Let's create some JavaScript destructuring assignments to practice extracting values from objects and arrays. These will cover basic destructuring, aliasing, nested destructuring, default values, and function parameter destructuring.

Basic Object Destructuring:

1. Swap two numbers using Destructuring
2. Extracting Name and Age:

JavaScript

const person = { name: "Alice", age: 30, city: "New York" };

// Use destructuring to create variables 'name' and 'age' from the person object.

1. Aliasing and Default Values:

JavaScript

const product = { id: 123, name: "Laptop", price: 999 };

// Use destructuring to create a variable 'productName' (aliased from 'name') and a variable 'discount' with a default value of 0.

1. Nested Object Destructuring:

JavaScript

const user = {

address: {

street: "123 Main St",

city: "Anytown",

zip: "12345"

},

name: "Bob"

};

// Use destructuring to create variables 'street', 'city', and 'zip' from the nested address object.

Basic Array Destructuring:

1. Extracting First Two Elements:

JavaScript

const numbers = [1, 2, 3, 4, 5];

// Use destructuring to create variables 'first' and 'second' containing the first two elements of the array.

1. Rest Operator with Array Destructuring:

JavaScript

const colors = ["red", "green", "blue", "yellow"];

// Use destructuring and the rest operator to create variables 'firstColor', 'secondColor', and an array 'remainingColors' containing the rest of the elements.

1. Skipping Elements in Array Destructuring:

JavaScript

const fruits = ["apple", "banana", "orange", "grape"];

// Use destructuring to extract the first and third elements into variables 'firstFruit' and 'thirdFruit', skipping the second element.

Combining Object and Array Destructuring:

1. Destructuring from an Array of Objects:

JavaScript

const books = [

{ title: "The Hobbit", author: "Tolkien" },

{ title: "Pride and Prejudice", author: "Austen" }

];

// Use destructuring to get the title of the first book and the author of the second book.

1. Destructuring with Nested Arrays:

JavaScript

const matrix = [

[1, 2, 3],

[4, 5, 6],

[7, 8, 9]

];

// Use nested destructuring to extract the number 5 from the matrix.

Function Parameter Destructuring:

1. Destructuring Object Arguments:

JavaScript

function greet({ name, age }) {

console.log(`Hello, ${name}! You are ${age} years old.`);

}

const person = { name: "Charlie", age: 35 };

greet(person);

// Call the greet function with the 'person' object.

1. Destructuring Array Arguments:

JavaScript

function displayCoordinates([x, y]) {

console.log(`X: ${x}, Y: ${y}`);

}

const coordinates = [10, 20];

displayCoordinates(coordinates);

// Call the displayCoordinates function with the 'coordinates' array.

Advanced Destructuring:

1. Dynamic Property Names:

JavaScript

const key = "city";

const { [key]: cityName } = person; // Use computed property name to destructure dynamically.

console.log(cityName);

1. Destructuring with Default Values and Nested Objects:

JavaScript

const product = {

name: "T-shirt",

details: {

color: "blue",

size: "M"

}

};

// Destructure 'name', 'details.color', and 'details.size'. Provide default values if any are missing.

These assignments provide a good range of practice with destructuring in JavaScript, covering various scenarios and levels of complexity. Remember to experiment and try different combinations to solidify your understanding. Let me know if you have any other questions.