

Exercise (Instructions):Node Modules: Callbacks and Error Handling

Objectives and Outcomes

In this exercise, you will learn about callbacks and error handling in Node applications. At the end of this exercise, you will be able to:

- Using Callbacks and error handling in Node applications
- Using external Node modules

Using Callbacks and Error Handling

- Create a file named *rectangle-2.js* and add the following code to this file:

```
1 module.exports = function(x,y,callback) {  
2   try {  
3     if (x < 0 || y < 0) {  
4       throw new Error("Rectangle dimensions should be greater than zero: l = "  
5         + x + ", and b = " + y);  
6     }  
7     else  
8       callback(null, {  
9         perimeter: function () {  
10           return (2*(x+y));  
11         },  
12         area:function () {  
13           return (x*y);  
14         }  
15       });  
16   }  
17   catch (error) {  
18     callback(error,null);  
19   }  
20 }
```

- Then, create a file named *solve-2.js* and include the following code in there:

```

1 var rect = require('./rectangle-2');
2
3 function solveRect(l,b) {
4     console.log("Solving for rectangle with l = "
5         + l + " and b = " + b);
6     rect(l,b, function(err,rectangle) {
7         if (err) {
8             console.log(err);
9         }
10        else {
11            console.log("The area of a rectangle of dimensions length = "
12                + l + " and breadth = " + b + " is " + rectangle.area());
13            console.log("The perimeter of a rectangle of dimensions length = "
14                + l + " and breadth = " + b + " is " + rectangle.perimeter());
15        }
16    });
17 };
18
19 solveRect(2,4);
20 solveRect(3,5);
21 solveRect(-3,5);

```

- To run the Node application, type the following at the prompt:

```
1 node solve-2
```

Using yargs External Node module

- Install the *yargs* Node module by typing the following at the prompt:

```
1 npm install yargs --save
```

- Then, create another file named *solve-3.js* and add the following code to it:

```

1 var argv = require('yargs')
2   .usage('Usage: node $0 --l=[num] --b=[num]')
3   .demand(['l','b'])
4   .argv;
5
6 var rect = require('./rectangle-2');
7
8 function solveRect(l,b) {
9     console.log("Solving for rectangle with l = "
10        + l + " and b = " + b);
11     rect(l,b, function(err,rectangle) {
12         if (err) {
13             console.log(err);
14         }
15         else {
16             console.log("The area of a rectangle of dimensions length = "
17                 + l + " and breadth = " + b + " is " + rectangle.area());
18             console.log("The perimeter of a rectangle of dimensions length = "
19                 + l + " and breadth = " + b + " is " + rectangle.perimeter());
20         }
21     });
22 };
23
24 solveRect(argv.l,argv.b);

```

- To run the Node application, type the following at the prompt:

```
1 node solve-3
```

Conclusions

In this exercise, you learnt about using Callbacks and error handling in Node applications. In addition you learnt about using external Node modules.

✓ Complete

