EECS 1012: Introduction to Computer Science

October 7, 2015

```
<!DOCTYPE html>
<html>
<head>
<script src="loop.js" type="text/javascript"> </script>
</head>
<body>
  <button onclick="go()">Go</button>
  <div id="output"> output<br> </div>
</body>
</html>
```

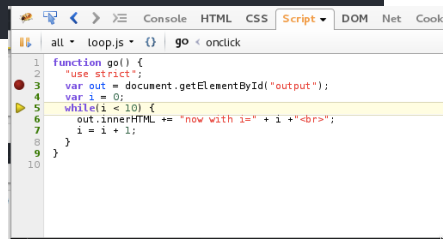
Basic Concept

- Repeat a collection of statements until some condition is met
- Why are there three different versions (btw, common in many programming languages)?
  - Because sometime one (or another) makes the code cleaner. **style**

while(expression) {}

- Evaluates the expression first
  - Then the body of the loop
- Imagine ‘read input’ while the input has not been validated, complain to the user

```
function go() {
  var out = document.getElementById("output");
  var x = 0;
  while(x < 10) {}
    out.innerHTML += x + "<br>";
    x = x + 1;
  }
}
```



do {…} while(expr);

- Evaluates the expression after doing the loop
  - Always does it once.

```
function go() {
  var out = document.getElementById("output");
  var x = 0;
  do {
    out.innerHTML += x + "<br>";
    x = x + 1;
  } while(x < 10);
}
```

for(exp;cond;exp){..}

- Often (very often) loop over something.
  - n times

JSLint designers have an issue with for
(for some applications there is a better version of for)

```
function go() {
  var out = document.getElementById("output");
  for(var x=0;x<10;x++) {
    out.innerHTML += x + "<br>";
  }
}
```

Repetition is not just for constants

```
function go() {
  "use strict";
  var out = document.getElementById("output");
  var n = prompt("Enter a number", "10");
  if(n != null) {
    n = Number(n);
    if(isNaN(n) || (n < 0)) {
      alert("Invalid number using 10");
      n = 10;
    }
    for(var i=0;i<n;i++){
      out.innerHTML += "Look, a line of text " + i + "<br>";
    }
  }
}
```