map app style guide

- semicolons at end of statements
- naming
    - weUseCamelCaseHere
    - variable names begin with lower case letter
    - class names begin with uppercase letter
- variables
    - Variables declared at the top of the scope when appropriate
    - please use descriptive names. Just use a text editor that has good auto fill
- Comments refer to code beneath

```
// adds node to chunk
chunk.push(node);
```

- Every function (including classes) has a comment above it that explains the arguments it expects, ie: what data type is expected for each argument. If the function takes no arguments, you do not need to comment about the args. Beneath that is a comment describing the functions purpose. Brownie points if you comment where you expect this to be called from. If you expect the function to be called directly from the html page, you must comment that above everything else.

```
// num is a number, bob is a person object
// this function combines foo and bob, then returns that
function foo(num, bob) {
   return num + bob;
}


//called from html doc
//prints out the name of each teacher
function waa() {
   var teachersAsAString = "";
   for (var i = 0; i < teachers; i++){
   teachersAsAString += teachers[i] + " ";
   }
   console.log("the teachers employed at our school are " + teachersAsAString);
```

- Use class syntax. Capitalize class names. You do not need to comment the purpose of the constructor function, but do comment the arguments it takes. Class files should contain only the contents of that class (functions not belonging to the class do not belong here.)

```
class Node {
  // num will be a number
  constructor(num) {
  this.num = num;
 }
  foo() {}
}


//do not include this in this document
function calledFromAnywhere() {}
```

- Prefer this `function foo() {}` to `var foo = function(){}` if you violate this, comment why. You do not need to comment why if you do this in the html doc.

-