Arrays

# Lesson Notes

## What about repeating data?

* We have a way to group values (Object)
* What if we have repeating data?
* For example, schools in a school district.
* We can model it as an object:
  + var schoolDistrict = {
  + school1: school1,
  + school2: school2,
  + school3: school3
  + };
* We can count the number of students in a school:
  + function countStudentsInSchool(school) {
  + // (algorithm to calculate number of students in a school)
  + }
* But to count the student in the district, we have to count each school separately:
  + function countStudentsInDistrict(schoolDistrict) {
  + var districtTotal = countStudentsInSchool(schoolDistrict.school1) +
  + countStudentsInSchool(schoolDistrict.school2) +
  + countStudentsInSchool(schoolDistrict.school3);
  + return districtTotal;
  + }
* If we add another school to the district, we have to change our code.
* There has to be a better way!
* We want to be able to group these values in memory without having to name them.

## Arrays in Javascript

* An array is an ordered group of zero or more elements
  + Each element has a value
  + Each element has an integer index
* An array has a length property
  + The length is the number of elements in the array

## How to write an array

* Syntax for an array
* Syntax for accessing element
* Syntax for updating an element
* Syntax for accessing length

## School district example

* School district is an array of schools
  + var schoolDistrict = [school1, school2, school3];
* We can find the number of schools (3) using the length property:
  + var numberOfSchools = schoolDistrict.length;
* We can find the **first** school by writing:
  + var firstSchool = schoolDistrict[**0**];
* We can find the **second** school by writing:
  + var secondSchool = schoolDistrict[**1**];
* We could also have written:
  + var schoolDistrict = [];
  + schoolDistrict[0] = school1;
  + schoolDistrict[1] = school2;
  + schoolDistrict[2] = school3;

## Arrays represent a group of elements in memory

* The elements of an array are grouped together
  + We have a “reference” to those variables
  + When we pass these to a function, we are passing the reference
  + This means that functions can read the memory, write the memory
* This is just like object references
* Visual for this (like object references)
  + Show school district variable pointing to array, which is pointing to schools
  + Show school1..school3 pointing to schools

## Using iteration with arrays

* We want to count all the students in the district.
  + We need a way to apply our function to each school.
  + We need to build up a total of all schools in the district.
* An iterative algorithm:
  + Start with a district total 0
  + Iterate:
    - Start at first index of the array
      * 0
    - End at last index of the array
      * schoolDistrict.length – 1
    - Each iteration
      * Calculate school total
      * Add school total to district total
* The iterative algorithm in javascript:
  + function countStudentsInDistrict(schoolDistrict) {
  + var districtTotal = 0;
  + for (var index = 0; index < schoolDistrict.length; index = index + 1) {
  + var school = schoolDistrict[index];
  + var schoolTotal = countStudentsInSchool(school);
  + districtTotal = districtTotal + schoolTotal;
  + }
  + return districtTotal;
  + }

## Exercise

* Type in the school district example
* Use provided source code for the schools (which have school names in them)
* Step through the debugger
* Notice the school variable getting updated to each school
* Check the total result

## Appending an item to an array

* Appending means “adding to”, in CS, typically “adding to the end”
* Show diagram of an array with school1, school2,school2
* Add school4 at index == array.length
* Show before and after

## Inserting an item into an array

* Show diagram of an array with school1,school2,school3
* Show assignments necessary for each case
* Add school4 at index 0
* Show before and after
* Add school4 at index 1
* Show before and after
* Add school4 at index 2
* Show before and after

## Exercises:

* Append item:
  + Write an appendElement function that works for any length array
  + Start with an empty array
  + Use appendElement to build an array of three schools
* Insert item:
  + Write an insertElement function that inserts an element at an index
  + Using array of three schools
    - Insert school4 at index 0
    - Insert school4 at index 1
    - Insert school4 at index 2

## Removing all the elements of the array

* Show diagram of an array with school1, school2,school3
* Update length to be zero
* Show before and after

## Removing the last item of an array

* Show diagram of an array with school1, school2,school3
* Make length one less
* Show before and after

## Removing an item in the middle of an array

* Show diagram of an array with school1,school2,school3
* Remove index 0
* Remove index 1
* Remove index 2
* Show before and after
* Show assignments necessary for each case

## Exercises:

* Remove all elements
  + Write a removeAllElements function that removes all element from an array
  + Test this with an array of three schools
  + Verify no schools left in array in debugger
* Remove last element:
  + Write a removeLastElement function that removes the last element from an array
  + Call this 3 times with same array of three schools
  + Verify schools are being deleted
* Remove element:
  + Write a removeElement function that removes an element at an index
  + Test with array of three schools at index 0
  + Test with array of three schools at index 1
  + Test with array of three schools at index 2
  + Verify correct school is deleted

## Reversing the elements of an array

* Show diagram of an array with school1,school2,school3,school4,school5
  + Swap(0, 4)
  + Swap(1, 3)
  + Swap(2, 2)
* Show diagram of an array with school1,school2,school3,school4
  + Swap(0, 3)
  + Swap(1, 2)
* Show diagram of an array with school1,school2,school3
  + Swap(0, 2)
  + Swap(1, 1)
* Show swaps for each step
* Need a function
  + swap(array, index1, index2)

## Exercise:

* Build an array of 4 schools
* Reverse the array
* Verify
* Build an array of 3 schools
* Reverse the array
* Verify

## So now you tell me…

* There is an easier way… ☺
* Appending an element to an array
  + array.push(element);
* Inserting an element into an array at an index:
  + array.splice(index, 0, element);
* Removing an element from the end of an array
  + array.pop();
* Removing an element from an array at an index:
  + array.splice(index, 1);
* Reversing the elements of an array:
  + array.reverse();