**1.** Дана сторона квадрата а. Найти его периметр. **2.** Дана сторона квадрата а. Найти его площадь.

**3.** Дано расстояние L в см. Используя операцию деления нацело, найти количество полных метров в нем. ( 1м=100см). Использовать функцию Math.trunk()

**4.** Дано целое число А. Проверить истинность высказывания : “Число А является положительным”

**5.** Даны два целых числа А, В. Проверить истинность высказывания: “Справедливы неравенства А> 2 и B<=3”. Провести проверку в консоли и тремя способами в браузере.( консоль, страница и всплывающее окно.) (11, 12,13 задания)

**6.** Дано целое число. Если оно является положительным, то прибавить к нему 1, в противном случае не изменять его. Вывести полученное число.

**7.** Извлечь корень. Три способа.

1) let a=16; Math.sqrt(a) 2) Math.pow(a,0.5) 3) a\*\*0.5

**8**. Открыть консоль информация 1) набрать в поиске в браузере -- about: blank или 2) F 12 3) откроется консоль 4) при работе в консоли для переноса стоки без выполнения кода shift - enter

**9**. 1) Инициализация переменной,2) присваивание ей значения3) вывод на печать значения переменной, 4)изменение переменной, 5) вывод на печать нового значения переменной

  let x;

x = 5;

document.writeln(x);

x=x\*8 + 24;

document.write(x);

**10.** **Функция alert()**

Функция alert() выводит на экран окно с сообщением. Текст сообщения всегда пишется в кавычках. Все функции вывода могут выводить не только текст, но и значения переменных. Пример:

|  |  |
| --- | --- |
| 9 10 11 | ***let*** x = 10;  alert('Текст');  alert(x); |

**11.** **Метод document.write()**

Метод document.write() добавляет текст на страницу. Это может быть не просто текст, но и HTML тэги. Они будут работать. Однако, не рекомендуется использовать этот метод для добавления элементов страницы. Они могут отображаться неправильно. Для создания элементов применяются другие методы, которые рассмотрены в этом учебнике. А document.write() используется, в основном, для отслеживания работы программы. Добавить к предыдущему примеру №10

Пример:

|  |  |
| --- | --- |
| 12 13 | document.write('Текст на странице ');  document.write(x); |

**12. Метод console.log()**

Консоль браузера - это полезный инструмент отладки страницы, в том числе, скриптов. Он позволяет получить важную информацию о работе скрипта. Удобнее всего консоль сделана в браузере FireFox

Добавить к предыдущему примеру №10. Посмотреть в консоли как выполняется

console.log('Текст выведенный в консоль');

console.log(x);

13. **Проверка типа**

В JavaScript есть функция, которая определяет тип данных. В качестве аргумента функции можно указать переменную или просто данные. typeof(данные). Функция возвращает строку, которая содержит название типа данных. Для примера определим тип переменной, которую мы создавали ранее. Также выведем тип данных, которые укажем в функции.

|  |  |
| --- | --- |
| 11 12 | alert(typeof(30));  alert(typeof(x)); |

Попробуйте создать переменные разных типов и определить их тип.

Добавить к предыдущему примеру №10 или создать новый документ

**14.** Присваивание

|  |  |
| --- | --- |
| 9 10 | ***let*** x;  x=2; |
| 11 | ***let*** y = x \*\* 3 + 4\*x; | |
| 12 | x = x + 7; | |

Что будет в результате? Как записывается в алгебре выражение №11

Чему равно

|  |  |
| --- | --- |
| 12 | x += 7; |

Проверить

15. Найти чему будет равен остаток от деления 20 на 3. Вывести результат тремя известными способами.

|  |  |
| --- | --- |
| 11 12 | ***let*** n = 20%3;  alert(n); |

16. Посчитать

|  |  |
| --- | --- |
| 13 | ***let*** a = 18 \* (x + 7); |

( let x=2;

    a=18 \* (x + 7);

    document.write(a))

17. Что это? Что делает? Что происходит?

(**Инкремент** ++ увеличивает переменную на 1: let counter = 2; counter++; // работает как counter = counter + 1. Декремент уменьшает значение переменной на 1.

Эти операторы выглядят так:

Инкремент: ++

Декремент: --

Их можно писать как до переменной, так и после переменной )

Префиксная форма возвращает новое значение, в то время как постфиксная форма возвращает старое (до увеличения/уменьшения числа).

Чтобы увидеть разницу, вот небольшой пример:

let counter = 1;

let a = ++counter; // (\*)

alert(a); // 2

В строке (\*) *префиксная* форма ++counter увеличивает counter и возвращает новое значение 2. Так что alert покажет 2.

Теперь посмотрим на постфиксную форму:

let counter = 1;

let a = counter++; // (\*) меняем ++counter на counter++

alert(a); // 1

В строке (\*) *постфиксная* форма counter++ также увеличивает counter, но возвращает *старое* значение (которое было до увеличения). Так что alert покажет 1.

Подведём итоги:

* Если результат оператора не используется, а нужно только увеличить/уменьшить переменную, тогда без разницы, какую форму использовать:
* let counter = 0;
* counter++;
* ++counter;

alert( counter ); // 2, обе строки сделали одно и то же

* Если хочется тут же использовать результат, то нужна префиксная форма:
* let counter = 0;

alert( ++counter ); // 1

* Если нужно увеличить и при этом получить значение переменной *до увеличения* – нужна постфиксная форма:
* let counter = 0;

alert( counter++ ); // 0

**Инкремент/декремент можно использовать в любых выражениях**

Операторы ++/-- могут также использоваться внутри выражений. Их приоритет выше, чем у большинства других арифметических операций.

Например:

let counter = 1;

alert( 2 \* ++counter ); // 4

Сравните с:

let counter = 1;

alert( 2 \* counter++ ); // 2, потому что counter++ возвращает "старое" значение

ПРИМЕР

         x = 7;

       let d=x++;

document.writeln(x);

document.writeln(d);

x = 7;

       let f=++x;

document.writeln(x);

document.write(f);

x = 7;

x++;

document.write (x);

18. Что это? Что делает? Что происходит? Надо ли корректировать , как?

(Оператор "+" может применяться к строкам. Он объединяет строки в одну строку. Как и у любого оператора, операндами у него могут быть переменные и строки внутри кода программы. Напомню, что строки всегда пишутся в кавычках. )

let m = 'меню';

let srt = 'главное ' + m;

let an = 'лиса ' + 'и ' + 'медведь';

document.write(srt);

document.write(an);

document.write(m)

19. Что это? Что делает?

(У этого оператора есть ещё одно применение: с одной переменной. Если переменная содержит строку, состоящую из цифр, то оператор (+) преобразует значение переменной в число. И переменную можно использовать в арифметических расчётах.)

**(!!!! ‘17’ строковое значение)**

let n = '17';

+n;

document.write(typeof(n));

document.write(typeof(+n));

let f = n \* 5;

document.write(f);

20. (В JavaScript операторы сравнения возвращают значения логического типа. Если значения, которые сравниваются, соответствуют оператору, то он возвращает true, а если не соответствуют, то он возвращает false. Именно это значение в дальнейшем используется программой. Если оператор сравнения вернул true, выполняются одни действия, а если вернул false, то другие)

***let*** x = 5;

alert(x > 2);

alert(x == 14);

21. Объяснение

(Иногда нужно провести не одно сравнение, а сразу несколько. В этом случае для связи сравнений в JavaScript используются логические операторы. Они объединяют два сравнения и возвращают один результат.

&& - логическое "и". Возвращает true только если оба сравнения верны. Если хотя бы одно из сравнений неверно, то возвращает false:)

|  |  |
| --- | --- |
| 12 13 | alert(7>2 && 5<16);  alert(5>1 && 10>18) |

Строка 12 выведет true, потому что оба сравнения верны. Строка 13 выведет false, потому что одно из сравнений не верно.

22. || - логическое "или". Возвращает true, если хотя бы одно сравнение верно. Возвращает false, только если оба сравнения неверны:

|  |  |
| --- | --- |
| 14 15 | alert(5>30 || 1<8);  alert(2>15 || 7>8072); |

Строка 14 вернёт true, даже если учитывать, что одно из сравнений неверно. Потому что второе сравнение верно и этого достаточно. Строка 15 вернёт false, потому что оба сравнения неверны.

23. 23-26 Выполнить в одной работе и задать х один раз в начале программы

Условие в JavaScript задаётся с помощью оператора if. Оно указывается в скобках. Для указания условий [используются операторы сравнения](https://basecourse.ru/javascript/compare.php), которые рассматривались в прошлой теме. После условия в фигурных скобках идут строки, которые выполняются, только если условие соблюдается.

|  |  |
| --- | --- |
| 9 10 11 12 13 14 15 | document.write('Начало программы ');  ***let*** x = 8;  ***if*** (x < 10)  {  document.write('Строка выполнится, если x < 10 ');  }  document.write('Продолжение '); |

Запустите страницу и Вы увидите, что на экран вывелись строки "Начало программы" и "Продолжение", а между ними текст из оператора условия. Теперь сделаем так, чтобы условие не выполнялось. Для этого изменим значение переменной x:

|  |  |
| --- | --- |
| 10 | ***let*** x = 30; |

Теперь тест из оператора условия не выводится,значит, строки внутри фигурных скобок не выполняются.

24. Задать х и проверить условие.

В фигурных скобках оператора условия может находиться очень много строк. Это может быть основная часть программы. При этом, если существуют переменные, которые используются не только внутри оператора, но и в дальнейшем, то объявлять эти переменные нужно до оператора.

Если условие состоит из нескольких сравнений, то для этого используются [логические операторы](https://basecourse.ru/javascript/compare.php#logoperator).

***if*** (x<100 && x>2)

{

document.write('Строка выполнится, если оба сравнения верны ');

}

Что значит &&

25. **Конструкция if else** позволяет написать строки, которые будут работать, если условие не выполняется. После фигурных скобок оператора if пишется cлово else, за ним ставятся фигурные скобки и в них содержится код, который будет выполняеться, если условие не соблюдается:

|  |  |
| --- | --- |
| 25 26 27 28 29 30 31 32 33 | document.write('Новая часть ');  ***if*** (x==7)  {  document.write('Выполнится, если сравнение верно ');  }  ***else***  {  document.write('Выполнится, если сравнение не верно ');  } |

26. **Оператор условия "?"**

Существует ещё один оператор условия. Он выглядит как знак вопроса. Его можно применять в тех случаях, когда в зависимости от условий, какой-то переменной нужно присвоить либо одно значение, либо другое.

***let*** ans = (x < 7) ? 'Условие выполняется' : 'Условие не выполняется';

document.write(ans);

Что значит знак вопроса, объяснение (Этот оператор работает так: проверяет условие в скобках, если оно выполняется, то оператор возвращает значение до двоеточия. А если оно не выполняется, то оператор возвращает значение после двоеточия. В качестве возвращаемых значений могут выступать не только данные, записанные в коде скрипта, но и значения переменных и результат работы функций или операторов.)

27.

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 | <!DOCTYPE html>  <html>  <head>  <title>Страница</title>  <meta charset="utf-8">  <style>  div  {  border: 1px solid Red;  }  </style>  </head>  <body>  <div id="block">Текст</div>  <div id="newblock">Другой блок</div>  <script>  let block = document.getElementById('block');  block.style.padding = '10px';  </script>  </body>  </html> |

1.let a=4; let P=4\*a;console.log(P)

2. let a=4; let S=a\*a;console.log(S). let S=a\*\*2;console.log(S) let S=Math.pow(a,2);console.log(S)

3. let L=399; let M=L/100;   let result=Math.trunc(M); document.write(result)

4,  <script>  let a=-17;  if (a>0) {document.write("положительное")  } else{document.write("отрицательное") }  </script>

5. let a=4, b=3 в консоли вывести a> 2 && b<=**3 shift + enter**

 <script>

        let a=4, b=3;

        console.log(a> 2 && b<=3);

        document.write(a> 2 && b<=3);

        alert(a> 2 && b<=3)

    </script>

6. let n=53;

    if (n>0) {

        n++

        document.write(n)

    } else {document.write(n)}

// BEGIN

const mySubstr = (string, length) => {

let resultString = '';

let i = 0;

while (i < length) {

resultString += string[i];

i += 1;

}

return resultString;

};

// END

5- 18

 <p>

     Задача 5

    </p>

    <p>

    <script>

        let a=4, b=3;

        console.log(a> 2 && b<=3);

        document.write(a> 2 && b<=3);

        alert(a> 2 && b<=3)

    </script></p>

    <p>

        Задача 6

    </p>

    <p>Дано целое число. Если оно является положительным, то прибавить к нему 1, в противном случае не  изменять его. Вывести полученное число.</p>

    <script>

        let n=53;

    if (n>0) {

        n++

        document.write(n)

    } else {document.write(n)}

    </script>

    <p>

        Задача 9

    </p>

    <script>

        let x;

x = 5;

document.writeln(x);

x=x\*8 + 24;

document.write(x);

    </script>

    <P>Задача 10</P>

        <script>

let e=10;

alert('Текст');

alert(e);

        </script>

    </p>

    <p>Задача 16</p>

    <p>

<P>

    <script>

    let y=2;

    a=18\*(y + 7);

    document.write(a)

    </script>

    <p>

        Задача17

    </p>

    <script>

         x = 7;

       let d=x++;

document.writeln(x);

document.writeln(d);

x = 7;

       let f=++x;

document.writeln(x);

document.write(f);

    </script>

    <p>Задача 18</p>

        <script>

            let m = 'меню';

let srt = 'главное ' + m;

let an = 'лиса ' + 'и ' + 'медведь';

document.write(srt);

document.write(an);

document.write(m)

        </script>

**10**. <body>

    <P>Задача 10</P>

    <script>

let b = 10;

alert('Текст');

alert(b);

    </script>

</p>

</body>

**15.**

|  |  |
| --- | --- |
| **11 12** | ***let* n = 20%3;**  **alert(n);** |

23-27…… <script>

        document.write('Начало программы ');

let x = 4;

if (x < 10)

  {

  document.write('Строка выполнится, если x < 10 ');

  }

document.write('Продолжение ');

</script>

<p>

    <p>Задача 24</p>

    <script>

        if (x<100 && x>2)

  {

  document.write('Строка выполнится, если оба сравнения верны ');

  }

    </script>

</p>

<p>

        Задача 25

    </p>

    <script>

document.write('Новая часть ');

if (x==7)

  {

  document.write('Выполнится, если сравнение верно ');

  }

else

  {

  document.write('Выполнится, если сравнение не верно ');

  }

    </script>

    <p>

      Задача26

    </p>

    <script>

      let ans = (x < 7) ? 'Условие выполняется' : 'Условие не выполняется';

document.write(ans);

    </script>