**Переменные и типы данных в JavaScript**

**2.1. Что такое переменные?**

**Переменная** — это "контейнер" для хранения данных.  
Ты можешь сохранить в переменной любое значение (число, текст, объект, функцию), а потом использовать его снова и снова.

📦 Пример:

js

КопироватьРедактировать

let name = "Гриша";

let age = 25;

**2.2. Как объявлять переменные**

В JS есть три способа:

js

var x = 10; // устаревший

let y = 20; // современный

const z = 30; // постоянная переменная

**📌 let — основная переменная, которую можно менять**

js

КопироватьРедактировать

let counter = 1;

counter = counter + 1;

**📌 const — переменная, которую нельзя переопределить**

js

const pi = 3.14;

**⚠️ var — устарел. Не используй без крайней необходимости.**

**2.3. Имена переменных: как правильно называть**

* Только латиница, цифры, символы $ и \_
* **Нельзя** начинать с цифры
* **Нельзя** использовать зарезервированные слова (let, if, return, ...)

✅ Хорошо: userName, totalPrice, max\_score  
❌ Плохо: 1user, let, имя

**2.4. Типы данных в JavaScript**

**📊 Примитивы:**

| **Тип** | **Пример** |
| --- | --- |
| number | 42, 3.14, -10 |
| string | "текст", 'слово' |
| boolean | true, false |
| undefined | значение не задано |
| null | "пустое" значение |
| bigint | большие числа (n) |
| symbol | уникальные идентификаторы |

**2.5. Оператор typeof**

Проверяет тип значения:

js

typeof "Гриша" // "string"

typeof 42 // "number"

typeof true // "boolean"

typeof null // "object" (! особенность языка)

typeof undefined // "undefined"

**2.6. Работа со строками**

js

let name = "Гриша";

let message = "Привет, " + name + "!"; // конкатенация

let welcome = `Добро пожаловать, ${name}!`; // шаблонная строка (``)

**2.7. Арифметические операции**

js

let a = 10;

let b = 5;

let sum = a + b; // 15

let diff = a - b; // 5

let product = a \* b; // 50

let div = a / b; // 2

let mod = a % 3; // остаток от деления

**2.8. Присваивание и изменение**

js

let x = 10;

x += 5; // 15

x \*= 2; // 30

**2.9. Логические значения и сравнение**

js

true, false

let isAdult = age >= 18;

console.log(10 > 5); // true

console.log(10 == "10"); // true (сравнение значений)

console.log(10 === "10");// false (сравнение по типу и значению)

**2.10. null и undefined — в чём разница?**

* undefined: переменная объявлена, но не имеет значения
* null: переменная намеренно "очищена"

js

КопироватьРедактировать

let a;

console.log(a); // undefined

let b = null;

console.log(b); // null

**2.11. Константы и неизменяемость**

js

const name = "Гриша";

name = "Аня"; // ❌ ошибка: нельзя переопределить const

Но: **объекты и массивы** внутри const можно изменять!  
Только нельзя переприсваивать саму ссылку.

**2.12. Глоссарий**

| **Термин** | **Значение** |
| --- | --- |
| Переменная | Контейнер для хранения данных |
| Константа | Переменная, которую нельзя изменить |
| Примитив | Простой тип данных |
| typeof | Оператор определения типа |
| null | Специальное значение "ничего" |
| undefined | Значение переменной по умолчанию |
| Арифметика | Матем. операции: +, -, \*, / |
| Конкатенация | Склеивание строк |

**2.13. Вопросы для самопроверки**

1. В чём разница между let, const и var?
2. Что произойдёт при попытке изменить const?
3. Что вернёт typeof null? Почему?
4. Какое значение по умолчанию у необъявленной переменной?
5. В чём разница между == и ===?