

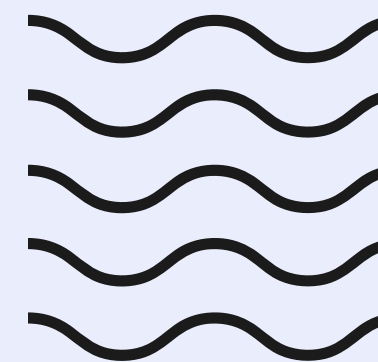
char vs String



ПОДГОТОВИЛ: АЛИШЕР ХАМИДОВ

План лекции

Основные моменты:



1. Примитивные и ссылочные типы
2. String vs char
3. Создание переменных
4. Преобразование типов
5. Конкатенация vs Сложение
6. equals()



Какие вы знаете примитивные типы?

Ваши ассоциации



String vs char

"A" - строка из
одного символа

'1' - СИМВОЛ В
одинарных
кавычках

String vs char

"1"

"A"

"Hello"




"john67@gmail.com"

'1'

'A'

'@'

'p'



```
char[] symbols = { 'H', 'e', 'l', 'l', 'o' };  
String word1 = String.valueOf(symbols);  
System.out.println(word1); // Hello
```

```
char x = 'a';  
int y = x;  
System.out.println(x); // a  
System.out.println(y); // 97
```



char x = 'a';

Character y = 'z'; // если хотим использовать
методы





American standard code for information
interchange (**ASCII**)



<https://theasciicode.com.ar/>

Unicode



https://en.wikipedia.org/wiki/List_of_Unicode_characters



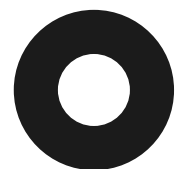
‘A’ – символный литерал, заданный напрямую любым отображаемым символом Unicode

‘\uxxxx’ – символ Unicode, где xxxx цифровой код символа Unicode✕ в шестнадцатеричной форме

‘\xxx’ – символ кодовой таблицы Latin-1, где xxx восьмеричный код символа Latin-1

1046 – код символа Unicode в десятичном исчислении

0x0950 – код символа Unicode в шестнадцатеричном формате





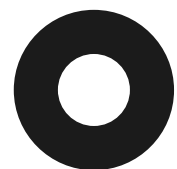
‘A’ – символный литерал, заданный напрямую любым отображаемым символом Unicode

‘\uxxxx’ – символ Unicode, где xxxx цифровой код символа Unicode✖ в шестнадцатеричной форме

‘\xxx’ – символ кодовой таблицы Latin-1, где xxx восьмеричный код символа Latin-1

1046 – код символа Unicode в десятичном исчислении

0x0950 – код символа Unicode в шестнадцатеричном формате



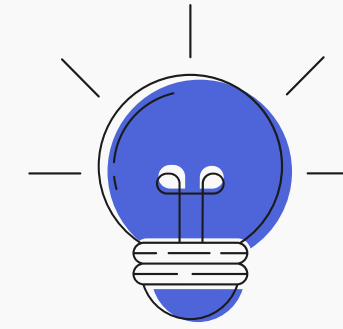


Управляющие символы (также должны быть заключены в одинарные кавычки):

- `\b` – backspace BS – забой (`\u0008` в кодировке Unicode и 8 в десятичной)
- `\t` – horizontal tab HT – табуляция (`\u0009` в кодировке Unicode и 9 в десятичной)
- `\n` – line feed LF – конец строки (`\u000a` в кодировке Unicode и 10 в десятичной)
- `\f` – form feed FF – конец страницы (`\u000c` в кодировке Unicode и 12 в десятичной)
- `\r` – carriage return CR – возврат каретки (`\u000d` в кодировке Unicode и 13 в десятичной)
- `\"` – двойная кавычка (`\u0022` в кодировке Unicode и 34 в десятичной)
- `'` – одинарная кавычка (`\u0027` в кодировке Unicode и 39 в десятичной)
- `\\` – backslash \ – обратная косая черта (`\u005c` в кодировке Unicode и 92 в десятичной)



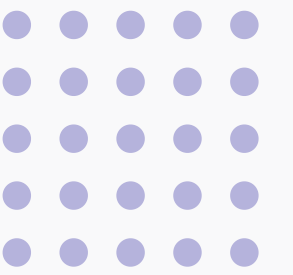
Как отличить?



Если слева или справа строка, то это конкатенация.

Только если оба операнда являются числами – это сложение.

Важна последовательность выполнения операций,
а значит, нужно обращать внимание на скобки.
Если их нет, то читаем слева направо.



`==` *vs* `equals()`

Для сравнения чисел используем `==`

Для сравнения строк используем `equals()`

`str1.equals(str2)`

`==` **vs** `equals()`

```
String str1 = "Bob";
```

```
String str2 = "Blob";
```

```
boolean result = str1.equals(str2);
```

```
System.out.println(result) // false
```