Review!
As mentioned last week, my goal is to review during this class

• We'll do most of this with exercises
• I'll add in a bit of explanation where necessary
• The things that I heard the most that people wanted reviewed were functions, conditionals, and for loops
• That's not to say that other materials won't be on the final, or aren't important…
As a brief recap, we've covered

- computing basics and html
- interfaces
- programming concepts
- art & image representation
- networking
- minds & machines
- inside the machine
- digital DNA
- diversity
There are MANY practice questions on the web

- If you try to do them by looking at the solution and making sure that you understand the solution rather than trying to solve the problem, you are in for a world of pain on the final.
- The best way to study is to do the practice problems
- That's what we'll do in class today based on feedback on what people would like to cover.
Let's start with functions

- A function is a bit of code that we **declare** in one place, and then **call** in another.
- You can think of this as kind of like the declaration creates a blueprint for a house, and then calling it actually creates the house.
Anatomy of a Javascript Function Declaration

A **function declaration** includes a **name** and possibly **parameters**. A parameter is a variable that is passed in to the function.

The **body** includes **variable declarations**, **other instructions**, and sometimes a **return** statement to return information back to whatever called it.

```javascript
function calculateTip(amount) {
  var tip;
  tip = amount * 0.12;
  return(tip);
}
```
Goal #1

• Snow is in the forecast. For all the Americans here who are wishing that they were at home for Thanksgiving, teach them how to translate snowfall levels in cm to snowfall levels in inches.

• To do this, create a function `convertCMtoIN`, which takes a parameter and returns the amount in inches.

• For example, `convertCMtoIN(10)` should return 3.937007874015748
Another similar practice question:

• Help the poor Americans translate temperatures from Celsius to Fahrenheit. To do so, multiply the number by 9/5 and add 32.

• For example, translateCtoF(25) should return 77
Next let’s help those poor Americans with the blinking green traffic lights

- In BC, a blinking green light means that it’s controlled by pedestrians
- In Ontario, a blinking green light means that it’s a protected left term
- Elsewhere, it means the light’s broken
- How can we write a function that captures that?

if/else Statements

if (<Boolean expression>)
  <then-statement>;
else
  <else-statement>;

• If the <Boolean expression>’s outcome is **true**:  
  • The <then-statement> is executed  
  • The <else-statement> is skipped  

• If the <Boolean expression>’s outcome is **false**:  
  • The <then-statement> is skipped  
  • The <else-statement> is executed
The blinkingLightExplainer

- Write a function blinkingLightExplainer that takes as a parameter the two letter province code and behaves appropriately.
  - blinkingLightExplainer('BC') returns “pedestrian controlled”
  - blinkingLightExplainer('ON') returns “protected left”
  - All other calls return “broken light”
Next, we’ll help those poor Americans go to the grocery store

- Most grocery store items are called the same, but some are different. Write a function that returns the same input unless the input is one of the following exceptions:
  hazelnuts → filberts
  powdered sugar → icing sugar
- For example `groceriesUSToCDN("milk")` should return "milk", but
  `groceriesUSToCDN("hazelnuts")` should return "filberts".

(http://www.americansguide.ca/isms.html)
Let’s count those seconds

• Next, let’s help Americans get out of the habit of counting seconds in Mississippi (e.g., “one Mississippi, two Mississippi”…)
• Instead, help them count in one-thousands or steamboats ([http://boards.straightdope.com/sdmb/archive/index.php/t-381546.html](http://boards.straightdope.com/sdmb/archive/index.php/t-381546.html))
  “one steamboat, two steamboat”
• For that, we need for loops
### for Loop Syntax

```plaintext
for(<initialization>;<continuation>;<next iteration>)
{
   < statement list>
}
```

*<continuation>* has the same form as the predicate in a conditional statement.

If the *<continuation>* test is false outcome, the loop terminates and *<statement list>* is skipped.

If *<continuation>* has a true outcome, the *<statement list>* is performed.

*<next iteration>* defines what happens at the end of the loop to start the next round.
Write a function to help people count their seconds

• `countSeconds(number, countingWord)` should help people count out the right number of `countingWords` by returning the right thing to say.

• E.g., `countSeconds(3, 'steamboat')` should return
  1 steamboat
  2 steamboat
  3 steamboat