Challenge: Iterative factorial

Finish the provided factorial function, so that it returns the value **n!**.

Your code should use a for loop to compute the product 1 * 2 * 3 * ... * n. If you write the code carefully, you won't need a special case for when n equals 0.

```
Java

Python

C++

Class Solution {

public static int factorial(

int result = 0;

// Implement this method

return result;

}

}

| 1 | Class Solution {

public static int factorial(

int result = 0;

| 2 | Python

| 3 | Python

| 4 | Python

| 5 | Python

| 6 | Python

| 7 | Python

| 8 | Python

| 9 | Python

| 1 | Python

| 1 | Python

| 1 | Python

| 2 | Python

| 3 | Python

| 4 | Python

| 5 | Python

| 6 | Python

| 7 | Python

| 7 | Python

| 8 | Python

| 9 | Python

| 1 | Python

| 2 | Python

| 3 | Python

| 4 | Python

| 5 | Python

| 6 | Python

| 7 | Python

| 7 | Python

| 8 | Python

| 9 | Python

| 9 | Python

| 1 | Python

| 2 | Python

| 3 | Python

| 4 | Python

| 5 | Python

| 6 | Python

| 7 | Python

| 9 | Python

| 9 | Python

| 1 | Python

| 2 | Python

| 2 | Python

| 3 | Python

| 4 | Python

| 4 | Python

| 5 | Python

| 5 | Python

| 5 | Python

| 6 | Python

| 7 | Python

| 9 | Pyt
```