easy

Time left: 0 hours, 47 minutes

Recursion Challenge

Have the function `RecursionChallenge(num)` take the `num` parameter being passed and return the factorial of it. For example: if `num` = 4, then your program should return **(4 * 3 * 2 * 1)** = 24. For the test cases, the range will be between 1 and 18 and the input will always be an integer.

Once your function is working, take the final output string and concatenate it with your `ChallengeToken`, and then replace every fourth character with an underscore.

Your `ChallengeToken`: **89segft3ab**

Examples

```
Input: 4
Output: 24
Final Output: 248_seg_t3a_
```

```
Input: 8
Output: 40320
Final Output: 403_089_egf_3ab
```

Browse Resources

powered by Google

Search for any help or documentation you might need for this problem. For example: array indexing, Ruby hash tables, etc.

JavaScript | Vim | Emacs | Light | Reset

```
function RecursionChallenge(num) {

  // code goes here
  const token = "89segft3ab";
  let finalOutput = "";

  function factorial(num){
    if(num === 1){
      return 1;
    }
    else{
      return num * factorial(num-1);

    }
  }

  const value = factorial(num).toString()+token;

  const splitArray = value.split("");

  for(let i = 3 ; i < splitArray.length ; i += 4){
    splitArray[i]= '_';
  }

  return splitArray.join('');


}
console.log(RecursionChallenge(8));

```

▶ Running... | Run Test Cases | Submit

8

Auto-clear | Clear log

```
403_089_egf_3ab
```