**JAVASCRIPT**

**1. What is your name?**

"John"

**2. Discover the Length**

"John".length

**3. Basic Math**

3+7

**4. Numbers and More**

100\*100

**5. Error: does not compute**

eggplant

**6. Editor and Comments**

// This is a comment that the computer will ignore. // It is for your eyes only! "cake".length

**7. What am I learning?**

confirm('This is an example of using JS to create some interaction on a website. Click OK to continue!');

**8. Interactive JavaScript**

// Also try the Q&A forum to get help // The link is on the bottom left of the page! confirm("Hello, world!")

**9. What is programming?**

prompt("What is your name?")

**10. Data Types I & II: Numbers & Strings**

"Codecademy is really cool!".length

**11. Data Types III: Booleans**

"I'm coding like a champ!".length > 10

**12. Using console.log**

console.log(2\*5) console.log("Hello")

**13. Comparisons**

// Here is an example of using the greater than (>) operator. console.log(15 > 4); // 15 > 4 evaluates to true, so true is printed. // Fill in with >, <, === so that the following print out true: console.log("Xiao Hui".length < 122); console.log("Goody Donaldson".length > 8); console.log(8\*2 === 16);

**14. Decisions, decisions**

if(prompt("What is your name?".length < 4)) { console.log("Your name has less characters than mine :P"); }

**15. Computers are smart**

if (prompt("What road do you want to go down?") === "One") { console.log("Let's go down the first road!"); } else { console.log("Let's go down the second road!"); }

**16. More Practice with conditionals**

if(prompt("Is Codecademy the best?") === "No") { console.log("The condition is true"); } else { console.log("The condition is false"); }

**17. Computers aren't that smart**

if (10 === 10) { console.log("You got a true!"); } else { console.log("You got a false!"); }

**18. Mid-lesson breather**

if (12 / 4 === "Ari".length) { confirm("Will this run the first block?"); } else { confirm("Or the second block?"); }

**19. Math**

if("Jon".length \* 2 / (2+1) === 2) { console.log("The answer makes sense!"); } else { console.log("Error Error Error") }

**20. Math and the modulo**

console.log(14%3) console.log(99%8) console.log(11%3)

**21. Modulo and if/else**

if(5===5) { console.log("The first number is even"); } else { console.log("The first number is odd"); } 10%5===0

**22. Substrings**

"wonderful day".substring(3,7)

**23. More substring practice**

console.log("January".substring(0,3)); console.log("Melbourne is great".substring(0,12)); console.log("Hamburgers".substring(3,10));

**24. Variables**

var myAge = 100; console.log(myAge);

**25. More Variable Practice**

// Declare a variable on line 3 called // myCountry and give it a string value. var myCountry = "United States of America"; // Use console.log to print out the length of the variable myCountry. console.log(myCountry.length); // Use console.log to print out the first three letters of myCountry. console.log(myCountry.substring(0,3));

**26. Change variable names**

// On line 2, declare a variable myName and give it your name. var myName = "John"; // On line 4, use console.log to print out the myName variable. console.log(myName); // On line 7, change the value of myName to be just the first 2 // letters of your name. myName = "Jo"; // On line 9, use console.log to print out the myName variable. console.log(myName);

**27. Conclusion: Part 1**

var myColor = "Blue"; console.log(myColor.length);

**28. Conclusion: Part 2**

if(5===5) { console.log("I finished my first course!"); } else { console.log("Impossible!"); }

**Building Rock, Paper Scissor**

## 2. User Choice

var userChoice = prompt("Do you choose rock, paper or scissors?");

## 3. Computer Choice: Part 1

var userChoice = prompt("Do you choose rock, paper or scissors?"); var computerChoice = Math.random(); console.log(computerChoice);

## 4. Computer Choice: Part 2

var userChoice = prompt("Do you choose rock, paper or scissors?"); var computerChoice = Math.random(); //ComputerChoice Random Number to Selection Conversion Statements if(computerChoice <= 0.33) { computerChoice = "rock"; } else if(computerChoice <= 0.66) { computerChoice = "paper"; } else { computerChoice = "scissors"; }

## 5. Both choices are the same!

/\*var userChoice = prompt("Do you choose rock, paper or scissors?"); var computerChoice = Math.random(); if (computerChoice < 0.34) { computerChoice = "rock"; } else if(computerChoice <= 0.67) { computerChoice = "paper"; } else { computerChoice = "scissors"; } console.log("Computer: " + computerChoice);\*/ var compare = function(choice1, choice2) { if(choice1 === choice2) { return "The result is a tie!"; } }

## 6. What if choice1 is rock?

/\*var userChoice = prompt("Do you choose rock, paper or scissors?"); var computerChoice = Math.random(); if (computerChoice < 0.34) { computerChoice = "rock"; } else if(computerChoice <= 0.67) { computerChoice = "paper"; } else { computerChoice = "scissors"; } console.log("Computer: " + computerChoice);\*/ var compare = function(choice1, choice2) { if(choice1 === choice2) { return "The result is a tie!"; } else if(choice1 === "rock") { if(choice2 === "scissors") { return "rock wins"; } else { return "paper wins"; } } }

## 7. What if choice1 is paper?

/\*var userChoice = prompt("Do you choose rock, paper or scissors?"); var computerChoice = Math.random(); if (computerChoice < 0.34) { computerChoice = "rock"; } else if(computerChoice <= 0.67) { computerChoice = "paper"; } else { computerChoice = "scissors"; } console.log("Computer: " + computerChoice);\*/ var compare = function(choice1, choice2) { if(choice1 === choice2) { return "The result is a tie!"; } else if(choice1 === "rock") { if(choice2 === "scissors") { return "rock wins"; } else { return "paper wins"; } } else if(choice1 === "paper") { if(choice2 === "rock") { return "paper wins"; } else { return "scissors wins"; } } }

## 8. What if choice1 is scissors?

var userChoice = prompt("Do you choose rock, paper or scissors?"); var computerChoice = Math.random(); if (computerChoice < 0.34) { computerChoice = "rock"; } else if(computerChoice <= 0.67) { computerChoice = "paper"; } else { computerChoice = "scissors"; } console.log("Computer: " + computerChoice); compare(userChoice, computerChoice); var compare = function(choice1, choice2) { if(choice1 === choice2) { return "The result is a tie!"; } else if(choice1 === "rock") { if(choice2 === "scissors") { return "rock wins"; } else { return "paper wins"; } } else if(choice1 === "paper") { if(choice2 === "rock") { return "paper wins"; } else { return "scissors wins"; } } }

## 9. Next Steps

var userChoice = prompt("Do you choose rock, paper or scissors?"); var computerChoice = Math.random(); if (computerChoice < 0.34) { computerChoice = "rock"; } else if(computerChoice <= 0.67) { computerChoice = "paper"; } else { computerChoice = "scissors"; } console.log("Computer: " + computerChoice); compare(userChoice, computerChoice); var compare = function(choice1, choice2) { if(choice1 === choice2) { return "The result is a tie!"; } else if(choice1 === "rock") { if(choice2 === "scissors") { return "rock wins"; } else { return "paper wins"; } } else if(choice1 === "paper") { if(choice2 === "rock") { return "paper wins"; } else { return "scissors wins"; } } }

Loops

## 1. Why use for loops?

console.log("1"); console.log("2"); console.log("3"); console.log("4"); console.log("5");

## 2. First for loop

for (var counter = 1; counter < 11; counter++) { console.log(counter); }

## 3. Starting the for loop

// Change where the for loop starts. for (var i = 5; i < 11; i = i + 1){ console.log(i); }

## 4. Ending the for loop

// Edit this for loop for (var i = 4; i < 24; i = i + 1) { console.log(i); }

## 5. Controlling the for loop

for (var i = 5; i < 51; i+=5) { console.log(i); }

## 6. How does it work?

for (var i = 8 ; i < 120; i+=12) { console.log(i); }

## 7. Practice counting down

// Example of infinite loop. THIS WILL CRASH YOUR // BROWSER. Don't run the code without changing it! for (var i = 10; i >= 0; i--) { console.log(i); }

## 8. Last practice for loop

for(var i=100; i>0; i-=5) { console.log(i); }

## 9. Meet arrays

var junk = ["Bunk","Trunk",4,6]; console.log(junk)

## 10. Array positions

var junkData = ["Eddie Murphy", 49, "peanuts", 31]; console.log(junkData[3])

## 11. Loops and arrays I

var cities = ["Melbourne", "Amman", "Helsinki", "NYC", "Boston", "Chicago"]; for (var i = 0; i < cities.length; i++) { console.log("I would like to visit " + cities[i]); }

## 12. Loops and arrays II

var names = ["Steven","Jacob","Brent","Jack","John"]; for(var i=0; i < 5;i++) { console.log("I know someone called " + names[i]) }

## 1. What You'll be Building

/\*jshint multistr:true \*/ text = "Blah blah blah blah blah blah Eric \ blah blah blah Eric blah blah Eric blah blah \ blah blah blah blah blah Eric"; var myName = "Eric"; var hits = []; // Look for "E" in the text for(var i = 0; i < text.length; i++) { if (text[i] === "E") { // If we find it, add characters up to // the length of my name to the array for(var j = i; j < (myName.length + i); j++) { hits.push(text[j]); } } } if (hits.length === 0) { console.log("Your name wasn't found!"); } else { console.log(hits); }

## 2. Declare your variables

/\*jshint multistr:true \*/ var text = "Lorem ipsum dolor sit amet, John consectetur adipiscing elit. Praesent quis dictum nulla. Nullam condimentum lacinia turpis a semper."; var myName = "John"; var hits = [];

## 3. Your first "for" loop

/\*jshint multistr:true \*/ var text = "Lorem ipsum dolor sit amet, John consectetur adipiscing elit. Praesent quis dictum nulla. Nullam condimentum lacinia turpis a semper."; var myName = "John"; var hits = []; for(var i =0;i <= text.length; i++) { }

## 4. Your "if" statement

/\*jshint multistr:true \*/ var text = "Lorem ipsum dolor sit amet, John consectetur adipiscing elit. Praesent quis dictum nulla. Nullam condimentum lacinia turpis a semper."; var myName = "John"; var hits = []; for(var i =0;i <= text.length; i++) { if(text[i] === "J") { } }

## 5. Your second "for" loop

/\*jshint multistr:true \*/ var text = "Stewart containing some text for Stewart written and developed by Stewart"; var myName = "Stewart"; var hits = []; /\* assign the value of 0 to i and continue to the loop while i is less than the length of the variable 'text', and add 1 to i after executing the code block \*/ for (i = 0; i < text.length + 1; i++) { /\* if the array value of the 'text' variable is equal to the 0 point of the myName variable \*/ if(text[i] === myName[0]) { /\* then assign the value of i to j and continue the loop while j is less than the length of the myName variable plus the value of i \*/ for(var j = i; j < (myName.length + i); j++) { /\* if the array value of the variable 'text' is equal to the array value of j minus i for the 'myName' variable, push the value of j on to the hits array \*/ if(text[j] === myName[j-i]) { hits.push(text[j]); } } } } if(hits.length === 0){ console.log("Your name wasn't found!"); } else { console.log(hits); }

## 6. Log it!

/\*jshint multistr:true \*/ var text = "Stewart containing some text for Stewart written and developed by Stewart"; var myName = "Stewart"; var hits = []; /\* assign the value of 0 to i and continue to the loop while i is less than the length of the variable 'text', and add 1 to i after executing the code block \*/ for (i = 0; i < text.length + 1; i++) { /\* if the array value of the 'text' variable is equal to the 0 point of the myName variable \*/ if(text[i] === myName[0]) { /\* then assign the value of i to j and continue the loop while j is less than the length of the myName variable plus the value of i \*/ for(var j = i; j < (myName.length + i); j++) { /\* if the array value of the variable 'text' is equal to the array value of j minus i for the 'myName' variable, push the value of j on to the hits array \*/ if(text[j] === myName[j-i]) { hits.push(text[j]); } } } } if(hits.length === 0){ console.log("Your name wasn't found!"); } else { console.log(hits); }

## 7. Victory!

/\*jshint multistr:true \*/ var text = "Stewart containing some text for Stewart written and developed by Stewart"; var myName = "Stewart"; var hits = []; /\* assign the value of 0 to i and continue to the loop while i is less than the length of the variable 'text', and add 1 to i after executing the code block \*/ for (i = 0; i < text.length + 1; i++) { /\* if the array value of the 'text' variable is equal to the 0 point of the myName variable \*/ if(text[i] === myName[0]) { /\* then assign the value of i to j and continue the loop while j is less than the length of the myName variable plus the value of i \*/ for(var j = i; j < (myName.length + i); j++) { /\* if the array value of the variable 'text' is equal to the array value of j minus i for the 'myName' variable, push the value of j on to the hits array \*/ if(text[j] === myName[j-i]) { hits.push(text[j]); } } } } if(hits.length === 0){ console.log("Your name wasn't found!"); } else { console.log(hits); }

## 1. While we're at it

var coinFace = Math.floor(Math.random() \* 2); while(coinFace === 0){ console.log("Heads! Flipping again..."); var coinFace = Math.floor(Math.random() \* 2); } console.log("Tails! Done flipping.");

## 2. While syntax

var understand = true; while(understand === true){ console.log("I'm learning while loops!"); understand = false; }

## 3. A fellow of infinite loops

understand = true; while(understand){ console.log("I'm learning while loops!"); understand = false; }

## 4. Brevity is the soul of programming

var bool = true; while(bool){ console.log("Less is more!"); bool = false; }

## 5. Practice makes perfect

var loop = function(){ var count = 0; while(count < 3){ console.log("I'm looping!"); count++ } }; loop();

## 6. Solo flight

var swag = true; var soloLoop = function(){ while(swag === true) { console.log("Looped once!"); swag = false; } }; soloLoop();

## 7. Mid-lesson Breather

//No code necessary

## 8. When to 'while' and when to 'for'

var count = 1; while(count < 10) { count++; console.log("This is a for loop and the count is..."); console.log("The count is " + count); } for(;count < 10;) { count++; console.log("This is a for loop and the count is..."); console.log("The count is " + count); }

## 9. The 'do'/'while' loop

var loopCondition = false; do { console.log("I'm gonna stop looping 'cause my condition is " + loopCondition + "!"); } while (loopCondition);

## 10. To learn it, you gotta 'do' it

var count = 1; var getToDaChoppa = function(){ do{ console.log("Welcome to 'Nam"); count++; } while(count<2); }; getToDaChoppa();

## 11. Review

for(var count = 1; count<2;count++) { console.log("Hello, world!") } do{ console.log("Hello, world!") count++; } while(count<2);

## 1. What you'll be building

var slaying = true; // A bit of new math magic to calculate the odds // of hitting the dragon. We'll cover this soon! var youHit = Math.floor(Math.random() \* 2); var damageThisRound = Math.floor(Math.random() \* 5 + 1); var totalDamage = 0; while (slaying) { if (youHit) { console.log("You hit the dragon and did " + damageThisRound + " damage!"); totalDamage += damageThisRound; if (totalDamage >= 4) { console.log("You did it! You slew the dragon!"); slaying = false; } else { youHit = Math.floor(Math.random() \* 2); } } else { console.log("The dragon burninates you! You're toast."); slaying = false; } }

## 2. Declare your variables

var slaying = true; var youHit = Math.floor(Math.random() \* 2); var damageThisRound = Math.floor(Math.random() \* 5 + 1); var totalDamage = 0;

## 3. The 'while' loop

var slaying = true; var youHit = Math.floor(Math.random() \* 2); var damageThisRound = Math.floor(Math.random() \* 5 + 1); var totalDamage = 0; while(slaying) { slaying = false; }

## 4. The first 'if' statement

var slaying = true; var youHit = Math.floor(Math.random() \* 2); var damageThisRound = Math.floor(Math.random() \* 5 + 1); var totalDamage = 0; while(slaying) { if(youHit === 1) { console.log("You have hit the Dragon!") } else { console.log("The Dragon has slain you!") } slaying = false; }

## 5. The second 'if' statement

var slaying = true; var youHit = Math.floor(Math.random() \* 2); var damageThisRound = Math.floor(Math.random() \* 5 + 1); var totalDamage = 0; while(slaying) { if(youHit === 1) { console.log("You have hit the Dragon!") totalDamage += damageThisRound; if(totalDamage >= 4) { console.log("You have slain the Dragon!"); } } else { console.log("The Dragon has slain you!") } slaying = false; }

## 6. Well done!

var slaying = true; var youHit = Math.floor(Math.random() \* 10); var damageThisRound = Math.floor(Math.random() \* 5 + 1); var totalDamage = 4; while(slaying) { if(youHit === 1) { console.log("You have hit the Dragon!") totalDamage += damageThisRound; if(totalDamage >= 4) { console.log("You have slain the Dragon! You have won!"); slaying = false; } else { youHit = Math.floor(Math.random() \* 2); } } else { console.log("The Dragon has slain you!") } slaying = false; }

Control Flow

**1. If/else**

var isEven = function(number) { if(number % 2 === 0) { return true; } else { return false; } };

**2. If/else if/else**

var isEven = function(number) { if(number % 2 === 0) { return true; } else if(isNaN(number) === true) { return "That is not a number, you must input a number!"; } else { return false; } };

**3. For or while**

for(var counter = 1;counter <= 5;counter++) { console.log("This is a for loop"); }

**4. Sneak preview: the switch statement**

var lunch = prompt("What do you want for lunch?","Type your lunch choice here"); switch(lunch){ case 'sandwich': console.log("Sure thing! One sandwich, coming up."); break; case 'soup': console.log("Got it! Tomato's my favorite."); break; case 'salad': console.log("Sounds good! How about a caesar salad?"); break; case 'pie': console.log("Pie's not a meal!"); break; default: console.log("Huh! I'm not sure what " + lunch + " is. How does a sandwich sound?"); }

**5. Adding to an existing switch**

var color = prompt("What's your favorite primary color?","Type your favorite color here"); switch(color) { case 'red': console.log("Red's a good color!"); break; case 'blue': console.log("That's my favorite color, too!"); break; case 'yellow': console.log("Yellow is okay...") break; default: console.log("I don't think that's a primary color!"); }

**6. Practice with switch**

var candy = prompt("What's your favorite candy?","Type your favorite candy here."); switch(candy) { case 'licorice': console.log("Gross!"); break; case 'gum': console.log("I like gum!"); break; case 'beets': console.log("...is that even a candy?"); break; default: console.log("I don't recognize that candy D:"); break; }

**7. More practice with switch**

scvar answer = prompt("How are you feeling","Insert Feels Here"); switch(answer) { case 'Happy': console.log("Yay, I'm happy too!"); break; case 'Sad': console.log("Aww, too bad!"); break; case 'Meh': console.log("Have a smile to brighten up your day!"); break; default: console.log("You're a robot..."); break; }

**8. All on your own**

var answer = prompt("What color is #TheDress?"); switch(answer) { case 'Blue and Black': console.log("Yeah!"); break; case 'Gold and White': console.log("No!"); break; case 'It does not Matter': console.log("Whatever"); break; default: console.log("That's not a valid answer."); break; }

**9. Mid-lesson breather**

var answer = prompt("What color is #TheDress?"); switch(answer) { case 'Blue and Black': console.log("Yeah!"); break; case 'Gold and White': console.log("No!"); break; case 'It does not Matter': console.log("Whatever"); break; default: console.log("That's not a valid answer."); break; }

**10. Overview**

// Complete lines 3 and 4! var iLoveJavaScript = true; var iLoveLearning = true; if(iLoveJavaScript && iLoveLearning) { // if iLoveJavaScript AND iLoveLearning: console.log("Awesome! Let's keep learning!"); } else if(!(iLoveJavaScript || iLoveLearning)) { // if NOT iLoveJavaScript OR iLoveLearning: console.log("Let's see if we can change your mind."); } else { console.log("You only like one but not the other? We'll work on it."); }

**11. And**

var hungry = true; var foodHere= true; var eat = function() { if(hungry && foodHere){ return true; } else{ return false; } };

**12. Or**

var tired = true; var bored = false; var nap = function() { if(tired || bored){ return true; } else{ return false; } }

**13. Not**

var programming = false; var happy = function() { if(programming != false){ return false; } else{ return true; } }

## 1. What you'll be building

var troll = prompt("You're walking through the forest, minding your own business, and you run into a troll! Do you FIGHT him, PAY him, or RUN?").toUpperCase(); switch(troll) { case 'FIGHT': var strong = prompt("How courageous! Are you strong (YES or NO)?").toUpperCase(); var smart = prompt("Are you smart?").toUpperCase(); if(strong === 'YES' || smart === 'YES') { console.log("You only need one of the two! You beat the troll--nice work!"); } else { console.log("You're not strong OR smart? Well, if you were smarter, you probably wouldn't have tried to fight a troll. You lose!"); } break; case 'PAY': var money = prompt("All right, we'll pay the troll. Do you have any money (YES or NO)?").toUpperCase(); var dollars = prompt("Is your money in Troll Dollars?").toUpperCase(); if(money === 'YES' && dollars === 'YES') { console.log("Great! You pay the troll and continue on your merry way."); } else { console.log("Dang! This troll only takes Troll Dollars. You get whomped!"); } break; case 'RUN': var fast = prompt("Let's book it! Are you fast (YES or NO)?").toUpperCase(); var headStart = prompt("Did you get a head start?").toUpperCase(); if(fast === 'YES' || headStart === 'YES') { console.log("You got away--barely! You live to stroll through the forest another day."); } else { console.log("You're not fast and you didn't get a head start? You never had a chance! The troll eats you."); } break; default: console.log("I didn't understand your choice. Hit Run and try again, this time picking FIGHT, PAY, or RUN!"); }

## 2. Prompt

var user = prompt("What is your name?");

## 3. .toUpperCase() and .toLowerCase()

var user = prompt("What is your name?").toUpperCase();

## 4. Switch

var user = prompt("What is your name?").toUpperCase(); switch(user){ case 'JOHN': console.log("I've been waiting"); break; case 'BRENT': console.log("I've been waiting"); break; case 'JACK': console.log("I've been waiting"); break; default: console.log("I've been waiting"); break; }

## 5. Logical Operators

var user = prompt("What is your name?").toUpperCase(); switch(user) { case 'JOHN': if(1+1==2 && 1 == 1) { console.log("I've been waiting for you..."); } else { console.log("Oops! You broke the game!"); } break; case 'BRENT': if(1-1==0 || 1 == 1) { console.log("I've been waiting for you..."); } break; case 'JACK': console.log("I've been waiting for you..."); break; default: console.log("I've been waiting for you..."); break; }

## 6. You did it!

var user = prompt("What is your name?").toUpperCase(); switch(user) { case 'JOHN': if(1+1==2 && 1 == 1) { console.log("I've been waiting for you..."); } else { console.log("Oops! You broke the game!"); } break; case 'BRENT': if(1-1==0 || 1 == 1) { console.log("I've been waiting for you..."); } break; case 'JACK': console.log("I've been waiting for you..."); break; default: console.log("I've been waiting for you..."); break; }

Data Structure

## 1. You know this!

var list = ["Bread", "Milk", "Peanut Butter"];

## 2. Access by offset

var languages = ["HTML", "CSS", "JavaScript", "Python", "Ruby"]; console.log(languages[2]);

## 3. Array Properties

var languages = ["HTML", "CSS", "JavaScript", "Python", "Ruby"]; console.log(languages[2]); console.log(languages.length);

## 4. Iterating over an array

var languages = ["HTML", "CSS", "JavaScript", "Python", "Ruby"]; for(var counter = 0; counter<5;counter++) { console.log(languages[counter]) }

## 5. Heterogeneous arrays

var myArray = [010101100110,false,"Test"];

## 6. Arrays of arrays

var newArray = [[1,2,3],[4,5,6],[7,8,9]];

## 7. Jagged arrays

var jagged = [[15,67,82],[56,45]];

## 8. Mid-lesson breather

//No code necessary

## 9. Nouns and verbs together

var phonebookEntry = {}; phonebookEntry.name = 'Oxnard Montalvo'; phonebookEntry.number = '(555) 555-5555'; phonebookEntry.phone = function() { console.log('Calling ' + this.name + ' at ' + this.number + '...'); }; phonebookEntry.phone();

## 10. Object Syntax

var me = { name: "John", age: 100 }

## 11. Creating a new object

var me = new Object(); me.name = "John"; me.age = 100;

## 12. Practice makes perfect

var object1 = new Object(); var object2 = new Object(); var object3 = new Object(); object1.name = "Object One" object2.name = "Object Two" object3.name = "Object Three"

## 13. Review

//No code necessary

## 14. Heterogenous arrays

var myArray = [15,true,"John",myObj]; var myObj = { type: 'fancy', disposition: 'sunny' };

## 15. Multidimensional arrays

var newArray = [[15,67,81],[myObj, 17]]; var myObj = { name: 'John', Relationship: 'Friend' };

## 16. Editing an existing object

var myObject = { name: 'Eduardo', type: 'Most excellent', interests: ['Sailing','Skiing','Programming'] };

## 17. Creating your own objects

var myOwnObject = new Object(); myOwnObject.name = "John G"; myOwnObject.gender = "Male"; myOwnObject.number = "555-555-555";

## 1. What you'll be building

var friends = {}; friends.bill = { firstName: "Bill", lastName: "Gates", number: "(206) 555-5555", address: ['One Microsoft Way','Redmond','WA','98052'] }; friends.steve = { firstName: "Steve", lastName: "Jobs", number: "(408) 555-5555", address: ['1 Infinite Loop','Cupertino','CA','95014'] }; var list = function(obj) { for(var prop in obj) { console.log(prop); } }; var search = function(name) { for(var prop in friends) { if(friends[prop].firstName === name) { console.log(friends[prop]); return friends[prop]; } } }; list(friends); search("Steve");

## 2. Creating your contact object

var friends = new Object();

## 3. Adding your friends

var friends = new Object(); friends.steve = {} friends.bill = {};

## 4. Adding properties

var friends = new Object(); friends.steve = { firstName: "Steve", lastName: "Smith", number: "666-666-666" }; friends.bill = { firstName: "Bill", lastName: "Smith", number: "666-666-666" };

## 5. Tossing in an array

var friends = new Object(); friends.steve = { firstName: "Steve", lastName: "Smith", number: "666-666-666", address: [] }; friends.bill = { firstName: "Bill", lastName: "Smith", number: "666-666-666", address: [] };

## 6. List 'em all!

var friends = new Object(); friends.steve = { firstName: "Steve", lastName: "Smith", number: "666-666-666", address: [] }; friends.bill = { firstName: "Bill", lastName: "Smith", number: "666-666-666", address: [] }; var list = function (friends) { for(var firstName in friends) { console.log(firstName); } }

## 7. Search for a friend

var friends = new Object(); friends.steve = { firstName: "Steve", lastName: "Smith", number: "666-666-666", address: [] }; friends.bill = { firstName: "Bill", lastName: "Smith", number: "666-666-666", address: [] }; var list = function (friends) { for(var firstName in friends) { console.log(firstName); } } var search = function (name) { for(var firstName in friends) { if(friends[firstName].firstName === name) { console.log(friends[firstName]); return friends[firstName]; } } }

## 8. Victory!

var friends = new Object(); friends.steve = { firstName: "Steve", lastName: "Smith", number: "666-666-666", address: [] }; friends.bill = { firstName: "Bill", lastName: "Smith", number: "666-666-666", address: [] }; var list = function (friends) { for(var firstName in friends) { console.log(firstName); } } var search = function (name) { for(var firstName in friends) { if(friends[firstName].firstName === name) { console.log(friends[firstName]); return friends[firstName]; } } }

objects

## . We've come a long, long...

(((3 \* 90) === 270) || !(false && (!false)) || "bex".toUpperCase() === "BEX"); answer = true;

## 2. Through the hard times...

// Here is an array of multiples of 8. But is it correct? var multiplesOfEight = [8,16,24,32,40,48]; // Test to see if a number from the array is NOT a true // multiple of eight. Real multiples will return false. var answer = multiplesOfEight[6] % 8 !== 0;

## 3. ...And the good!

for(var counter = 1;counter<21;counter++){ if(counter%5===0&&counter%3===0){ console.log("FizzBuzz"); } else{ if(counter%3===0){ console.log("Fizz"); } if(counter%5===0){ console.log("Buzz"); } if(counter%5!==0&&counter%3!==0){ console.log(counter); } } }

## 4. I have to celebrate you baby

var getReview = function (movie){ switch(movie) { case "Toy Story 2": return("Great story. Mean prospector."); break; case "Finding Nemo": return("Cool animation, and funny turtles."); break; case "The Lion King": return("Great songs."); break; default: console.log("I don't know!"); } }

## 5. I have to praise you like I should!

console.log("I'm ready for Objects!");

## 6. Intro

var bob = new Object()

## 7. Properties

var me = { age: 22, country: "United States" };

## 8. Accessing Properties

var bob = { name: "Bob Smith", age: 30 }; var susan = { name: "Susan Jordan", age: 25 }; // here we save Bob's information var name1 = bob.name; var age1 = bob.age; // finish this code by saving Susan's information var name2 = susan.name var age2 = susan.age

## 9. Accessing Properties, Part 2

// Take a look at our next example object, a dog var dog = { species: "greyhound", weight: 60, age: 4 }; var species = dog["species"]; // fill in the code to save the weight and age using bracket notation var weight = dog["weight"]; var age = dog["age"];

## 10. Another Way to Create

var bob = new Object(); bob.name = "Bob Smith"; bob.age = 30; var susan1 = { name: "Susan Jordan", age: 24 }; var susan2 = new Object(); susan2.name = "Susan Jordan"; susan2.age = 24;

## 11. Putting it all together

var snoopy = { species: "beagle", age: 10 } var buddy = new Object(); buddy.species = "golden retriever"; buddy.age = 5;

## 12. More Practicing Making Objects

var bicycle = new Object(); bicycle.speed = 0; bicycle.gear = 1; bicycle.frame\_material = "carbon fiber";

## 13. Function Review

// Accepts a number x as input and returns its square var square = function (x) { return x \* x; }; // Write the function multiply below // It should take two parameters and return the product var multiply = function (x,y) { return x\*y; }; multiply(15,10);

## 14. So What's a Method

// here is bob again, with his usual properties var bob = new Object(); bob.name = "Bob Smith"; bob.age = 30; // this time we have added a method, setAge bob.setAge = function (newAge){ bob.age = newAge; }; // here we set bob's age to 40 bob.setAge(40); // bob's feeling old. Use our method to set bob's age to 20 bob.setAge(20);

## 15. Why Are Method's Important?

var bob = new Object(); bob.age = 17; // this time we have added a method, setAge bob.setAge = function (newAge){ bob.age = newAge; }; bob.getYearOfBirth = function () { return 2014 - bob.age; }; console.log(bob.getYearOfBirth());

## 16. The 'this' Keyword

// here we define our method using "this", before we even introduce bob var setAge = function (newAge) { this.age = newAge; }; // now we make bob var bob = new Object(); bob.age = 30; // and down here we just use the method we already made bob.setAge = setAge; // change bob's age to 50 here bob.setAge(50);

## 17. "This" Works for Everyone

// here we define our method using "this", before we even introduce bob var setAge = function (newAge) { this.age = newAge; }; // now we make bob var bob = new Object(); bob.age = 30; bob.setAge = setAge; // make susan here, and first give her an age of 25 var susan = new Object(); susan.age = 25; susan.setAge = setAge; // here, update Susan's age to 35 using the method susan.setAge(35);

## 18. Make Your Own Method

var rectangle = new Object(); rectangle.height = 3; rectangle.width = 4; // here is our method to set the height rectangle.setHeight = function (newHeight) { this.height = newHeight; }; // help by finishing this method rectangle.setWidth = function (newWidth){ this.width = newWidth; } rectangle.setWidth(8); rectangle.setHeight(6); // here change the width to 8 and height to 6 using our new methods

## 19. More Kinds of Methods

var square = new Object(); square.sideLength = 6; square.calcPerimeter = function() { return this.sideLength \* 4; }; square.calcArea = function() { return Math.pow(this.sideLength, 2); }; var p = square.calcPerimeter(); var a = square.calcArea();

## 20. The Object Constructor

// here we make bob using the Object constructor var bob = new Object(); bob.name = "Bob Smith"; bob.age = 20; // add bob's age here and set it equal to 20

## 21. Custom Constructors

function Person(name,age) { this.name = name; this.age = age; } // Let's make bob and susan again, using our constructor var bob = new Person("Bob Smith", 30); var susan = new Person("Susan Jordan", 25); var george = new Person("George Washington", 275);

## 22. Try it Out!

function Cat(age, color) { this.age = age; this.color = color; } // make a Dog constructor here function Dog(age, breed, color) { this.age = age; this.breed = breed; this.color = color; }

## 23. More Options

function Person(name,age) { this.name = name; this.age = age; this.species = "Homo Sapiens"; } var sally = new Person("Sally Bowles", 39); var holden = new Person("Holden Caulfield", 16); console.log("sally's species is " + sally.species + " and she is " + sally.age); console.log("holden's species is " + holden.species + " and he is " + holden.age);

## 24. Constructors With Methods

function Rectangle(height, width) { this.height = height; this.width = width; this.calcArea = function() { return this.height \* this.width; }; this.calcPerimeter = function() { return this.height\*2+this.width\*2; } } var rex = new Rectangle(7,3); var area = rex.calcArea(); var perimeter = rex.calcPerimeter();

## 25. Constructors in Review

function Rabbit(adjective) { this.adjective = adjective; this.describeMyself = function() { console.log("I am a " + this.adjective + " rabbit"); }; } var rabbit1 = new Rabbit("fluffy"); var rabbit2 = new Rabbit("happy"); var rabbit3 = new Rabbit("sleepy");

## 26. Arrays of Objects

// Our person constructor function Person (name, age) { this.name = name; this.age = age; } // Now we can make an array of people var family = new Array(); family[0] = new Person("alice", 40); family[1] = new Person("bob", 42); family[2] = new Person("michelle", 8); family[3] = new Person("timmy", 6); // add the last family member, "timmy", who is 6 years old

## 27. Loop the Loop

//Person Constructor function Person(name,age) { this.name = name; this.age = age; } //Array of People var family = new Array(); family[0] = new Person("alice",40); family[1] = new Person("bob",42); family[2] = new Person("michelle",8); family[3] = new Person("timmy",6); //Loop Through New Array for(i=0;i<4;i++) { console.log(family[i].name) }

## 28. Passing Objects into Functions

// Our person constructor function Person (name, age) { this.name = name; this.age = age; } // We can make a function which takes persons as arguments // This one computes the difference in ages between two people var ageDifference = function(person1, person2) { return person1.age - person2.age; } var alice = new Person("Alice", 30); var billy = new Person("Billy", 25); // get the difference in age between alice and billy using our function var diff = ageDifference(alice,billy);

## 29. Try it Out!

// Our person constructor function Person (name, age) { this.name = name; this.age = age; } // We can make a function which takes persons as arguments // This one computes the difference in ages between two people var ageDifference = function(person1, person2) { return person1.age - person2.age; }; var olderAge = function(person1, person2) { if(person1.age>person2.age) { return person1.age; } else{ return person2.age; } } // Let's bring back alice and billy to test our new function var alice = new Person("Alice", 30); var billy = new Person("Billy", 25); console.log("The older person is " + olderAge(alice, billy));

## 30. What are Objects For?

var spencer = { age: 22, country: "United States" }; var spencer2 = new Object(); spencer2.age = 22; spencer2.country = "United States";

## 31. Properties

var snoopy = new Object(); snoopy.species = "beagle"; snoopy.age = 10; // save Snoopy's age and species into variables // use dot notation for snoopy's species var species = snoopy.species = "beagle"; // use bracket notation for snoopy's age var age = snoopy["age"] = 10;

## 32. Customizing Constructors

// 3 lines required to make harry\_potter var harry\_potter = new Object(); harry\_potter.pages = 350; harry\_potter.author = "J.K. Rowling"; // A custom constructor for book function Book (pages, author) { this.pages = pages; this.author = author; } // Use our new constructor to make the\_hobbit in one line var the\_hobbit = new Book(320,"J.R.R. Tolkien");

## 33. Methods

function Circle (radius) { this.radius = radius; this.area = function () { return Math.PI \* this.radius \* this.radius; }; this.perimeter = function () { return 2 \* Math.PI \* this.radius }; };

## 1. Digitizing People

var bob = { firstName: "Bob", lastName: "Jones", phoneNumber: "(650) 777-777", email: "bob.jones@example.com" }; console.log(bob.firstName); console.log(bob.lastName); console.log(bob.email);

## 2. More People

var bob = { firstName: "Bob", lastName: "Jones", phoneNumber: "(650) 777-7777", email: "bob.jones@example.com" }; var mary = { firstName: "Mary", lastName: "Johnson", phoneNumber:"(650) 888-8888", email: "mary.johnson@example.com" }; var contacts = [bob,mary]; console.log(contacts[1].phoneNumber);

## 3. Displaying People

var bob = { firstName: "Bob", lastName: "Jones", phoneNumber: "(650) 777-7777", email: "bob.jones@example.com" }; var mary = { firstName: "Mary", lastName: "Johnson", phoneNumber: "(650) 888-8888", email: "mary.johnson@example.com" }; var contacts = [bob, mary]; var printPerson = function(person){ console.log(person.firstName + " " + person.lastName); }; printPerson(contacts[0]); printPerson(contacts[1]);

## 4. Listing Everybody

var bob = { firstName: "Bob", lastName: "Jones", phoneNumber: "(650) 777-7777", email: "bob.jones@example.com" }; var mary = { firstName: "Mary", lastName: "Johnson", phoneNumber: "(650) 888-8888", email: "mary.johnson@example.com" }; var contacts = [bob, mary]; function printPerson(person) { console.log(person.firstName + " " + person.lastName); } function list(){ var contactsLength = contacts.length; for(i = 0;i<contactsLength;i++){ printPerson(contacts[i]) } } list();

## 5. Finding that Special Someone

var bob = { firstName: "Bob", lastName: "Jones", phoneNumber: "(650) 777-7777", email: "bob.jones@example.com" }; var mary = { firstName: "Mary", lastName: "Johnson", phoneNumber: "(650) 888-8888", email: "mary.johnson@example.com" }; var contacts = [bob, mary]; function printPerson(person) { console.log(person.firstName + " " + person.lastName); } function list() { var contactsLength = contacts.length; for (var i = 0; i < contactsLength; i++) { printPerson(contacts[i]); } } function search(lastName){ var contactsLength = contacts.length; for(i=0;i<contactsLength;i++){ if(this.lastName=contacts[i].lastName){ printPerson(contacts[i]); } } } search("Jones");

## 6. We Made a Friend!

var bob = { firstName: "Bob", lastName: "Jones", phoneNumber: "(650) 777-7777", email: "bob.jones@example.com" }; var mary = { firstName: "Mary", lastName: "Johnson", phoneNumber: "(650) 888-8888", email: "mary.johnson@example.com" }; var contacts = [bob, mary]; function printPerson(person) { console.log(person.firstName + " " + person.lastName); } function list() { var contactsLength = contacts.length; for (var i = 0; i < contactsLength; i++) { printPerson(contacts[i]); } } function search(lastName){ var contactsLength = contacts.length; for(i=0;i<contactsLength;i++){ if(this.lastName=contacts[i].lastName){ printPerson(contacts[i]); } } } function add(firstName,lastName,phoneNumber,email){ contacts[contacts.length] = { firstName: firstName, lastName: lastName, phoneNumber: phoneNumber, email: email }; } add("John","Gaidimas",2222222222,"john@gaidimas.com"); search("Jones");

objects II

## 1. An Objective Review

var james = { job:"programmer", married:false }; function Person(job, married) { this.job = job; this.married = married; } var gabby = new Person("student", true);

## 2. Fun with Functions

function Person(job, married) { this.job = job; this.married = married; this.speak = function(){ console.log ("Hello!"); return; }; } var user = new Person("Codecademy Student",false); user.speak();

## 3. Literally Speaking

var james = { job: "programmer", married: false, speak: function() { console.Log("Hello, I am feeling great"); console.Log("Hello, I am feeling okay"); } }; james.speak("great"); james.speak("just okay");

## 4. Can I See Your References?

var james = { job: "programmer", married: false, sayJob: function() { console.log("Hi, I work as a "+this.job); } }; // james' first job james.sayJob(); // change james' job to "super programmer" here james.job = "super programmer"; // james' second job james.sayJob();

## 5. Who's in Your Bracket?

var james = { job: "programmer", married: false }; // set to the first property name of "james" var aProperty = ["job"]; // print the value of the first property of "james" // using the variable "aProperty" console.log(james[aProperty]);

## 6. I.D. Please

// complete these definitions so that they will have // the appropriate types var anObj = { job: "I'm an object!" }; var aNumber = 42; var aString = "I'm a string!"; console.log(typeof anObj); // should print "object" console.log(typeof aNumber); // should print "number" console.log(typeof aString); // should print "string"

## 7. Know Thyself

var myObj = { name:"John" }; console.log( myObj.hasOwnProperty('name') ); // should print true console.log( myObj.hasOwnProperty('nickname') ); // should print false

## 8. Dressed to Impress

var suitcase = { shirt: "Hawaiian" }; if(suitcase.hasOwnProperty('shorts')) { console.log(suitcase.shorts) } else { suitcase.shorts = "red"; }

## 9. Getting IN=timate

var nyc = { fullName: "New York City", mayor: "Bill de Blasio", population: 8000000, boroughs: 5 }; for(var property in nyc) { console.log(property) }

## 10. List ALL the Properties!

var nyc = { fullName: "New York City", mayor: "Bill de Blasio", population: 8000000, boroughs: 5 }; for(var x in nyc){ console.log(nyc[x]); }

## 11. Class is in Session

function Person(name,age) { this.name = name; this.age = age; } // Let's make bob again, using our constructor var bob = new Person("Bob Smith", 30); var susan = new Person("Susan Jordan", 35); // make your own class here function Circle(radius) { this.radius = radius; }

## 12. Teach Snoopy

function Dog (breed) { this.breed = breed; } // here we make buddy and teach him how to bark var buddy = new Dog("Golden Retriever"); buddy.bark = function() { console.log("Woof"); }; buddy.bark(); // here we make snoopy var snoopy = new Dog("Beagle"); snoopy.bark = function() { console.log("Hello"); }; snoopy.bark();

## 13. How do Classes Help Us?

function Person(name,age) { this.name = name; this.age = age; } // a function that prints the name of any given person var printPersonName = function (p) { console.log(p.name); }; var bob = new Person("Bob Smith", 30); printPersonName(bob); var me = new Person("John Gaidimas", 150); printPersonName(me);

## 14. Prototype to the Rescue

function Dog (breed) { this.breed = breed; }; // here we make buddy and teach him how to bark var buddy = new Dog("golden Retriever"); Dog.prototype.bark = function() { console.log("Woof"); }; buddy.bark(); // here we make snoopy var snoopy = new Dog("Beagle"); /// this time it works! snoopy.bark();

## 15. Prototype Practice

function Cat(name, breed) { this.name = name; this.breed = breed; } // let's make some cats! var cheshire = new Cat("Cheshire Cat", "British Shorthair"); var gary = new Cat("Gary", "Domestic Shorthair"); Cat.prototype.meow = function() { console.log("Meow!"); } gary.meow()

## 16. It's All in the Genes

function Animal(name,numLegs) { this.name = name; this.numLegs= numLegs; } Animal.prototype.sayName = function() { console.log("Hi my name is " + this.name); } // provided code to test above constructor and method var penguin = new Animal("Captain Cook", 2); penguin.sayName();

## 17. Marching Penguins

function Animal(name, numLegs) { this.name = name; this.numLegs = numLegs; } Animal.prototype.sayName = function() { console.log("Hi my name is " + this.name); }; // create a Penguin constructor here function Penguin(name, numLegs) { this.name = name; this.numLegs = numLegs; } Penguin.prototype.sayName = function() { console.log("Hi my name is " + this.name); }; // our test code var theCaptain = new Penguin("Captain Cook", 2); theCaptain.sayName();

## 18. DRY Penguins

// the original Animal class and sayName method function Animal(name, numLegs) { this.name = name; this.numLegs = numLegs; } Animal.prototype.sayName = function() { console.log("Hi my name is " + this.name); }; // define a Penguin class function Penguin(name) { this.name = name; this.numLegs = 2; } Penguin.prototype = new Animal(); // set its prototype to be a new instance of Animal

## 19. Black (and White) Penguin Magic

// the original Animal class and sayName method function Animal(name, numLegs) { this.name = name; this.numLegs = numLegs; } Animal.prototype.sayName = function() { console.log("Hi my name is " + this.name); }; // define a Penguin class function Penguin(name) { this.name = name; this.numLegs = 2; } Penguin.prototype = new Animal(); // set its prototype to be a new instance of Animal var penguin = new Penguin("Admiral Ackbar"); penguin.sayName();

## 20. Penguins, Properties, and the Prototype

function Penguin(name) { this.name = name; this.numLegs = 2; } // create your Emperor class here and make it inherit from Penguin function Emperor(name) { this.name = name; } Emperor.prototype = new Penguin(); // create an "emperor" object and print the number of legs it has var emperor = new Emperor("Han Shot First"); console.log(emperor.numLegs);

## 21. Up the Food0I-mean-Prototype-Chain

// original classes function Animal(name, numLegs) { this.name = name; this.numLegs = numLegs; this.isAlive = true; } function Penguin(name) { this.name = name; this.numLegs = 2; } function Emperor(name) { this.name = name; this.saying = "Waddle waddle"; } // set up the prototype chain Penguin.prototype = new Animal(); Emperor.prototype = new Penguin(); var myEmperor = new Emperor("Jules"); console.log(myEmperor.saying); // should print "Waddle waddle" console.log(myEmperor.numLegs); // should print 2 console.log(myEmperor.isAlive); // should print true

## 22. Open to the Public

function Person(first,last,age) { this.firstName = first; this.lastName = last; this.age = age; } var john = new Person('John','Smith',30); var myFirst = john.firstName; var myLast = john.lastName; var myAge = john.age; //declare variable myAge set to the age of the john object.

## 23. Private Variables

function Person(first,last,age) { this.firstname = first; this.lastname = last; this.age = age; var bankBalance = 7500; } // create your Person var john = new Person("John","G",150); // try to print his bankBalance console.log(john.bankBalance)

## 24. Accessing Private Variables

function Person(first,last,age) { this.firstname = first; this.lastname = last; this.age = age; var bankBalance = 7500; this.getBalance = function() { return bankBalance; }; } var john = new Person('John','Smith',30); console.log(john.bankBalance); var myBalance = john.getBalance(); console.log(myBalance);

## 25. Private Methods

function Person(first,last,age) { this.firstname = first; this.lastname = last; this.age = age; var bankBalance = 7500; var returnBalance = function() { return bankBalance; }; this.askTeller = function() { return returnBalance; }; } var john = new Person('John','Smith',30); console.log(john.returnBalance); var myBalanceMethod = john.askTeller(); var myBalance = myBalanceMethod(); console.log(myBalance);

## 26. Passing Arguements

function Person(first,last,age) { this.firstname = first; this.lastname = last; this.age = age; var bankBalance = 7500; this.askTeller = function(pass) { if (pass == 1234) return bankBalance; else return "Wrong password."; }; } var john = new Person('John','Smith',30); var myBalance = john.askTeller(1234);

## 27. Looks For-in to Me

var languages = { english: "Hello!", french: "Bonjour!", notALanguage: 4, spanish: "Hola!" }; for(var x in languages) { if(typeof languages[x] == "string"){ console.log(languages[x]); } };

## 28. Hello? Yes, This is Dog

function Dog (breed) { this.breed = breed; }; Dog.prototype.sayHello = function() { console.log("Hello this is a " + this.breed + " dog") } var yourDog = new Dog("golden retriever"); yourDog.sayHello(); var myDog = new Dog("dachshund"); myDog.sayHello();

## 29. So Meta I can't Take It!

// what is this "Object.prototype" anyway...? var prototypeType = typeof Object.prototype; console.log(prototypeType); // now let's examine it! var hasOwn = Object.prototype.hasOwnProperty("hasOwnProperty"); console.log(hasOwn);

## 30. Private Eye

function StudentReport() { var grade1 = 4; var grade2 = 2; var grade3 = 1; this.getGPA = function() { return (grade1 + grade2 + grade3) / 3; }; } var myStudentReport = new StudentReport(); for(var x in myStudentReport) { if(typeof myStudentReport[x] !== "function") { console.log("Muahaha! " + myStudentReport[x]); } } console.log("Your overall GPA is " + myStudentReport.getGPA());

## 1. Shut the Shop!

//Create the object called cashRegister //and initialize its total property var cashRegister = { total: 0, } //Using dot notation change the total property cashRegister.total = 2.99;

## 2. Manually Add It Up?

var cashRegister = { total:0, add: function(itemCost){ this.total += itemCost; } }; cashRegister.add(0.98); cashRegister.add(1.23); cashRegister.add(4.99); cashRegister.add(0.45); //Show the total bill console.log('Your bill is '+cashRegister.total);

## 3. Short-Term Memory

var cashRegister = { total: 0, add: function(itemCost) { this.total += itemCost; }, scan: function (item) { switch (item) { case "eggs": this.add(0.98); break; case "milk": this.add(1.23); break; case "magazine": this.add(4.99); break; case "chocolate": this.add(0.45); break; } return true; } }; cashRegister.scan("eggs"); cashRegister.scan("eggs"); cashRegister.scan("magazine"); cashRegister.scan("magazine"); cashRegister.scan("magazine"); console.log('Your bill is '+cashRegister.total);

## 4. I Have to Scan It More Than Once?

var cashRegister = { total:0, add: function(itemCost){ this.total += itemCost; }, scan: function(item,num) { switch (item) { case "eggs": this.add(0.98\*num); break; case "milk": this.add(1.23\*num); break; case "magazine": this.add(4.99\*num); break; case "chocolate": this.add(0.45\*num); break; } } }; cashRegister.scan("eggs",4); cashRegister.scan("milk",4); cashRegister.scan("magazine",4); cashRegister.scan("chocolate",4); //Show the total bill console.log('Your bill is '+cashRegister.total);

## 5. Bleep Bleep

var cashRegister = { total:0, //Dont forget to add your property add: function(itemCost) { this.total += itemCost; this.lastTransactionAmount = itemCost; }, scan: function(item,quantity) { switch (item) { case "eggs": this.add(0.98 \* quantity); break; case "milk": this.add(1.23 \* quantity); break; case "magazine": this.add(4.99 \* quantity); break; case "chocolate": this.add(0.45 \* quantity); break; } return true; }, voidLastTransaction : function() { this.total -= this.lastTransactionAmount; this.lastTransactionAmount = 0; } }; cashRegister.scan('eggs',1); cashRegister.scan('milk',1); cashRegister.scan('magazine',1); cashRegister.scan('chocolate',4); cashRegister.voidLastTransaction(); cashRegister.scan("chocolate", 3); //Show the total bill console.log('Your bill is '+cashRegister.total);

## 6. Over the Moon

function StaffMember(name,discountPercent) { this.name = name; this.discountPercent = discountPercent; } var sally = new StaffMember("Sally",5); var bob = new StaffMember("Bob",10); var me = new StaffMember("John",20);

## 7. You Deserved It!

function StaffMember(name,discountPercent){ this.name = name; this.discountPercent = discountPercent; } var sally = new StaffMember("Sally",5); var bob = new StaffMember("Bob",10); var me = new StaffMember("John",20); var cashRegister = { total:0, lastTransactionAmount: 0, add: function(itemCost){ this.total += (itemCost || 0); this.lastTransactionAmount = itemCost; }, scan: function(item,quantity){ switch (item){ case "eggs": this.add(0.98 \* quantity); break; case "milk": this.add(1.23 \* quantity); break; case "magazine": this.add(4.99 \* quantity); break; case "chocolate": this.add(0.45 \* quantity); break; } return true; }, voidLastTransaction: function(){ this.total -= this.lastTransactionAmount; this.lastTransactionAmount = 0; }, applyStaffDiscount: function(employee) { this.total -= this.total\*employee.discountPercent/100; } }; cashRegister.scan('eggs',1); cashRegister.scan('milk',1); cashRegister.scan('magazine',3); cashRegister.applyStaffDiscount(me); // Show the total bill console.log('Your bill is '+cashRegister.total.toFixed(2));