1. Переменные

"use strict";//директива которая указывает скриптовому файлу,

//что мы работаем в современном режиме, !использовать всегда!

//прописывать в начале скриптового файла

let number = 5; // = - присваивание //let - переменная изменяемая

const leftBorderWidth = 1; // цифра не может быть первой в названиии

//переменной,нельзя использовать зарезирвированные слова.

number = 10;

console.log(number);

const obj = { //прямых констант в js небывает. что такое прямая константа

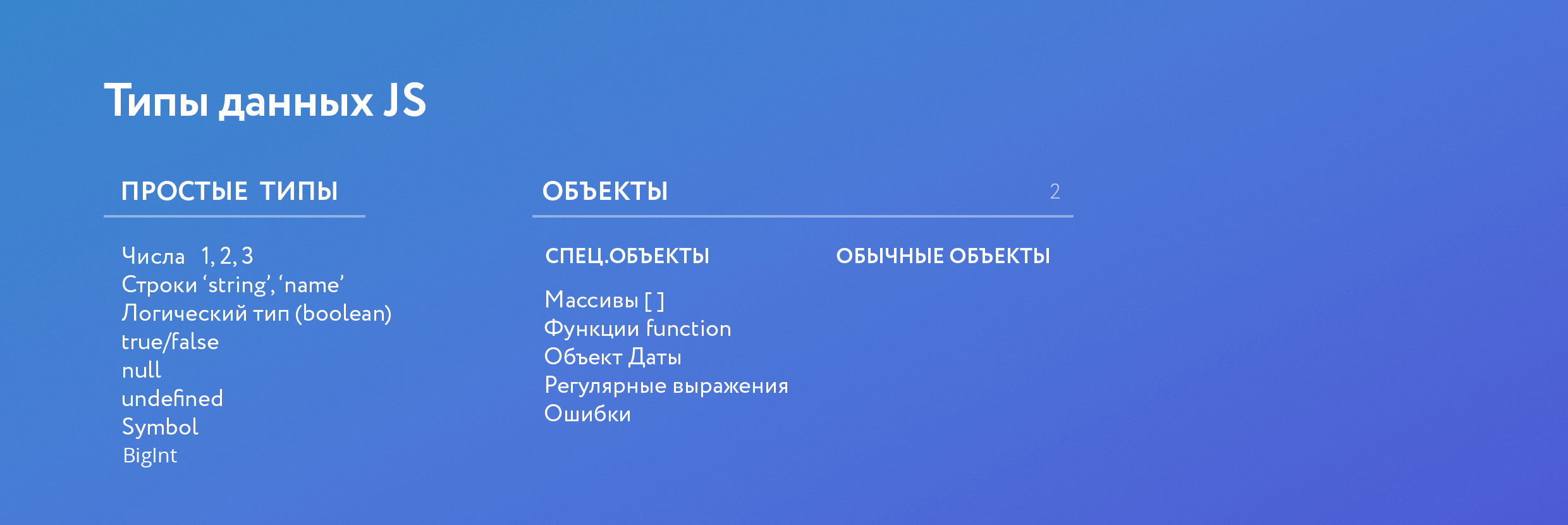
    a:50

};

obj.a = 10;

console.log(obj);

1. Типы данных



Undefined – что-то существует, но этого нет (стоит холодильник, но он пустой )

BigInt – работа с большими числами. (ограничение с обычными числами. Мы не можем задать число, которое больше чем 2 в 53 степени)

"use strict";

//тип данных - число

let number = 4.6; //числа, даже дробные

console.log(4/0);//infinity

console.log('string'\*9);//nan - не число

//типданных - строка

const persone = "5";

//логичесий тип данных (boolean)

const bool = true;

//тип анных null И undefined

console.log(something);

let und;

console.log(und);

//тип данных - объект

const obj = {// это объект

    name: "john",// свойства объекта

    age: 25,// свойства объекта

    isMarried: false// свойства объекта

};

console.log(obj.name); // обратились к значению свойства name  объекта

//массив это частный случай объекта, о не является типом данных

console.log(obj["name"]);//конструкция в квадратных скобках

//используется для хранения данных идущих по порядку

let arr = ['plum.png', 'orange.jpg', 'apple.bmp', 6, {}, []];

console.log(arr[1]);// в программировании исчисление начинается с 0

//[Running] node "e:\3.Projects JS\first\tempCodeRunnerFile.js"

//orange.jpg

1. Простое общение с пользователем

"use strict";

//alert('hello');

//const result = confirm('are you here?');

//console.log(result);

//происходит запись результата

//const answer = prompt("Вам есть 18?", "17");

//console.log(answer); // ответ записывает от пользователя, тип данных строка

//вся информация от пользователя птип данных является строка,

//если не преобразован тип данных программистом

//const answer = +prompt("Вам есть 18?", "18");

//console.log(typeof(answer)); в этом случаи будет number тип данных

const answers = [];

//answers[0]=prompt ('как ваше имя', '');

//answers[1]=prompt ('ваша ффамилия', '');

//answers[2]=prompt ('сколько вам лет', '');

//document.write(answers);

console.log(typeof(answers));

console.log(typeof(null));//ошибка, но такая вот особенность

4.Интерполяция

Это значит, что мы можем вставлять внутри строки значение переменной.

const category = 'toys';

//console.log('https://someurl.com/' + category); // конкоцинация (https://someurl.com/toys)

console.log(`https://someurl.com/${category}/5`);//не забывать про апострофы

const user = 'Ivan';

console.log(`Privet, ${user}`);

5.Операторы

console.log('arr' + '- object'); // конкоцинация, строки сложатся и правратятся

//в большую строку -  arr- object;

console.log(4 + '- object'); // тоже самое

console.log(4 + +'5');// 9, унарный плюс +"5", использует один аргумент

//икримент и дикримент

let incr = 10,

    decr = 10;

    incr++;//увеличение на единицу(постфикснаяя форма записи)

    decr--;//уменьшение знач на ед.(постфикснаяя форма записи)

   // ++incr;//(префиксная форма записи)

// --decr;//(префиксная форма записи)

    console.log(incr); //11 тоже самое будет и при префиксной форме

    console.log(decr); // 9

    console.log(incr++); //10

    console.log(decr--); //10

    /\*в постфиксной фоме возвращается старое значение,

     а после произвоит действие\*/

     console.log(5%2); // 1 - выводит остаток

     console.log(2\*4 == 8); //true сравнение происходит по значению, а не по типу данных

     // если сравнивать по типу данных то

     console.log(2\*4 === '8'); // false, строгое сравнение, сравнивает и по тиру данных

     /\* && - оператор и - работает когда два либо более значения являются правдивыми

        || - оператор или - работает когда хотябы одно значение является правдивым

      \*/

    const isChecked = true;

          isClose = false;

          console.log(isChecked || isClose); // выдает true

          const isChecked = true;

          isClose = false;

          console.log(isChecked && isClose); // выдает false, true выдаст, когда оба значения будут true

          // ! - оператор отрицания

           console.log(2 + 2 \* 2 === 8);//false

           console.log(2 + 2 \* 2 != 8); // true

           console.log(2 + 2 \* 2 !== '6');//true