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

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:

```
module.exports = function(x,y,callback) {
  try {
    if (x < 0 || y < 0) {
        throw new Error("Rectangle dimensions should be greater than zero: l
= "
                            + x + ", and b = " + y);
    }
    else
         callback(null, {
            perimeter: function () {
                          return (2*(x+y));
                        },
            area:function () {
                           return (x*y);
                        }
    });
  }
  catch (error) {
        callback(error,null);
  }
}
```

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

```
var rect = require('./rectangle-2');
```

```
function solveRect(l,b) {
    console.log("Solving for rectangle with l = "
                + l + " and b = " + b);
    rect(l,b, function(err,rectangle) {
        if (err) {
            console.log(err);
        }
        else {
            console.log("The area of a rectangle of dimensions length = "
                 + l + " and breadth = " + b + " is " + rectangle.area());
            console.log("The perimeter of a rectangle of dimensions length =
"
                 + l + " and breadth = " + b + " is " +
rectangle.perimeter());
        }
    });
};

solveRect(2,4);
solveRect(3,5);
solveRect(-3,5);
```

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

```
    node solve-2
```

Using yargs External Node module

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

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

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

```
var argv = require('yargs')
    .usage('Usage: node $0 --l=[num] --b=[num]')
    .demand(['l','b'])
    .argv;

var rect = require('./rectangle-2');
```

```
function solveRect(l,b) {
    console.log("Solving for rectangle with l = "
                + l + " and b = " + b);
    rect(l,b, function(err,rectangle) {
        if (err) {
            console.log(err);
        }
        else {
            console.log("The area of a rectangle of dimensions length = "
                  + l + " and breadth = " + b + " is " + rectangle.area());
            console.log("The perimeter of a rectangle of dimensions length =
"
                  + l + " and breadth = " + b + " is " +
rectangle.perimeter());
        }
    });
};

solveRect(argv.l,argv.b);
```

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

```
     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.