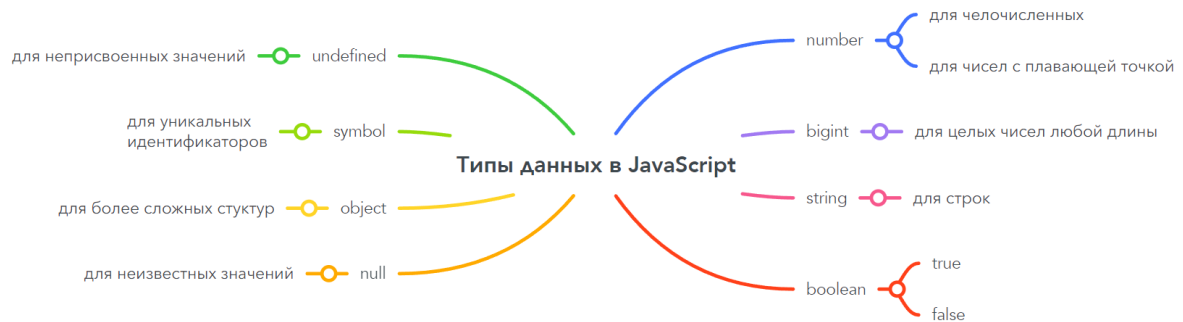


ВОПРОСЫ.

1. Основы JavaScript:

- типы данных в JS;



- разница между let, const и var;

var: ограничивается только глобальной областью видимости и областью видимости функции (блоками не ограничивается – for/if); к ней можно обращаться до объявления, так как интерпритатор JS выполняет «Поднятие» и её значение равно undefined.

let: он уже может ограничиваться областью видимости блока кода и нельзя обращаться к переменной до её объявления.

const: тоже самое, что и let, но объекты, созданные с помощью него, не могут перезаписываться.

- что такое hoisting?

Поднятие (hoisting) – процесс, выполняющийся компилятором для объектов, объявленных с помощью var. Он условно ставит объявление переменных в самое начало кода, хоть фактически это и не происходит.

2. Функции:

- Как объявить функцию в JavaScript?

Чтобы объявить функцию нужно написать ключевое слово «function», затем имя функции, в круглых скобках параметры функции (либо пустые круглые скобки), а в конце фигурные скобки – тело функции (её определение). Если хотите, чтобы функция возвращала нужное вам значение, то используйте слово «return».

- В чем разница между function declaration и function expression?

function declaration – классическая инициализация функции. function expression – инициализация функции как переменной.

Разница между ними в том, что function declaration компилятор ищет в первую очередь и компилирует сразу, а function expression как обычную переменную, то есть нельзя обратиться до объявления.

- что такое стрелочные функции?

Это функции, у которых нет имени, но если нужно его задать, то необходимо присвоить их каким-либо переменным. Объявление таких функций происходит сразу с параметров (круглых скобок), затем стрелочка («=>» как раз поэтому и назвали стрелочной функций) и потом тело функции. Фишка в коротком синтаксисе.

3. Объекты и массивы:

- Как создать объект в JavaScript?

Есть два варианта: с помощью литеральной нотации (название объекта = { }, а потом добавлять ему всякое), либо с помощью конструктора Object() (название объекта = new Object()).

- Как добавить новый элемент в конец массива?

Для добавление элемента в конец массива используется функция push().

- Как перебрать все элементы массива?

С помощью метода forEach(), цикла for(), цикла for ... in(), цикла for of () и обычного итератора.

4. Обработка событий:

- Как назначить обработчик события элементу?

Есть два варианта: либо указать в свойстве объекта разметки название обработчика события (функция), либо добавить объекту разметки идентификатор и уже в коде его событию присвоить функцию.

- Что такое event delegation?

Делегирование события (event delegation) – это назначение обработчика события объекту разметки, внутри которого имеется множество дочерних элементов, тогда при нажатии на них будет также срабатывать этот обработчик.

- Как отменить стандартное поведение события?

Есть два варианта: первый – возвращать false при нужном событии; второй – вызвать у события event.preventDefault().

5. Асинхронность:

- Как работает синхронный и асинхронный код в JavaScript?

Синхронное выполнение кода – код выполняется в строго указанном порядке (строка за строкой), а при асинхронном выполнении кода – порядок может нарушаться (например первая строка выполняется и начинает выполняться вторая строка без ожидания завершения первой). Реализовано же асинхронное программирование за счёт промисов. Функции, которые объявлены асинхронными возвращают эти промисы. Раньше реализовывали такое за счёт WebAPI браузера через `setTimeout()`;

- Что такое Promise?

Промис – это средство реализации асинхронного программирования в JS. Он является объектом, который содержит своё состояние. На него навешивают обратные вызовы для двух состояний: выполнен успешно и выполнен с ошибкой.

- Как использовать `async/await`?

`async` используется при объявлении функции и значит, что она будет возвращать промис. `await` используется при вызове функции, заставляет подождать её выполнение, а возвращает то, что асинхронная функция отдаёт при завершении.

ЗАДАНИЯ.

Они находятся в моём репозитории GitHub.

<https://github.com/TheSanKris/JavaScript>