JAVASCRIPT

RECURSION

- Technique for iterating over an operation by having a function call itself repeatedly until it arrives at a result.
- Simply? When a function calls itself.

THREE KEY FEATURES

- **Termination Condition** - if (something bad happened) { STOP };
  - In factorial example, if (x < 0) return; is our termination condition. It's not possible to factorial a negative number.
- **Base Case** - if (this happens) { Good! We're done }; - also stops our recursion.
  - In factorial example, if (x === 0) return 1; is our base case. We know that once we've gotten x down to zero, we've succeeded in determining our factorial!
- **Recursion** - our function calling itself.
  - In the factorial example, return x * factorial(x – 1); is where the recursion actually happens. We're returning the value of the number x multiplied by the value of whatever factorial(x-1) evaluates to.

EXAMPLE

```
function factorial(x) {
  // TERMINATION
  if (x < 0) return;


  // BASE
  if (x === 0) return 1;


  // RECURSION
  return x * factorial(x - 1);
}


factorial(3);
// 6
```

EXAMPLE #2

```
> const qq = a => c => c ? a + c : c;
<· undefined
> qq(1)(0) == 0
<· true
> qq(3)(5)
<· 8
> 
```

NPM - REASONS FOR APPEARANCE

- The code is difficult to communicate to others.
- Each developer has his own copy of the same or nearly identical solutions to the same problems.
- Only the author is engaged in revision.
- There are many different copies with no quick update option.
- Due to the fact that the code is copied directly into another project, as a rule, it is modified and becomes specific to a particular project.

TERMINOLOGY

- **Package** - basic unit that NPM manages as a whole. It can contain any number of files and code. NPM allows you to install packages, upgrade, or remove.

- **Registry** - repository for NPM packages. Anyone can publish a package in the npm registry, spending just a minute, and the rest can use it. The repository currently contains hundreds of thousands of packages and their number is growing rapidly. The source code of the packages is usually stored on the github. Regardless, the packages in npm have nothing to do with git and github.

PACKAGE.JSON

- **dependencies** field is used to list all the dependencies of your project that are available on npm.
- devDependencies field to your package.json - these are dependencies not required for normal operation, but required/ recommended if you want to patch or modify the project