# JAVASCRIPT PRE-MADE FUNCTIONS

## Arrays

Intersection between 2

function findIntersection(line1, line2){

return line1.filter(function(item) {

return (line2.indexOf(item) != -1);

});

};

## Object

Browse through long object and log all the sub-objects which are the values for the key ‘title’

For…in iterates through properties of an object

If object[prop] isn’t an object, then there cannot be a key-value pair inside so we skip it

If object[prop] is an object, then we keep looking for the key “title”

function iterateForTitle(obj) {

for (var property in obj) {

if (typeof obj[property] === "object") {

iterateForTitle(obj[property])

} else if (property === 'title'){

console.log(property + " " + obj[property]);

}

}

}

List ALL properties of an object, as in a tree

Same as above, itereates through objects

* keep iterating as long as the sub-obj is an object
* when sub is not object, this is the end of the line so we log it
* “stack” keeps storing the full path the item
* 🡪 run iterate(obj, “”) *// Key to start with stack empty string*

function iterate(obj, stack) {

for (var property in obj) {

if (obj.hasOwnProperty(property)) {

if (typeof obj[property] == "object") {

iterate(obj[property], stack + '.' + property);

} else {

console.log(property + " " + obj[property]);

$('#output').append($("<div/>").text(stack + '.' + property))

} } } }

<http://jsfiddle.net/tbynA/1/>

<http://stackoverflow.com/questions/15690706/recursively-looping-through-an-object-to-build-a-property-list>