

ЗАДАНИЕ НА ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 1

Синтаксис языка TypeScript. Стрелочные функции. Встроенные и особые типы. Формат JSON.

Задание № 1 включает в себя 4 раздела:

- 1) Установка Node.js, создание проекта (1 вариант)
 - 2) Реализация стрелочных функций
 - 3) Использование встроенных и специальных типов при реализации объектов
 - 4) Преобразование объектов к формату JSON.
-

- 1) Перейдите на сайт [Node.js](https://nodejs.org/). Выполните установку программы версии, рекомендованной для большинства пользователей.
- 2) Реализуйте стрелочную функцию и выполните её вызов для подтверждения корректности логики её работы
 - Вариант №1.** Реализуйте стрелочную функцию, возвращающую строку, составленную (см. конкатенация) из 2х чисел, передаваемых в качестве параметров.
 - Вариант №2.** Реализуйте стрелочную функцию, возвращающую строку, составленную из 3х строк (см. конкатенация), передаваемых в качестве параметров.
 - Вариант №3.** Реализуйте стрелочную функцию, возвращающую сумму 3х чисел, передаваемых в качестве параметров.
 - Вариант №4.** Реализуйте стрелочную функцию, возвращающую произведение 2х чисел, передаваемых в качестве параметров.
 - Вариант №5.** Реализуйте стрелочную функцию, возвращающую разность 2х чисел, передаваемых в качестве параметров.
 - Вариант №6.** Реализуйте стрелочную функцию, возвращающую остаток от деления 2х чисел, передаваемых в качестве параметров
- 3) Выполните создание объектов – констант (const) и переменных (let) с указанием различных встроенных и специальных типов. Название переменной может быть любым. Количество объектов не более 8.
- 4) Скопируйте и выполните преобразование объекта data к формату JSON в виде строки. Результат преобразования следует вывести в консоль.

Вариант № 1	<pre> interface Entity { id: number; } interface ToJsonStringify extends Entity { name: string; surname?: string; } const data: ToJsonStringify = { id: 1, name: "Василий", } </pre>
Вариант № 2	<pre> interface Entity { id: number; } interface ToJsonStringify extends Entity { e1: string; e2: string; } const data: ToJsonStringify = { id: 2, e1: "Поле 1", e2: "Поле 2", } </pre>
Вариант № 3	<pre> interface Entity { id: number; } interface ToJsonStringify extends Entity { e1?: number; e2: string; } const data: ToJsonStringify = { id: 3, e1: 12345, e2: "Поле 2", } </pre>

Вариант № 4	<pre> interface Entity { id: number; } interface ToJsonStringify extends Entity { e1?: number; e2?: string; } const data: ToJsonStringify = { id: 4, e1: null, e2: null, } </pre>
Вариант № 5	<pre> interface Entity { id: number; } interface ToJsonStringify extends Entity { e1: number; e2: number; } const data: ToJsonStringify = { id: 5, e1: 12345, e2: 999, } </pre>
Вариант № 6	<pre> interface Entity { id: number; } interface ToJsonStringify extends Entity { e1?: number; e2?: number; } const data: ToJsonStringify = { id: 6, e1: null, e2: 28, } </pre>

	}
--	---