inspirito do your business and we'll do the rest

User Interface Engineering

Javascript Coding Exercise

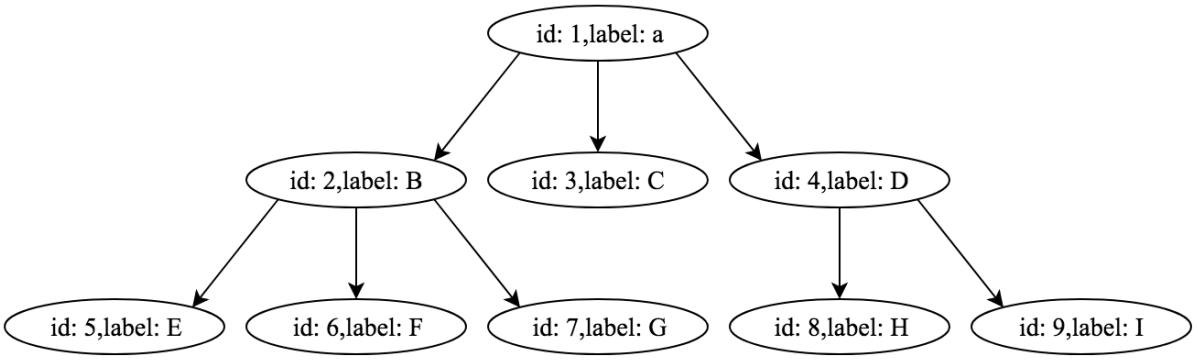
Candidate Name:

Instructions

Please complete on your own, using only standard reference materials and documentation found on the web. Include reasonable test input examples and the corresponding results (ie. send either a file that can be run, or a link to jsfiddle or codepen with your code).

Problem 1

Give the JSON representation of the following tree structure. Each node has an id (an integer, required), a label (a string, optional), and an array of child nodes (optional).



Problem 2

Write a JavaScript function that uses AJAX to retrieve the JSON string in Problem 1 and converts it to a JavaScript object. Stub out or mock an imaginary server, the point is to see the response handler and AJAX code. You can use vanilla JavaScript or your favorite framework for the AJAX function.

Problem 3

Using the object you created in Problem 2, write a recursive JavaScript function that, given a node id, returns the label string of the node with the matching id.

Problem 4

Create an HTML page with a form containing the following fields:

Field	Element type	Required?	Minimum	Maximum	Data type
Name	Input	Yes	1	100	Alpha
Address 1	Input	Yes	1	100	Alphanumeric
Address 2	Input	No	0	100	Alphanumeric
City	Input	Yes	1	50	Alphanumeric
State	Select	Yes	2	2	Alpha
Zip Code	Input	Yes	5	5	Numeric

Add OK and Cancel buttons to the bottom of the form. Clicking on the OK button validates the field values and displays any errors. Clicking on the Cancel button resets the form. Use CSS to nicely style the labels and layout, use JavaScript to handle the OK and Cancel button click events. When validating each field value, highlight all fields in error and display appropriate error messages in any way you wish.

Resources

To generate the tree above, use http://webgraphviz.com/ , with the following:

```
digraph tree {
        "id: 1,label: a" -> "id: 2,label: B";
        "id: 1,label: a" -> "id: 3,label: C";
        "id: 1,label: a" -> "id: 4,label: D";

        "id: 2,label: B" -> "id: 5,label: E";
        "id: 2,label: B" -> "id: 6,label: F";
        "id: 2,label: B" -> "id: 7,label: G";

        "id: 4,label: D" -> "id: 8,label: H";
        "id: 4,label: D" -> "id: 9,label: I";
};
```