Adding a Child

7 min

The next task is to add a child to our tree. Each child in our children

Preview: Docs Loading link description

[array](#)

 has to be an instance of a TreeNode, however we want to allow our user interface to accept adding data in other forms as well.

For instance, if our

Preview: Docs Loading link description

[method](#)

 to add a child is .addChild(), we want to accommodate calling tree.addChild(3) as well as tree.addChild(new TreeNode(3)).

Instructions

1.   Checkpoint 1 Passed

1.

Below the constructor, define another method, .addChild() which takes one parameter, child.

2.   Checkpoint 2 Passed

2.

Inside .addChild(), check if child is an instance of TreeNode. If it is, add child to the end of the children array. Otherwise, create a TreeNode instance for it before adding it to the children array.

Hint

To check if a variable is an instance of something, use the instanceof keyword.

```
let myerror= new Error('Whoops');
console.log(myerror instanceof Error); // returns true
```

To add an item to an array, use .push().

```
let colors = [];
colors.push('blue');
```

3.   Checkpoint 3 Passed

3.

Open **script.js**, and do the following:

- Add a child of value 15 to the tree object.

- Display the output of tree in the terminal

4.   Checkpoint 4 Passed

4.

In **script.js**, do the following:

- o Add another child by creating a TreeNode for it with value 30
- o Add this child to the tree object.
- o Display the output of tree on the terminal

TreeNode.js

```
class TreeNode {
 constructor(data) {
  this.data = data;
  this.children = [];
 }

 addChild(child) {
  if (child instanceof TreeNode) {
   this.children.push(child)
  }
  else {
   this.children.push(new TreeNode(child));
  }
 }
};

module.exports = TreeNode;
```

script.js

```
const TreeNode = require('./TreeNode');

const tree = new TreeNode(1);

console.log(tree);
```

```
tree.addChild(15);

console.log(tree);


tree.addChild(new TreeNode(30));

console.log(tree);
```

>>Output

```
TreeNode { data: 1, children: [] }

TreeNode { data: 1, children: [ TreeNode { data: 15, children: [] } ] }

TreeNode {
  data: 1,
  children:
   [ TreeNode { data: 15, children: [] },
     TreeNode { data: 30, children: [] } ] }
```