**Object Destructuring 1**

What does the following code return/print?

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

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

console.log(numPlanets); *// 8*

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

*// {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"})

*// Your name is Alejandro and you like purple*

getUserData({firstName: "Melissa"})

*// Your name is Melissa and you like green*

getUserData({})

*// 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); *// Maya*

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

console.log(third); *// 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); *// "Raindrops on roses"*

console.log(whiskers); *// "whiskers on kittens"*

console.log(aFewOfMyFavoriteThings);

*// ["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) *// [10,30,20]*

**ES2015 Refactoring**

**ES2015 Object Destructuring**

**const** obj = {

numbers: {

a: 1,

b: 2

}

};

**const** {a,b} = obj.numbers

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

[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***.

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