

Поради щодо іменування змінних у JavaScript та PHP

JavaScript

1. Використовуй camelCase

Це стандарт для змінних, функцій, аргументів:

```
let userName = 'Andriy';  
function getUserAge() {}
```

⚠ Не використовуй **snake_case**, **PascalCase** чи **UPPER_CASE** для звичайних змінних — це стилістична плутанина.

2. Називай змінну по суті її значення

Погано:

```
let x = 25;
```

Добре:

```
let userAge = 25;
```

Ім'я змінної має відповідати тому, **що вона зберігає**, а не тому, як її легше набрати на клавіатурі.

3. Будь конкретним, але не багатослівним

Погано:

```
let d; // якась дата? яка?
```

Добре:

```
let userRegistrationDate;
```

Ще краще, якщо контекст очевидний:

```
let registrationDate;
```

4. Булеві змінні — як запитання

```
let isLoggedIn = true;  
let hasPermission = false;  
let canEdit = true;
```

Так ясно, що це список, а не один об'єкт.

5. Не починай з _ чи \$, якщо це не спеціальний випадок

```
let $el = document.querySelector('#btn'); // ок, якщо це елемент  
let _internalVar = 5; // ок у класах, як приватна змінна
```

Але не зловживай. \$ — ок тільки для DOM-елементів чи jQuery.

6. Уникай аббревіатур і скорочень, якщо вони не очевидні

Погано:

```
let prsnNm = 'Oleh';
```

Добре:

```
let personName = 'Oleh';
```

Методика: "Називай так, ніби ти повернешся до цього коду за 6 місяців"

Ідеальний код читається як історія:

```
let commentList = getComments();  
let hasNewComments = commentList.length > 0;
```

Уникай розмитих назв типу **data, info, temp, thing, stuff**. Це все сміття.

Таблиця стилів найменування для JavaScript

Стиль	Приклад	Використання	Коментар
camelCase	username	Змінні, параметри та методи	Це стандарт для змінних та методів
PascalCase	UserName	Імена класів, конструкторів	Для класів та конструкторів, але не для змінних
snake_case	user_name	Рідко використовується в JavaScript	Не є стандартом у JavaScript, але може бути в проєктах, де використовуються бази даних чи сервери.
UPPER_CASE	MAX_USERS	Константи, глобальні налаштування	Для констант (особливо глобальних). Все велике з підкресленням.
kebab-case	form-validator.js	Імена файлів, URL	Не використовується для змінних у JS, але часто використовується для імен файлів чи маршрутів.


Основні моменти:

- 1. **camelCase** — стиль для змінних, функцій та методів. У JavaScript це стандарт.
- 2. **PascalCase** — використовується для назв класів та конструкторів.
- 3. **UPPER_CASE** — для **констант**. Кожен символ великої літери і підкреслення.
- 4. **snake_case** — у JavaScript рідко використовується, але може бути у випадку інтеграції з іншими технологіями або базами даних.
- 5. **kebab-case** — не застосовується для змінних, але корисний для імен файлів або маршрутів.

PHP

1. camelCase — стандарт для змінних і методів

```
$userName = 'Andriy';
$registrationDate = '2024-05-01';
```

 Якщо не дотримуватись стилю — код виглядає як різномісна каша.
camelCase — це стандарт у більшості сучасних PHP-проєктів (WordPress, Laravel, Symfony).

2. snake_case — використовується в старих проєктах або для БД

```
$user_name = 'Andriy'; // допустимо, якщо API або БД диктує такий стиль
```

Якщо ти працюєш із WordPress — там часто використовується **snake_case**. Наприклад: `get_post_meta()`, `the_title()`.

3. PascalCase — для класів

```
class UserProfile {}  
class OrderManager {}
```

Назви класів завжди з великої літери й кожне нове слово з великої — **PascalCase**.

4. UPPER_CASE — для констант

```
define('MAX_USERS', 100);  
const SITE_URL = 'https://example.com';
```

Константи — весь текст великими літерами з підкресленням.

5. Ім'я має відповідати суті

Погано:

```
$a = 'Hello';  
$data = '???';
```

Добре:

```
$greetingText = 'Hello';  
$productPrice = 199;
```

6. Булеві змінні — як запитання

```
$isAdmin = true;  
$hasAccess = false;  
$canEdit = true;
```

Як і в JS: якщо можна відповісти "так/ні" — починай із **is**, **has**, **can**, **should**.

7. Масиви — у множині

```
$users = ['Anna', 'Bohdan'];  
$products = ['Phone', 'Laptop'];
```

Так ясно, що це список, а не окремий об'єкт.

8. Не зловживай скороченнями

Погано:

```
$usrnm = 'Andriy';
```

Добре:

```
$userName = 'Andriy';
```

Методика: "Ім'я змінної має бути самодокументуючим"

```
$totalCartPrice = $productPrice + $shippingFee;
```

Якщо ти можеш зрозуміти, що робить код **без коментарів** — значить, імена підібрані вдало.

Таблиця стилів найменування для PHP

Стиль	Приклад	Використання	Коментар
camelCase	\$userName	Змінні, параметри, методи	Найпоширеніший стиль для змінних і методів
snake_case	\$user_name	Змінні, в старих проектах, бази даних	Зазвичай для змінних або таблиць в базах даних
PascalCase	\$UserName	Імена класів, простори імен	Не для змінних. Використовувати для класів та інтерфейсів
UPPER_CASE	MAX_USERS	Константи, глобальні налаштування	Тільки для констант, часто з підкресленням
kebab-case	template-about-us.php	Використовується в URL, імена файлів	Не для змінних у кодї, тільки для веб-ресурсів або файлів

Основні моменти:

1. **camelCase** — найкращий варіант для змінних і методів.
2. **snake_case** — може бути корисним для старих проектів або баз даних.
3. **PascalCase** — використовувати тільки для класів, інтерфейсів, простори імен.
4. **UPPER_CASE** — для **констант**. Все велике, з підкресленнями.
5. **kebab-case** — використовують лише для імен файлів або URL, але не в PHP коді.