



Arrow Functions ES6



Arrow Functions

- Traditional function expressions can be verbose and cumbersome, especially when used as callbacks.
- Arrow function's concise syntax makes them perfect for streamlining callback functions

```
JS

// Traditional function expression as a callback
array.map(function(item) {
  return item * 2;
});

// Arrow function simplifies the callback
array.map(item => item * 2);
```

Swipe



Event Handling

- Arrow functions shine in event handling scenarios, where 'this' context can be tricky to manage.
- With arrow functions, you can maintain the lexical scope of 'this' without resorting to workarounds like 'bind()'

```
JS

// Traditional event handler using function expression
button.addEventListener('click', function() {
  console.log('Button clicked');
});

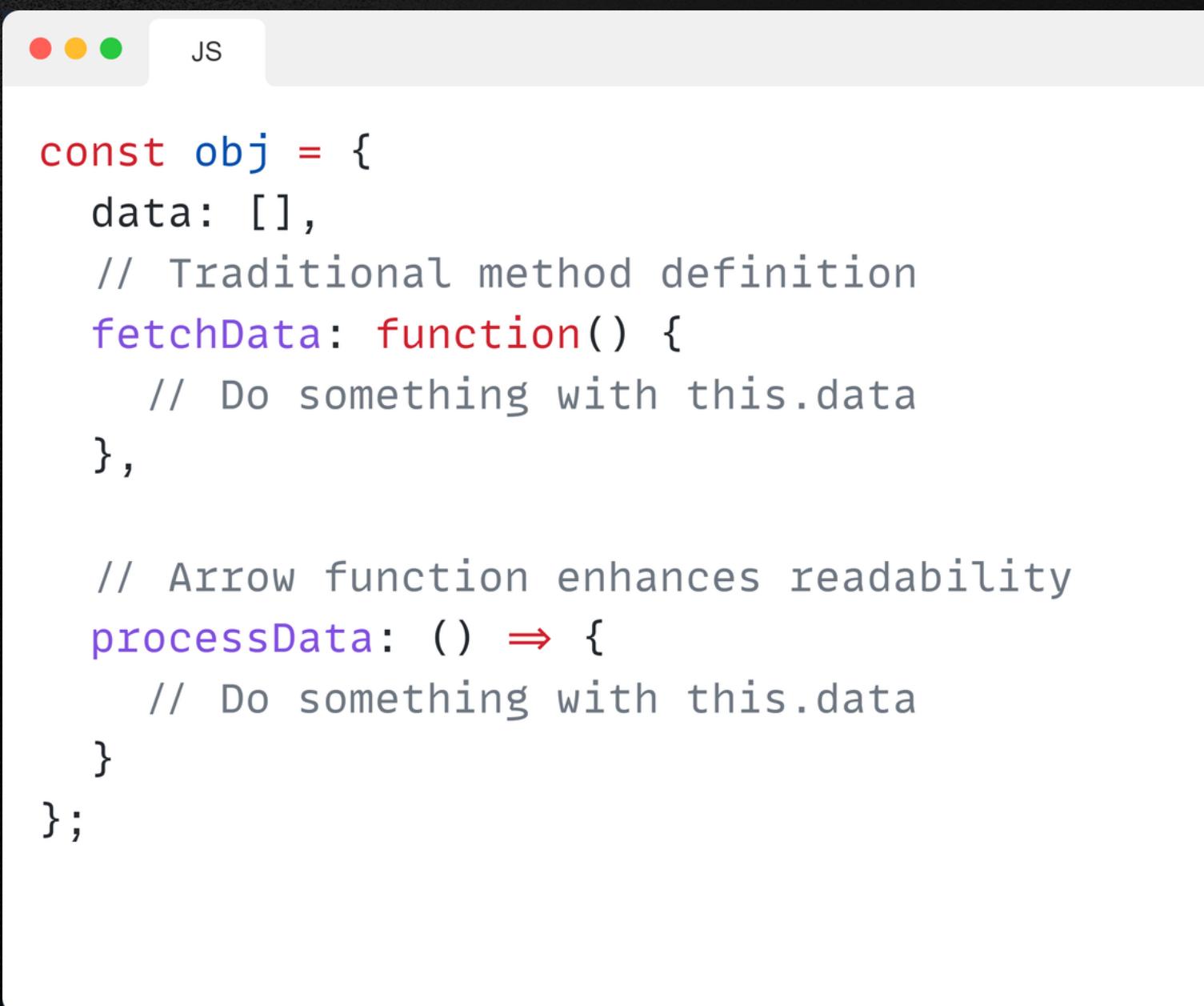
// Arrow function simplifies event handling
button.addEventListener('click', () => {
  console.log('Button clicked');
});
```

Swipe 



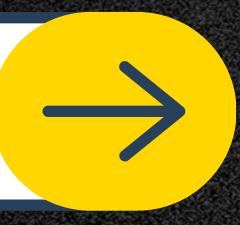
Object Method Definitions

- Defining methods within objects is smoother than ever with arrow functions.



A screenshot of a code editor window titled "JS". The code defines an object "obj" with two methods: "fetchData" and "processData". The "fetchData" method is defined using a traditional function expression, while the "processData" method is defined using an arrow function, which is noted to enhance readability.

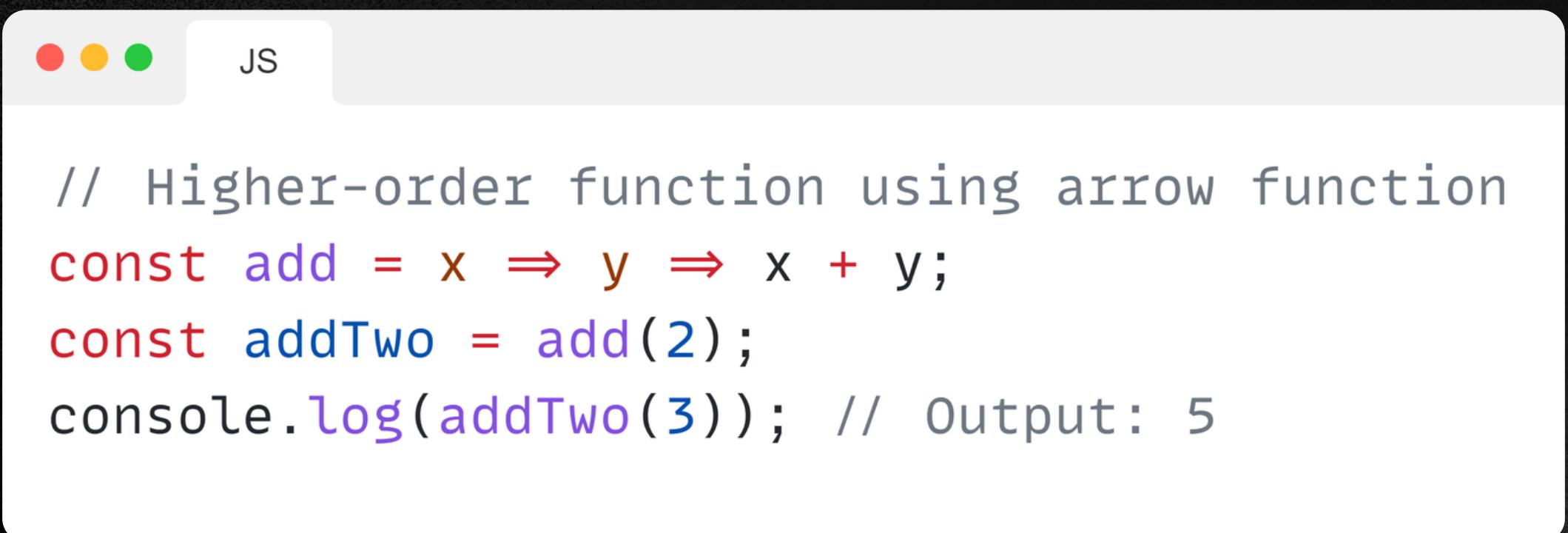
```
const obj = {
  data: [],
  // Traditional method definition
  fetchData: function() {
    // Do something with this.data
  },
  // Arrow function enhances readability
  processData: () => {
    // Do something with this.data
  }
};
```

Swipe 



Less Code

- Arrow functions promote functional programming principles by offering concise syntax for higher-order functions and method chaining.



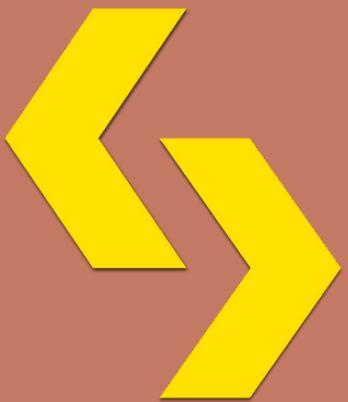
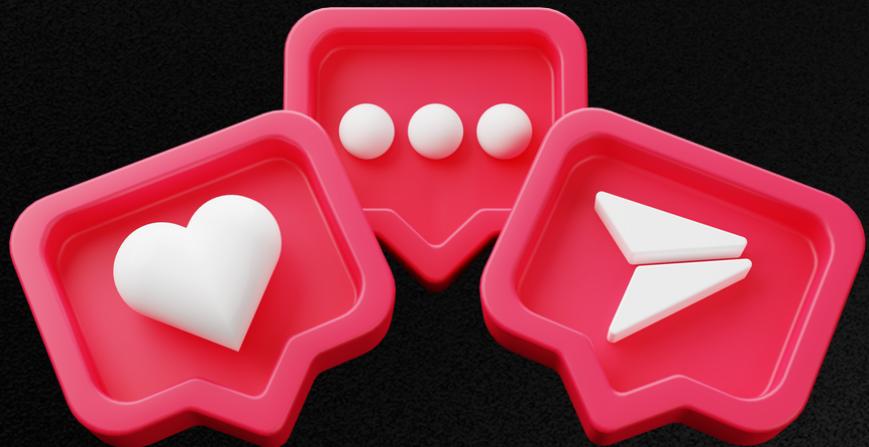
```
// Higher-order function using arrow function
const add = x => y => x + y;
const addTwo = add(2);
console.log(addTwo(3)); // Output: 5
```

Swipe 



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