**Advanced Functions: Map, filter, find, some, every**

**Instructions**

To complete this Practice problem, you will need to get all the tests to pass. To do so, complete the following:

* Complete each function as described below.
* Use the suggested native array methods in each function, as described.

This practice problem should not take you longer than about 25 minutes. If you spend longer than that, reach out for help!

**Dataset**

Assume for all of the following problems that parks refers to a dataset that looks similar to the following.

const parks = [

{

name: "Canyonlands",

areaInSquareKm: 1366.2,

location: { state: "Utah" },

},

{

name: "Crater Lake",

areaInSquareKm: 741.5,

location: { state: "Oregon" },

},

{

name: "Zion",

areaInSquareKm: 595.9,

location: { state: "Utah" },

},

];

**findParkByName()**

Use the .find() method to return the object of the park with the matching name. Otherwise, the function should return null.

findParkByName(parks, "Zion"); *//> { name: "Zion", ... }*

**allParksAboveCertainSize()**

Use the .every() method to return true if all parks are greater than the given size. Otherwise, return false.

allParksAboveCertainSize(parks, 500); *//> true*

allParksAboveCertainSize(parks, 1000); *//> false*

**getBigParkNames()**

Use the .filter() *and* the .map() method to return the names of all parks with a size greater than or equal to the given size.

getBigParkNames(parks, 700); *//> [ "Canyonlands", "Crater Lake" ]*

**doesStateHaveOneBigPark()**

Use the .filter() *and* the .some() method to return true if any park from the given state has a size greater than or equal to the given size. Otherwise, return false.

doesStateHaveOneBigPark(parks, 700, "Oregon"); *//> false*

doesStateHaveOneBigPark(parks, 700, "Utah"); *//> true*

**Tips**

* You may complete this challenge on your own machine before uploading it to Qualified.
* Reference the related checkpoint for help on completing this Practice problem.
* If you need help, contact your mentor or speak with your peers in Slack.