"Fizzbuzz" is a famous interview question used in programming interviews. It goes something like this:

* Loop through the numbers 1 to 100
* If the number is divisible by 3, print "Fizz"
* If the number is divisible by 5, print "Buzz"
* If the number is divisible by both 3 and 5, print "FizzBuzz"
* If the number is **not** divisible by 3 or 5, print the number

***TIP:****A number x is divisible by a number y if the answer to x / y has a remainder of 0. For example, 10 is divisible by 2 because 10 / 2 = 5 with no remainder. You can check if a number is divisible by another number by checking if x % y === 0.*

We're going to have you program your own version of FizzBuzz called "JuliaJames" (yes, imaginative, right?) Keep in mind that in an interview, you would want to write efficient code with very little duplication. We don't want you to worry about that for this question. Just focus on practicing using loops.

**Directions:**

Write a while loop that:

* Loop through the numbers 1 to 20
* If the number is divisible by 3, print "Julia"
* If the number is divisible by 5, print "James"
* If the number is divisible by 3 and 5, print "JuliaJames"
* If the number is **not** divisible by 3 or 5, print the number

/\*

\* Programming Quiz: JuliaJames (4-1)

\*/

/\*

\* QUIZ REQUIREMENTS

\* - Your code should have a variable `x` with a starting value of `1`

\* - Your code should include a `while` loop

\* - Your `while` loop should have a stop condition

\* - Your code should use a conditional statement

\* - Your code should increment `x` by `1` each time the loop executes

\* - Your code should produce the expected output

\* - Your code should not be empty

\* - BE CAREFUL ABOUT THE PUNCTUATION AND THE EXACT WORDS TO BE PRINTED.

\*/

var x = 1;

while (x<=20) {

if(x % 3 === 0 && x % 5 === 0){

console.log("JuliaJames");

}else if(x%3===0){

console.log("Julia");

}else if(x%5===0){

console.log("James");

}else {

console.log(x);

}

x++;

}

/\* your stop condition goes here \*/

// check divisibility

// print Julia, James, or JuliaJames

// increment x