Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и кибербезопасности

Высшая школа «Компьютерных технологий и информационных систем»

**ОТЧЕТ по лабораторной работе №5**

по дисциплине «Системный подход в разработке программного обеспечения»

**Выполнил:**

студент группы 5130902/20201 \_\_\_\_\_\_\_\_\_\_\_\_\_ А. И. Сафонов

подпись

**Проверил:**

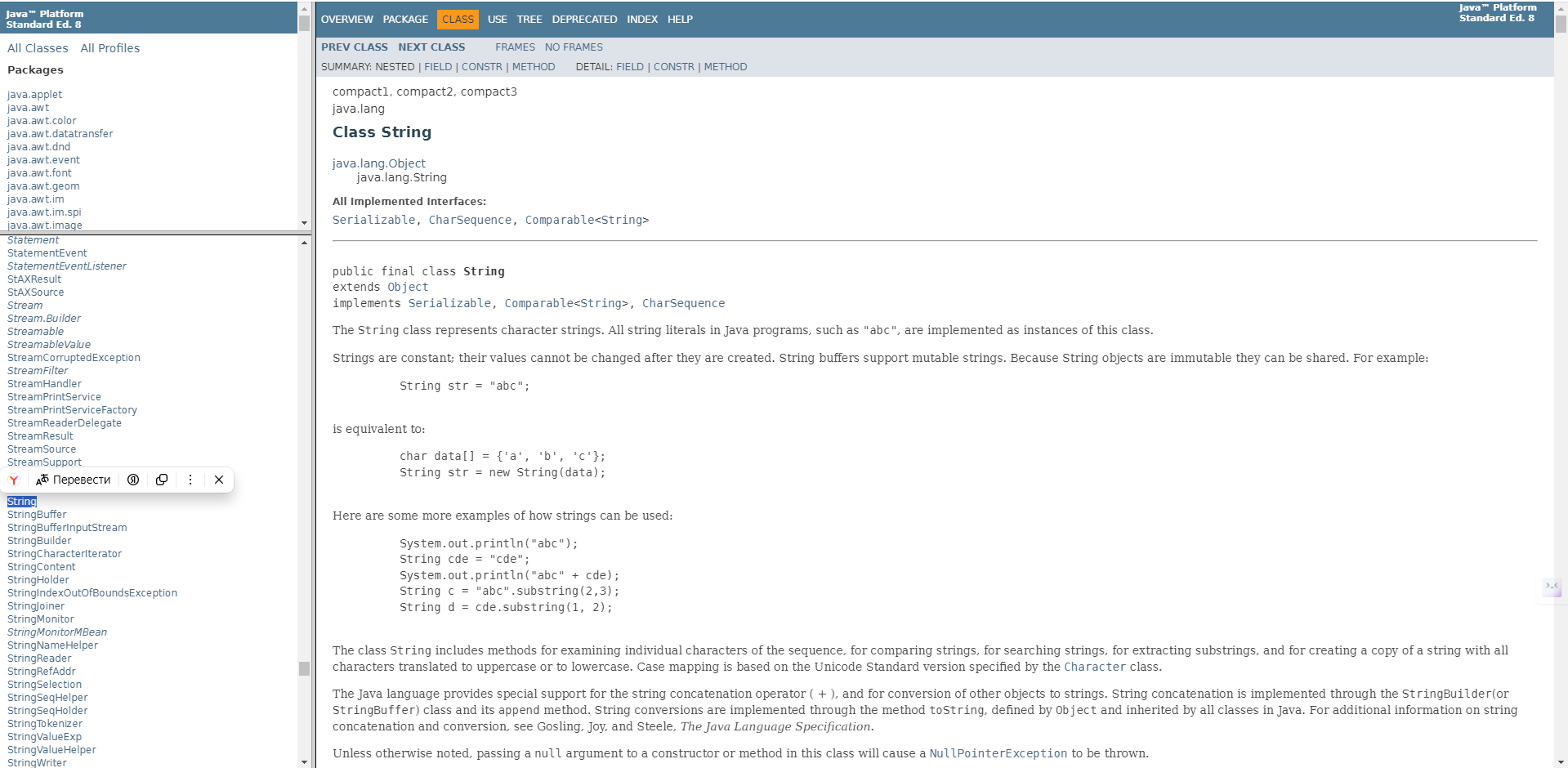
Кандитад тех. Наук., доцент \_\_\_\_\_\_\_\_\_\_\_\_ С. А. Нестеров

подпись

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_ 2024г.

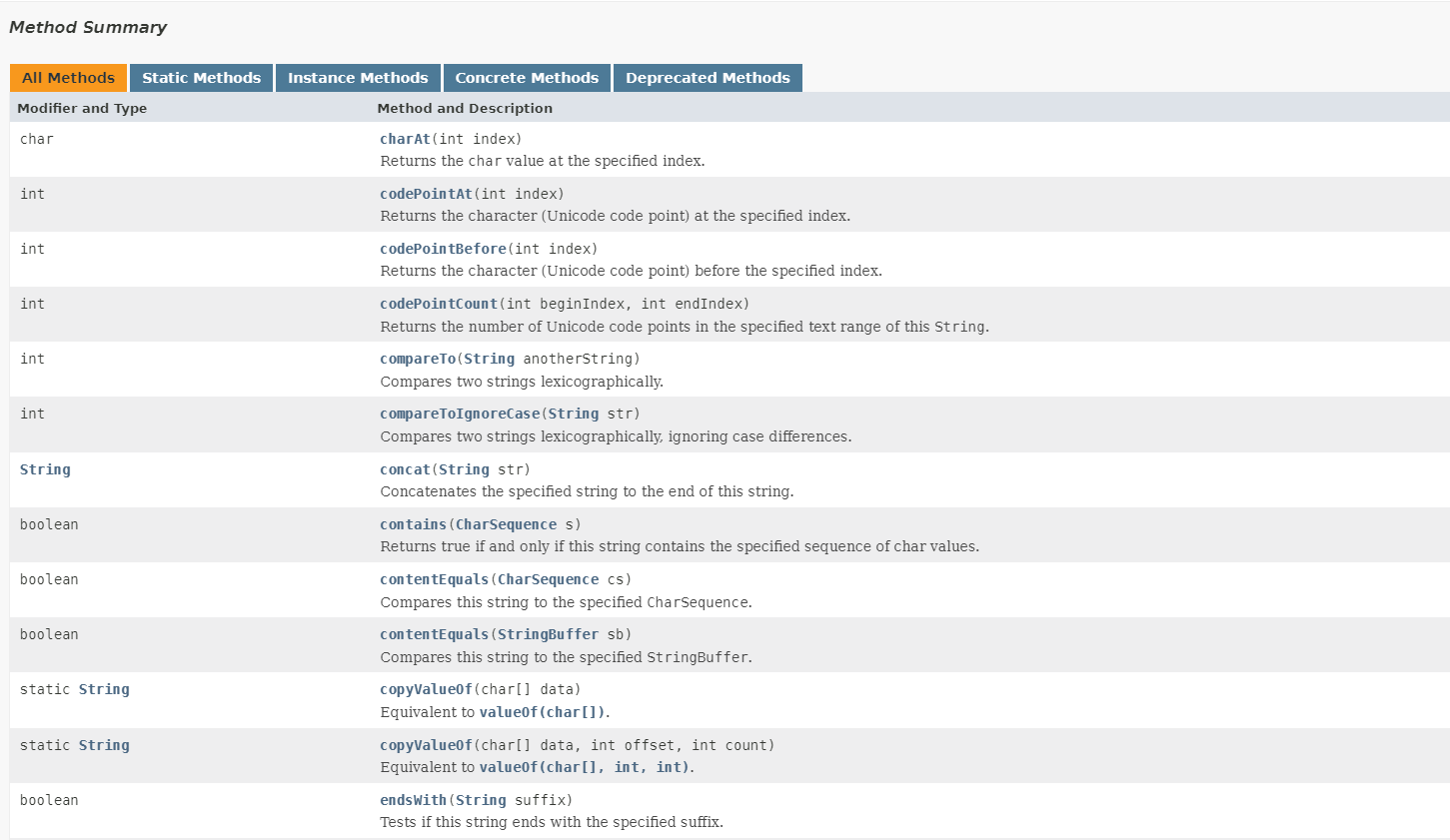
Санкт-Петербург, 2024

1. Задания по уроку 4.3.
2. Работа с документацией класса String.

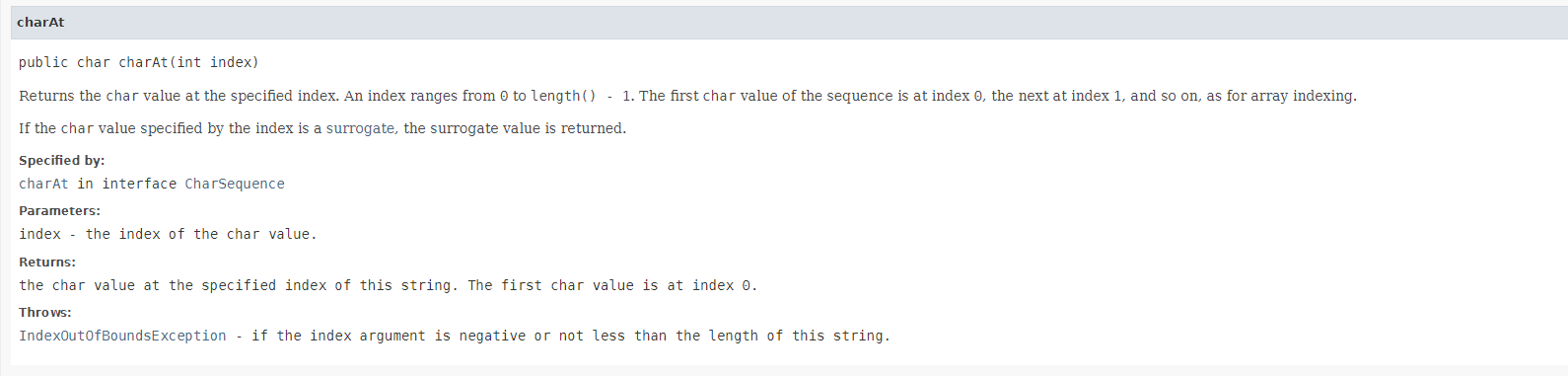


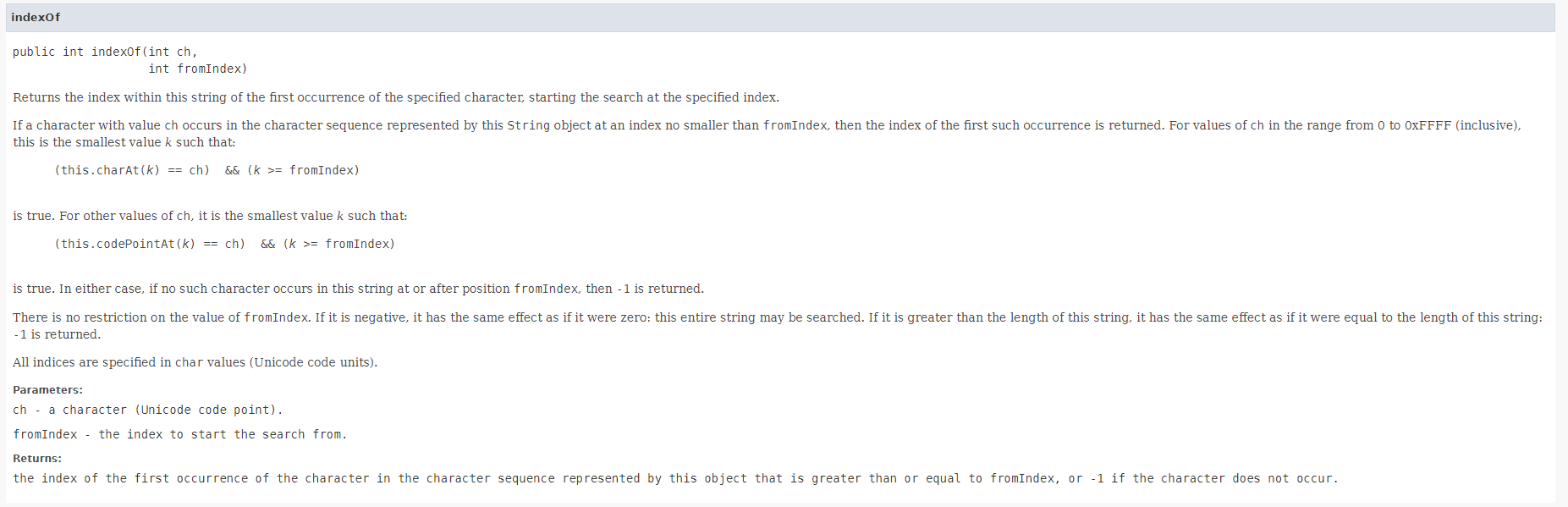
На снимке экрана показаны три главных панели веб-страницы:

* на левой верхней панели можно выбрать пакет;
* на левой нижней панели перечислены классы пакета, на этой панели выбран класс String;
* на главной панели справа отображаются сведения о классе String.



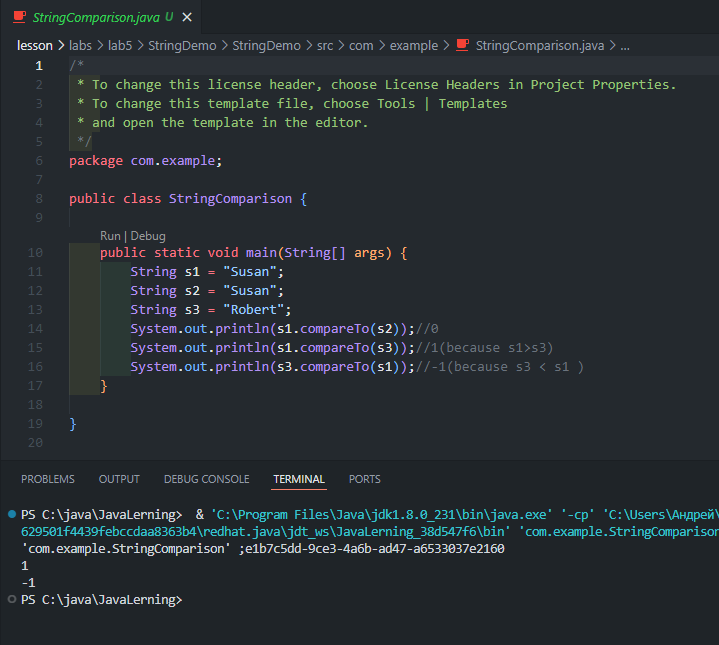
Также на главной панели представлен список методов класса. Имя каждого метода и типы его параметров представляют собой гиперссылки для просмотра дополнительных сведений.





1. Проект StringDemo.

Метод compareTo:



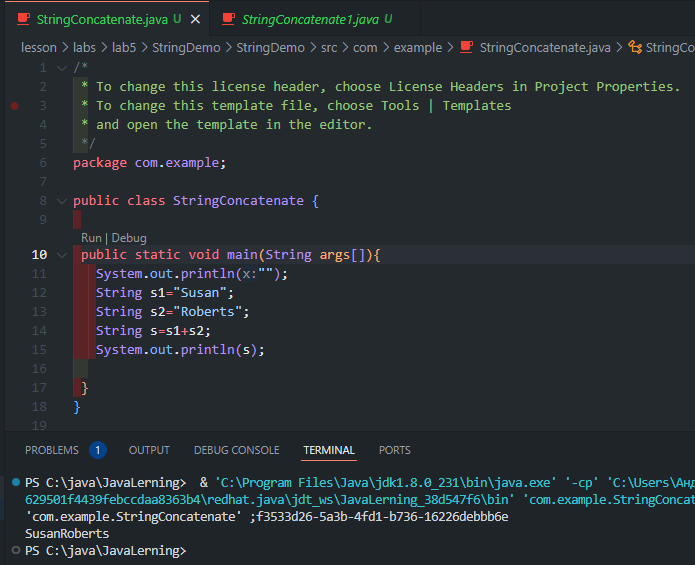
Сравнивает две строки лексикографически. Сравнение основано на значении Unicode каждого символа в строках. Возвращается:

* «0» если строки равны;
* значение меньше нуля если строка лексикографически меньше аргумента String;
* значение больше нуля если строка лексикографически больше аргумента String;

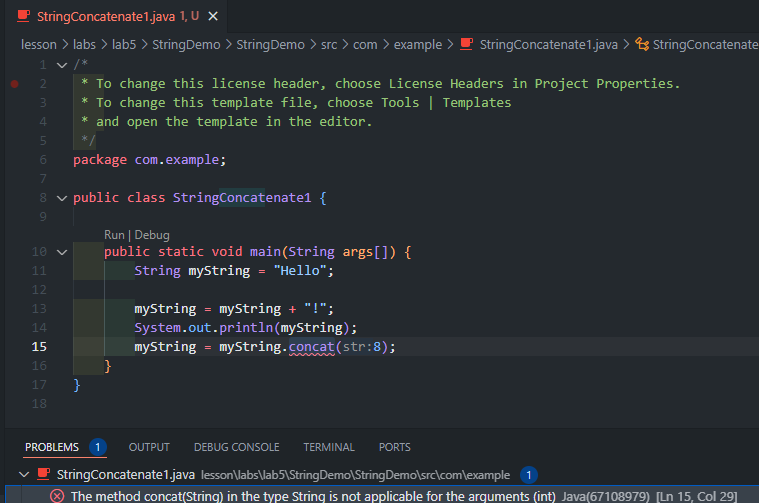
Если строки различны, то метод возвращает разницу двух символьных значений в позиции k (наименьший индекс, в котором символы строк отличаются). Если строки одинаковы, то метод вернет разницу длин строк.

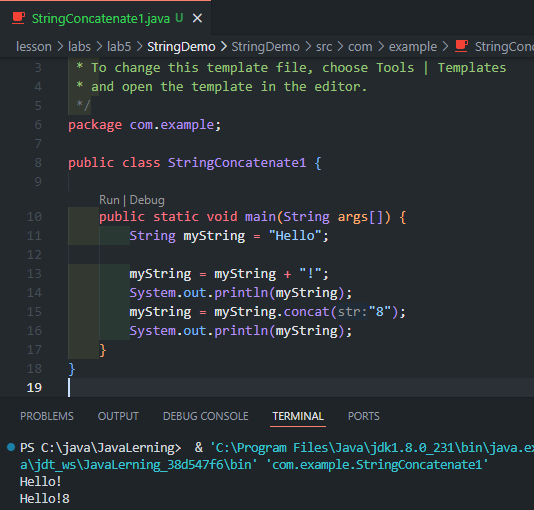
Метод concat:

Конкатенация строк — это операция склеивания двух строк с использованием знака «+». Может объединять строки и значения с типом данных char, int, double или float. Преобразует значение в формат строки перед объединением.

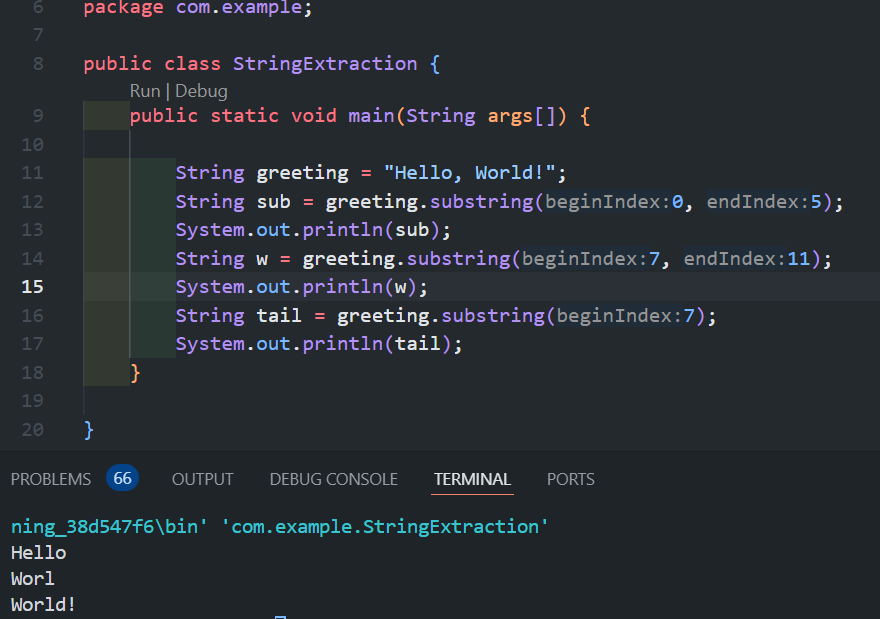


Метод concat объединяет строки (вызывается только для строк: проверяет совместимость типов данных, и в случае обнаружения несоответствий в среде компиляции отображается ошибка).





Метод substring:



Метод substring возвращает строку, которая является подстрокой данной строки. Может содержать один или два параметра типа int:

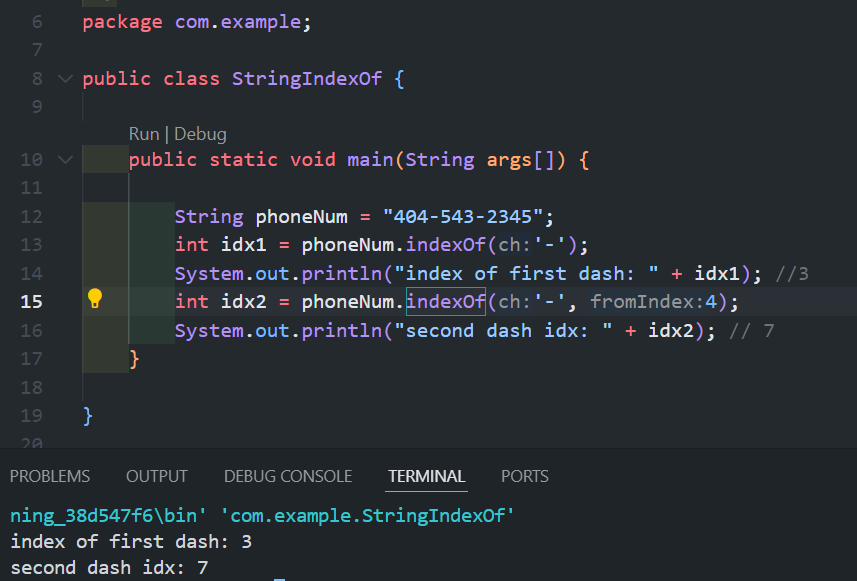
* один параметр — начальный индекс включительно: подстрока начинается с символа с указанным индексом и продолжается до конца данной строки;
* два параметра — начальный индекс включительно, конечный индекс не включительно: подстрока начинается с начального индекса и продолжается до конечного индекса.

Возникает ошибка IndexOutOfBoundsException: если индекс больше длины строки, индекс отрицательный или конечный индекс меньше начального.

Метод indexOf:

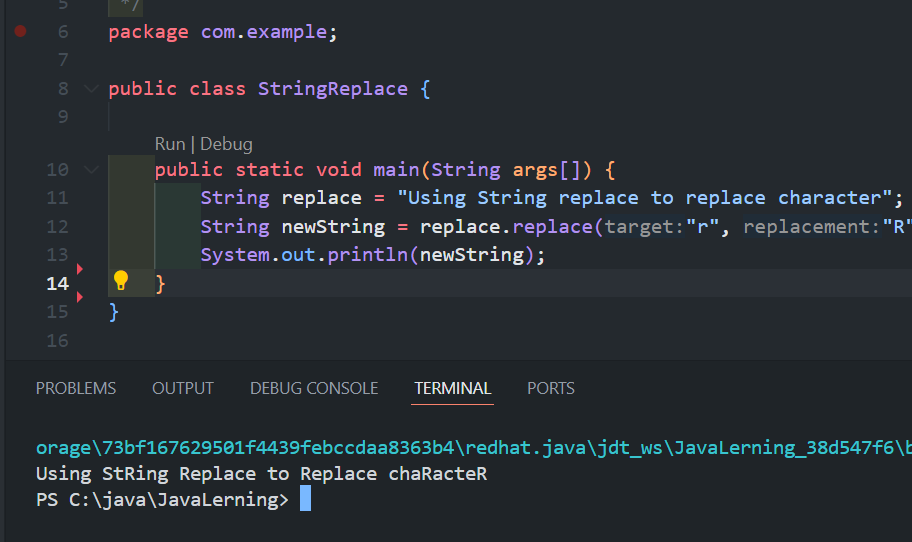
Метод indexOf возвращает индекс первого вхождения символа в строке. Может иметь один или два параметра:

* один параметр типа int (кодировка символов Unicode) или char: возвращает индекс первого вхождения символа в строке или «-1», если символа в строке нет;
* два параметра — один типа int (char) и один типа int (индекс, с которого следует начать поиск): возвращает индекс первого вхождения символа в строке, начиная со второго параметра и до конца строки, возвращает «-1», если такого нет.



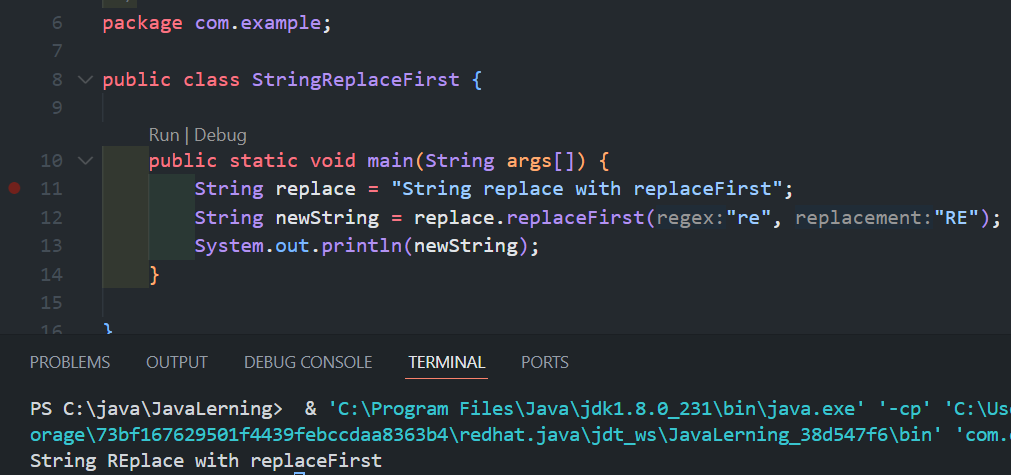
Метод replace:

Метод replace содержит два параметра типа char (String): старый символ и новый символ. Метод возвращает строку, полученную из данной строки путем замены ВСЕХ вхождений первого указанного параметра на второй.



Метод replaceFirst:

Метод replaceFirst содержит два параметра типа String. Заменяет подстроку, представленную первым параметром, на другую подстроку, заданную вторым параметром.

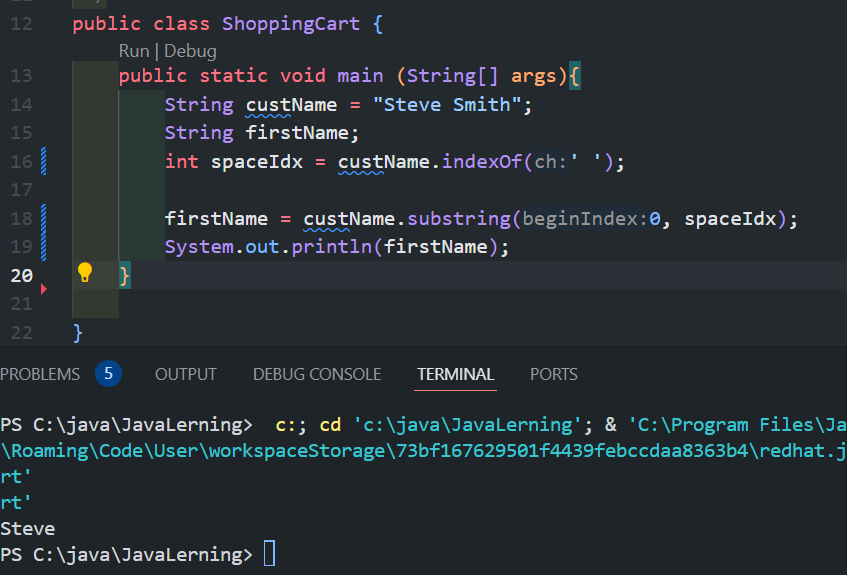


Рассмотрим еще два метода:

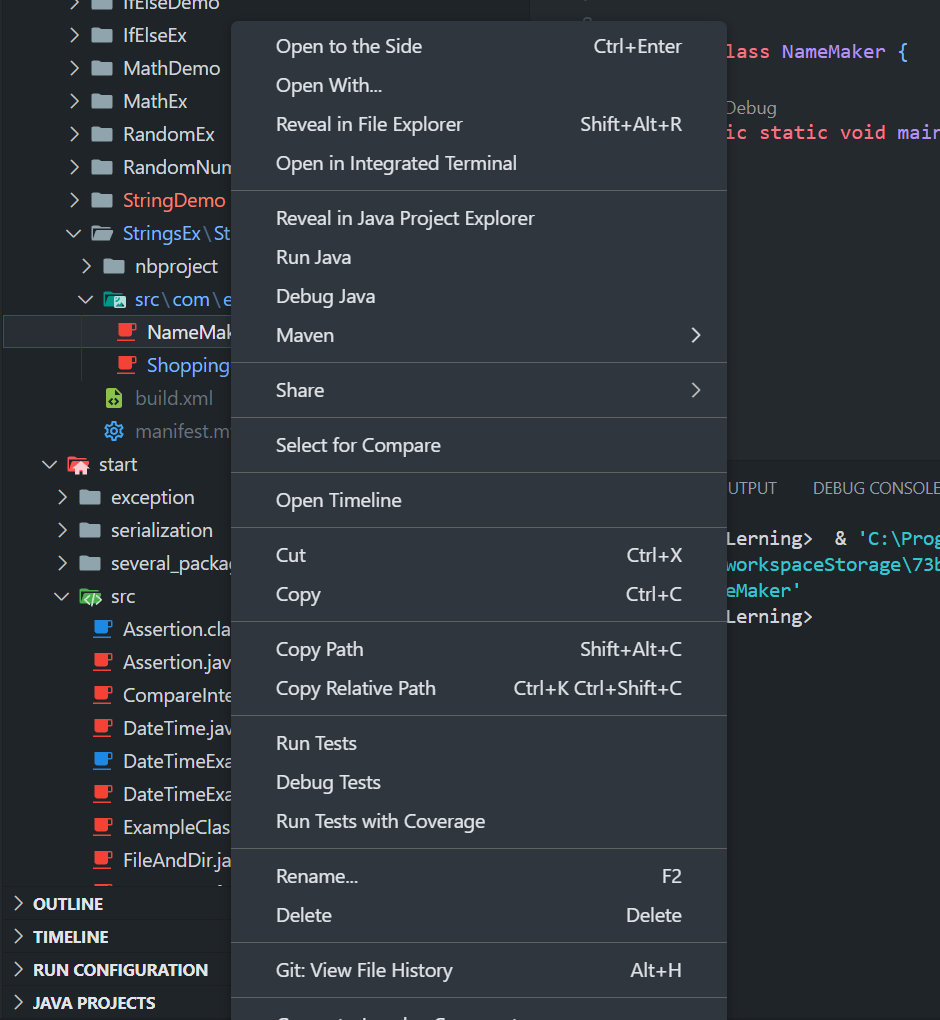
* Метод charAt возвращает символ строки, находящийся в индексе, переданном в качестве параметра.
* С помощью метода length в классе String можно вычислить длину строки. Возвращает длину или количество символов в имени в виде целочисленного значения.

1. Проект StringEx.

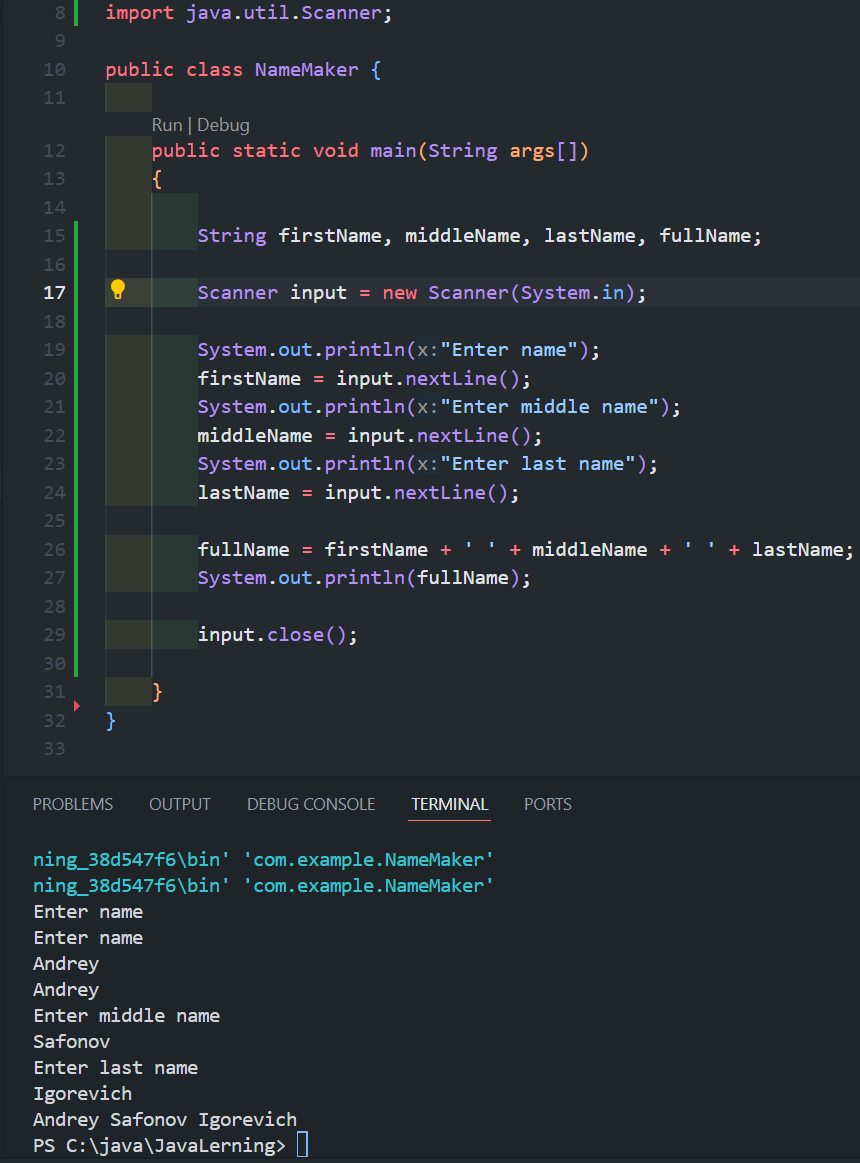
* Импортируйте и откройте проект StringsEx.
* Проверьте ShoppingCart.java. Выполните следующие действия.
* С помощью метода indexOf получите индекс символа пробела (" ") в custName. Присвойте его значение индексу spaceIdx.
* С помощью метода substring и spaceIdx получите первую часть имени custName. Присвойте это значение переменной firstName и выведите firstName.



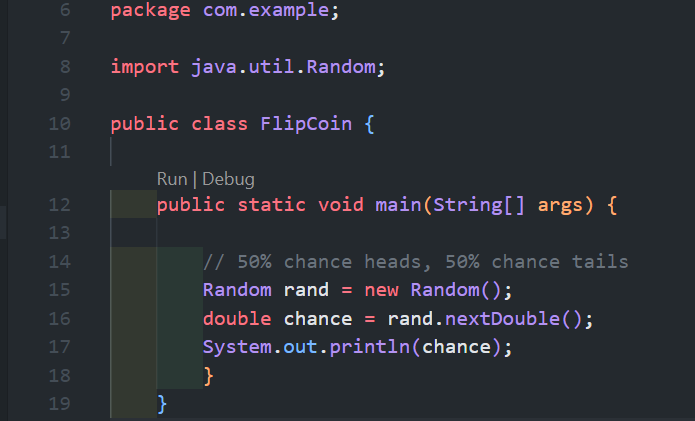
Если в проекте есть два файла .java с главными методами, то запуск через кнопку Run Java в VS Code (или сочетание клавиш Ctrl+F5).



* Изучите NameMaker.java.
* Выполните следующие действия.
* Объявите переменные String: firstName, middleName, lastName и fullName
* Создайте запрос, чтобы пользователи могли ввести свое имя, отчество и фамилию, и считайте эти данные с клавиатуры.
* Задайте и выведите fullName как firstName+a blank char+middleName+a blank char+lastName.

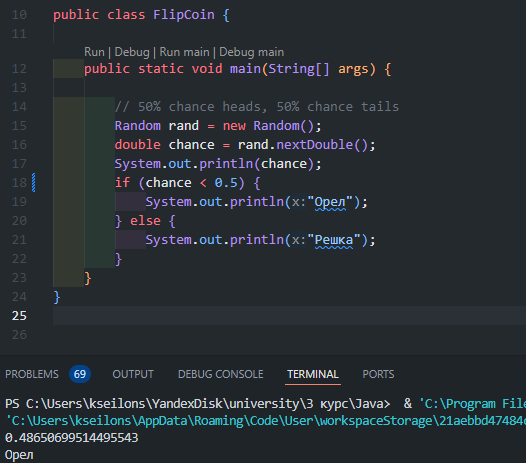


1. Задания по уроку 4.4.
2. Программа генерирует случайное число типа double.

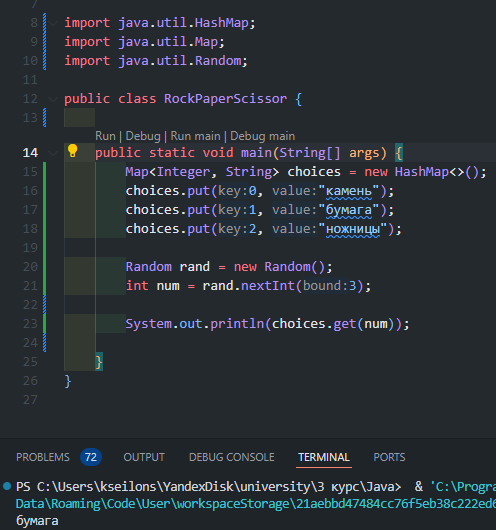


Добавим условие:

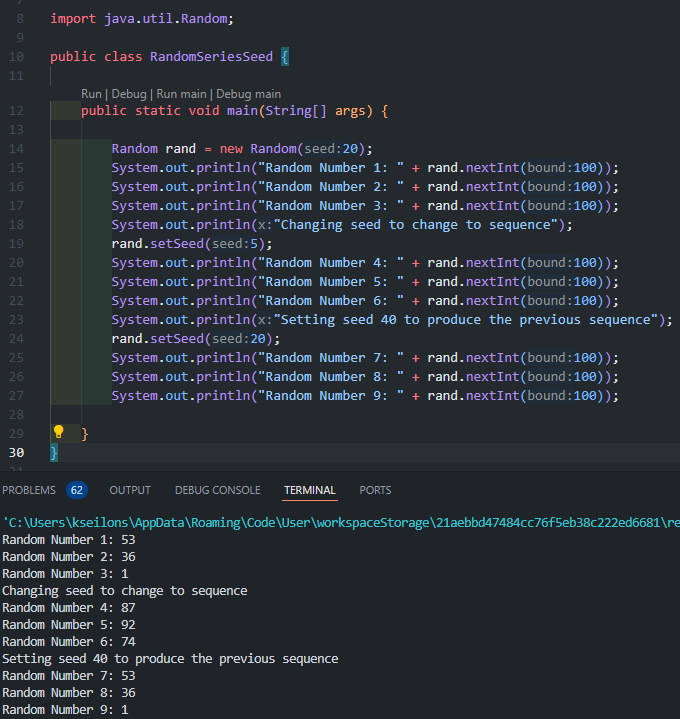
* Если вероятность < 0,5, получается результат "орел". В противном случае получается результат "решка".



1. Генерируется случайное целое число в диапазоне от 0 до 3 (не включительно), и в зависимости от сгенерированного числа программа выводит «камень», «бумага» или «ножницы».



1. RandomSeriesSeed.



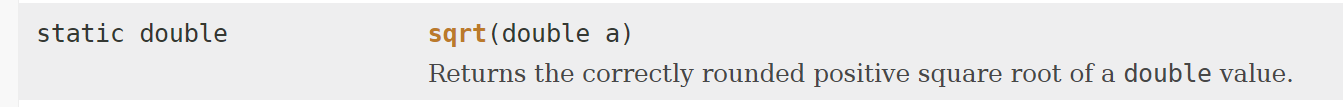
Во время создания экземпляра класса Random передаем постоянное целое число, чтобы задать начальное значение. Для того, чтобы изменить начальное значение, вызываем метод setSeed().

При задании одинакового начального значения мы получаем одинаковые последовательности и одну отличающуюся от них последовательность (начально значение равно 20).

Т.е. каждый раз, когда передается одно и то же начальное значение, возвращается одна и та же случайная последовательность.

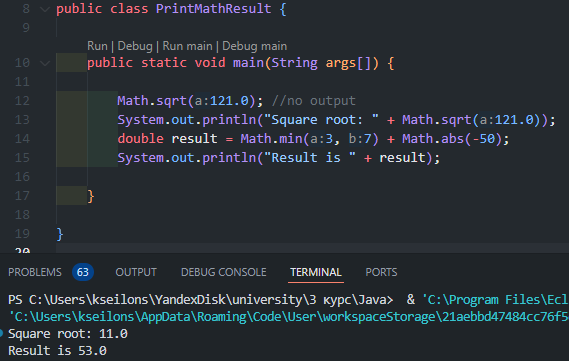
1. Задания по уроку 4.5.
2. Класс Java Math содержит методы для выполнения математических расчетов. Методы класса Math являются статическими. Это означает, что вам не придется создавать объект класса Math для вызова методов. Входит в пакет java.lang. Содержит следующие статистические поля:

* PI – значение типа double, которое ближе чем любое другое к *pi* (отношение длины окружности к ее диаметру). Число «Пи» равно 3.141592653589793.

 Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

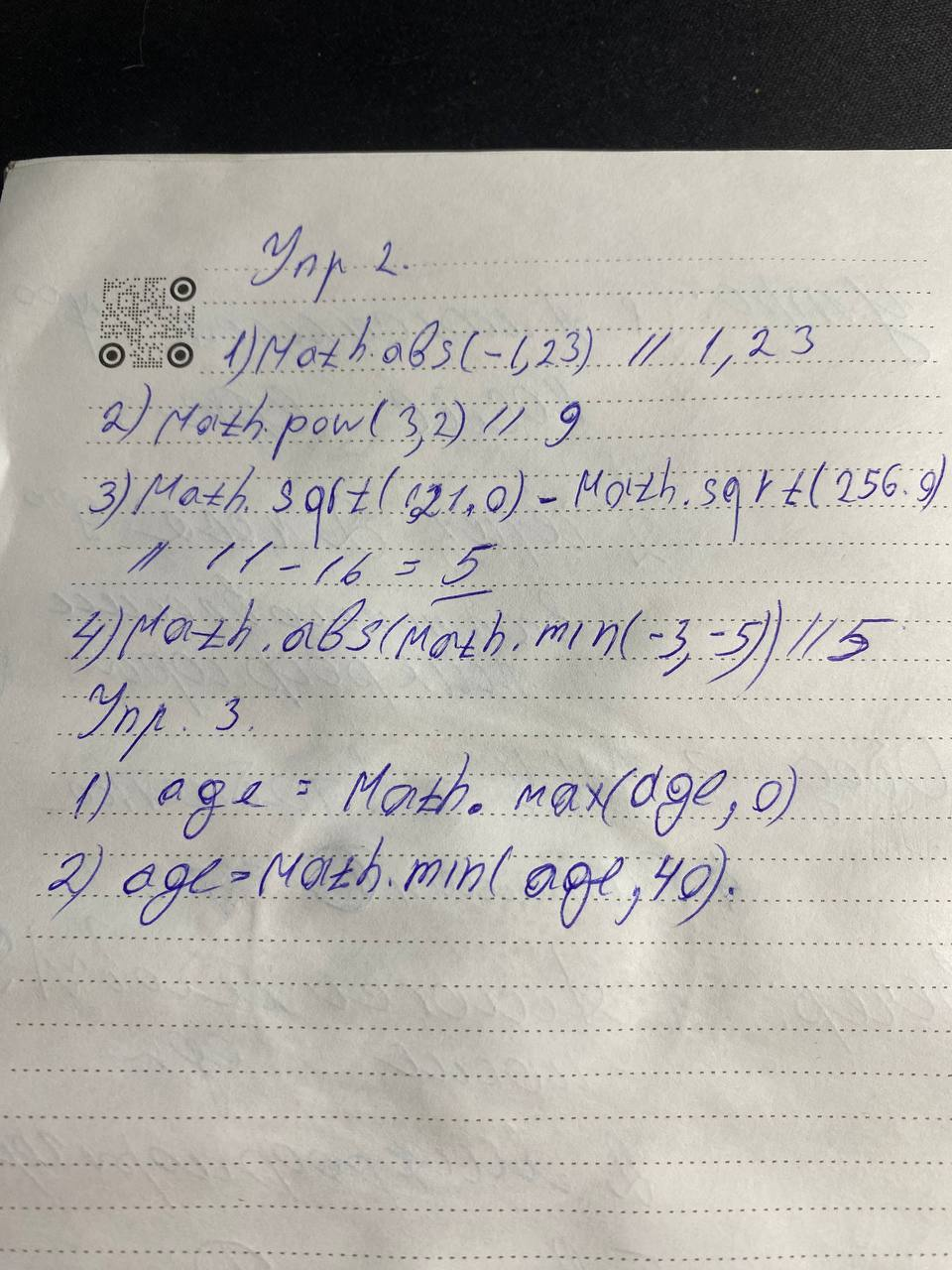
1. PrintMathResult.



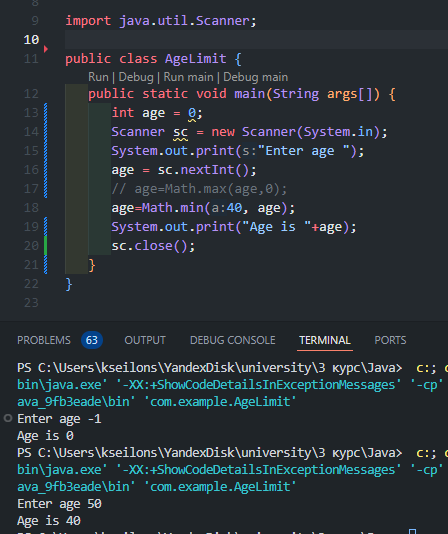
Методы класса Math не выводят результаты на консоль. Каждый метод возвращает числовой результат. С возвращаемым значением можно выполнять многие другие действия, помимо вывода.

Чтобы увидеть результат, необходимо вывести его или сохранить в переменной (первый вывод), а также результаты можно объединить и использовать в более крупном выражении (второй вывод).

1. «На бумаге».

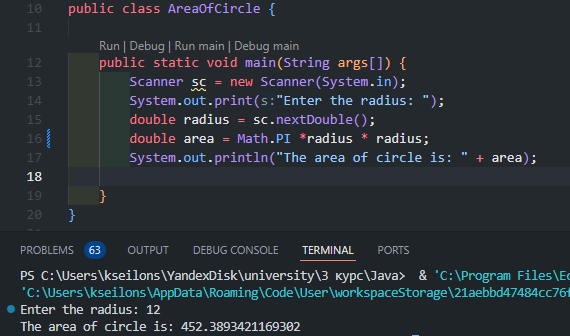


Сравним с AgeLimit:



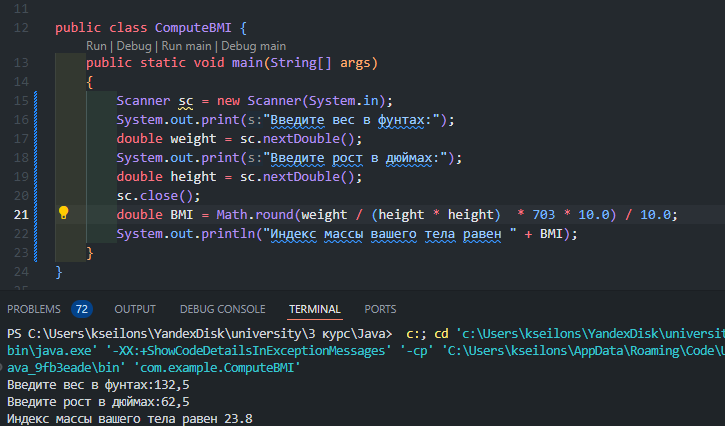
Действительно, выражение math.max(age,0) меняет отрицательные значения возраста на 0, а math.min(40,age) - ограничивает максимальный возраст значением 40.

1. Площадь круга.



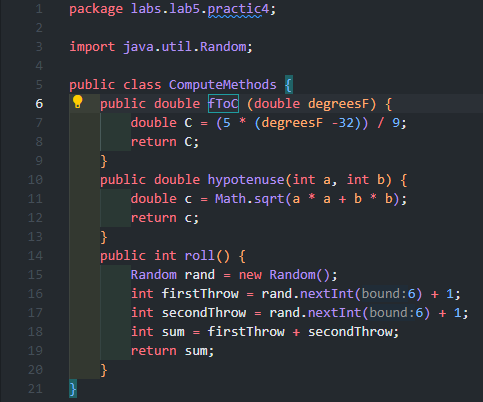
Использование поля Math.PI для расчета площади дает более точный результат, чем использование постоянного значения pi, например 3,14.

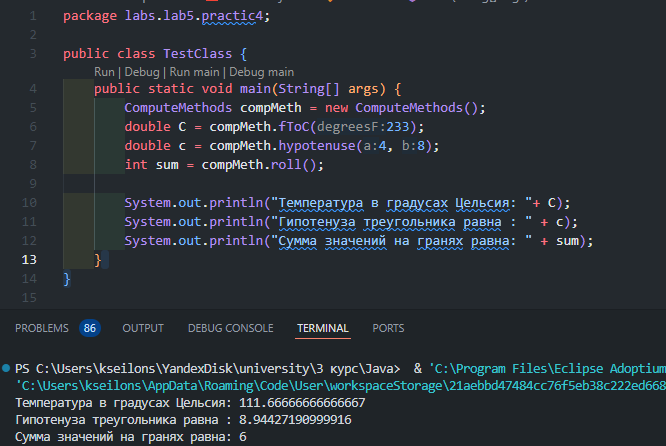
1. Индекс массы тела человека:



1. Упражнение 4.
2. Проблема 1.

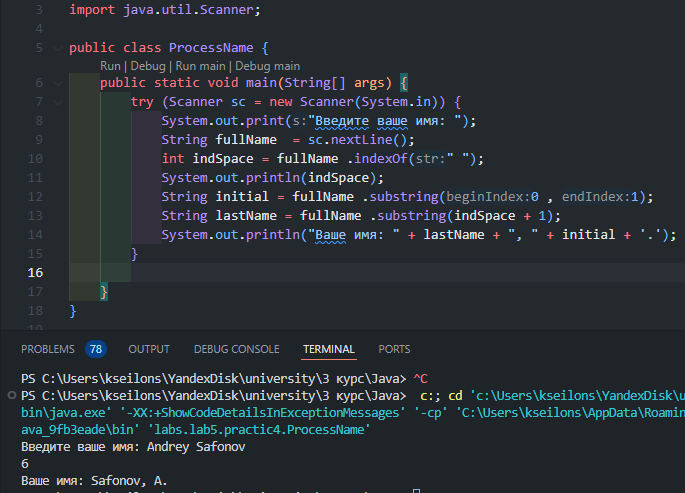
* Создайте java-файл ComputeMethods.java и определите следующие три метода: public double fToC(double degreesF) public double hypotenuse(int a, int b) public int roll()
* Создайте второй java-файл TestClass.java и выполните следующее: добавьте метод main в метод main: создайте экземпляр ComputeMethods, вызовите методы, определенные в файле ComputeMethods.java этого экземпляра, и отобразите их результаты.





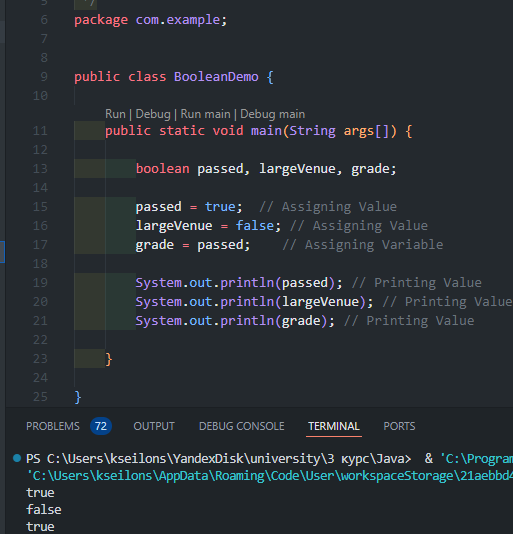
Методы fToC (преобразует температуру, заданную в градусах по Фаренгейту, в градусы по Цельсию), hupotenuse (вычисляет длину гипотенузы треугольника на основе длин его катетов) и roll (моделирует бросок двух кубиков и отображает сумму значений на их верхних гранях.).

1. Проблема 2.

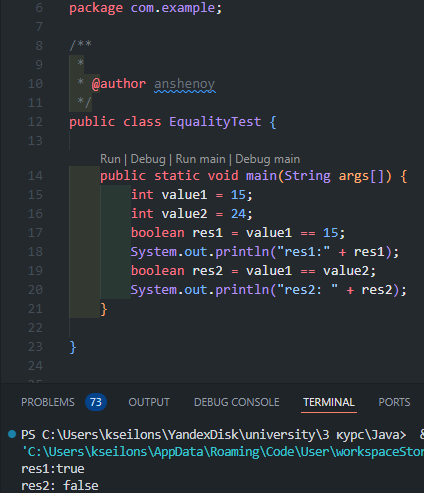


1. IfElseDemo.

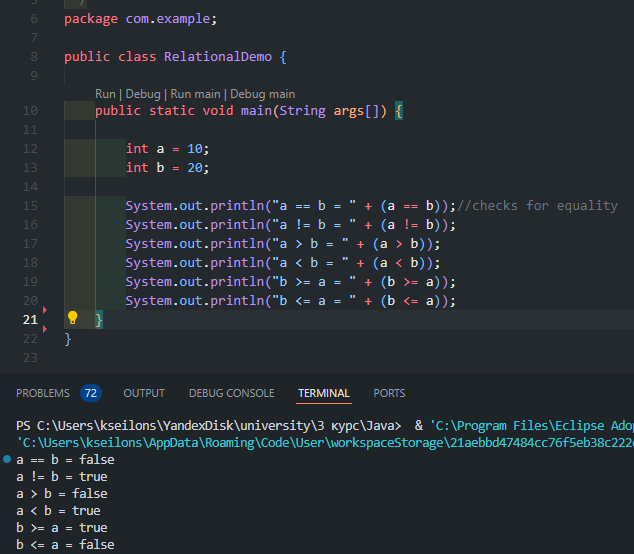
1) Тип данных boolean объявляется с помощью ключевого слова boolean и может иметь только два возможных значения: true и false.



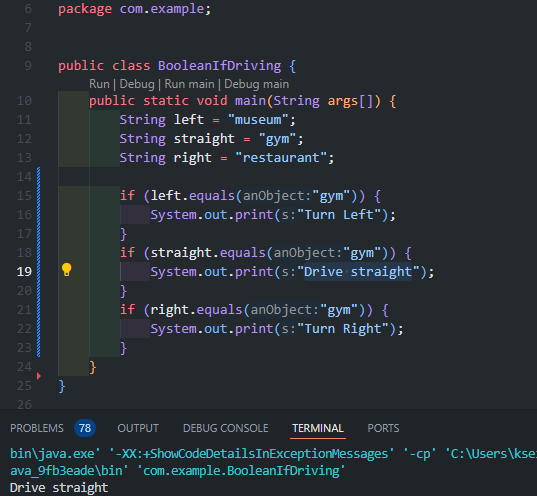
2) == является оператором отношения. Этот оператор проверяет, являются ли обе части выражения boolean равными. Переменной boolean можно присвоить любое значение, которое возвращает выражение boolean. В приведенном ниже примере переменной res1 присваивается значение, возвращающее true (value1 =15 и оно сравнивается со значением 15), а переменной res 2 – значение, возвращающее false (сравниваются value1 и value2, а они не равны).



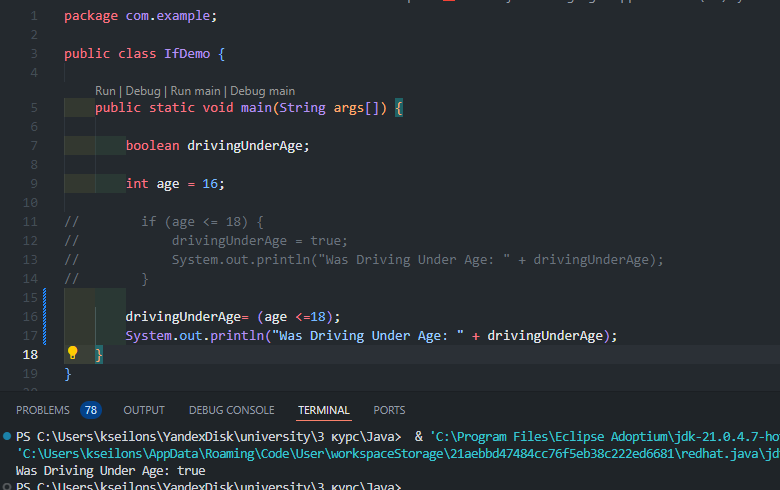
3) Работа с операторами отношения:



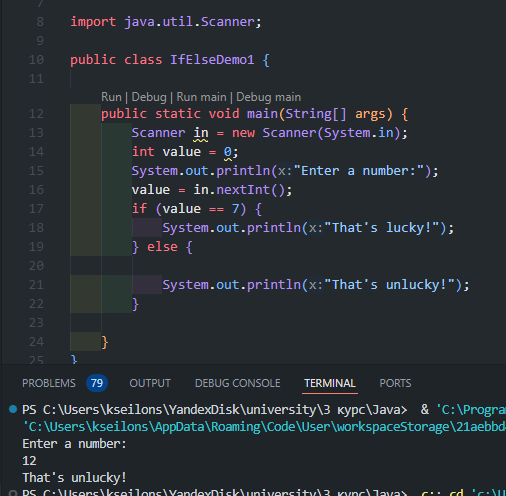
4) Оператор if состоит из выражения boolean, после которого идет один или несколько операторов. Если выражение boolean имеет значение true: Оператор выполнен. Если выражение boolean имеет значение false: Оператор пропущен. В примере выражение boolean во втором операторе if возвращает выражение true. Поэтому в консоли выводится "Drive straight".



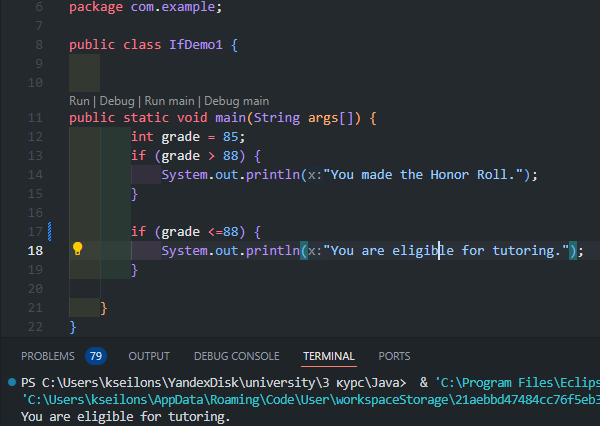
5) В примере проверяется возраст человека, и если он ниже 18, то выводит предупреждение о несовершеннолетнем за рулем (можно делать через if или просто присвоением значения переменной типа boolean):



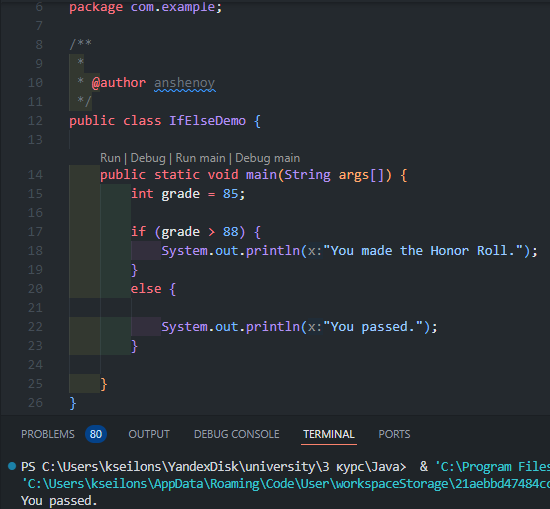
6) В данной программе введенное пользователем число сравнивается с «7»: если оно равно «7», то это удачное число, иначе – неудачное.



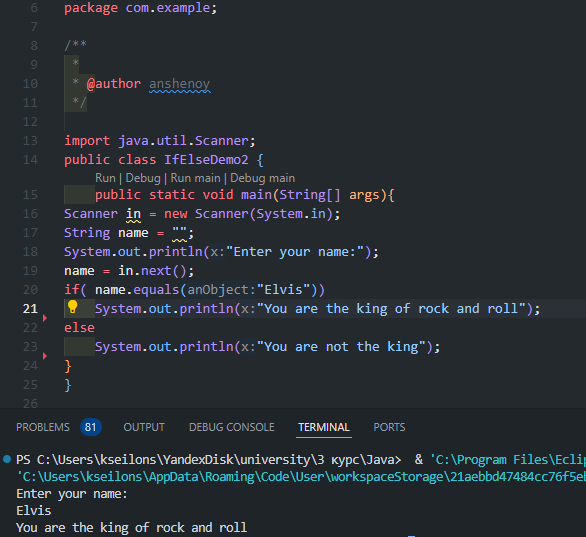
7) В примере представлены два оператора if: первый выполняет проверку значений, которые больше 88, второй выполняет проверку значений, которые меньше или равны 88. Оба оператора if оценены, даже если первый из них имеет значение true.



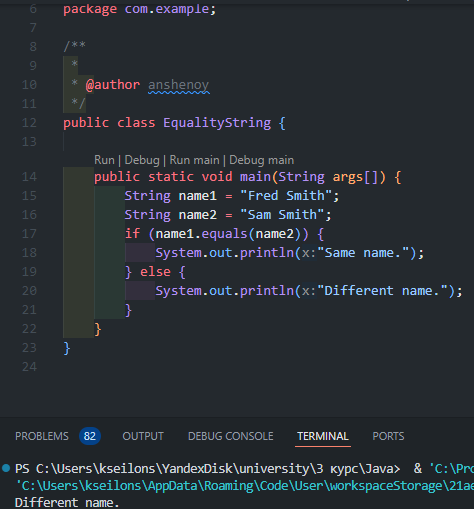
8) Можно заменить два оператора if на оператор if/else. Оператор if/else более эффективен, так как выполняется только одно сравнение.



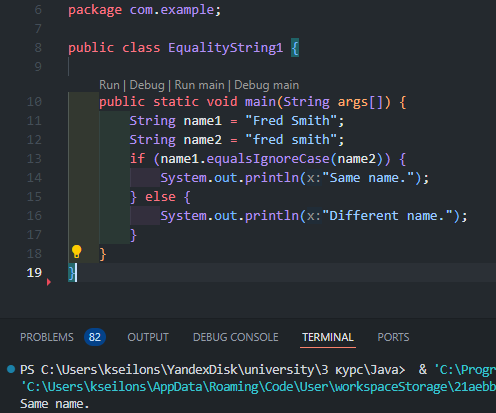
9) Сравнивается введенная пользователем строка со строкой «Elvis»: если равны, то выводит «You are the king of rock and roll», иначе «You are not the king». Сравниваем переменные String с помощью метода equals(). Он принимает аргумент String, проверяет равенство содержимого String, затем возвращает выражение boolean.



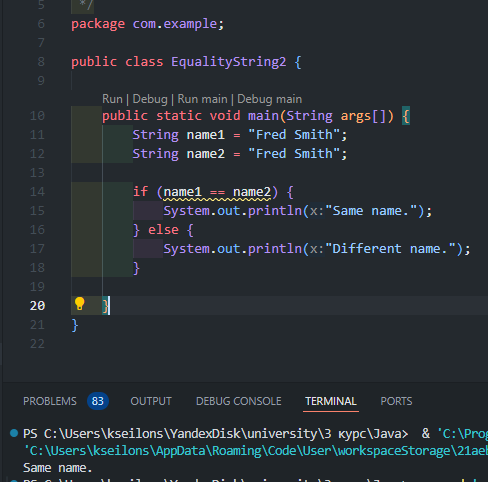
10) Сравниваем переменные String с помощью метода equals().



11) Сравниваем переменные String с помощью метода equalsIgnoreCase(). Он выполняет лексикографическое сравнение строк, при котором различия регистра игнорируются.

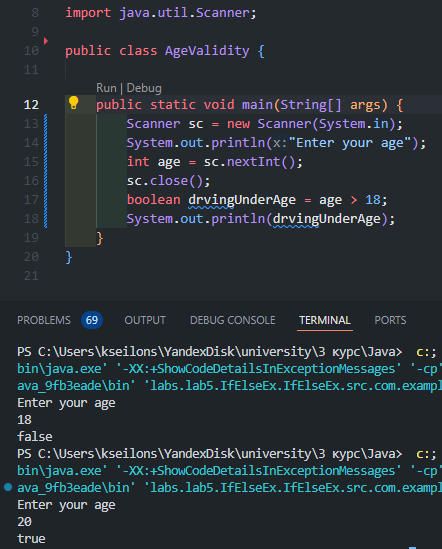


12) Почти никогда не следует выполнять сравнение переменных String с помощью ==. Оператор == в Java сравнивает ссылки на объекты, а не сами объекты. В случае строк это означает, что == проверяет, указывают ли две переменные на один и тот же объект в памяти, а не на то, содержат ли они одинаковую последовательность символов.



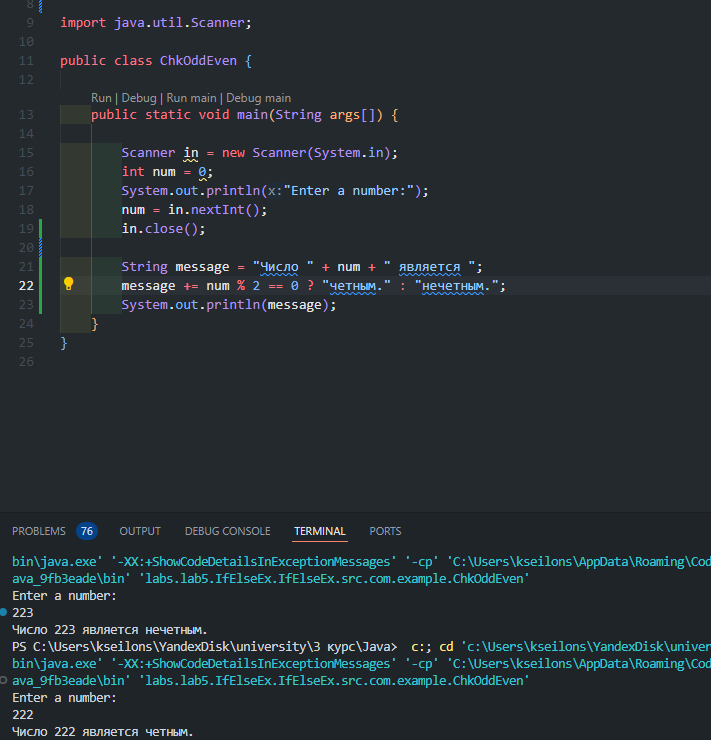
1. IfElseEx.
2. Упражнение 1.

Ввод пользователями своего возраста. Объявление переменной boolean, drivingUnderAge. Инициализация drivingUnderAge до значения false. Создание выражения boolean для проверки, является ли введенный пользователем возраст меньше или равен 18, затем присвоение переменной drivingUnderAge значение true. Вывод значения drivingUnderAge.



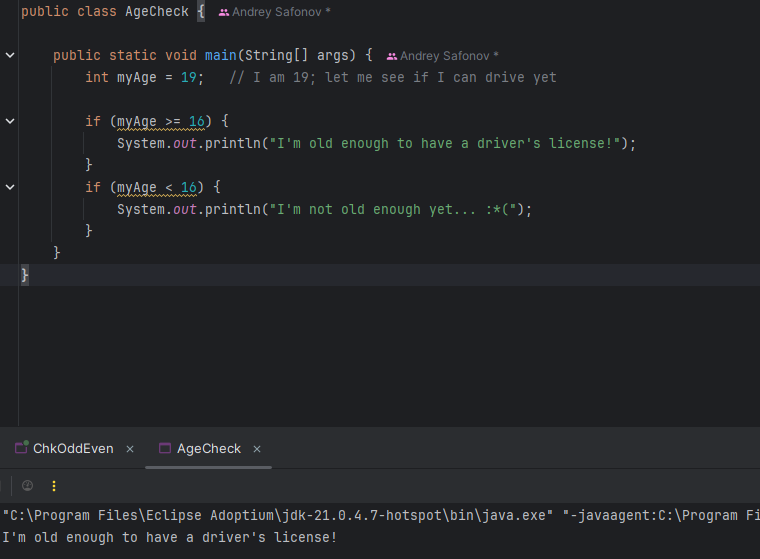
1. Упражнение 2.

Проверка числа на четность.

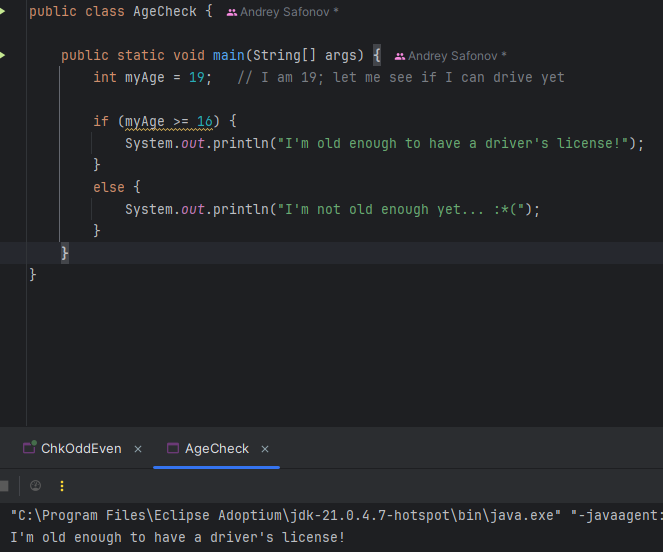


1. Упражнение 3.

Первое решение проблемы:

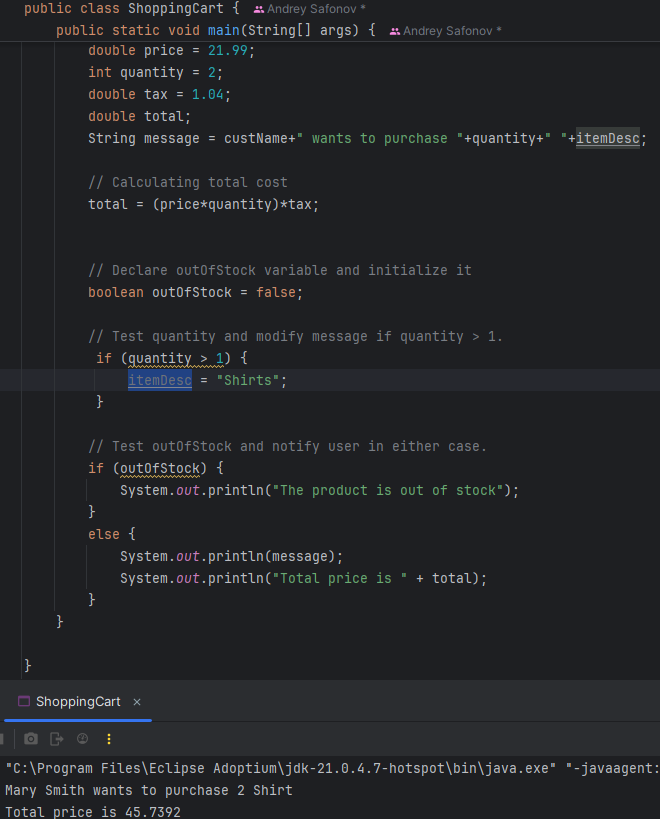


Второе:



1. Упражнение 4.

Объявление и инициализация переменной boolean outOfStock. Если количество > 1, меняем переменную сообщения, чтобы указать множественное число. Если позиция отсутствует, сообщаем пользователю о ее недоступности. Или выводим сообщение и общую стоимость.



1. Упражнение 5.

* Объявите имя переменной String.
* Пользователь должен ввести значение для имени. –
* Убедитесь, что указано имя "Moe", затем введите текст "Вы король рок-н-ролла".
* Или введите текст "Вы не король".
* Не используйте ==

