**Содержание урока**

1. [Про выделение и освобождение памяти](https://stepik.org/lesson/%D0%9C%D0%B0%D1%81%D1%81%D0%B8%D0%B2%D1%8B-%D0%B8-%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8-12761/step/2?course=Java-%D0%91%D0%B0%D0%B7%D0%BE%D0%B2%D1%8B%D0%B9-%D0%BA%D1%83%D1%80%D1%81&unit=3109)
2. [Массивы](https://stepik.org/lesson/%D0%9C%D0%B0%D1%81%D1%81%D0%B8%D0%B2%D1%8B-%D0%B8-%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8-12761/step/3?course=Java-%D0%91%D0%B0%D0%B7%D0%BE%D0%B2%D1%8B%D0%B9-%D0%BA%D1%83%D1%80%D1%81&unit=3109)
3. [Varargs](https://stepik.org/lesson/%D0%9C%D0%B0%D1%81%D1%81%D0%B8%D0%B2%D1%8B-%D0%B8-%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8-12761/step/4?course=Java-%D0%91%D0%B0%D0%B7%D0%BE%D0%B2%D1%8B%D0%B9-%D0%BA%D1%83%D1%80%D1%81&unit=3109)
4. [Сравнение массивов](https://stepik.org/lesson/%D0%9C%D0%B0%D1%81%D1%81%D0%B8%D0%B2%D1%8B-%D0%B8-%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8-12761/step/5?course=Java-%D0%91%D0%B0%D0%B7%D0%BE%D0%B2%D1%8B%D0%B9-%D0%BA%D1%83%D1%80%D1%81&unit=3109)
5. [Печать массивов](https://stepik.org/lesson/%D0%9C%D0%B0%D1%81%D1%81%D0%B8%D0%B2%D1%8B-%D0%B8-%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8-12761/step/6?course=Java-%D0%91%D0%B0%D0%B7%D0%BE%D0%B2%D1%8B%D0%B9-%D0%BA%D1%83%D1%80%D1%81&unit=3109)
6. [Строки](https://stepik.org/lesson/%D0%9C%D0%B0%D1%81%D1%81%D0%B8%D0%B2%D1%8B-%D0%B8-%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8-12761/step/7?course=Java-%D0%91%D0%B0%D0%B7%D0%BE%D0%B2%D1%8B%D0%B9-%D0%BA%D1%83%D1%80%D1%81&unit=3109)
7. [Вопрос](https://stepik.org/lesson/%D0%9C%D0%B0%D1%81%D1%81%D0%B8%D0%B2%D1%8B-%D0%B8-%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8-12761/step/8?course=Java-%D0%91%D0%B0%D0%B7%D0%BE%D0%B2%D1%8B%D0%B9-%D0%BA%D1%83%D1%80%D1%81&unit=3109)
8. [Вопрос](https://stepik.org/lesson/%D0%9C%D0%B0%D1%81%D1%81%D0%B8%D0%B2%D1%8B-%D0%B8-%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8-12761/step/9?course=Java-%D0%91%D0%B0%D0%B7%D0%BE%D0%B2%D1%8B%D0%B9-%D0%BA%D1%83%D1%80%D1%81&unit=3109)
9. [Задача](https://stepik.org/lesson/%D0%9C%D0%B0%D1%81%D1%81%D0%B8%D0%B2%D1%8B-%D0%B8-%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8-12761/step/10?course=Java-%D0%91%D0%B0%D0%B7%D0%BE%D0%B2%D1%8B%D0%B9-%D0%BA%D1%83%D1%80%D1%81&unit=3109)

Реализуйте метод, проверяющий, является ли заданная строка палиндромом. Палиндромом называется строка, которая читается одинаково слева направо и справа налево (в том числе пустая). При определении "палиндромности" строки должны учитываться только буквы и цифры. А пробелы, знаки препинания, а также регистр символов должны игнорироваться. Гарантируется, что в метод попадают только строки, состоящие из символов ASCII (цифры, латинские буквы, знаки препинания). Т.е. русских, китайских и прочих экзотических символов в строке не будет.

Воспользуйтесь предоставленным шаблоном. Декларацию класса, метод main и обработку ввода-вывода добавит проверяющая система.

Подсказки (не читайте, если хотите решить сами):

* для удаления из строки всех символов, не являющихся буквами и цифрами, можно воспользоваться регулярным выражением "[^a-zA-Z0-9]"; найдите в классе String метод, выполняющий замену по регулярному выражению;
* для перестановки символов строки в обратном порядке можно воспользоваться методом reverse(), который находится в классе StringBuilder;
* в классе String есть методы для преобразования всей строки в верхний и нижний регистр.

**Sample Input:**

Madam, I'm Adam!

**Sample Output:**

True

===========================

public class Palindrome {

boolean isPalindrome(String text){

String txt1 = text.replaceAll("[^a-zA-Z0-9]", "");

String txt2 = new StringBuilder(txt1).reverse().toString();

return txt1.equalsIgnoreCase(txt2);

/\* return text.replaceAll("[^a-zA-Z0-9]", "").equalsIgnoreCase

(String.valueOf(new StringBuilder(text.replaceAll("[^a-zA-Z0-9]", ""))

.reverse()));\*/

}

}

class PalindromeDemo {

public static void main(String[] args) {

String txt = "Madam, I'm Adam!";

Palindrome palindrome = new Palindrome();

System.out.println(palindrome.isPalindrome(txt));

}

}