрах и возвращает результат
2. Вызовите все методы в `main`
3. Результат распечатайте в консоль



# Полезные ссылки

- [String \(computer science\) - Wikipedia](#)
- [String \(Java Platform SE 7 \) \(oracle.com\)](#)

# Дополнительная практика

Напишите программу, которая запрашивает имя пользователя и выводит на экран «Добрый день <..... !>».

Улучшите программу, которая запрашивает имя пользователя и выводит на экран «Добрый день <..... !>».

«Ваше имя начинается с символа <.> и заканчивается на символ <.>».