

# Type Conversion



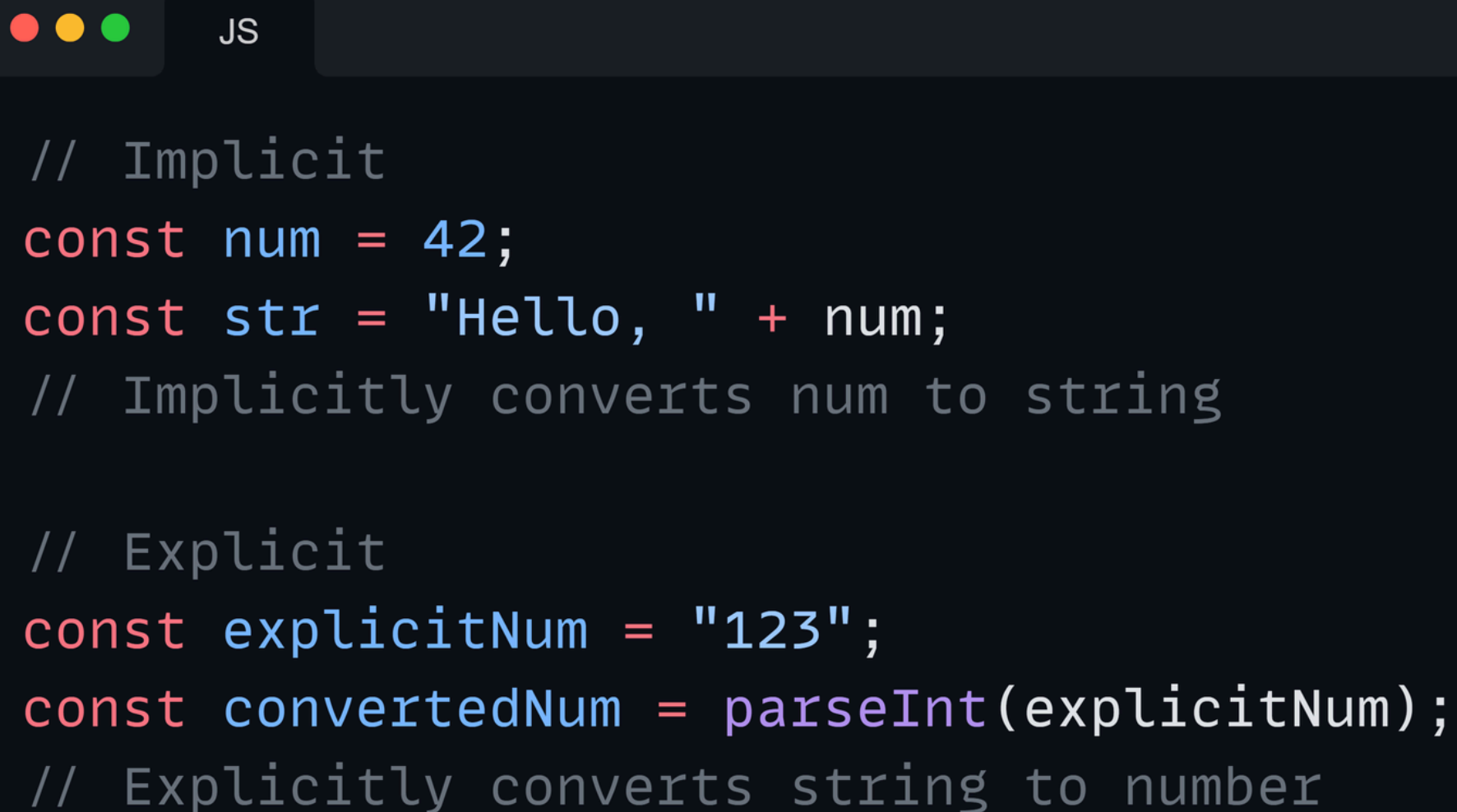
## in JavaScript

[www.scribbler.live](http://www.scribbler.live)



# Implicit vs Explicit

- JavaScript boasts both implicit and explicit type conversion.
- Implicit: Automatic conversion done by JavaScript during operations.
- Explicit: Manually initiated by developers using functions like `parseInt()` or `String()`.



```
// Implicit
const num = 42;
const str = "Hello, " + num;
// Implicitly converts num to string

// Explicit
const explicitNum = "123";
const convertedNum = parseInt(explicitNum);
// Explicitly converts string to number
```

# The Concatenation Effect

- When strings encounter other data types, they tend to pull them into their world through implicit conversion.

```
JS  
  
const age = 25;  
const message = "I am " + age + " years old.";  
// Implicitly converts age to string
```



# parseInt() method

- Use parseInt() to convert a string to an integer, providing control over the parsing process.



JS

```
const numericString = "42";  
const numericValue = parseInt(numericString);  
// Explicitly converts string to number
```

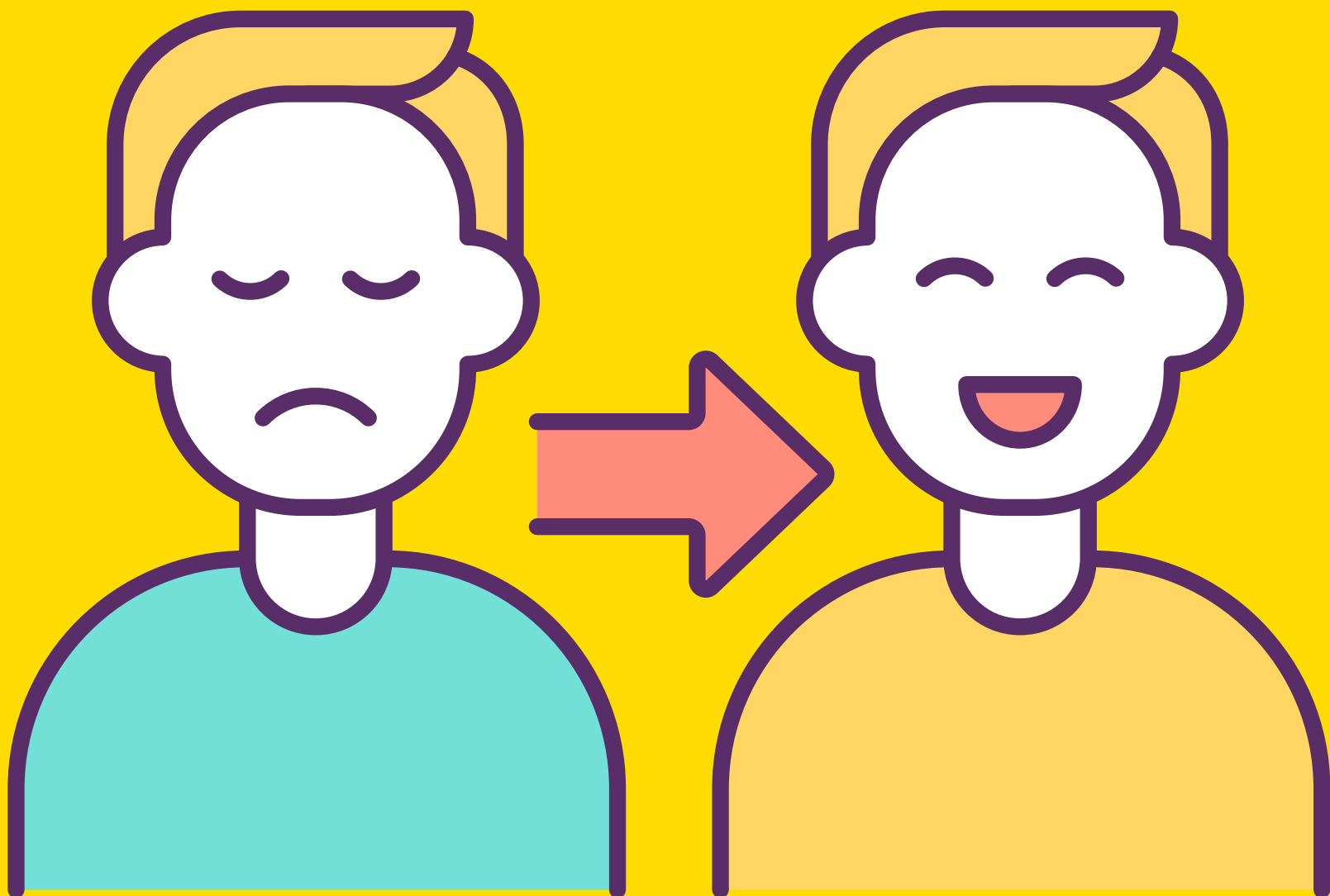




# String()

- Transform any data type into a string using the String() function.

```
JS
const numValue = 42;
const strValue = String(numValue);
// Explicitly converts number to string
```



# === Vs ==

- The strict equality operator (===) compares both value and type.
- Implicit type conversion may occur with the loose equality operator (==), leading to unexpected results.

```
JS

const num1 = 42;
const str1 = "42";

console.log(num1 === str1);
// Outputs: false (strict equality)
console.log(num1 == str1);
// Outputs: true
// (loose equality with implicit conversion)
```



# Keep Exploring Javascript with us!

Share this with a friend who needs it and  
make sure to practice these in scribbler.



## Scribbler.live

Free and Open Interface to  
experiment JavaScript