

Assignment operators (=, +=, -=, etc.)

Absolutely! Let's dive deep into **Assignment Operators** in JavaScript.



Assignment Operators in JavaScript

Assignment operators assign values to variables and can also perform operations before assigning.



Basic Syntax

```
variable operator value;
```

For example:

```
x += 5; // same as x = x + 5;
```



Common Assignment Operators

Operator	Description	Equivalent To	Example	Result (if <code>x = 10</code>)
<code>=</code>	Assignment	–	<code>x = 10</code>	10
<code>+=</code>	Add and assign	<code>x = x + y</code>	<code>x += 5</code>	15
<code>-=</code>	Subtract and assign	<code>x = x - y</code>	<code>x -= 3</code>	7
<code>*=</code>	Multiply and assign	<code>x = x * y</code>	<code>x *= 2</code>	20
<code>/=</code>	Divide and assign	<code>x = x / y</code>	<code>x /= 2</code>	5
<code>%=</code>	Modulo and assign	<code>x = x % y</code>	<code>x %= 3</code>	1

<code>**=</code>	Exponentiation and assign	<code>x = x ** y</code>	<code>x **= 2</code>	<code>100</code>
<code><<=</code>	Left shift and assign	<code>x = x << y</code>	<code>x <<= 2</code>	<code>40</code> (10 << 2 = 40)
<code>>>=</code>	Right shift and assign	<code>x = x >> y</code>	<code>x >>= 1</code>	<code>5</code> (10 >> 1 = 5)
<code>&=</code>	Bitwise AND and assign	<code>x = x & y</code>	<code>x &= 5</code>	<code>0</code> (10 & 5 = 0)
<code> </code>	<code>=</code>	Bitwise OR and assign	<code>x = x</code>	<code>y</code>
<code>^=</code>	Bitwise XOR and assign	<code>x = x ^ y</code>	<code>x ^= 6</code>	<code>12</code> (10 ^ 6 = 12)

✓ Examples

Basic Assignment (`=`)

```
let name = "Abhi"; // Assigns the string "Abhi" to variable `name`
```

Add and Assign (`+=`)

```
let total = 100;
total += 50; // total becomes 150
```

Subtract and Assign (`-=`)

```
let balance = 80;
balance -= 20; // balance becomes 60
```

Multiply and Assign (`*=`)

```
let score = 10;
score *= 3; // score becomes 30
```

Divide and Assign (`/=`)

```
let amount = 20;  
amount /= 4; // amount becomes 5
```

Modulo and Assign (`%=`)

```
let num = 10;  
num %= 3; // num becomes 1
```

Exponentiation and Assign (`**=`)

```
let power = 2;  
power **= 3; // power becomes 8 (23)
```



Bitwise Assignment Examples (Advanced)

```
let a = 5;    // 0101  
let b = 3;    // 0011  
  
a &= b;       // 0001 → a becomes 1  
a |= b;       // 0111 → a becomes 7  
a ^= b;       // 0100 → a becomes 4
```



Tips

- These operators **modify** the variable in place.
- They're commonly used in **loops**, **accumulators**, and **state updates**.

Let me know if you want a cheat sheet or quiz based on this!