**Destructuring Exercise**

**Object Destructuring 1**

What does the following code return/print?

**let** facts = {numPlanets: 8, yearNeptuneDiscovered: 1846};

**let** {numPlanets, yearNeptuneDiscovered} = facts;

console.log(numPlanets); *// ?*

console.log(yearNeptuneDiscovered); *// ?*

**1.) 8**

**2.) 1846**

**Object Destructuring 2**

What does the following code return/print?

**let** planetFacts = {

numPlanets: 8,

yearNeptuneDiscovered: 1846,

yearMarsDiscovered: 1659

};

**let** {numPlanets, ...discoveryYears} = planetFacts;

console.log(discoveryYears); *// ?*

**1.) {yearNeptuneDiscovered: 1846, yearMarsDiscovered: 1659}**

**Object Destructuring 3**

What does the following code return/print?

**function** getUserData({firstName, favoriteColor="green"}){

**return** `Your name is *${*firstName*}* and you like *${*favoriteColor*}*`;

}

getUserData({firstName: "Alejandro", favoriteColor: "purple"}) *// ?*

getUserData({firstName: "Melissa"}) *// ?*

getUserData({}) *// ?*

***1.) Your name is Alejandro and you like purple***

***2.) Your name is Melissa and you like green***

***3.) Your name is undefined and you like green***

**Array Destructuring 1**

What does the following code return/print?

**let** [first, second, third] = ["Maya", "Marisa", "Chi"];

console.log(first); *// ?*

console.log(second); *// ?*

console.log(third); *// ?*

***1.) Maya***

***2.) Marisa***

***3.) Chi***

**Array Destructuring 2**

What does the following code return/print?

**let** [raindrops, whiskers, ...aFewOfMyFavoriteThings] = [

"Raindrops on roses",

"whiskers on kittens",

"Bright copper kettles",

"warm woolen mittens",

"Brown paper packages tied up with strings"

]

console.log(raindrops); *// ?*

console.log(whiskers); *// ?*

console.log(aFewOfMyFavoriteThings); *// ?*

***1.) Raindrops on roses***

***2.) whiskers on kittens***

***3.) [Bright copper kettles, warm woolen mittens, Brown paper packages tied up with strings]***

**Array Destructuring 3**

What does the following code return/print?

**let** numbers = [10, 20, 30];

[numbers[1], numbers[2]] = [numbers[2], numbers[1]]

console.log(numbers) *// ?*

***1.) [10, 30, 20]***

**ES2015 Refactoring**

In this exercise, you’ll refactor some ES5 code into ES2015.

**ES5 Assigning Variables to Object Properties**

**var** obj = {

numbers: {

a: 1,

b: 2

}

};

**var** a = obj.numbers.a;

**var** b = obj.numbers.b;

**ES2015 Object Destructuring**

*/\* Write an ES2015 Version \*/*

**let obj = {**

**numbers: {**

**a: 1,**

**b: 2**

**}**

**};**

**let {a, b} = obj;**

**ES5 Array Swap**

**var** arr = [1, 2];

**var** temp = arr[0];

arr[0] = arr[1];

arr[1] = temp;

**ES2015 One-Line Array Swap with Destructuring**

*/\* Write an ES2015 Version \*/*

**[arr[0], arr[1]] = [arr[1], arr[0]];**

**raceResults()**

Write a function called ***raceResults*** which accepts a single array argument. It should return an object with the keys ***first***, ***second***, ***third***, and ***rest***.

* *first: the first element in the array*
* *second: the second element in the array*
* *third: the third element in the array*
* *rest: all other elements in the array*

**Write a *one line* function to make this work using**

* An arrow function
* Destructuring
* ‘Enhanced’ object assignment (same key/value shortcut)

raceResults(['Tom', 'Margaret', 'Allison', 'David', 'Pierre'])

*/\**

*{*

*first: "Tom",*

*second: "Margaret",*

*third: "Allison",*

*rest: ["David", "Pierre"]*

*}*

*\*/*

**const raceResults = ([first, second, third, …rest]) => ({first, second, third, rest});**