

[Back to Dashboard](#)[Status](#)[Changes](#)**Build History** [\(trend\)](#)[RSS for all](#) [RSS for failures](#)

## Project task-jtr-practice1

Практическое задание №1

### Задание 1

-----  
Название класса: ua.nure.your\_last\_name.Practice1.Part1  
-----

Написать класс, который реализует функциональность вывода в консоль строки "hello, world".  
Написать командный файл (bat файл windows или shell скрипт Linux), который компилирует и запускает данное приложение из консоли.  
Командный файл сохранить в каталоге src проекта.

### Задание 2

-----  
Название класса: ua.nure.your\_last\_name.Practice1.Part2  
-----

Написать класс, который реализует функциональность сложения двух чисел, переданных в приложение как параметры командной строки.

### Задание 3

-----  
Название класса: ua.nure.your\_last\_name.Practice1.Part3  
-----

Написать класс, который реализует функциональность вывода параметров командной строки в консоль (выводить через пробел).

### Задание 4

-----  
Название класса: ua.nure.your\_last\_name.Practice1.Part4  
-----

Написать класс, который реализует функциональность определения наибольшего общего делителя двух целых положительных чисел, переданных в приложение как параметры командной строки.

### Задание 5

-----  
Название класса: ua.nure.your\_last\_name.Practice1.Part5  
-----

Написать класс, который реализует функциональность определения суммы цифр целого положительного числа, переданного в приложение как параметр командной строки.

### Задание 6

-----  
Название класса: ua.nure.your\_last\_name.Practice1.Part6  
-----

Написать класс, который создает массив из n элементов и заполняет его восходящим рядом простых чисел (2, 3, 5, 7, ...). Число n передавать как параметр командной строки

### Задание 7

-----  
Название класса: ua.nure.your\_last\_name.Practice1.Part7  
-----

Колонки электронных таблиц (например Excel) имеют буквенную нумерацию в виде больших букв латинского алфавита (последовательно, слева направо):

A, B, ..., Y, Z, AA, AB, ..., AY, AZ, BA, BB, ... и т.д.

В то же время каждая колонка имеет порядковый номер: A - 1; B - 2; ... ; Y - 25; Z - 26; AA - 27; AB - 28; ... и т.д.

Написать класс, который содержит следующие три метода:

1) метод определения порядкового номера колонки по ее буквенному номеру  
[ public static int chars2digits(String number): A => 1; B => 2; ...; Z => 26; AA => 27; AB => 28; ... ];

2) метод определения буквенного номера колонки по ее порядковому номеру  
[ public static String digits2chars(int number): 1 => A; 2 => B; ...; 26 => Z; 27 ==> AA; 28 ==> AB; ... ]

```
};
```

```
3) метод overriding по буквенному начеру, корень, начер, корень, который начерится, сводя от лачный
```

Jenkins

All

task-jtr-practice1

[ENABLE AUTO REFRESH](#)

Работу методов проверить на следующих данных

```
-----  
A ==> 1 ==> A  
B ==> 2 ==> B  
Z ==> 26 ==> Z  
AA ==> 27 ==> AA  
AZ ==> 52 ==> AZ  
BA ==> 53 ==> BA  
ZZ ==> 702 ==> ZZ  
AAA ==> 703 ==> AAA  
-----
```

Замечания.

1. Результат должен быть представлен в виде проекта с именем Practice1.
2. Корневой пакет для всех классов и прочих пакетов (если они потребуются):  
ua.nure.your\_last\_name.Practice1, где your\_last\_name - ваш логин без кода проекта.  
Если логин=ivanovjtr, то your\_last\_name=ivanov
3. Дополнительно в корневой пакет положить класс Demo, который демонстрирует работу всех 5 подзадач.
4. Проект загрузить в репозиторий, проверить, успешность сборки в Jenkins, оптимизировать метрики в Sonar (после того, как будет дан доступ к этим системам).

Вопросы.

1. Какие категории типов данных существуют в Java?
2. Перечислите примитивные типы данных.
3. Тип данных char, что хранит, область определения.
4. Напишите метод main (два варианта).
5. Укажите автоматические преобразования между примитивными типами
6. При каких преобразованиях между примитивными типами возможна утрата информации?
7. Какие преобразования между типами вы знаете.
8. Что такое wrappers, autoboxing, autounboxing.
9. Напишите анонимный массив, массив константу, в чем отличие.
10. Напишите пример двумерного массива.

[Recent Changes](#)

Project disk usage information + trend graph



**Disk Usage:** Workspace 0, Builds {all=0, locked=0}, Job directory 16146

## Permalinks

 [Help us localize this page](#)

Page generated: 21 жовт 2018 15:27:07

[REST API](#)[Jenkins ver. 1.540](#)