****Закрепите полученные знания, выполнив практическую работу. Все три задания обязательны для выполнения и сдачи на проверку.****

Выполнить работу будет гораздо проще после самостоятельной практики. Задания для неё мы разместили под некоторыми видео этого модуля.

#### Задание 1

****Цель****

* Научиться пользоваться методами работы с подстроками.

****Что нужно сделать****

Выполните задание в отдельном репозитории “java\_Strings” в классе main/java/practice/strings/SequentialWordsNumbers.java.

Нужно написать метод sequentialWordsNumbers, который в качестве аргумента принимает текст и перед каждым словом пишет его порядковый номер в круглых скобках и возвращает строку. Метод необходимо написать ****без использования регулярных выражений****. Пример того, как оно должно работать:

Введите текст:  
Это просто текст, для примера работы программы  
Результат:  
(1) Это (2) просто (3) текст, (4) для (5) примера (6) работы (7) программы

Запустите в проекте тесты и убедитесь в том, что они выполняются успешно. Если тесты не выполняются и выдают информацию об ошибках, постарайтесь самостоятельно исправить эти ошибки и добиться успешного выполнения тестов перед отправкой результатов преподавателю.

****Советы и рекомендации****

* Используйте поиск пробелов в тексте методом indexOf в цикле.
* Сохраняйте новую строку в объект класса StringBuilder, а затем верните объект класса String

****Критерии оценки работы****

Принято:

* Программа работает так, как показано на примере.
* В программе не используются регулярные выражения.
* Автотесты выполняются без ошибок.

На доработку: задание не выполнено, выполнено неточно либо частично.

****Как отправить работу на проверку****

Выполните также второе и третье задания (смотрите ниже) и отправьте куратору результаты сразу по всем заданиям.

#### Задание 2

****Цель****

* Научиться писать и использовать несложные регулярные выражения.

****Что нужно сделать****

Выполните задание в отдельном репозитории “java\_Strings” в классе main/java/practice/regex/SplitText.java.

Реализуйте метод, который принимает в качестве аргумента строку, содержащую английский текст (не менее 100 слов). Метод должен разделить текст на слова и вернуть строку, состоящую из полученных слов, где каждое слово будет на отдельной строке. Знаки препинания и числа не являются словами.

Запустите в проекте тесты и убедитесь в том, что они выполняются успешно. Если тесты не выполняются и выдают информацию об ошибках, постарайтесь самостоятельно исправить эти ошибки и добиться успешного выполнения тестов перед отправкой результатов преподавателю.

****Критерии оценки работы****

Принято:

* Корректность работы программы независимо от передаваемого ей на вход текста.
* Автотесты выполняются без ошибок.

На доработку: текст разделяется на слова неверно.

****Как отправить работу на проверку****

Выполните также третье задание (смотрите ниже) и отправьте куратору результаты сразу по всем заданиям.

#### Задание 3

****Цель****

* Научиться писать и использовать несложные регулярные выражения.

****Что нужно сделать****

* Выполните задание в отдельном репозитории “java\_Strings” в классе main/java/practice/regex/PhoneCleanerRegex.java.
* Реализуйте удаление лишних символов при вводе номера телефона в консоли и проверку соответствия номера формату мобильных номеров в России. Если введённую строку нельзя привести к формату мобильного номера, выводите сообщение о неверном вводе. Телефон может быть введён не только в формате 79091234567, но и с лишними символами.

Примеры вводимых значений и результаты вывода программы:

|  |  |  |
| --- | --- | --- |
| Вводимое значение | Вывод программы | Примечание |
| +7 909 123-45-67 | 79091234567 | В номере 11 цифр, код страны 7, номер верный |
| +7 (909) 1234567 | 79091234567 | В номере 11 цифр, код страны 7, номер верный |
| 8-905-1234567 | 79051234567 | В номере 11 цифр, первая цифра 8 (заменяем на код страны 7) — номер верный |
| 9-453-1234567 | Неверный формат номера | В номере 11 цифр, первый символ после очистки 9, это не 7 и не 8 — формат номера неверный |
| 8-905-123 | Неверный формат номера | В номере 7 цифр — номер неверный |
| 905-1234567 | 79051234567 | В номере 10 цифр. Добавляем в начале код страны 7 |
| 8-905-12345672342 | Неверный формат номера | В номере более 11 цифр — номер неверный |

* Запустите в проекте тесты и убедитесь в том, что они выполняются успешно. Если тесты не выполняются и выдают информацию об ошибках, постарайтесь самостоятельно исправить эти ошибки и добиться успешного выполнения тестов перед отправкой результатов преподавателю.

****Критерии оценки работы****

Принято:

* Корректность работы программы независимо от передаваемого ей на вход текста.
* Автотесты выполняются без ошибок.

На доработку:

* Верные номера не распознаются как верные и не конвертируются в единый формат.
* Неверные номера распознаются как верные и конвертируются в числовой формат вместо отображения ошибки