Lecture Handout
Week 5: Sept 28-30, 2015

In the previous two weeks, we have learned about properties and methods on data types:

If the variable is a **String**:	If the variable is an **Array**:
`s.length` returns the number of characters in the string (Number)	`a.length` returns the number of elements in the array (Number)
`s[index]` returns the character at index `index` (String)	`a[index]` returns the character at index `index` (String)
`s.charAt(index)` returns the character at index `index` (String)	

Today there is one last property for Arrays: **push(value) adds value at array end**

If the variable is an **Array**:
var a = []; **a.push("Anything");** **a.push(42);**

CS 105's online store is doing great; it is time for an upgrade! Suppose we want to store the online cart in an updated format:

```
var cart = [
  { item: "light coat", price: 100.00 },
  { item: "leather gloves", price: 50.00 },
  { item: "cs105 sticker", price: 0.99 }
];
```

Puzzle #1: The data type of `cart` is: ___**array of objects**_____

Whenever an item needs to be added to the cart, our website calls the function `addItem`. This function is passed two parameters: the name of the item and the price. For example, the function calls to generate the above cart would be:

```
addItem("light coat", 100.00);
addItem("leather gloves", 50.00);
addItem("cs105 sticker", 0.99);
```

Puzzle #2: Define an empty Array for cart:

```
var cart = [];
```

(Assume this is in a global scope.

Puzzle #3: Write the `addItem` function to `push` an object into `cart`:

```
// Declare the function variable:
var addItem = function (itemName, itemPrice) {

    // Create the object:

var obj = {
  item: itemName;,
  price: itemPrice
};
    // Add it to the cart array:

cart.push( obj);
}
```

Suppose we want to give a **DOUBLE SALE**! This sale involves a discount for individual items over a certain price **and** a discount if the grand total is over a certain amount:

- If an item is over $ _____**3**_____, take __**10**___% off!
- If the grand total is over $ _____**150**_____, take __**20**___% off!

Puzzle #4: Write the JavaScript to loop through the array to visit each object. In each object, create a new property `discountPrice` that has the discounted price of the item *(even if there is no discount!)*.

```
// Loop through the Array
for (var i=0; i<cart.length; i++) { var item = cart[i];
//For each item, check if price>3 //if so, apply a 10% discount
var newPrice = item.price;
if (item.price >3) {
newPrice = newPrice*0.9;
}

//Set .discountPrice to the price
```

```
item.discountPrice = newPrice;
alert(JSON.stringify(Cart));
}
```

Before the code for Puzzle #4 runs, our cart had the following value:

```
var cart = [
  { item: "light coat", price: 100.00 },
  { item: "leather gloves", price: 50.00 },
  { item: "cs105 sticker", price: 0.99 }
];
```

Puzzle #5: What is the value of `cart` after running the code in Puzzle #4?

```
cart = [
{item: “leather coat”, price: 100.00, discountPrice: 90.00}, {item:
“leather gloves”, price: 50.00, discountPrice: 45.00}, {item: cs105
sticker”, price: 0.99, discountPrice: 0.99}
];
```

Puzzle #6: Write a JavaScript function that returns the total price of the cart, applying any grand total discounts before returning the value.

```
var total = function (cart) { var sum = 0;
for (var i=0; i<cart.length; i++) { var item = cart[i];
sum = sum + item.discountPrice; }
if (sum > 150) {
sum = sum * 0.8;
}
return sum; }
```

One more example, this time using weather data:

```
var weatherData = [
  { week: 1, temps: [82, 84, 85, 72, 73, 75, 79] },
  { week: 2, temps: [84, 87, 89, 89, 76, 78, 81] },
  { week: 3, temps: [83, 74, 72, 71, 72, 74, 76] }
];
```

We want to create a function that returns a new Array of Objects where each element in the array contains a week and `highTemp`, where the `highTemp` was the high temperature for the week.

Puzzle #7: First, will the return value look like?

```
var new_weatherdata= [
{week: 1, highTemp: 85},
{week: 2, highTemp: 89},
{week: 3, highTemp: 83}
];
```

Puzzle #8: Write the JavaScript function:

```
var highTempdata = function(weatherData) {
  var new_array = [];
  for (var i=0; i<weatherData.length; i++) {
    var week1 = weatherData[i].week;
    var highTemp1 = getmax(weatherData[i].temps);
    var new_object = {
      week: week1,
      highTemp:  highTemp1
    };
    new_array.push(new_object);
  }
  return (new_array);
};
function getmax(array) {
  var max = array[0];
  for (var i = 0; i < array.length; i++) {
    if (array[i] > max) {
      max = numbers[i];
    }
  }
return (max); }
```

Reminder: CS 105 Midterm Exam

Monday, October 12, 2015, 7:30pm – 9:00pm

(Cannot make it? Ensure you sign up for a conflict exam starting Sept. 28)