**Hoisting**

Sometimes your JavaScript code will produce errors that may seem counterintuitive at first. **Hoisting** is another one of those topics that might be the cause of some of these tricky errors you're debugging.

### QUESTION 1 OF 3

What value will be printed to the console?

sayHi("Julia");

**function** **sayHi**(name) {

console.log(greeting + " " + name);

**var** greeting;

}

* 

error

* 

undefined

* 

Julia

* undefined Julia
* 

null

* 

null Julia

* 

NaN

* 

NaN Julia

* 

Hello Julia

SUBMIT

### QUESTION 2 OF 3

What value will be printed to the console?

sayHi("Julia");

**function** **sayHi**(name) {

console.log(greeting + " " + name);

**var** greeting = "Hello";

}

* 

error

* 

undefined

* 

Julia

* undefined Julia
* 

null

* 

null Julia

* 

NaN

* 

NaN Julia

* 

Hello Julia

SUBMIT

### QUESTION 3 OF 3

What value will be printed to the console?

**function** **sayHi**(name) {

**var** greeting = "Hello";

console.log(greeting + " " + name);

}

sayHi("Julia");

* 

error

* 

undefined

* 

Julia

* 

undefined Julia

* 

null

* 

null Julia

* 

NaN

* 

NaN Julia

* Hello Julia

**What you've learned so far:**

* JavaScript hoists function declarations and variable declarations to the top of the current scope.
* Variable *assignments* are not hoisted.
* Declare functions and variables at the top of your scripts, so the syntax and behavior are consistent with each other.