JavaScript ES6+ Features — Lesson Notes

Destructuring

```
// Array destructuring
const [first, second] = [10, 20];

// Object destructuring
const { name, age } = { name: "Sara", age: 22 };
```

Spread & Rest Operators

```
const arr = [1, 2, 3];
const copy = [...arr, 4, 5];       // [1, 2, 3, 4, 5]

const { id, ...rest } = user;      // rest = everything except id
```

Modules (import / export)

```
// math.js
export const add = (a, b) => a + b;
export default class Calculator { /* ... */ }

// app.js
import Calculator, { add } from './math.js';
```

Optional Chaining & Nullish Coalescing

```
const city =
        user?.address?.city;           // undefined if any is null
const name = user?.name ?? "Anonymous";     // fallback only for
        null/undefined
```

Map, Filter, Reduce

```
const nums = [1, 2, 3, 4, 5];
const doubled = nums.map(n => n * 2);           // [2, 4, 6, 8,
        10]
const evens = nums.filter(n => n % 2 === 0);    // [2, 4]
const sum = nums.reduce((acc, n) => acc + n, 0); // 15
```

Key Takeaways

1. Destructuring makes extracting values from objects/arrays concise.
2. The spread operator is essential for immutable data patterns.
3. Use optional chaining (`?.`) to safely access nested properties.
4. `map`, `filter`, `reduce` are the backbone of functional JS.
