**Тема №2. Функции, контекст выполнения.**

Вторая тема посвящена функциям и контексту выполнению (ключевое слово *this*), так же немного затрагивается ООП.

Что нужно знать:

* Чем являются функции в JS? Области применения функций.
* Контекст функции, когда определяется, от чего зависит. Привести примеры, когда функция теряет контекст.
* Вызов функции как конструктора, особенности, связь с прототипами.
* Методы *bind*, *call* и *apply*.
* Стрелочные функции, отличия от обычных.
* Замыкания, лексическая зона видимости.
* Что почитать:
* [https://google.ru](https://google.ru/)
* [https://learn.javascript.ru](https://learn.javascript.ru/), пункты 2.14-2.16 (дополнительно пункт 6)
* Книга “You don’t know JS. Замыкания и объекты”, К. Симпсон. Главы 6-7, стр. 133-176. (pdf лежит в Dropbox/QPD/Книги/JavaScript)
* Книга “You don’t know JS. ES6 & не только”, К. Симпсон. Глава 2, стр. 85-93. (pdf лежит в Dropbox/QPD/Книги/JS, React)
* Задания для темы:
* Реализовать функцию Calculator, с помощью которой можно производить вычисления.
* Функция должна вызываться только как конструктор, при вызове без ключевого слова *new* должно выбрасываться исключение. В аргументе функция принимает начальное значение для вычисления.
* После вызова функция возвращает объект с 4 методами: *plus*, *minus*, *multiply*, *divide*. Каждый из них принимает число для совершения операции и возвращает этот же объект (чтобы было можно строить цепочки вызовов).
* Также объект имеет пятый метод *calculate*, который возвращает результат вычисления всей цепочки.
* Результат цепочки должен высчитываться с учетом приоритетов операторов, то есть *2 + 2 \* 2*, записанное методами *plus* и т.д. должно вернуть *6*, а не *8*.
* Пример работы готовой программы:
* 