To get a handle on how this works differently with arrow functions, let's do a quick recap of how this works in a standard function. If you have a solid grasp of how this works already, feel free to [jump over this section](https://classroom.udacity.com/courses/ud356/lessons/3925704a-be38-4b70-8c8b-a4a812b6a309/concepts/32f53f68-4b31-40fb-b136-4eda28b8ff57).

The value of the this keyword is based completely on how its function (or method) is called. this could be any of the following:

**1. A new object**

If the function is called with new:

**const** mySundae = **new** Sundae('Chocolate', ['Sprinkles', 'Hot Fudge']);

In the code above, the value of this inside the Sundae constructor function is a new object because it was called with new.

**2. A specified object**

If the function is invoked with call/apply:

**const** result = obj1.printName.call(obj2);

In the code above, the value of this inside printName() will refer to obj2 since the first parameter of call() is to explicitly set what this refers to.

**3. A context object**

If the function is a method of an object:

data.teleport();

In the code above, the value of this inside teleport() will refer to data.

**4. The global object or undefined**

If the function is called with no context:

teleport();

In the code above, the value of this inside teleport() is either the global object or, if in strict mode, it's undefined.

***TIP:****this in JavaScript is a complicated topic. We just did a quick overview, but for an in-depth look at how this is determined, check out*[*this All Makes Sense Now!*](https://github.com/getify/You-Dont-Know-JS/blob/master/this%20%26%20object%20prototypes/ch2.md)*from Kyle Simpson's book series*[*You Don't Know JS*](https://github.com/getify/You-Dont-Know-JS/blob/master/README.md)*.*

**QUESTION 1 OF 2**

What is the value of this inside the Train constructor function below?

**const** redTrain = **new** Train('red');

* 

the window object

* a new object
* 

undefined

SUBMIT

**QUESTION 2 OF 2**

What is the value of this inside the increaseSpeed() function below?

**const** redTrain = **new** Train('red');

redTrain.increaseSpeed(25);

* 

the window object

* 

a new object

* the redTrain object
* 

undefined

SUBMIT

NEXT