Kata 7 - In the Air Tonight

Assignment

30 - 45 minutes

Status | Complete

The new smart parking lot in Codeville was a big hit! Next on your list to tackle is the air quality. You've decided that you want to install air pollution sensors around the city to monitor air quality and identify problem areas. We need to write the code for the sensors to trigger a special message when the air is too polluted.

> For this challenge we will implement a function called `checkAir()`, which will check a collection of air samples. The function will take in two arguments. The first argument is an array of strings, where each string represents a small air sample that is either `clean` or `dirty`. The second argument is a number representing the highest acceptable amount of dirty samples. For example, a threshold of 0.4 means that there must be less than 40% of total samples classified as dirty for our air to be considered clean. Our function must return `Polluted` if there are too many dirty air samples, or `Clean` if the proportion of dirty samples is below the threshold.

Input

```
const checkAir = function (samples, threshold) {
  // Code here!
};

console.log(checkAir(
  ['clean', 'clean', 'dirty', 'clean', 'dirty', 'clean', 'clean', 'dirty', 'clean', 'dirty'],
  0.3
));

console.log(checkAir(
  ['dirty', 'dirty', 'dirty', 'dirty', 'clean'],
  0.25
));

console.log(checkAir(
  ['clean', 'dirty', 'clean', 'dirty', 'clean', 'dirty', 'clean'],
  0.9
))
```

Expected Output

```
Polluted
Polluted
Clean
```

Not sure where to get started? It might be useful to start by creating a variable to keep track of how many dirty samples we find as we loop through the array of samples. We can increment or add to this number using the `++` operator. Once you know how many dirty samples there are, we just need to do some simple math to determine if it exceeds the threshold.

Submit Your Work

- Browse to gist.github.com and create a new gist.
- Copy-and-paste your code into the form
- Name the gist and the file appropriately and click `Create secret gist`.
- Finally, mark this activity as completed (at the bottom of this page) and please copy/paste the *entire* browser URL for your gist (from *gist.github.com*) into the text field.

---

https://gist.github.com/sa ✕ Cancel Submission

← Previous
Kata 6 - SmartParking

Next
Kata 8 - Repeating Numbers (Stretch) →

How well did you understand this content?

Thank you for your feedback

★ ★ ★ ★ ★

Totally got it!

Please give us some written insight into your feedback

Prep Work

- Katas ✓
- </> Kata 1 - Sum the Largest Numbers ✓
- </> Kata 2 - Conditional sums ✓
- </> Kata 3 - Vowels ✓
- </> Kata 4 - Instructors Names ✓
- </> Kata 5 - Percent Encoded String ✓
- </> Kata 6 - SmartParking ✓
- </> Kata 7 - In the Air Tonight ✓
- </> *Kata 8 - Repeating Numbers* ✓
- </> *Kata 9 - Case Maker* ✓
- </> *Kata 10 - Multiplication Table* ✓
- </> *Kata 11 - Bouncy Castles* ✓
- </> *Kata 12 - The Great Codeville Bake-off* ✓
- </> *Kata 13 - Talking Calendar* ✓
- </> *Kata 14 - Change Calculator* ✓
- </> *Kata 15 - Organizing Instructors* ✓
- </> *Kata 16 - Case Maker II* ✓
- </> *Kata 17 - JS Object From URL Encoded String* ✓
- </> *Kata 18 - Square Code* ✓
- </> *Kata 19 - Queen Threat Detector* ✓
- </> *Kata 20 - Taxicab Geometry* ✓
- </> *Kata 21 - Number Guesser* ✓

› 7: Stretch Project

› 8: The Lab Manual

› 9: Day One Prep

› 10: Collab Tools Setup

POWERED BY

LIGHTHOUSE LABS