Министерство науки и высшего образования Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем и программной инженерии

**Лабораторная работа №1**

**по дисциплине**

**«Графический и веб-дизайн»**

**ЯЗЫК JAVASCRIPT**

**Выполнил**:

ст. гр. ПРИ-120

Д. А. Грачев

**Принял**:

Шамышев А. А.

Владимир, 2023

ЦЕЛЬ РАБОТЫ

Получить навыки работы с языком JavaScript; освоить основные типы данных и операторы JavaScript; изучить механизм прототипирования.

ВЫПОЛНЕНИЕ РАБОТЫ

1. Был создан HTML файл, к которому был подключен файл со скриптами, ниже представлен листинг данного файла, где в тэге script можно увидеть подключение файла javascript

<!doctype *html*>  
<html *lang*="en">  
<head>  
 <meta *charset*="UTF-8">  
 <meta *name*="viewport"  
 *content*="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">  
 <meta *http-equiv*="X-UA-Compatible" *content*="ie=edge">  
 <title>Лабораторная работа 1</title>  
 <script *src*="script.js"></script>  
</head>  
<body>  
 <div *class*="task2">  
 <button *onclick*="task2()">Показать задание 2</button>  
 <div *class*="display"></div>  
 </div>  
 <br>  
 <div *class*="task3">  
 <button *onclick*="task3(20)">Показать задание 3</button>  
 <div *class*="display"></div>  
 </div>  
 <br>  
 <div *class*="task4">  
 <button *onclick*="task4()">Показать задание 4</button>  
 <div *class*="display"></div>  
 </div>  
 <br>  
 <div *class*="task5">  
 <button *onclick*="task5()">Показать задание 5</button>  
 <div *class*="display"></div>  
 </div>  
 <br>  
 <div *class*="task6">  
 <button *onclick*="task6()">Показать задание 6</button>  
 <div *class*="display1"></div>  
 <div *class*="display2"></div>  
 </div>  
 <br>  
 <div *class*="task7">  
 <button *onclick*="task7()">Показать задание 7</button>  
 <div *class*="display"></div>  
 <div *class*="display2"></div>  
 </div>  
 <br>  
 <div *class*="task8">  
 <button *onclick*="task8()">Показать задание 8</button>  
 <div *class*="display"></div>  
 <div *class*="display2"></div>  
 </div>  
 <br>  
 <div *class*="task9">  
 <button *onclick*="task9()">Показать задание 9</button>  
 <div *class*="display"></div>  
 <div *class*="display2"></div>  
 </div>  
</body>  
</html>

1. Создать собственный JS-скрипт с использованием переменных и условным оператором («if» и «?»). Запустить скрипт в браузере. JS-скрипт должен быть создан отдельным файлом и подключен к HTML-странице.
   1. Листинг

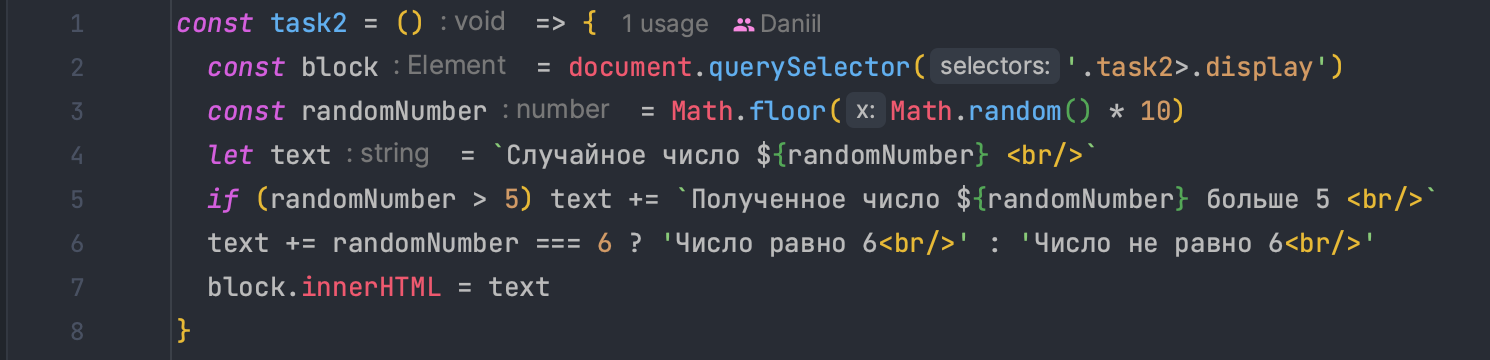


Рисунок . Задание 2 - листинг

* 1. Описание работы скрипта: во второй строке с помощью css-селектора выбирается блок на странице на который в дальнейшем будет вставлен результат работы скрипта, далее происходит рандомизация выполнения скрипта, в зависимости от результата выполнения функции random() выводится различный результат на экран с помощью условных конструкций: условного оператора if и тернарного оператора
  2. Результат

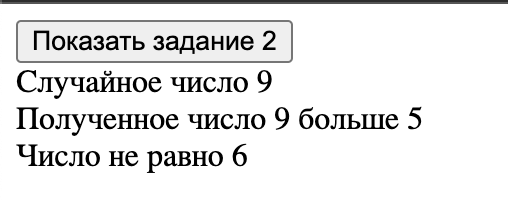


Рисунок . Задание 2 - результат

1. Выведите простые числа с помощью цикла for на интервале от 2 до n, где n любое число больше 10. (Натуральное число, большее 1, называется простым, если оно ни на что не делится, кроме себя и 1)
   1. Листинг

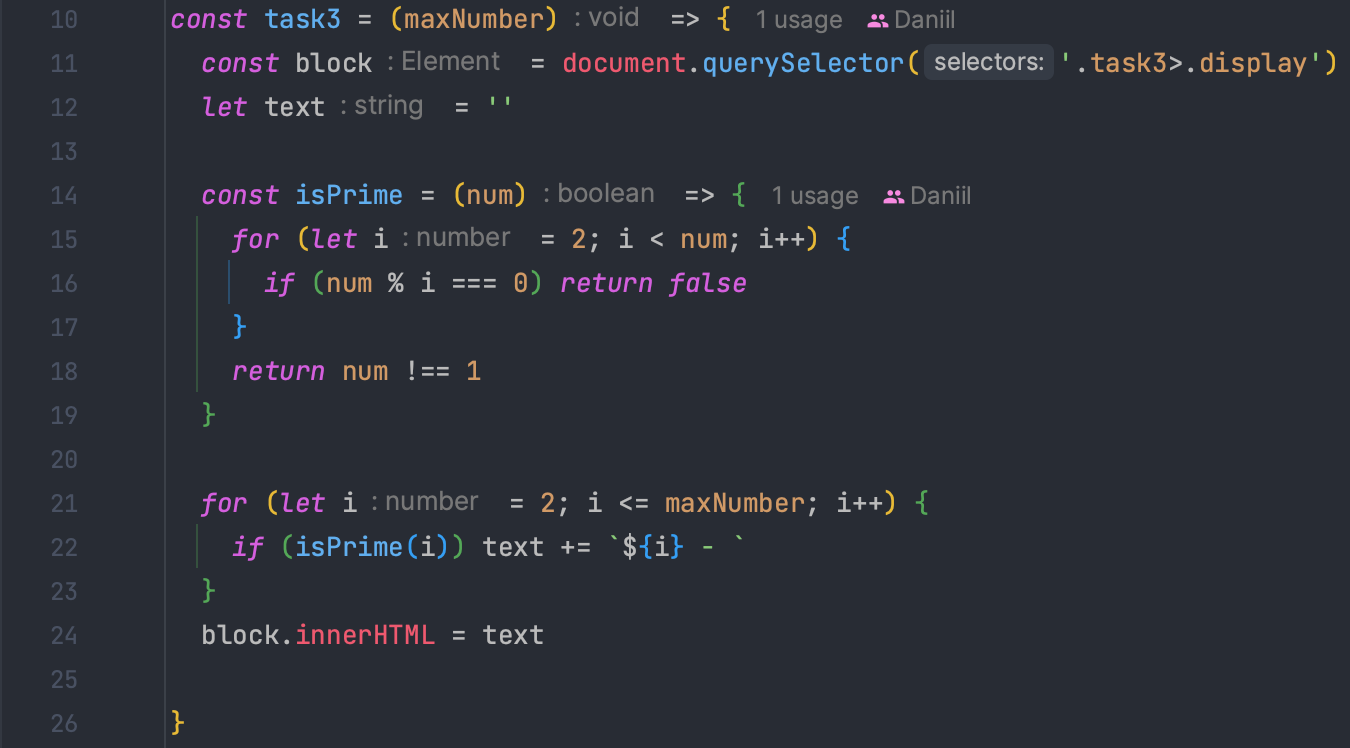


Рисунок . Задание 3 - листинг

* 1. Описание работы скрипта: в начале так же выбирается блок для дальнейшей вставки результата работы и отображения в браузере, далее числа до введенного maxNumber проверяются на то, являются ли они простыми путем проверки делимости на все возможные делители
  2. Результат

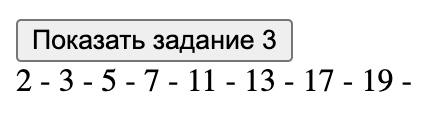


Рисунок . Задание 3 - результат

1. Создать собственный JS-скрипт с примерами использования forEach(), map(), some(), reduce(). Запустить скрипт в браузере. JSскрипт должен быть создан отдельным файлом и подключен к HTMLстранице;
   1. Листинг

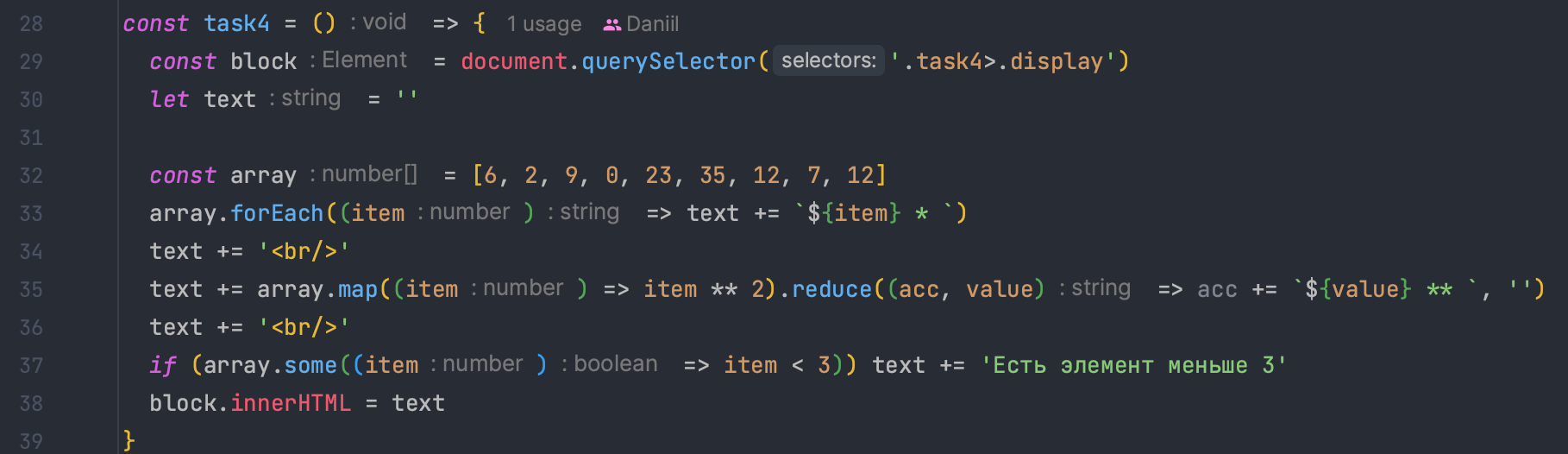


Рисунок . Задание 4 - листинг

* 1. Описание работы скрипта: создается массив чисел, после чего из этого массива генерируется строка с помощью метода forEach(), содержащая все эти числа, далее с помощью метода map() создается массив квадратов первоначальных чисел, затем так же создается строка, содержащая эти квадраты, но уже с помощью метода reduce(), и, в конце, с помощью метода some() проверяется есть ли элементы меньшие 3 в массиве и добавляется в вывод на экран
  2. Результат

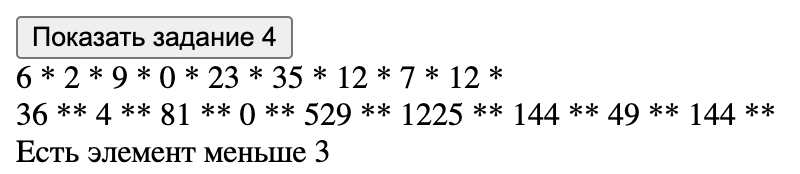


Рисунок . Задание 4 - результат

1. Создайте JS-скрипт с примером использованием объектов и их прототипов;
   1. Листинг

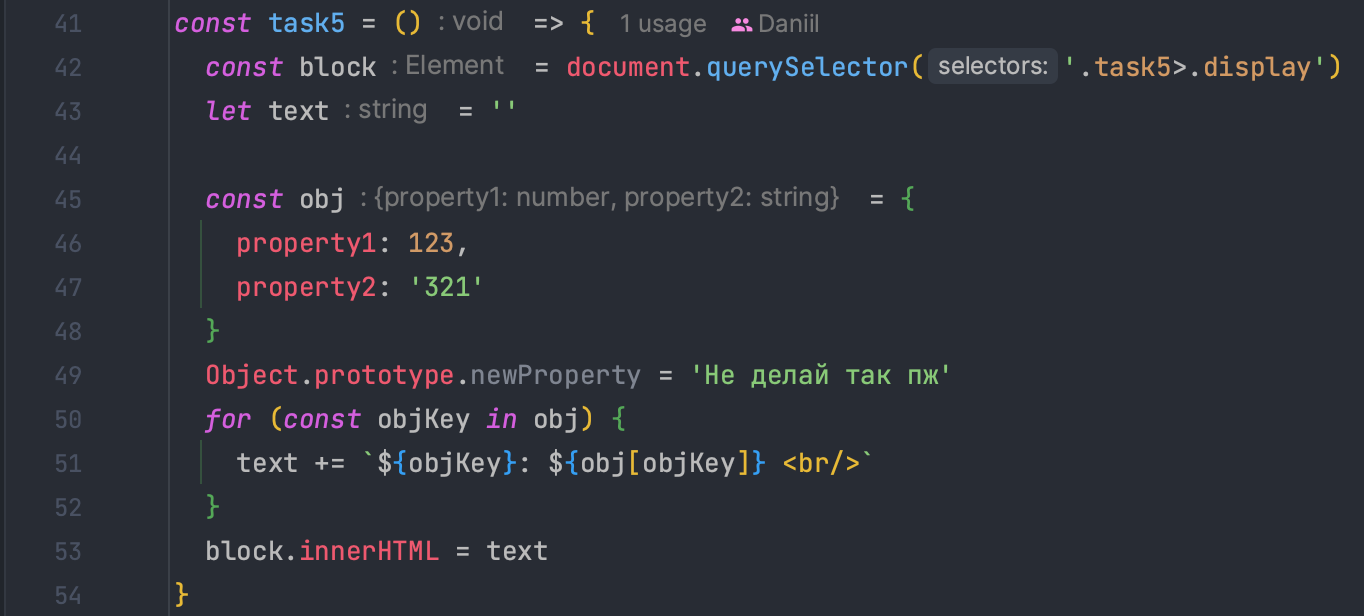


Рисунок . Задание 5 - листинг

* 1. Описание работы скрипта: создается объект, после чего ему добавляется свойство newProperty с помощью прототипа, после чего все свойства объекта выводятся с помощью цикла for…in
  2. Результат

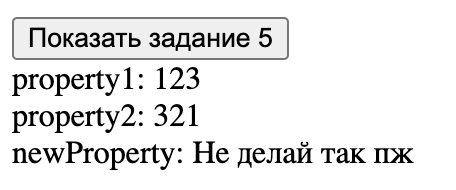


Рисунок . Задание 5 - результат

1. Реализуйте несколько (2 или больше) классов и продемонстрируйте на них 3 основные парадигмы ООП (наследование, инкапсуляция и полиморфизм);
   1. Листинг



Рисунок . Задание 6 - листинг

* 1. Описание работы скрипта: был создан класс Man, в котором сразу же используется инкапсуляция в виде создания приватного поля logCount, которое будет увеличиваться при повторном выводе результатов, далее применено наследование для создания класса Driver и тут же полиморфизм для переопределения методов родительского класса, после чего создаются экземпляры данных классов и выводятся на экран
  2. Результат

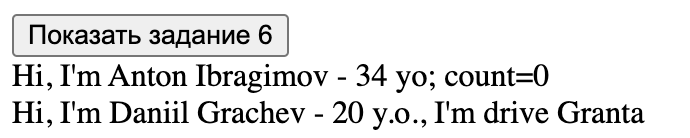


Рисунок . Задание 6 - результат

1. Добавьте метод и поле в уже созданный класс из задания 2 с помощью prototype
   1. Листинг

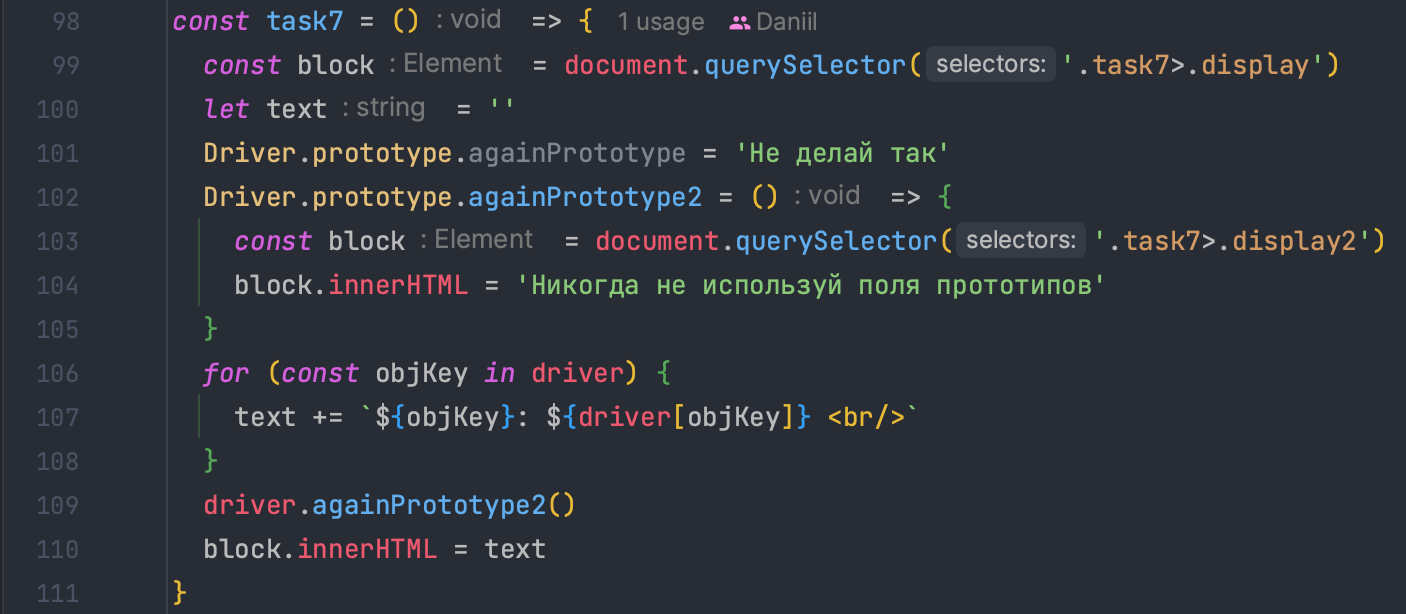


Рисунок . Задание 7 - листинг

* 1. Описание работы скрипта: с помощью прототипа к классу Driver добавляется поле и метод, которые после выводятся в браузер
  2. Результат

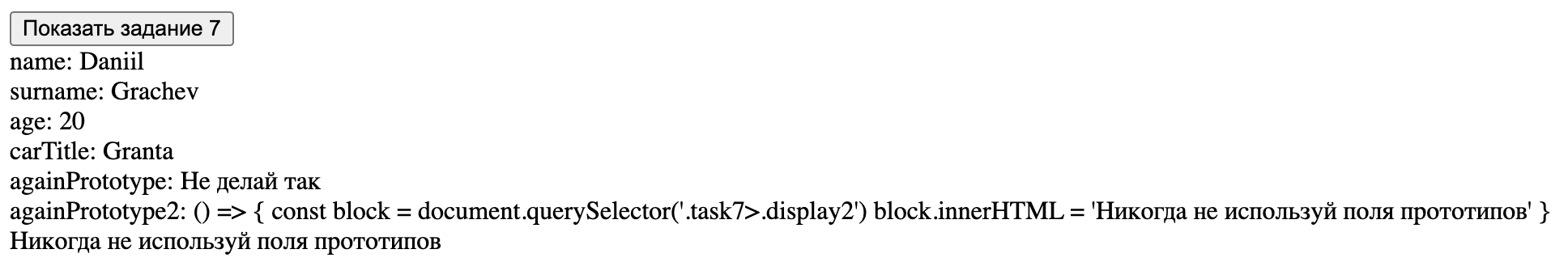


Рисунок . Задание 7 - результат

1. Определить класс «Трапеция», имеющий координаты 4-х точек, конструктор и метод для расчета площади трапеции. Создать массив экземпляров этого класса и определить, в скольких трапециях площадь больше заданного числа x
   1. Листинг



Рисунок . Задание 8 - листинг класса трапеции



Рисунок . Задание 8 - листинг скрипта

* 1. Описание работы скрипта: создан класс трапеции с нужными полями и методами, далее создается рандомное количество трапеций(от 0 до 10) с рандомными координатами(от 0 до 100), после этого считается площадь каждой трапеции и смотрится в скольких из них площадь больше рандомного числа(от 0 до 1000)
  2. Результат

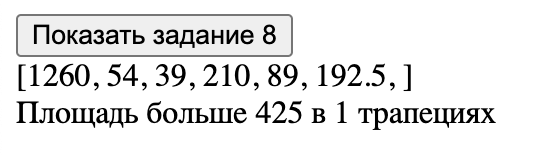


Рисунок . Задание 8 – результат

1. Выполнить задание 8 с использованием объектов;
   1. Листинг



Рисунок . Задание 9 - листинг объекта трапеции

* 1. Описание работы скрипта: был создан объект трапеции с полями и методом получения площади, далее алгоритм работы идентичен предыдущему заданию
  2. Результат

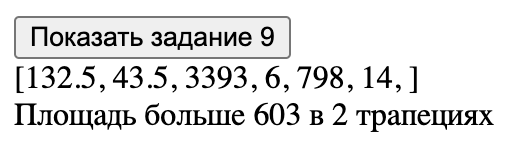


Рисунок . Задание 9 - результат

ВЫВОД

В ходе выполнения работы были получены навыки работы с языком JavaScript; освоены основные типы данных и операторы JavaScript; изучены механизм прототипирования.