

Внимание! Задачи отсортированы по порядку тем и по возрастанию уровня сложности. Каждый может определить для себя необходимый уровень. Если задача кажется вам слишком простой – смело переходите к следующей.

Естественно, к ближайшему занятию я не жду от вас решения всех задач, но надеюсь, что каждый к ближайшему занятию решит хотя бы 3-5 задач. Успехов!

P.S. Да, если будет необходимость, список дополним!

## УСЛОВИЯ

1. Реализовать метод **boolean isEven(int number)** который возвращает true если переданное число четное иначе возвращает false. Реализовать метод **void printCheckResultMessage(int number, boolean result)** который выводит на экран строку вида «10 это четное число» или «15 это не четное число»  
Подсказка: Используем конструкцию if и оператор % modulo
2. Реализовать метод **boolean isDivisible(int number1, int number2)** который возвращает true если **number1** делится на **number2** без остатка. Аналогично первой задаче реализовать метод печати результата.
3. Реализовать метод, который в качестве параметров получает 3 числа типа int. Метод должен возвращать true если и первое и второе число делятся без остатка на третье.  
Подсказка: Используйте метод, написанный в пункте 2
4. Реализовать метод **int max3(int num1, int num2, int num3)** который возвращает число, наибольшее из трех переданных чисел:  
Например: **max3(10,19,0) -> 19**
5. Реализовать метод **String longestString (String str1, String str2, String str2)** который возвращает самую длинную строку из трех заданных строк:  
Подсказка: Используйте метод, написанный в пункте 4  
Например: **longestString ("java","welcome","hello") -> "welcome"**

## ЦЫКЛЫ

6. Реализовать метод, который печатает все числа в диапазоне от 0 до 100, которые не делятся на 4
7. Реализовать метод, который принимает 2 параметра *int start* и *int finish* и возвращает сумму всех чисел от start до finish включительно.  
Например: **sum(10,15) -> 75**
8. \*Реализовать метод, который принимает 3 параметра *start*, *finish*, *digit* типа int. *digit* находится в диапазоне ( $0 \leq \text{digit} < 9$ ), если нет, то считаем *digit*=0. Метод должен вернуть сумму всех чисел от start до finish исключая те числа, которые заканчиваются на *digit*.  
Например: **sum(10,15,3) -> 62** (в данном случае, в сумму входят числа 10,11,12,14 и 15. Число 13 не входит.)

## ПЕРЕБОР СТРОК ПО СИМВОЛАМ

Для решения задач данного раздела, не используются ни какие стандартные методы класса String кроме **length()** и **charAt()**

9. Реализовать метод, который **int findFirstCharIndex(String str, char ch)** который возвращает позицию первого вхождения (индекс) символа ch в строке str. Если символа в строке нет, возвращаем -1.  
Например: find("let's talk about java",'t') -> 2
10. Реализовать метод, который **int findLastCharIndex(String str, char ch)** который возвращает позицию последнего вхождения (индекс) символа ch в строке str. Если символа в строке нет, возвращаем -1.  
Например: find("let's talk about java",'t') -> 15
11. Реализовать метод, который возвращает длину общего префикса двух строк, т.е. сколько символов начиная с начала одинаковые у обеих строк.
12. Реализовать метод, который возвращает длину общего постфикса двух строк, т.е. сколько символов начиная с конца одинаковые у обеих строк.
13. Реализовать метод, который печатает заданную строку, при этом каждый символ печатается в [] Например: "Hello" -> [H][e][l][l][o]
14. Реализовать метод, который возвращает часть заданной строки, начиная с позиции start, заканчивая позицией finish. (разбирали в классе, но можно реализовать самостоятельно).  
Например: substring("let's talk about java", 7,20) -> "talk about ja"
15. Представьте, что вы пишете банковскую программу. Вам нужно реализовать метод, который вместо заданной строки с номером счета (например "DE5128279087265") возвращает строку вида "DE51\*\*\*\*\*65" (количество звездочек соответствует количеству засекреченных цифр)